University of Venda

SHORT TERM WIND POWER FORECASTING IN SOUTH AFRICA USING NEURAL NETWORKS

By

Lucky Oghenechodja Daniel

17004991

submitted in fulfilment of the requirements for the degree of Master of Science in Statistics

in the

Department of Statistics
School of Mathematical and Natural Sciences
University of Venda
Thohoyandou, Limpopo
South Africa

Supervisor : Dr. Caston Sigauke

Co-Supervisor : Dr. Colin Chibaya

Co-Supervisor : Mr. Rendani Mbuvha

Date submitted: 11 August 2020

Abstract

Wind offers an environmentally sustainable energy resource that has seen increasing global adoption in recent years. However, its intermittent, unstable and stochastic nature hampers its representation among other renewable energy sources. This work addresses the forecasting of wind speed, a primary input needed for wind energy generation, using data obtained from the South African Wind Atlas Project. Forecasting is carried out on a two days ahead time horizon. We investigate the predictive performance of artificial neural networks (ANN) trained with Bayesian regularisation, decision trees based stochastic gradient boosting (SGB) and generalised additive models (GAMs). The results of the comparative analysis suggest that ANN displays superior predictive performance based on root mean square error (RMSE). In contrast, SGB shows outperformance in terms of mean average error (MAE) and the related mean average percentage error (MAPE). A further comparison of two forecast combination methods involving the linear and additive quantile regression averaging show the latter forecast combination method as yielding lower prediction accuracy. The additive quantile regression averaging based prediction intervals also show outperformance in terms of validity, reliability, quality and accuracy. Interval combination methods show the median method as better than its pure average counterpart. Point forecasts combination and interval forecasting methods are found to improve forecast performance.
Keywords: Additive quantile regression averaging; Forecasts combination; Machine learning; Point and interval forecasting; Renewable energy; Wind energy.

Declaration

I, Lucky O. Daniel, [student Number: 17004991], hereby declare that the dissertation titled: “Short Term Wind Power Forecasting in South Africa Using Neural Networks” for the Master of Science degree in Statistics at the University of Venda, hereby submitted by me, has not been submitted for any degree at this or any other university, that it is my own work in design and in execution, and that all reference material contained therein has been duly acknowledged.

Signature:  Date: 11 August 2020

Dedication

This research is dedicated to the glory of God Almighty the giver of life, health, wisdom and strength. To Him alone be all the glory.

I dedicate this research also in loving memory of my late sister Patience Ejiroghene Nelson (Nee Daniel) for her support and believe in me. To my Late Dad, Pa Daniel Ikpesu Owhoka, your love for education and your resolve to see your children through school despite your meagre earnings has set me on this path and I am not yet done dad, I am going all the way to make your name great. I thank God for you always Mba (my father) as we used to call you. We lost both of you in the month of November, we receive God's consolation for these great loses.

I dedicate this work most especially to my Mum, Mrs Elizabeth Oturaerevwe Daniel. I cannot thank God enough for you Mummy, for your consolation and constant faith in God for this your sick child. Your prayers over me and my siblings has shown forth in this great feat and more to come. Thank you Mummy. I love you Mum.

Acknowledgements

I want to thank the DST-CSIR for providing the funding for this Master of Science degree in e-Science programme for the duration of the course work at the University of Witwatersrand and research at the University of Venda. The support of the DST-CSIR National e-Science Postgraduate Teaching and Training Platform (NEPTTP) towards this research is hereby acknowledged.

My profound and heartfelt gratitude goes to Dr Moyo (NEPTTP node director, Univen) and Dr Chibaya for motivating for my enrolment on the first e-Science cohort and also to Prof Celik, Ms Casey Sparks and Ms Caz for accepting me in the programme. I appreciate the University of Venda and the University of Witwatersrand, for this collaboration and for accepting me as their student.

I acknowledge the help, assistance and encouragements from my colleagues at Wits in persons of Oni Oluwabamigbe Oghenetega, Jeremiah Olamijuwon, Witness Maarke, Mrs Zakka, Ozioma, Mr Gbenga, Nurcia, Neshunda, Tabang and others.

In the same manner, My Univen colleagues Tendani Mutavhatsindi (Prof), Masala Netshiozwi (Slayer), Maduvha Thayani (strong woman) are worthy of being acknowledged. We are a very strong and indivisible team. Your help is quite remarkable and thank God we made it through. Thank you guys for the gift of your friendship, help

and supports. Greater heights guys. My CT family, My friends and their spouses, Funsho Ogunshola and Jide Runsewe, Ms Onoji. My siblings and their spouses. My Mum, nieces and nephews. Thank you all so much.

Special acknowledgement to my Supervisor Dr. Caston Sigauke. Your highly skilled and spirited approach to research has birthed this work. To my Co-Supervisors Dr Colin Chibaya, and Mr Rendani Mbuvha, you are a great gift to me and I thank you so much sirs for your constructive criticisms and directions in this research.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AI– Artificial Intelligence

ANN –Artificial Neural Network

BNN - Bayesian Neural Network

ML – Machine Learning

SL – Statistical Learning

SGB – Stochastic Gradient Boosting

GAM – Generalized Additive Model

NNA – Neural Network Algorithm

LQRA – Linear Quantile Regression Averaging

Lasso – Least Absolute Shrinkage and Selector Operator

AQRA – Additive Quantile Regression Averaging

ARIMA – Auto-Regressive Integrated Moving Average

SARIMA– Seasonal Auto-Regressive Integrated Moving Average

SARIMAX– Seasonal Auto-Regressive Integrated Moving Average for Exogenous Variable

PI – Prediction Interval

PIW- Prediction interval Width

PICP - Prediction Interval Coverage Probability

PINC – Prediction Interval Nominal Coverage

PINAW – Prediction Interval Normalised Average Width

PICAW – Prediction Interval Coverage Average -Normalised Average

PINAD – Prediction Interval Normalised Average Deviation

RMSE – Root Mean Square Error

MAE – Mean Average Error

MAPE– Mean Average Percentage Error

ABLL – Actual Below Lower Limit

AAUP – Actual Above Upper Limit

Research Output

Peer Reviewed Journal Publication

Daniel, L. O., Sigauke, C., Chibaya, C., Mbuvha, R. (2020). Short-Term Wind Speed Forecasting Using Statistical and Machine Learning Methods. Algorithms, 13(6), 132.

Chapter 1

Introduction

Warnings about the various possible consequences of global warming across the planet as reported in research from several nations has encouraged the use of renewable energy resources such as wind energy (Ferreira et al., 2019). This is a viable strategy for mitigating the problem of greenhouse gases (GHG) in the atmosphere from various human activities (Crate and Nuttall, 2016). GHG, amongst many other effects of global warming such as global climate change, has its attendant effect evident in environmental degradation and pollution (Ferreira et al., 2019; Crate and Nuttall, 2016). Apart from these, reduction in and skyrocketing prices of conventional energy sources necessitates the use of renewable energy sources (Lin, 2007; Chen and Folly, 2018).

There are various natural processes for generating renewable energy. These natural processes involves Wind, Solar, Biomass, Geothermal, Hydro-power among others. They generate renewable energy through the use of heat from the sun, heat and speed of wind in the atmosphere, various anthropological activities and the ocean waves (Lin, 2007). The wind is an efficient, affordable, pollution free, renewable and abundant energy source (Pinson et al., 2013; Barbosa de Alencar et al., 2017). In recent times, the rising level of the energy generated from the wind has been witnessed in developed countries and in research publications (Pinson et al., 2013). As of 2012, 'the world wind

energy association (WWEA) informed that the world wide cumulative installed wind power capacity reached 254GW (Pinson et al., 2013). Countries such as Denmark and Uruguay use 40 and 20 percent of this energy respectively for their power generation (Barbosa de Alencar et al., 2017). Electric energy generated from wind energy installed capacity is expected to reach over 800GW before the end of 2021 (Barbosa de Alencar et al., 2017; Pinson et al., 2013). Wind energy thus presents prospects for an alternative power source in curbing the power crises South Africa is facing and also a viable power export to markets deficient of electric energy in the African continent.

The earth's surface and atmosphere being hit by Solar energy along with planetary rotation which results in uneven heating of the earth's atmosphere and differentials in atmospheric pressures are causalities for wind formation (Pinto et al., 2013; Ferreira et al., 2019). Wind is thus formed from the displacement in air masses, due to the difference in atmospheric pressure of two distinct regions, affected by natural effects of the continent, sea level, heights, longitude, and soil roughness (Barbosa de Alencar et al., 2017). Wind energy however, is generated through the contact of the wind with blades of wind devices such as wind turbines (WT). The blades of the WT when rotating converts wind speed into mechanical energy. This in turn drives the rotor of a wind generator to produce electricity (Pinson et al., 2013; Barbosa de Alencar et al., 2017). According to the kinetic energy equation, the quantity of energy generated from wind is a function of its speed (v) and mass (m). Hence, for generating electric power from wind, it is very important to make a good wind speed prediction (Chen and Folly, 2018; Barbosa de Alencar et al., 2017) & (Zhu et al., 2012).

Wind energy generation with many promising prospects, however, is faced with the challenge of variability of the wind speed. The fluctuating, intermittent, and stochastic nature of the wind makes predicting power generation a Herculean task (Abhinav et al., 2017). Also, the non-linear, non stationary characteristics of the wind power temporal series makes accurately forecasting power generation difficult (Barbosa de

Alencar et al., 2017). The predictability of the wind speed amongst other wind information is essential for the assessment of wind energy exploitation purposes such as wind power generation. Hence, accurate wind speed prediction helps in maximizing wind power generating facilities by reducing mistakes and economic cost involved in the planning and effective running of such facilities (Ferreira et al., 2019).

Wind speed prediction methods for electrical energy exploitation purpose gained attention in recent research. These methods are broadly classified into deterministic and probabilistic forecasting models (Yan et al., 2014). Most literature lists different methods for wind speed/power forecasting some of which are the persistence, numerical, statistical and hybrid methods (Morina et al., 2016; Giebel and Kariniotakis, 2017). The statistical methods are seen as including the artificial intelligence (AI) methods, especially the artificial neural networks (ANN) method. ANN is a black-box statistical method and non-ANN methods are seen as grey-box statistical methods (Giebel and Kariniotakis, 2017). Other authors, however, classify the AI methods as being non-statistical and broadly classify wind speed/power forecasting methods into conventional statistical methods and AI methods (Verma et al., 2018; Barbosa de Alencar et al., 2017; Chen and Folly, 2018). Statistical methods are based on statistical time series using the previous history of wind data to forecast over the next short period of time say 1 hour. Models based on statistical methods are easy to use and develop. They adjust their parameters through the difference between their predicted and actual wind speed (Barbosa de Alencar et al., 2017). Artificial intelligence methods on the other hand, make use of Machine learning (ML) models such as Neural Networks (NNs) and Gradient Boosting Machines (GBM) (Chen and Folly, 2018). ML based models make use of non-statistical approaches in knowing the relation between input and output. It makes use of the training ability to learn mappings between output and input (Barbosa de Alencar et al., 2017; Sebitosi and Pillay, 2008). Statistical methods and machine learning methods especially the neural network methods are both suited for

short term prediction. The persistence and numerical methods are known as physical models (Barbosa de Alencar et al., 2017). The physical models make use of statistical distributions on physical quantities such as barometric pressure for forecasting. The hybrid models use a combination of two or more models such as the statistical models and machine learning models for forecasting (Barbosa de Alencar et al., 2017).

An investigation into techniques for predicting wind speed for wind power forecasting is herein reported. Statistical methods along with machine learning methods were used for forecasting wind speed. These methods were evaluated individually and combined in order to quantify their ability to carry out the forecasting. Statistical methods, such as statistical learning, makes use of models such as Auto Regressive (AR), Moving Average (MA) or both (ARMA) and for non stationary data makes use of Auto Regressive Integrated Moving Average models (ARIMA). The conventional statistical learning techniques is the ARIMA (Chen and Folly, 2018). Other statistical learning techniques involving Generalised Additive models (GAMs) can also be used for wind speed forecasting (Sigauke et al., 2018). We present the use of GAMs and ML models for forecasting wind speed using South Africa wind data set. The rest of this chapter presents the problem statement, research focus where we give the research aim and objectives, the research significance and scope, contributions envisaged from the study and a detailed outline of the dissertation.

1.1 Problem Statement

South Africa is an emerging and developing economy, though the most developed and industrialized in the African Continent (Odhiambo, 2009). South Africa stands as having more than half of the electricity generated from the continent being the highest producer and consumer of electricity in Africa, majority of which is being generated from coal (Odhiambo, 2009; Sebitosi and Pillay, 2008). However, this fossil fuel has had its toll on the environment through the emission of carbon gases, making the country the seventh largest emitter of GHG (Sebitosi and Pillay, 2008). There is an urgent need

for the main Electricity company of South Africa (Eskom) to source for alternative distributed energy resources such as renewable energy sources in which, wind energy serves as the most viable. The unpredictability of the wind has been a deterring factor in achieving this goal. However, many developed and developing economies in the world over such as China, USA, India, Denmark among others have successfully embraced and implemented electricity generation from wind energy (Barbosa de Alencar et al., 2017). Wind energy thus contributes a chunk of proportion into their electricity grid in annual increasing rate (Barbosa de Alencar et al., 2017; Sebitosi and Pillay, 2008).

Sebitosi and Pillay (2008) describes the reasons for the reluctance of Eskom in embracing this 21st-century trend. Most power systems are largely unpredictable, hence, the need for load and demand forecasting. Majority of Eskom's forecasting for load balancing has focused largely on Solar power (Buhari et al., 2012). Despite the challenge with the wind for electricity generation, the results from other developed countries are appalling. This suggests that there is a need for more research to be carried out, in the South African context, in increasing the current level of electricity integrated into the grid from wind sources. Methods that ensure accurate wind forecasting with minimum error in short term are crucial to maintain effective load balancing and stability of the grid system (Abhinav et al., 2017). This research thus presents its investigation into three of such methods and compare their performance in forecasting wind speed, a driver of wind power generation using South Africa Wind data set. Also, a combination of forecasts from these methods, with its evaluated level of certainty will be the main deliverable of this research.

1.2 Focus of the Study

The main aim and objectives of this study are given in Sections (1.2.1) and (1.2.2), respectively.

1.2.1 Research Aim

The research was aimed at performing a comparative analysis of neural network algorithm (NNA), stochastic gradient boosting (SGB) algorithm, generalised additive model (GAM) with their combination in short-term forecasting of wind speed. This aim will be achieved by the following objectives:

1.2.2 Research Objectives

The objectives of this study are to:

- make use of Least absolute shrinkage and selector operator (Lasso) for feature / variable selection from the Wind data set,
- use NNA, SGB, GAM and their combination for short-term point forecasting of wind speed,
- evaluate and compare the accuracy of results for the NNA, SGB, GAM along with their combination using three evaluation metrics involving RMSE, MAE and MAPE,
- evaluate individual and combined interval forecasts for uncertainty measures.

1.2.3 Research Questions

The following research questions drive this research

- Given selected variables from the Lasso, how does the point forecasts from the NNA performs in comparison to those of the benchmark models (SGB and GAMs) using the evaluation metrics?
- Which forecasts combination method out performs the other and the individual models?
- How can we quantify the uncertainties in the point and interval forecasting models?

1.3 Significance of the Study

As mentioned in Section (1.1), the unstable nature of the wind has made wind speed forecasting difficult. In order to have electric power generation from wind incorporated into the grid, there is the need to study various methods that can forecast wind speed and minimise forecasting errors. Thus, the importance of making use of methods with little or no error for wind speed forecasting cannot be overemphasized. The use of both Machine Learning, Statistical learning models and a hybrid model formed from their combination gives significance to the research. It presents point forecasts with quantified levels of uncertainty.

1.4 Scope of the Study

This research involve the use of wind meteorology (time series) data set obtained from the Wind Atlas South Africa (WASA) website. The response variable is the wind speed and the predictor variables used in this study are: wind direction (WD), humidity (H), temperature (T), lagged wind speed (LWS), and barometric pressure, depending on the variables selected by the Lasso. The NNA was implemented using Python while the SGB and the GAM were implemented on R respectively. These platforms are usually the work tool in the field of Data Science for extracting actionable insights from data.

1.5 Contributions of the Study

Due to the stochastic nature and unpredictability of the wind, majority of forecasting problems for renewable energy resources have been skewed towards solar energy and solar irradiance forecasting in South Africa (Olaofe, 2013). The study is geared towards adding to the few studies on renewable energy as it pertains to the wind, in the South African context. The research findings from this research serves as a contribution to the knowledge regarding how renewable energy, wind energy, can be used for electric power generation in South Africa by forecasting wind speed.

Majority of the methods used in time series forecasting involves the ARIMA, SARIMA and SARIMAX models (Chen and Folly, 2018). To the best of our knowledge, this research serves as the first time GAM is used for forecasting wind speed using South African data. GAM was used for other types of forecasting but not for the wind forecasting in particular. An investigation into the reliability of GAM for time series forecasting especially for the wind speed forecasting introduces GAM as an alternative model for future research.

Another notable contribution of this research is the choice of models, along with the techniques for combining forecasts. Individual forecasts from the 3 models are combined using both linear and additive quantile regression averaging in order to have an improved forecast with smaller errors. Lastly, We hope to report our findings in reputable journals accredited by the Department of Higher Education and Technology.

1.6 Structure of the Dissertation

The rest of this report is organised as follows: Chapter 2 presents a brief literature review as it relates to wind speed/power forecasting and various techniques that have been used for forecasting. Methodology along with the theoretical formulations of our methods are given in Chapter 3. Empirical results and discussions are presented in Chapter 4. Chapter 5 concludes the mini-dissertation presenting conclusions, summary and recommendations for future researches.

Chapter 2

Literature Review

2.1 Introduction

This chapter summarize what is known on wind speed/power forecasting and various methods and techniques that have been employed in various works. The first three aspects of the discussion will focus on the wind speed/power forecast relations, types of forecast in time scales and forecasting techniques in related works. A brief discussion on the GAM is presented and the gaps in research. The chapter ends with a summary.

2.2 Wind Speed / Power Relations

Unconditional distribution of wind speed and the potential power generation is a classical statistical problem dealing with wind energy resource assessments (Brown et al., 1984). Transition from limiting distribution to dynamic models or linear time series were first seen in the work of Brown et al. (1984). This is useful for wind farm siting and design only and not in the operational management of wind power generation. This is due largely in part to it lacking information on the volatile and conditional dynamics of wind power generation (Pinson et al., 2013). Dynamic modelling were majorly based on physical deterministic frameworks involving complex atmospheric

processes, physical quantities such as pressure, instrumental to understanding wind origination (Landberg, 1999). The models were useful in generating state variables for global weather. The evolution of the dynamics of these state variables result in wind components such as wind speed and wind direction. Recently, wind power generation is modelled as a stochastic process with the wind speed being the most relevant meteorological variable (Pinson et al., 2013).

Wind speed is the main variable upon which the power generated from a wind turbine (WT) depends (Abhinav et al., 2017; Barbosa de Alencar et al., 2017). The WT converts the wind speed (and other variables such as density) into wind power through its power curve shown in Figure 2.1. C_p is the power coefficient denoting the wind power that can be extracted from the rotor blade of the WT. The maximum wind power a WT can generate is given according to Beltz's law $C_p \leq 0.593\%$. The relationship between wind and rotor speed, blade design and the blade tip angle determines the C_p for a particular turbine (Thomas and Cheriyan, 2012; Chen and Folly, 2018). Three regions are worthy of note in Figure 2.1:

- The Linearly C_p progressing region. This region has the optimum constant C_p region where increasing wind speed increases wind power.
- A constant power region, produces limited power even at high wind speed, due to the decrease in C_p, the rotor efficiency.
- Power shut down region, where generation power decreases to zero and the cut-out point of wind speed is reached between 25-26m/s.

The relationship between wind speed and wind power generation conversion is given in equations (2.1) and (2.2), respectively as given in Chen and Folly (2018) & Olaofe (2013).

$$P_{rb}(v) = \frac{1}{2}\rho(t)Av^3 \tag{2.1}$$

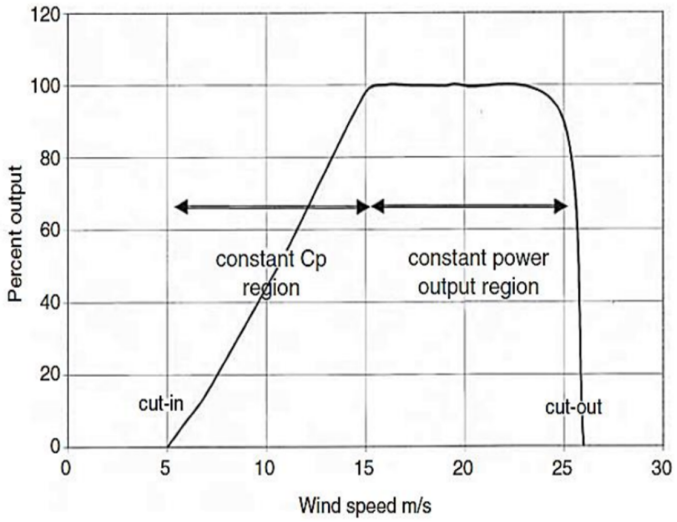

Figure 2.1: Sample Power Curve of a Wind Turbine.
(Source: (Barbosa de Alencar et al., 2017)).

$$P_{rp}(v) = \frac{1}{2}\rho(t)Av^3C_p(v), \tag{2.2}$$

where P_{rb} is the ideal, available power, while P_{rp} is the realizable power (in Watts (W)) the wind turbine generates, $\rho(t)$ is the air density at varying time, which depends on the surrounding temperature and atmospheric pressure, A is the sweep area of blade in (m^3), C_p is the power coefficient.

Air density, temperature and pressure are related as follows:

$$\rho(t) = \Big(\frac{P}{RT}\Big)e^{-\left(\frac{gh}{RT}\right)}, \tag{2.3}$$

where $\rho(t)$ is the air density in varying time measured in kg/m^3, P is the barometric pressure in Pascal (pa), T is the temperature of air in kelvin (K), R is the dry air specific gas constant, 287.058 (j/(KgK)), g is the Earth gravity, 9.8 (m/s^2) and h is the hub altitude above the ground level in (m) (Olaofe, 2013; Chen and Folly, 2018). Lastly, we consider the power law equation. This applies when the hub altitude of the wind turbine is higher than hub height of the data. The dataset we have from the WASA is

at 62m above ground level, in most cases, the height of the WT is higher, hence we use these formulation for the conversion in order to have the correct wind speed,

$$v_1 = (v_0)(\frac{h_1}{h_0})^{\alpha}, \tag{2.4}$$

where v_1 is the wind speed measured at altitude h_1 above the ground, v_0 is the wind speed at height h_0 usually of the wind speed data point. We can use h_1 as 90 and h_0 as 62 with α being the surface roughness as 0.089 (Chen and Folly, 2018). These relations shows how the wind speed at a particular WT hub can be interpolated along with other quantities of interest to estimate the wind power realizable.

2.3 Forecast Time Scales

There are majorly four (4) classifications for the time scale forecasting given in Barbosa de Alencar et al. (2017); Chen and Folly (2018) regarding range and application. The very short/ultra short time forecast ranges from minutes to an hour ahead, and useful in real time grid operation, regulation action and electricity market clearing. There is also the short term forecast which deals with forecasts done in the range of 30 minutes to 1 hour and several hours say 6hours ahead, which is useful in applications such as alternating load decision and economic load dispatch planning (Barbosa de Alencar et al., 2017).

The medium term forecasting is another time scale classification that ranges from several hours, or 6hours to 1 day or 1 week ahead forecast and useful in applications such as reserve requirement decision, operational security in electricity market and generator online/offline decisions. Lastly, the long term forecast is the longest which has ranged in multiple days ahead to 1 year or more ahead forecast and useful in applications such as operation management and optimal operation cost and maintenance planning decisions and for the feasibility study of wind farm design (Chen and Folly, 2018). However, authors such as Giebel and Kariniotakis (2017); Verma et al. (2018) classify forecasts ranging from hours to few days as a short-term forecasting time hori-

zon . Hence, this method is termed Short term because it involves a two day ahead wind speed forecasting. To achieve this, we used lagged variables that lies in the ultra, small and medium term time scales. The response variable was lagged in ultra, small and medium term time scale horizon for the short term wind speed forecasting.

2.4 Related Works on Techniques

This section gave a brief review of the literature surveyed for this mini-dissertation. The methods used by various authors and their research findings based on their results, are reviewed in this section.

The first critical literature to this research is the work of Chen and Folly (2018). With the same wind meteorology dataset used in this research, they made use of three methods to forecast the wind power generation. Their methods involve one statistical approach, the ARIMA model and two machine learning methods, the Artificial Neural Network and the Adaptive Neuro-fuzzy inference systems. They evaluated their results using Root Mean Squared Error (RMSE) and Mean Average Error (MAE). The results showed that the machine learning approach performed better for the ultra short, while the ARIMA model performed better for the short term, 1 hour ahead wind speed and wind power forecasting. However our approach evaluates the methods using the MAE, MAPE and the RMSE for the two days short term wind speed forecasting.

A comparison of four wind forecasting models was done by Barbosa de Alencar et al. (2017) involving the ARIMA, hybrid ARIMA with a neural network, ARIMA hybridized with two neural networks and a neural network with a SONDA data set in all time scale forecast horizons. Using four evaluation metrics, the hybridized ARIMA with two neural network outperformed the rest of the models. In Abhinav et al. (2017), a wavelet based neural network forecast model, applicable to all seasons of the year, was used to predict the short-term wind power. Significant accuracy values using the normalized MAE and normalized RMSE was recorded from the use of less historical data and a less complex model.

A hybrid wind power forecasting model constructed from fitting extracted data by the Boltzmann function, a hybrid neural network comprising of ANN and genetic algorithm was adopted in Lin (2007). Forecasting error was identified from the power output trend and the target curvature change, whose relative difference gave the error and its variation can be seen in time series. A statistical approach to the wind power grid forecasting is discussed in Ying et al. (2012), through the use of wind scale, wind forecast modeling technology involving correlation matrix of output power and forecast accuracy coefficient. RMSE and MAE were used to evaluate the forecast errors. A short-term wind farm power output prediction model using fuzzy modeling derived from raw data of wind farm is presented in Zhu et al. (2012). This model was validated using the RMSE of the train set and the test set. The fuzzy model outperformed the NN model and was also able to provide interpretable structure which reveals rules for the qualitative description of the prediction system.

An investigation comprising of three types of back propagation neural network variants, Lavenberg Marquardt, Scaled conjugate gradient (SCG) and Bayesian regularization, for a feed forward multilayer perceptron was carried out in Baghirli (2015). Using the statistical metrics of MAPE, SCG was found to outperform the rest. The work of Mbuvha (2017) on the use of Bayesian Regularization Back propagation algorithm to short term wind power forecasting was seen as a viable technique for reducing model overfitting.

An approach which provides additional information on the variability and uncertainty of forecasts values is seen in Liu et al. (2015). Their method involve the use of forecast combination based approach to generating prediction intervals (PIs). These PIs leverage the development in point forecasting and less reliance on high-quality expert forecast in probabilistic load forecasting on sister forecasts. The authors made use of Quantile regression averaging (QRA) method for combining point forecasts from sister forecasts, generated from regression models on publicly available data set from

Global Energy Forecasting Competition 2014 (GEFC0m2014). Pinball loss function and Wrinkler score were used to evaluate the performance of this model for a day ahead forecast. According to these evaluation metrics, the methodology generates better PIs than did the benchmark Vanilla methods (Liu et al., 2015).

Combining probabilistic load forecasts using a constrained quantile regression averaging (CQRA) method, an ensemble, formulated as a linear programming (LP) problem, each corresponding to a quantile, that minimises a pinball loss is seen in Wang, Wang, Qu and Liu (2018). This study was conducted using publicly available datasets from ISO NE and CER in Ireland. The results showed that the ensemble method performed better in improving the performance in terms of the pinball loss compared to the individual models. The work of Nowotarski and Weron (2015) investigates interval or density forecast on forecasting electricity spot prices. They made use of QRA for constructing PIs on data downloaded from GDF Suez website. The QRA method outperformed the best of the individual 12 models in terms of the PI coverage percentage and PI width.

The need to quantify the uncertainty and risk associated with point forecasts through probabilistic forecasting drives the work of Abuella and Chowdhury (2017). Using an ensemble learning tool, the random forest for combining individual models and hourly-ahead combined point forecasts were obtained. This was used for obtaining the ensemble based probabilistic solar power forecasts. The comparison carried out on an Australian 1 year data showed that the ensemble based and analog ensemble based probabilistic forecast have similar accuracy using the pinball loss function. Using Extreme learning machine, optimised with a two step symmetric weighted objective function and particle swarm optimization, a deterministic forecast with a quantifiable prediction uncertainty was seen in Sun et al. (2017). This was carried out on benchmark datasets and real world byproduct gas datasets. The results using Prediction Interval Coverage Probability (PICP), Prediction Interval Normalised Average Width (PINAW)

and Prediction Interval Normalised Average Deviation (PINAD) showed high quality PIs constructed for byproduct gas forecasting application.

The challenge of resolving intermittency and uncertainty in renewable energy especially wind power forecasting was the sole aim of the work of Shen et al. (2018). These authors achieved this aim by quantifying the potential uncertainties of wind power via constructing prediction intervals and prediction models using Wavelet Neural Network (WNN) optimized by the Evolutionary knowledge Multi-objective Artificial Bee Colony (EKMOABC) algorthm on an Alberta interconnected electric system 2015 wind power data. The prediction interval constructed was evaluated using PICP and PI covered-normalized average width (PICAW). Indices such as Coverage width-based criterion (CWC) and PI multi-objective criterion (PIMOC) based on certain relations between, the PINAW, PICAW and the PICP were also used to evaluate their models, showing the PI constructed from PIMOC as more reliable. The EKMOABC optimised WNN is the proposed method in this reference and was contrasted with WNN optimised with multi objective particle swarm optimization (MOPSO) and non-dominated sorting genetic algorithm II (NSGAII), showing the wind power PI forecasting by EKMOABC -WNN with higher accuracy and reliability (Shen et al., 2018).

To assess the predictability of wind information needed for evaluation of wind power generation projects, an investigation into wind speed prediction using mathematical models was reported in Ferreira et al. (2019). The mathematical models used were the Holt Winters, ANN, and the Hybrid time series models on SONDA and SEINFRA/CE data. Using the MAE and the RMSE for model evaluation, the hybrid model presented lesser errors amongst the other models. For extensive review on probabilistic forecasting of wind power generation and wind energy forecasting management and operational challenges, see Zhang et al. (2014); Giebel and Kariniotakis (2017); Pinson et al. (2013) respectively.

Given the approaches used in existing literature for various kinds of forecasting,

it is evident that the methods employed in this study had been used on different datasets around the world and evaluated, mostly using the same metrics employed in this study. This approach involves a comparative use of three of such methods for point forecasting and two other methods for interval forecasting on a dataset created and curated in South Africa. This was done to ascertain the viability of these methods for point forecasting of wind speed and interval forecasting with a step further into interval combination in order to have a forecast with small error values, whose uncertainties are also quantified. Considering the foregoing reviews on techniques, one begin to wonder if GAM is not been used for forecasting. A review on literature that has GAM used as a forecasting model is hereby presented in the next section.

2.5 The Generalized Additive Model

Linear regression models, a class of additive models are normally used with GAMs (Jones and Wrigley, 1995). GAMs are suitable for exploring the dataset and visualizing the relationship between the dependent and independent variables (Liu, 2008). Goude et al. (2014) use GAMs in modelling electricity demand for the French distribution network at both short and medium term time scales for more than 2200 substations. The relationship between the load and the explanatory variables was estimated by their model. The proposed model is given in equations (2.5) and (2.6), respectively.

$$y_t = \sum_{i=1}^{p} f_i(x_{it}) + \varepsilon_t, \quad t = 1, \cdots, n, \tag{2.5}$$

$$y_t = f_1(x_{1t}) f_2(x_{2t}) + \cdots + f_p(x_{pt}) + \varepsilon_t, \quad t = 1, \cdots, n, \tag{2.6}$$

where ϵ_t is the error term, y_t is a univariate response variable, x_{pt} are the covariates that drive f_i, the smooth functions. Non linear functions are meant to be smoothed that they can be relatively well estimated by penalized regression on a spline basis (Goude et al., 2014). Effects that drove the French hourly load consumption was modelled using GAM and compared with the operational one in Pierrot and Goude (2011). The

effect of different variables was estimated with GAM given in equation (2.7).

$$L_t = f_1(L_{t-24})f_2(L_{t-168}) + f_3(T_t) + f_4(\mu(T_{t-24}, T_{t-48})) + f_5(cc) + f_6(posit) + C + \varepsilon_t, \tag{2.7}$$

where L_t, L_{t-24} and L_{t-168} are the t instant load to forecast, one day lagged load and the one week lagged load respectively, T_{t-24}, T_{t-48} are the t instant temperature, one day lagged temperature and the two day lagged temperature, μ is the mean, cc represents the cloud cover, $posit$ is the position of the day through the year (September to August) C is the intercept and ϵ_t is the residual error.

GAM is good for interpretability, regularization, automation and flexibility (Liu, 2008). It finds a balance between the biased and yet interpretable algorithm, linear models and extremely flexible grey box learning algorithms composing the movement, seasonality and climate change variables. GAM was fitted on weekly load demand in Cho et al. (2013). The existence of functional form trend between two variables and their shape whether linear or non linear, should it exist was examined using GAMs in the work of Shadish et al. (2014).

The use of GAMs in forecasting wind speed is very rare, there is a dearth of GAMs to wind power forecasting especially, in literature. Hence, this research investigates the use of GAM as an alternative to the ARIMA models and its variants that have been the most used approach in wind power forecasting in the existing literature.

2.6 Identified Gap

This research investigates methods for wind speed forecasting with minimal error values along with the combination of these methods. The major gap in research identified from the foregoing literature review presents Generalised additive model and Stochastic gradient boosting (a Gradient boosting variant) as alternative methods for enhanced performance when combined with the Neural network algorithm using Additive quantile regression averaging method for forecasts combination. The worthiness

of this intervention was presented using these models. Evaluations of the approach by the error metrics are given in the results section in chapter 4 of this mini-dissertation.

2.7 Summary of Chapter

This chapter began with a brief introduction of what it would entail, in which the relationship between wind speed and the wind power along with other parameters of note was discussed. The time scale classification horizons and their applications was also presented. A brief perusal on techniques that have been used for various forms of forecasting and forecast combinations was featured. The chapter presented the theoretical formulation for GAMs that have been used for various works on the GAMs and identified the gap filled by the research. The Literature review concludes by summarising the chapter accordingly.

Chapter 3

Methodology

3.1 Introduction

This chapter gave the methods that was employed in carrying out the research. Three crucial aspects are laid out in order, to include the variable selection and data pre-processing, the Artificial neural network (ANN), and the benchmark models which are Generalized additive models (GAM) and the Stochastic gradient boosting (SGB). The report presents the ANN, GAM, SGB and the model for variable selection in the light of their theoretical and mathematical formulations for their working principles. Also, a brief discussion on forecast combination is also presented. This chapter concludes with model evaluation metrics in terms of point forecasts, prediction intervals evaluation metrics and prediction interval combination towards quantifying the uncertainties in the predictions made by the models,along with a summary of the overall methodology of the research.

3.2 Variable Selection and Data Pre-processing

Various data pre-processing methods were employed before feeding the data into the models. For a time series short term forecasting problem, the following data pre-processing methods are used: Data cleaning, data sampling, stationarity check, data

transformation, data de-trending, data de-seasonalization and data normalization (Baghirli, 2015). The main pre-processing process employed is the data cleaning. Missing values were eradicated so that data could be prepared for variable selection needed for implementing these models. Data was also checked to see that it exhibit properties needed for producing good forecasts. Not all the wind meteorological dataset available on WASA website could be used for forecasting. Data exhibiting needed properties such as trends, seasonality and having residuals were adjudged to yield good forecasts than those which do not have these properties from their plots. Selecting relevant features from a dataset needed for the forecasting task is the process of variable selection. Not all variables are necessary predictors, among such predictors, there is a need for selecting the best predictor variables. There are many methods for variable selection. Least absolute shrinkage and selection operator (Lasso) for variable selection was used in this study (Lim and Hastie, 2015).

3.2.1 The Least Absolute Shrinkage and Selection Operator, LASSO

Suppose there are N pair of predictor variables x and response variables y, i.e. $\{x_i, y_i\}_{i=1}^{N}$. The aim is to give an approximate value for the response variable y_i from the predictors linearly combined, as it is in linear regression such as given in equation (3.1)

$$\eta(x_i) = \beta_0 + \sum_{j=1}^{p} x_{i_j}\beta_j. \tag{3.1}$$

The vector $(\beta = (\beta_0, \beta_1, \cdots, \beta_p) \in \mathcal{R}^p$ of regression weights which parameterizes the model along with an intercept term $\beta_0 \in \mathcal{R}$. An estimation for (β) using least-squares method is based on minimizing squared error loss using equation (3.2)

$$\hat{\beta} = \min_{\beta} \left\{ \frac{1}{2N} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 \right\}. \tag{3.2}$$

Due to the problems of interpretability and prediction accuracy associated with least squares method, the need for the Lasso thus emerged (Hastie et al., 2015). Lasso performs better in prediction accuracy measured in terms of the mean squared error by shrinking regression coefficient values or setting some of them to zero thereby introduces bias and reduces the variance of predicted values. Lasso also helps with an improved interpretability by identifying smaller subsets of predictors with stronger effect from a large set of predictors. The Lasso provides an automatic way for variable selection in linear regression problems because it solves a convex, quadratic program with convex constraint optimization problem. It works by combining the least square loss of equation (3.2) and the $\ell_1 - constraint$ or bounded on sum of the absolute values of the coefficient (Hastie et al., 2015).

The Lasso solves $\hat{\beta}$ using the optimization problem of (3.2) subject to $\sum_{j=1}^{p}|\beta_j| \leq t$ which is the $\ell_1 - norm$ constraint written as $||\beta||_1 \leq t$. A comparable method before the Lasso is the ridge regression which solves (3.2) subject to $\sum_{j=1}^{p}\beta_j^2 \leq t^2$. The best form for estimating the Lasso problem is by having the predictors x_i standardized, making all the columns centered such that $(\frac{1}{N}\sum_{i=1}^{N}x_{ij} = 0)$ with unit variance $(\frac{i}{N}\sum_{i=1}^{N}x_{ij}^2 = 1)$ and the response values y_i centered such that $(\frac{1}{N}\sum_{i=1}^{N}y_i = 0)$ with an omitted intercept term β_0 produces the optimal solution in which equation (3.2) becomes equation (3.3)

$$\min_{\beta \in \mathcal{R}^p}\left\{\frac{1}{2N}\sum_{i=1}^{N}(y_i - \sum_{j=1}^{p}x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{p}|\beta_j|\right\} \tag{3.3}$$

Expression (3.3) is known as the Lagrangian form which produces an effective, convenient and simple computational algorithm for the numerical computation of the Lasso using coordinate descent procedure among other methods (Hastie et al., 2015).

The complexity of the Lasso is controlled by the value of its constraint, t. Smaller

values of t produces sparse and easily interpretable models less closely fitted to the training data whereas larger values of t free up more parameters, more closely fitted to the data. These two extremes of t hamper the generalization ability of the Lasso model by recording a large error value in the prediction error test set. A trade off between overfitting and sparsity is desirable for the Lasso generalization ability. This is carried out by the cross-validation procedure which strikes a balance in the value of t that gives the accurate model for predicting individual test data set. Hastie et al. (2015) give more details concerning the theoretical framework of the Lasso whereas, Plan and Vershynin (2016) explain how the Lasso could be used for non-linear observations as we have in this research.

3.2.2 Generalized Cross Validation

Cross-validation (CV) entails splitting the dataset into training or calibration and test or validation subset in order to evaluate models. Mpfumali (2019) identified some types of a CV such as; a hold out method or the 2-fold CV, repeated random data sampling, k-fold CV, leave-one-out CV which extends to the Generalised CV (Friedman et al., 2001). The model is split into 75/25% or 80/20% for the training and testing subset respectively and used for CV. One of the packages in R that implements cross validation is the "mgcv" developed by Hastie and Tibshirani (1990); Wood (2001). Generalised cross validation measures the goodness of fit considering the model complexity and residual error. GCV criterion is expressed as in equation (3.4) as given by Craven and Wahba (1978).

$$GCV(M) = \frac{\frac{i}{n}\sum_{t=1}^{n}[y_t - \hat{f}_M(X_t)]^2}{[]1 - \frac{G(M)}{n}]^2} \tag{3.4}$$

where n is the number of observation $G(M)$ estimates the cost penalty measure of a model with M basis function. Lack of fit on a M basis function model $\hat{f}_M(X_t)$ is determined by numerator while the denominator determines the penalty for model

complexity $G(M)$. The lower the GCV criterion value the preferred the model.

Mathematical underpinnings of the functions executed by the implementation platform is given in these last two sections. The Lasso, GCV and the rest of the theoretical explanations of the models are given as we proceed in this chapter on methods.

3.3 Artificial Neural Networks

Artificial Neural networks (ANNs) are mathematical means of computation inspired by the field of Biology, especially the nervous system. It simulates the biological neuron in the human brain (Lin, 2007; Zhang et al., 1998). ANNs can learn, be taught and generalise to new experiences. It is characterized as robust, parallelizable and self-co0rdinating and is popularly useful in most areas. The basic element of the ANNs is the neuron shown in Figure 3.1. Its architecture is built of many neurons organised in 3 major layers of input, hidden and output layers respectively. A simple ANN architecture is shown in Figure 3.2.

Mathematically, ANNs can be formulated as:

$$\begin{aligned} Y_k &= \varphi(\Sigma(x_1w_1 + x_2w_2 + x_3w_3 + \cdots + x_kw_k) + \beta_k) \\ Y_k &= \varphi(u_k - \theta_k) \\ u_k &= \sum_{i=1}^{k} x_i w_i \\ \beta_k &= -\theta_k \end{aligned} \tag{3.5}$$

From equation (3.5), a composite formulation of Figure 3.1 can be inferred. While the first two lines gives the overall formulation for ANN, the last two lines explains the unit terms. The symbol φ denotes the activation or transfer function, which can be sigmoid, threshold, piece-wise, identity function, β_k is the bias and θ_k is the threshold, u_k is the weighted sum, w_i is the weight at neuron i and x_i are data points.

The most commonly used ANN for forecasting is the feed forward multi-layer net-

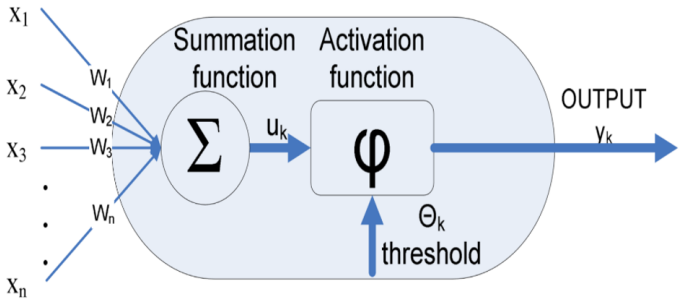

Figure 3.1: The schematic Neuron representation
(Source: Lin (2007)).

work because it does not allow for loops in the network. The ANN learns, learning is defined as the modifiable components of a system, to which applied change(s) is/are responsible for the success or failure of, and can improve performance of the system (Schmidhuber, 2015). The learning process in ANNs thus involve tuning parameters such as weights and thresholds. Learning involves basically, two types which are supervised learning in which case a dataset and the target output is provided to the ANN, while the unsupervised learning is a self organised learning with no input target data for classifying patterns (Schmidhuber, 2015).

The weights in the neurons (i, j) in the network at a given layer k is modified thus:

$$\Delta W_{ij}^{k}(t+1) = \eta \Delta W_{ij}^{k}(t) + \alpha \Delta W_{ij}^{k}(t-1), \tag{3.6}$$

where α is the momentum term which tracks the effects of the previous training iteration on the current one, while the η is the learning or convergence rate based on the learning rule such as gradient decent formulated as:

$$\Delta W_{ij}^{k} = \eta \frac{\delta E_k}{W_{ij}^{k}}, \tag{3.7}$$

where η is the learning rate and $\eta \frac{\delta E_k}{W_{ij}^{k}}$ is the derivative of the error gradient w.r.t. weight from neuron j to i at layer k. Other learning rules involves Delta rule and Hebb's rule (Haykin, 1994; Heaton, 2008).

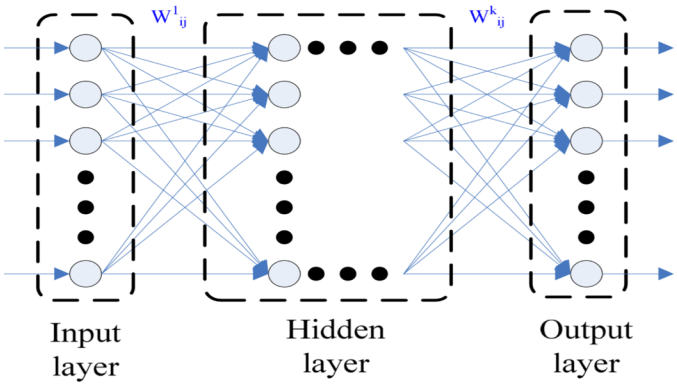

Figure 3.2: Neural Network Architecture

(Source: Lin (2007)).

3.4 Training Algorithm

The training algorithm is the main determinant of the ANN taxonomy and nomenclature (Lim and Hastie, 2015). This phase involves minimizing the error by updating the weight until the desired output is reached or a terminating criterion is satisfied. Training algorithm finds the function that evaluates the ANN error rate using the learning rule. Different types of training algorithms are used for the ANN such as the Back propagation Hunga (2018), Genetic algorithm Lin (2007), Gene expression programming Haykin (1994), Simulated annealing among others (Schmidhuber, 2015). The training algorithm used for this research is the Back propagation algorithm (with Bayesian regularised), used for training the multilayer perceptron neural networks. This algorithm is chosen because it generalises well with small and messy data sets and therefore reduces over fitting (Chen and Folly, 2018). The algorithm and mathematical formulations of its steps along with its flowchart are presented in this section.

3.4.1 Back Propagation Algorithm

The Back Propagation Neural Network (BPNN) is implemented in two phases which are the forward phase, the layer by layer introduction of the input pattern without

changing the weights and the backward phase which involves a layer by layer back propagating of error signals from comparing the target output with the network output by changing weights for each layer. BP is done in the following steps:

- Selection of the paired input-target vectors from the training data set; and application to the ANN input nodes.
- Process the network output.
- Compute the errors between the network output and the target.
- Minimise the errors by adjusting the neuron weights connection.
- Iterate 1 to 4 for all the paired input-target vectors in the training set until a reasonable error is reached for the entire set or terminating criterion is satisfied.

3.4.2 Mathematical Formulations for BP

In equation (3.8) w_{ij} interconnects the I^{th} output O_i from neuron i to its j^{th} input neuron, if the neuron at layer k is not an input neuron then its state can be formulated mathematically as:

$$O_k = f(\sum_i w_{ij} O_i), \tag{3.8}$$

where $f(x) = 1/(1 + e^{-x})$ is the sigmoid activation function, the summation is done in all neurons in all the layers with a target t the output of the neuron can be specified as:

$$E_k = \frac{1}{2(t_k - O_k)^2}, \tag{3.9}$$

where E is the error and k is the neuron on the output layer. Using the gradient error, the gradient descent algorithm modifies the weights thus:

$$\Delta W_{ij}^k = -(\frac{\delta E}{\delta O_j}) x (\frac{\delta O_j}{\delta W_{ij}^k}) = -(\frac{\delta E}{\delta W_{ij}^k}), \tag{3.10}$$

where $\delta_j = -(\frac{\delta E}{\delta O_j})$ is the error signal so that we have:

$$\Delta W_{ij}^k = \epsilon \delta_j O_i, \tag{3.11}$$

where ϵ is the learning rate and δ_k depends on whether neuron j is an output layer in which case we have:

$$\Delta_j = O_j(1 - O_j) \sum_k \delta w_{jk}, \tag{3.12}$$

in order to mitigate premature convergence, the momentum rate α is included so that we have:

$$\Delta W_{ij}^k(n+1) = \epsilon \delta_j O_i + \alpha \Delta W_{ij}^k(n), \tag{3.13}$$

n represents the number of iterations or epochs. The flowchart of a BP training algorithm for a NN is as shown in Figure 3.3.

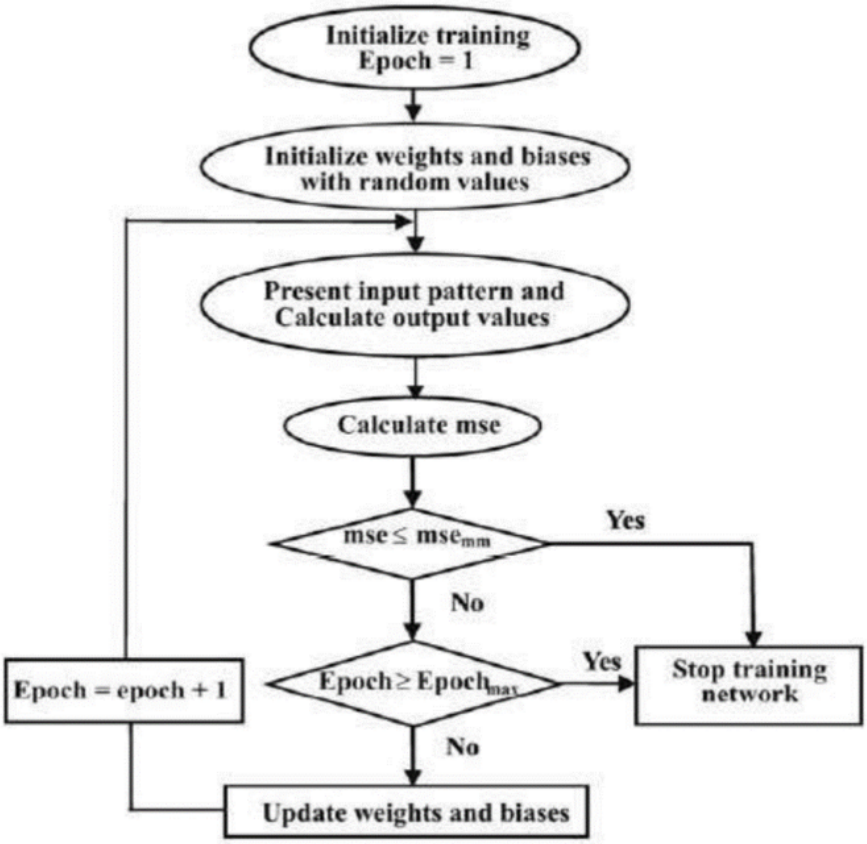

Figure 3.3: Flowchart for Back Propagation
(Source: Hunga (2018)).

3.4.3 Generalization

The ANNs learn in the training phase however, it should be able to generalize well when different sets of input data is used by producing output similar to the target. Hence, it means that the NN does not overfit its training data. There are variants of the BPNN among which Bayesian Regularization is best suited for generalization because it does not overfit its training data (Baghirli, 2015). The Bayesian Neural Network (BNN) as expressed in Mbuvha (2017) is the method towards which our ANN was

adapted.

3.5 Benchmark Models

3.5.1 Generalized Additive Models (GAMs)

GAMs are models which allow for an additive relationship between its dependent and independent variables (Hastie and Tibshirani, 1990). GAM is unlike the Generalized linear Model (GLM) which allows for a functional relationship between response and predictor variables, and it is more flexible to model. GAMs make use of non-linear form and smooth function of predictor variables in its modelling that is applicable in different forms. A GAM can be formulated in its simplest form as (3.14):

$$g(\mu_t) = \mathbb{A}\theta + \sum_{j=1}^{p} f_j(x_{jt}) + \varepsilon_t, \quad Y_t \sim EF(u_t, \phi) \quad t = 1, \cdots, n, \tag{3.14}$$

where Y_t represents the independent univariate response variable from an exponential family distribution having mean u_t, scale parameter ϕ, g represents the smooth monotonic link function, $\mathbb{A}$ is a design matrix, θ represents an unknown parameter vector, f_j is an unknown smooth function of the predictor variable x_j that may have a vector value, ε_t is an independent identical distribution random error (Hastie and Tibshirani, 1990; Pya and Wood, 2016).

3.5.2 Stochastic Gradient Boosting

The Gradient Boosting Machine or Gradient TreeBoost were the terms previously used for Gradient Boosting when it was implemented newly by Friedman (Friedman, 2002). Gradient Boosting is a Machine learning model used for classification and regression problems (Mpfumali et al., 2019). It stage-wisely builds weak predictive models generalised by optimization of arbitrary differentiateable function. The Statistical framework of gradient boosting describes it as an optimization problem which minimizes the loss in a model by a stage-wise addition of weak learners to the models using a gradient descent procedure (Friedman, 2002; Mpfumali et al., 2019). Gradient

descent traditionally minimizes set of parameters such as coefficient of regressors or ANN weights through loss or error calculation and weight update (Friedman, 2001). The weak learners are organised in substructures or decision trees that replaces the parameters. Parameterized tree is added to the model, thereby reducing the error and the residual losses using the parameters of the trees following the direction of the gradient (Friedman, 2001). The gradients spots the error in the weak learners. The major drawback to the gradient boosting is that it is a greedy algorithm that can easily over fit training data (Friedman, 2001; Hastie et al., 2005). One of the variants of gradient boosting is the stochastic gradient boosting (SGB) formed by taking a random sample of the training data set without replacement (Friedman, 2002, 2001). Its general formulation is given in equation (3.15).

Stochastic Gradient Boosting Algorithm

$$F_0(x) = argmin_\gamma \sum_{i=1}^{N} \phi(y_i \gamma)$$

for $m = 1, 2, \cdots, M$ **do**

$$\{\pi(i)\}_1^N = rand.perm\{i\}_1^N$$

$$\bar{y}_{\pi(i)m} = -\left[\frac{\delta\phi(y_{\pi(i)}, F(x_{\pi(i)}))}{\delta F(x_{\pi)(i)})}\right]_{F_{(x)=Fn-1(x)}}, i = 1, \cdots \bar{N}$$

$$\{R_{lm}\}_1^L = L - terminalnodetree(\{\bar{y}_{\pi(i)m}, X_{\pi(i)}\}_1^{\bar{N}})$$

$$\gamma_{lm} = argmin_\gamma \sum_{x\pi(i)\in R_{lm}} \phi(y_{\pi(i)}, F_{n-1}(x_{\pi(i)}) + \gamma)$$

$$F_n(x) = F_{n-1}(x) + V.\gamma_{lm} 1(x \in R_{lm})$$

end for

Figure 3.4: Stochastic gradient boosting algorithm (Friedman, 2002)

Stochastic gradient boosting can be expressed algorithmically as shown in the Algorithm in Figure 3.4.

$$F(x) = \sum_{m=1}^{M} \beta_m h(x; \gamma_m) \tag{3.15}$$

where $h(x; \gamma_m) \in \mathcal{R}$ are functions of x with characteristics of γ_m and β_m expansion parameters that limits over fitting (Friedman, 2001; Hastie et al., 2005).

The use of SGB in forecasting wind speed/wind power, like the GAMs, is also very rare. Hence, this research investigates the use of SGB specifically as a variant of the gradient boosting models that have been mostly used in forecasting problems in the literature. This will enhance the provision of knowledge to the forecasting and machine learning community about the problems SGB is capable of solving.

3.6 Forecasts Combination

Wind speed forecasting presented in this research report requires that the performance of individual forecasts be improved. The overarching forecast which is the main deliverable of the research work will be an improved forecast obtained from forecasts using the ANN, SGB and the GAM. The combined forecast has a superior quality in terms of accuracy and reduced error when compared to its components. One viable way of achieving this is through the forecast combination first introduced by Bates and Granger (1969). Forecast combination is an approach for ensuring increased forecast accuracy and error variability. The combination of forecasts from different methods reduces the error margin. This is useful in uncertain and unpredictable situations as wind prediction. The theoretical justification for forecast combination involves testing and averaging individual forecast models according to their probabilities when the problem is viewed from the perspectives of Bayesian Model averaging (BMA) (Hoeting et al., 1999).

Forecast combination is important because it provides a means of compensating for the drawbacks in component forecasts. It helps to avoid the risk involved in using one forecast and also provides a way of benefiting from various interactions among

component forecasts (Wang, Wang, Qu and Liu, 2018). The motivation for forecasts combination lies in the following: possibility for insufficiency of forecasts from individual component models, forecasts from the components are from different and complementary perspectives, considerable grounds are covered from component forecasts which gives a complete picture of the forecasts when combined and improved forecasts accuracy due to the effects of model uncertainties, structural breaks, and model mis-specification being forestalled by forecasts combination (Wang, Wang, Qu and Liu, 2018).

Depending on the forecasting intervals, the loss function is an important parameter in forecast combination. It is the main performance evaluation criterion and the sole ingredient in forecast combination formulae (Sigauke, 2017). Forecasts combination schemes, involving adaptive forecast combination scheme and regression based combination make use of loss function for combining forecasts (Sigauke, 2017). A regression based approach is the quantile regression averaging (QRA) used for combining forecasts and computing prediction intervals (Sigauke, 2017). Convex combination of models make use of various algorithms based on minimizing losses from pinball, absolute error, percentage error and square losses. The forecast combination method presented in this study is according to Sigauke (2017); Mpfumali et al. (2019) as given in equation (3.16)

$$y_{i,\tau} = \sum_{k=1}^{K} w_{i,k,\tau}\hat{y}_{i,k} + \varepsilon_{i,\tau} \quad \tau \in (0,1) \quad i = 1, \cdots, m, \tag{3.16}$$

where $y_{i,\tau}$ is the combined forecast for the wind speed, k is the number of forecasting methods used to predict the next observation i of $y_{i,\tau}$, m is the total number of point forecast models, $w_{i,k,\tau}$ is the weight assigned to each forecast $\hat{y}_{i,k}$ and $\varepsilon_{i,\tau}$ is an error term having to do with the loss function.

3.7 Quantile Regression Averaging

Simple averaging methods of forecast combination from individual point forecasts use equal weight and had proven to be a viable means of improving forecast accuracy (Hoeting et al., 1999). However, the inability of the interval forecasts resulting from point forecast of simple averaging combined models to ensure a nominal coverage rate requires the application of unequal weights in the forecast combination (Nowotarski and Weron, 2015). The estimation of interval forecast follows a complex process and involves applying weights based on the quantiles (Taieb et al., 2016). Quantile Regression (QR) thus processes and applies quantile based weight to individual point forecasts from a number of forecasting models to give interval forecast with nominal coverage rate that can be used to ascertain the uncertainties in the combined forecast as well as in the individual forecasting models. In this QR setting, the individual point forecast and the combined interval forecasts models the predictor variables and the response variable respectively. In order to combine point forecasts from individual models, we present the use of Quantile regression averaging (QRA) methods to generate interval forecast for the forecasting process. Along with the combined interval forecasts, we also generate interval forecast from individual forecast models by using QRA to assess the uncertainties in each constituent model comparable to that of the combined model. The QR problem can be expressed as (3.17) as given in Nowotarski and Weron (2015)

$$Q_y(q|X_i) = X_i\beta_q, \tag{3.17}$$

where $Q_y(q|.)$ stands as the conditional qth quantile of the actual wind speed (y_i), X_i are the independent variables treated as regressors and β_q is a vector of quantile q

parameters estimated by minimizing a loss function for a qth quantile using (3.18)

$$\begin{aligned}\min_{\beta_q}\Bigg\{\sum_{(ny_i\geq X_i\beta_q)} q|y_i - X_i\beta_q| + \sum_{(ny_i<X_i\beta_q)}(1-q)|y_i - X_i\beta_q|\Bigg\} \\ = \min_{\beta_q}\Bigg\{\sum_i (q - 1_{y_i<X_i\beta_q})(y_i - X_i\beta_q)\Bigg\},\end{aligned} \tag{3.18}$$

where y_i is the actual wind speed and $X_i = [1, \hat{y}_1, \cdots \hat{y}_m.i]$ is the vector of point forecasts from m individual forecasting models.

3.7.1 Linear Quantile Regression Averaging

Linear qauntile regression averaging is the general form for QR averaging as defined in section (3.6). It contains a model involving the response variable and the independent variables of the combined forecast. Let $y_{i\tau}$ be the 2day wind speed, with M total number of methods for predicting the next observation, $y_{i+1}, y_{i+2\cdots}$ of $y_{i+M\tau}$, using $M = 1, \cdots M$ methods, the combined forecast is expressed as (3.19) as given in Sigauke et al. (2018).

$$\hat{y}_{i\tau}^{LQRA} = \beta_0 + \sum_{k=1}^{K}\beta_k\hat{y}_{ik} + \epsilon_{i\tau}, \tag{3.19}$$

where y_{ik} represent forecast from kth method, and $\hat{y}_{i\tau}^{LQRA}$ is the combined forecast while $\epsilon_{i\tau}$ is the error term we seek to minimize equation (3.20)

$$\arg\min_{\beta}\left(\sum_{i=1}^{N}\rho\tau(\hat{y}_i^{LQRA} - \beta_0 - \sum_{k=1}^{K}\beta_i\hat{y}_{ik})\right) \tag{3.20}$$

We express equation (3.20) in reduced matrix form (Maciejowska et al., 2016) analogous to equation (3.18) as seen in equation (3.21)

$$\arg\min_{\beta\in\mathbb{A}^\rho}\left[\sum_{i:\hat{y}_i^{LQRA}>x_i^\tau\beta}\tau(\hat{y}_i^{LQRA} - x_i^\tau\beta)\sum_{i:\hat{y}_i^{LQRA}<x_i^\tau\beta}(1-\tau)(\hat{y}_i^{LQRA} - x_i^\tau\beta)\right]. \tag{3.21}$$

3.7.2 Additive Quantile Regression Averaging

A hybrid regression model based on Additive Quantile Regression (AQR) model consists of GAM and QR. Its first use was seen in Gaillard et al. (2016), extended in Fasiolo et al. (2017), used in Sigauke et al. (2018) and given in equation (3.22).

$$y_{i\tau} = \sum_{k=1}^{p} s_{k,\tau}(x_{ik}) + \epsilon_{ik} \quad \tau \in (0,1), \tag{3.22}$$

where x_{ik} are p covariates terms, from $x_{i1}, x_{i2\ldots}, x_{ip}$, $S_{k,\tau}$ are smooth functions while $\epsilon_{i\tau}$ are the error terms. Smooth function (s) can be expressed as equation (3.23)

$$s_k(x) = \sum_{q=1}^{j} \beta_{q,k} bs_{qk}(x_{ik}) \tag{3.23}$$

where bs_k represents the k^{th} basis function with j dimension and β_k is the k^{th} parameter. Parameter of equation (3.22) are estimated by minimizing the expression in equation (3.24):

$$Q_{y|x}(\tau) = \sum_{i=1}^{N} \rho\tau \left(y_{i,\tau} - \sum_{k=1}^{p} s_{k,\tau}(x_{ik}) \right), \tag{3.24}$$

where $\rho\tau$ is the pinball loss function, however the loss function it minimises is that of equation (3.18).

3.8 Forecast Evaluation Metrics

This section discussed the three evaluation metrics for the individual and combined point forecasts. Mathematical formulation for these evaluation metrics are presented herein. Three main accuracy metrics for evaluation of the point forecasts made from the prediction models and their combination (ANN, SGB, GAM and QRA) are the mean absolute error, (MAE), mean absolute percentage error (MAPE) and the root mean squared error (RMSE). They are as formulated in equations (3.25 through 3.27),

where m is the number of observations in the test data set, y_t is the estimated values of the response variables, and ε_t is the residual of the i^{th} observation given as $\varepsilon_t = y_t - \hat{y}_t$

$$\text{MAE} = \frac{1}{m}\sum_{i=1}^{m}|\varepsilon_t| \tag{3.25}$$

$$\text{MAPE} = \frac{100}{m}\sum_{i=1}^{m}|\frac{\varepsilon_t}{\hat{y}_t}| \tag{3.26}$$

$$\text{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}|\varepsilon_t^2|} \tag{3.27}$$

3.9 Prediction Intervals Formulation and Evaluation Metrics

In order to ascertain the uncertainty in point forecasts, it is necessary to provide the prediction intervals (PIs) so as to quantify these uncertainties (Mpfumali et al., 2019). This section gives the formulation of the PIs and various metrics for evaluating the performance of estimated PIs:

3.9.1 Prediction Interval Formulation

Given a set of data containing point forecasts from different models at certain quantiles (say 90, 95 & 99) along with the actual prediction value such that $D = \{(x_i, y_i), i = 1, 2, \cdots m\}$. x_i represents an input vector corresponding to a particular variable in D and y_i represents the actual value. The PI with nominal confidence (PINC) $100(1-\alpha)\%$ for the y_i is given in equation (3.28)

$$Q^{\alpha}(x_i) = \left[L^{\alpha}(x_i), U^{\alpha}(x_i)\right] \tag{3.28}$$

where $Q^{\alpha}(x_i)$ represents the range of PI values within the actual y_i having $L^{\alpha}(x_i)$ and $U^{\alpha}(x_i)$ as its Lower and upper bound estimate (LUBE) values respectively. Thus the

probability that $Q^{\alpha}(x_i)$ lies within y_i is expected to be $100(1-\alpha)\%$ can be expressed in (3.29) (Sun et al., 2017).

$$PINC = Pr(Q^{\alpha}(x_i) \in y_i) = 100(1-\alpha)\% \tag{3.29}$$

3.9.2 Prediction Interval Width (PIW)

The first index for estimating PI is the PIW. It is estimated using Lower and upper bound estimate (LUBE) as seen in (Shen et al., 2018). The difference between the content of $Q^{\alpha}(x_i)$ i.e. the upper limit and the lower limit gives the PIW as expressed in equation (3.30) (Mpfumali et al., 2019).

$$PIW_i = U^{\alpha}(x_i) - L^{\alpha}(x_i) \qquad i = 1, \cdots, N \tag{3.30}$$

3.9.3 Prediction Interval Coverage Probability (PICP)

The PICP is an important index that evaluates the reliability of the formulated PIs. It can be expressed as equation (3.31)

$$PICP = \frac{1}{N}\sum_{i=1}^{N} q_i \tag{3.31}$$

Where N is the number of observations in the data set, q is defined in (3.32):

$$q_i = \begin{pmatrix} 1 & if & y_i \in Q^{\alpha}(x_i) \\ 0 & if & y_i \notin Q^{\alpha}(x_i) \end{pmatrix} \tag{3.32}$$

3.9.4 Prediction Intervals Normalized Average Width (PINAW)

The PINAW is one of the indices for assessing the PI. It gives quantitative width of the PIs and it is described as equation (3.33)

$$PINAW = \frac{1}{NR}\sum_{i=1}^{N} PIW_i \tag{3.33}$$

where $R = y_{max} - y_{min}$ represents the range of the highest and lowest values of the actual y_i.

3.9.5 Prediction interval Normalized Average Deviation (PINAD)

The PINAD is an index for describing the deviation of the PIs from the actual values quantitatively (Shen et al., 2018; Mpfumali et al., 2019). It is expressed as equation (3.34)

$$PINAD = \frac{1}{NR}\sum_{i=1}^{N} dv_i \tag{3.34}$$

dv_i is defined in (3.35)

$$dv_i = \begin{pmatrix} L_i^{\alpha} - y_i & if \quad y_i < L_i^{\alpha} \\ 0 & if \quad L_i^{\alpha} \leq y_i \leq U_i^{\alpha} \\ y_i - U_i^{\alpha} & if \quad y_i > U_i^{\alpha} \end{pmatrix} \tag{3.35}$$

3.9.6 Prediction Interval Covered-Normalized Average Width (PICAW)

The indices described above are only about the PIs covered by the actual values. The PICAW estimate involves the PIs not covered by the actual values, since this affects the PI widths negatively, it is therefore a new evaluation index for the width defined as equation (3.36) (Shen et al., 2018).

$$PICAW = \frac{1}{R}\left(\frac{1}{N_{p+}}\sum_{i=1}^{N_{p+}} PIW_i + \lambda \frac{1}{N_{p-}}\sum_{i=1}^{N_{p-}} PIW_i\right) \tag{3.36}$$

Where N_{p+}, N_{p-} respectively represents the number of the actual values that the PIs does or does not cover respectively. A control parameter that widens the difference between the PIs and the actual values is $\lambda > 1$ else if $\lambda = 1$, PICAW becomes PINAW. The PICAW gives more accurate PI construction evaluation when actual values are farther off from the PIs.

3.10 Combined Prediction Intervals

Combining point forecasts have the tendency to improve accuracy as shown in research (Nowotarski and Weron, 2015; Wang, Zhang, Tan, Hong, Kirschen and Kang,

2018). The combined point forecast can also be improved in terms of its prediction interval by combining its prediction limits. A comparison of the combined PI and the PIs from constituent models is presented in this section. Lower and upper prediction limits from N number of forecasting models can be represented as $[L_n, U_n]n = 1, \cdots, N$ to denote the resulting $100(1-\alpha)\%$ PI, the combined $100(1-\alpha)\%$ PI can also denoted as $[L_C, U_C]$ from PI combination method C. Two prediction interval combination methods (C) which are the Simple Averaging and the Median Method combining PIs was used as applied in (Mpfumali et al., 2019).

3.10.1 Simple Averaging PI Combination Method

This method makes use of the Arithmetic means of the prediction limits from the forecasting models. This can be expressed as (3.37)

$$L_{Av} = \frac{1}{N}\sum_{n=1}^{N} L_n \qquad U_{Av} = \frac{1}{N}\sum_{n=1}^{N} U_n \tag{3.37}$$

A very much robust interval is known to be produced from this fairly simple approach (Mpfumali et al., 2019)

3.10.2 Median PI Combination Method

A method that is less sensitive to outliers and has a considerable ease of use is expressed as equation (3.38) given in (Mpfumali et al., 2019).

$$L_{Md} = Median(L_1, \cdots, L_N) \quad U_{Md} = Median(U_1, \cdots, U_N) \tag{3.38}$$

The Mathematical formulae presented in this chapter fits into our study because these formulae are either implemented in the computational tools employed or they are explicitly programmed in order to measure the performance and the uncertainties in predicting wind speed. Results from these metrics are reported in chapter 4 accordingly.

3.11 Chapter Summary

This chapter began with stating what it will cover and proceeded to a discussion on variable selection and data preprocessing under which it presents the mathematical (theoretical) formulations for the Lasso and the Generalised CV. It further discusses the ANN and the BPNN as its training algorithm in terms of the mathematical framework of this blackbox model. GAM and SGB along with forecast combination are the successive points of discussion in terms of their theories and methods in which an in-depth study on the QRA methods involving LQRA and AQRA are also expressed methodically. The chapter summary is preceded by various formulae for point forecast and interval forecast evaluation metrics. Table 3.1 presents an overall summary of the methodology for the research. A summary of the models, description and motivations for their use is presented in Table 3.1.

Model	**Description**	**Motivation**
ANN, SGB and GAM	Machine and Statistical learning, Deterministic methods	Point forecasting
LQRA and AQRA	Statistical and Probabilistic Methods	Forecasts combinationand Prediction interval construction
The LASSO	Mathematical	Variable/feature selection
MAE, RMSE, and MAPE	Mathematical	Point forecast evaluation metrics
PIW, PICP, PINAW, PINAD and PICAW	Mathematical formulations and Interval forecasting metrics	To quantify uncertainties in the forecasts
Simple Average and Median PI	Mathematical	Prediction Interval combination (Improves forecast)

Table 3.1: Summary of models, description and motivations

Chapter 4

Analysis, Results and Discussions

4.1 Introduction

This chapter gave a brief discussion on the data source along with its features (or variables) and observations. It proceeds to give the exploratory data analysis in terms of the variables selected using Lasso. The approach employed in this study only uses other covariates in a wind meteorology dataset to forecast wind speed and does not extend to wind power forecasting using the power curve and other methods, as discussed in Giebel and Kariniotakis (2017); Yan et al. (2014). The report of the results of various point forecasts and PI accuracy values and compare models using these accuracy measures are presented in this chapter.

4.2 Source and Description of Dataset

The dataset contain meteorological data with features such as wind speed, wind direction, humidty, barometric pressure and air temperature (Brown et al., 1984; Landberg, 1999). The data was obtained from the Wind Atlas South Africa website (http://wasadata.csir.co.za/wasa1/WASAData). Much data for various locations where readings were carried out are available on this website. The data corresponding to the location 3 (WM03) which is the Vredendal in the Western Cape province of South

Africa was used for this research. WM03 was picked because it has small amount of null and nan values also known as missing data, they could be easily computed and cleaned up in data pre-processing. Another reason for chosing WM03 is because the needed properties for time series forecasting, such as trend, periodic seasonality, residuals were observed in its visualization compared to other points, See Appendix B for the visualization of the decomposition for the response variable of the dataset. Vredendal is located on Longitude 18.419916^0E and Latitude 31.730507^0S, respectively. The data were curated from 1st of January 2018 through 1st of March 2019. The dataset contains 61057 rows of observations and was reduced to 60769 by the final two days lagged variable added to the selected variables. The data was divided into training and test data. While training data corresponds to data from January - November 2018, taking 45576 observations of 10 minutes recordings, testing data on the other hand spans through Dec to 1st March 2019 with 15193 observations of 10 minutes interval recordings. A pictorial representation of the point location and map representation along with the mast used on the location corresponding to Vredendal is shown in Figures 4.1 and 4.2, respectively.

4.3 Exploratory Data Analysis

The meteorological dataset thus obtained contains dirty data. The messy data was cleaned up using data cleaning approaches in Python and R used for implementing the models. Nan and NULL values makes data dirty. Hence, our first exploratory data analysis focused on various methods of cleaning up data. We explore the summary statistics of the response variable (WS_62_mean) as shown in Table 4.1.

Table 4.1 shows that the distribution of the Wind speed at $62m/s$ is right skewed and platykurtic, as seen in the skewness and kurtosis values, hence not normally distributed. Another reason for the non normality assertion is that the mean and median have different values. Time series plot, density, QQ (normal quantile to quantile) and boxplots as shown in Figure 4.3 all show that the predictor variable (wind speed at

http://wasadata.csir.co.za/wasa1/WASAData

https://www.weather-forecast.com/locationmaps/Vredendal.10.gif

Figure 4.1: Vredendal Point and Map Location.

Figure 4.2: Site showing Mast for wind speed at 62m.
(Source: WASA station and site description report pg29).

62m) distribution is not normally distributed.

Min	1st Qu	Median	Mean	3rd Qu	Max	Std.	Skewness	Kurtosis
0.2134	4.3674	6.8670	7.1725	9.6473	20.8555	3.475177	0.3963677	-0.4654862

Table 4.1: Wind Speed 62m/s Summary Statistics.

We also visualized the variations in the wind speed measured at 62m, the predictor variable, in order to envisage the possibility of forecasting it. The visualization was carried out using a box plot as shown in Figure 4.4. As can be seen from Figure 4.4, various patterns that are expected from the unpredictability nature of the wind are

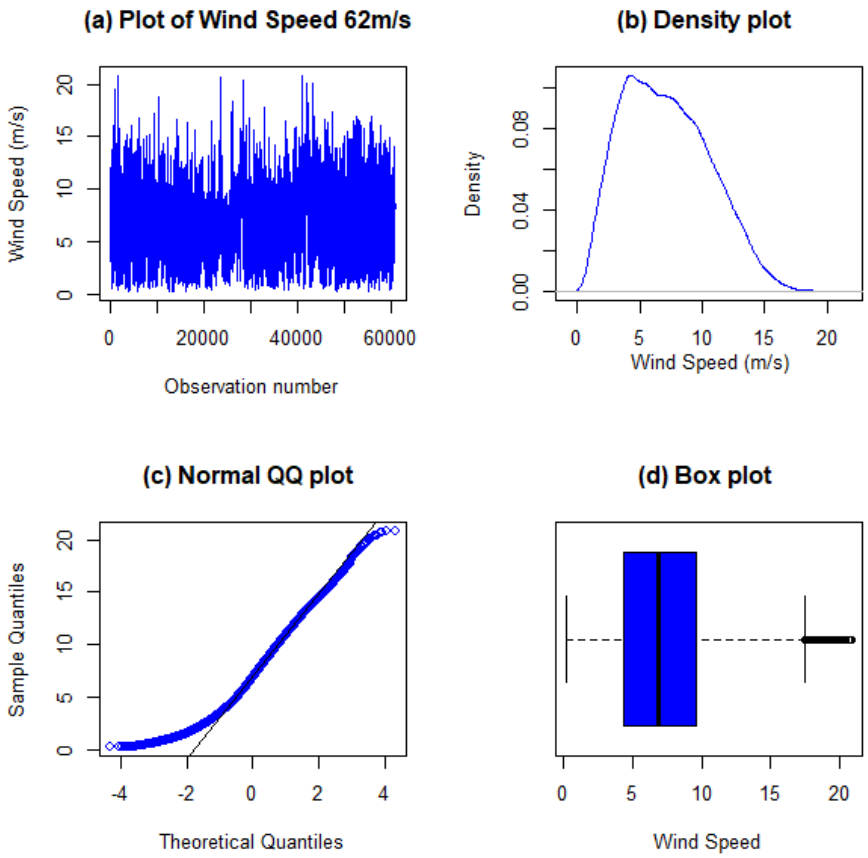

Figure 4.3: Diagnostic plots for the predictor variable, Wind Speed 62_mean (m/s).

visible. There are no clear patterns visible when visualized in terms of the weeks in the year and the days of the month, other than the availability of the wind. However, the hours of the day and the months of the year show some clear patterns. Seasonal variation can be inferred from the months of the year boxplots, and a trend can be seen from the hours of the day boxplots. The 10th month corresponding to the peak of the spring season has the largest wind speed as seen in the month of the year boxplots. Whereas, the wind speed is seen to progress from the early hours of the day to the later hours of the day. The box plots show that wind is available throughout the days,

however in varying and unpredictable quantity. This further shows how viable the wind is for generating renewable energy.

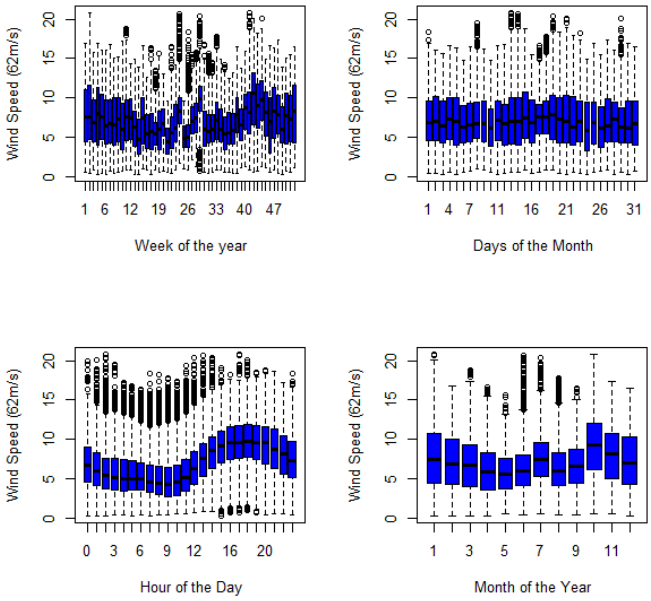

Figure 4.4: Distribution of Wind Speed(m/s) measured at 62m WT hub height across the week, month, day and year in the dataset.

4.4 Variable Selection using Lasso

This section of the chapter is discussed in two parts. The first part gives the variable selection for the ANN using the Python 3 computational tool. The second part, on the other hand, gives the variable selection from the benchmark models, i.e. the GAM and the SGB respectively on R software.

4.4.1 ANN Variable Selection

The variable selection for the ANN was done using the Python 3 programming language. We made use of the ordinary Lasso and the cross-validation Lasso (Lassocv)

in the variable selection process. As explained in chapter 3, the Lassocv proved more robust than the Lasso because more variables were selected using it compared to using the standard Lasso. We hereby feed the ANN model with the variables selected from the Lassocv. Table 4.2 shows the differences recorded from using Lasso and Lassocv for the ANN model variable selection process. With the same training and testing ratio of 75% and 25% for training and testing data respectively. Lassocv selected more variables with larger train and test score values along with a smaller training and testing data mean squared error (MSE) values, respectively, as seen in Table 4.2. The visualization of the variables against their coefficients along with the plot of the coefficient magnitude against the coefficient index (variables) using Lassocv of the python 3 computational tool for the process of variable selection for the ANN model is shown in Figure 4.5. From Figure 4.5, few variables or features within the coefficient index between 20 and 30 have negative coefficient magnitude, as shown by the small spike to the left in the left-hand side of Figure 4.5 and the spike below the zero line at the right-hand side of the figure. Majority of the variables have their magnitude on the zero line in the plot of coefficient magnitude and coefficient index. The various spikes in the right hand plot show the variables with non zero positive magnitude, which are represented by different colours of lines pointing in the right direction by the coefficient of Lasso model plot.

Metric	Lasso	Lassocv
Data Split ratio	0.75/0.25	0.75/0.25
Number of Variables selected	10 out of 47	14 out of 47
Training Score	0.9859	0.9964
Testing Score	0.98565	0.9963
Training data MSE	0.1704	0.0431
Testing data MSE	0.1736	0.0447

Table 4.2: Comparison between Lasso and Lassocv.

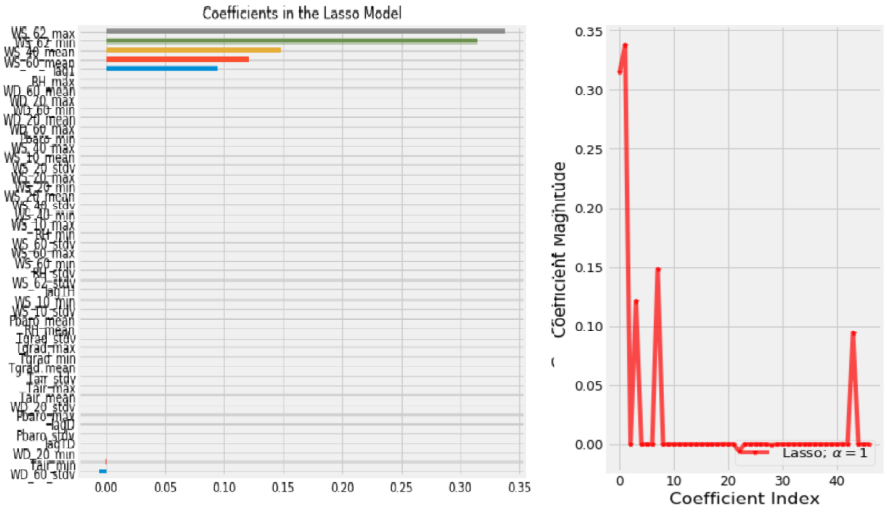

Figure 4.5: Variable selection and coeficient of LassocV plots.

4.4.2 Benchmark Variable Selection

The variable selection process for the benchmark, GAM and SGB models was done using the ”glmnet” library available on the R computational tool. This library implements both the ordinary Lasso and the Lasso cross-validation. A plot generated from the Lasso computation for the benchmark models on R is shown in Figure 4.6. It shows

the L_1 norm which is the Lasso along with the coefficient for selection of variables. The MSE plot against the log lambda shows that few variables lie between 0 and 1 on the log lambda axis and these are the selected variables for the variable selection process.

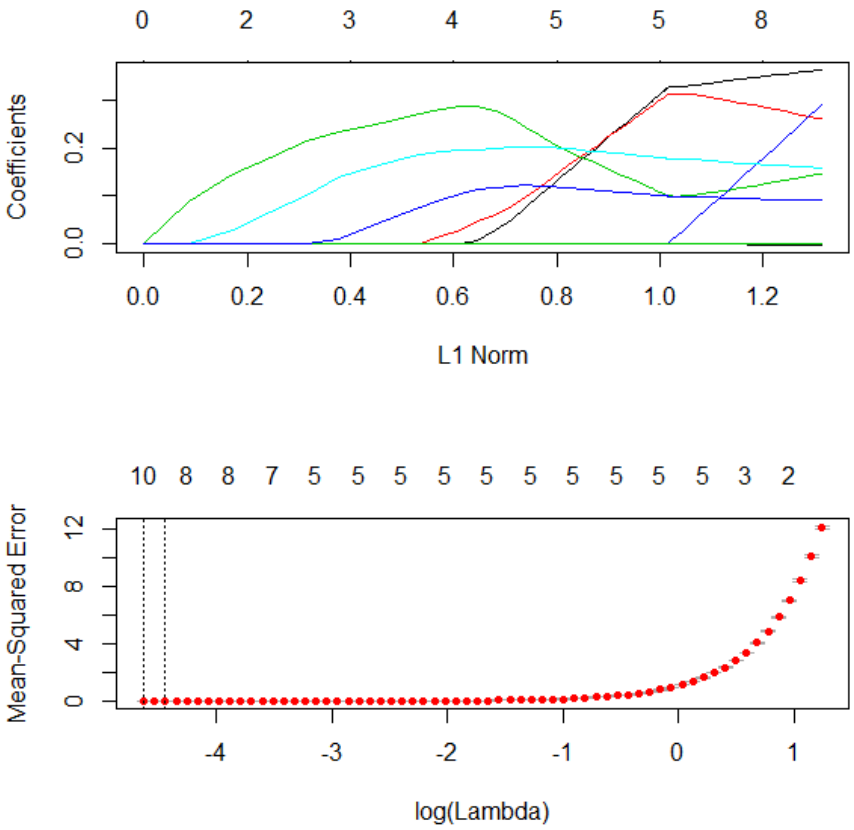

Figure 4.6: Benchmark Lasso plots.

Table 4.3 gives the selected variables along with their coefficients from these two programming tools. For the Lassocv from the python 3 kernel, the selected 14 variables were added with the other lagged variables and used for the ANN model. Also, the benchmark (BMlasso) model from the R kernel, selected 10 variables, lagged variables were added to the selected variables and used for both the GAM and the SGB models,

respectively. The lagging was done from 10 mintute lag, 1 hour lag, 1 day lag and 2 days lag accordingly. In all the variable selection process only the 10minute lag was selected by the various Lasso variable selection processes.

Variable	**ANN Lasso**	**ANN Lassocv**	**BMLasso**
WS_62_min	0.088897	0.314418	3.643221 e -01
WS_62_max	0.410781	0.337446	2.614169e-01
WS_60_mean	0.253464	0.120781	1.454732 e-01
WD_60_mean	0.000287	0.000134	9.436018 e-05
WD_60_min	–	0.000068	–
WD_60_max	-0.000591	0.000007	–
WD_60_stdv	-0.015175	-0.005907	-4.402683e-03
WD_20_mean	0.000834	0.000048	–
WD_20_stdv	-0.001630	—	–
RH_min	-0.000793	0.000169	–
lag1	0.119809	0.094290	8.940951e-02
WS_40_mean	–	0.148194	1.581057 e-01
WD_20_min	–	-0.000036	–
WD_20_max	–	0.000082	–
WS_60_stdv	–	–	2.898686e-01
Tair_min	–	0.000590	-7.744566e-05
Tair_mean	–	–	-2.311307e-04

Table 4.3: Variables and their co-efficients from the Lasso.

4.5 Benchmark Model Process Analysis

4.5.1 Stochastic Gradient Boosting

For this first benchmark model to perform forecasting, it's working principle entails some form of variable selection out of the variables fed into it. With each variable's relative importance, selection was made amongst the provided variables in its training data. Table 4.4 shows some of the variables picked by the model and their relative importance and visualised using Figure 4.7. The plot for this selection is given in Figure 4.7, where the vertical axis represents the variables selected by the model with incresing relative degree of importance. The SGB model run using the following final values n.trees = 150, interaction.depth = 3, shrinkage = 0.1 and n.minobsinnode = 10. Various computational metrics evaluating the SGB model given by the R computational tool is seen in Table 4.5.

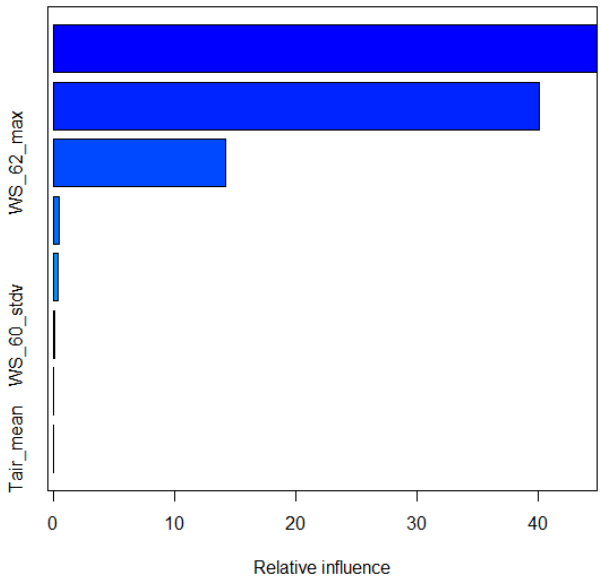

Figure 4.7: SGB variable selection plots.

Selected Variable	Relative Importance
WS_40_mean	4.379310e+01
WS_60_mean	4.117128e+01
WS_62_max	1.411519e+01
WD_60_mean	4.687078e-01
WD_60_stdv	3.892787e-01
WS_60_stdv	6.227314e-02
Tair_min	1.768753e-04
Tair_mean	0.000000e+00

Table 4.4: Variable selection by the SGB Model.

Interaction.depth	n.trees	RMSE	R-2quared	MAE
1	50	0.5661520	0.9800269	0.3959074
1	100	0.3781487	0.9883580	0.2662815
1	150	0.3302557	0.9908337	0.2339422
2	50	0.3850630	0.9885786	0.2712526
2	100	0.2699944	0.9938029	0.1930438
2	150	0.2418027	0.9949749	0.1750470
3	50	0.3110675	0.9922893	0.2177381
3	100	0.2260250	0.9956210	0.1625230
3	150	0.2000053	0.9965582	0.1454654

Table 4.5: SGB Computational Model Evaluation.

4.5.2 Generalised Additive Model, GAM

The same number of variables were used in the GAM as in the SGB model. The GAM as given in previous chapter uses crucial computational parameters. In fitting

the GAM model, these computational parameters were given along with various statistical error measure terms, in relation to the variables selected by the model from among the fitted variables for its computation. The values for these parameters and their statistical significance were given. The crucial parameters to the GAM model therefore involves the parametric coefficient for the intercept term, significance of the smooth terms, smooth parameter convergence and the basis terms. The 'mgcv' optimizer values, RMS gcv values, model rank and Hessian among other parameters were given. See Appendix A for values recorded for these parameters using the GAM.

The GAM and SGB being the benchmark models for this study produced point forecasts from the test set given the training set. The point forecasts from these two models are expressed graphically in Figure 4.8.

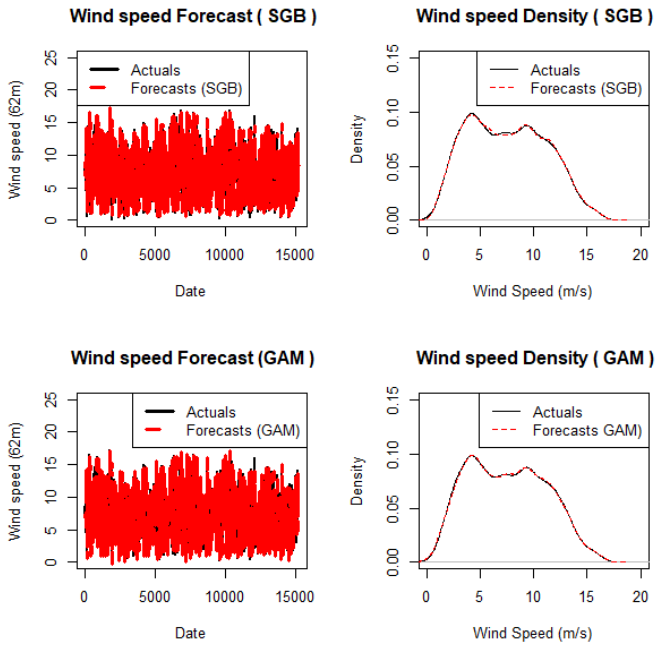

Figure 4.8: Benchmark Density and Point Forecasts Plots.

From Figure 4.8, the highest wind speed values both for the actual and forecasts

hovers close to 20 m/s. We can also infer that the wind speed at point very close to 5m/s is the densest plot of the wind speed with a density value very close to or a bit above 0.10. In the density plot the lines showing the actual point forecasts values and the lines showing the benchmark model's point forecasts values are very close showing the performance of the models in forecasting. This reflects a considerable acceptable performance in the two benchmark models. Table 4.6 presents a summary of the accuracy metrics.

4.6 Artificial Neural Networks and Additive Quantile Regression Averaging

This section provides the analysis and results from making use of the Artificial neural network (ANN) and the Additive quantile regression averaging (AQRA) model for point forcasting. The ANN was implemented using the computational tool known as python 3 programming language. Following the steps from data curation, to data explorations and visualisations along with data normalisation, this model is ready to carry out predictions from the variables selected by the Lassocv model. The ANN model was constructed from the SKLearn Library by importing MLPRegressor. The major determinant of an ANN is the number of layers in the network, the number of hidden layers and the number of nodes used for the hidden layer. Our ANN was constructed using three layers of input, hidden and output. The number of the hidden layer was made to be 1 by default, and the number of nodes in the hidden layer was given as 5. This makes the neural network thus constructed similar to the Bayesian Neural Network (BNN) in Mbuvha (2017). In a BNN, the values of the α and β are very crucial, in which case are between 0 and 1 for a BNN. We thus performed hyperparameter tuning to find the best α values, best parameter and best accuracy making use of the GridSearchCV from the SKLearn model selector library.

The AQRA on the other hand is a model formed from combining point forecasts

from the other models. The model is fitted with forecasts from the BNN, and the benchmark models (GAM and SGB) as its predictor variables, while the response variable are combined point forecasts of the AQRA model. The model was implemented using "ggam" library developed by Fasiolo et al. (2017) on the R computational tool. The model was set at a seed of 1000 and functions such as tune-learn-fast has its object containing the actual forecasts and the point forecasts from the other 3 models fitted at a quantile value of $\tau = 0.5$. A plot of the point forecast from the ANN and the AQRA models are shown in Figure 4.9.

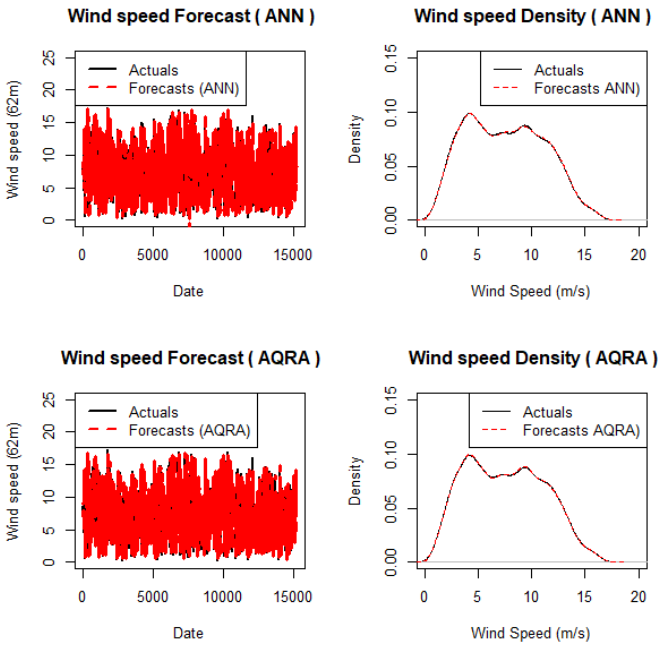

Figure 4.9: ANN & AQRA Density and Point Forecasts Plots.

From Figure 4.9, we found that the highest wind speed forecast lies between 15 and 20 m/s and the highest dense plot for the wind speed is at a position very close to 5m/s with density of a little above 0.10. These two density plots also presents these two models as viable models for wind speed forecasting because of the proximity of

the actual and point forecasts values as represented by the two lines in the density plots.

The forecasts from these four models show the wind speed values lies in range of 5-20m/s. This corresponds to the linearly progressing and constant power region of the graph in Figure 2.1 of chapter two. This shows that both the actual values and the values from forecasts are within the range of values where electric power is being generated from the wind turbine. A strong, increasing and constant electrical energy can thus be realized from the wind speed values, which depicts the richness of this renewable energy for generating electrical power.

4.7 Forecasts Accuracy Measures

4.7.1 Point Forecasts Accuracy Measures

This section presents the error measures for our point forecasting models. This study made use of a total of five (5) forecasting models. The main model is the ANN model and compared with two benchmark models which are the SGB (A machine learning model) and GAM (Statistical learning model). The other two (2) models are a statistical learning model called the Additive quantile regression averaging and the Linear quantile regression averaging (LQRA) formed by the point forecasts combination of the other three (3) models. We present three (3) accuracy metrics in order to evaluate these five (5) point forecasting models. The various accuracy of point forecasts from these models are presented in Table 4.6. These provide answers to the first two research questions in section 1.2.3.

Model	No of used Variables	RMSE	MAE	MAPE
BNN	14	0.2091	0.1526	2.7437
SGB	8	0.2553	0.1397	2.3754
GAM	8	0.2468	0.1579	3.0356
AQRA	3	0.1888	0.1167	2.0330
LQRA	3	0.1928	0.1204	2.0986

Table 4.6: Accuracy Measures for the Point Forecasting Models.

From Table 4.6, model BNN, is the first model whose forecasts is from the ANN, the second model is the SGB forecasts, the third model is the GAM forecasts, while the fourth and fifth models are the forcasts combination models which is the AQRA forecasts and the forecasts corresponding to the LQRA model respectively. Our discussion evaluates Table 4.6 in the lines of individual model and combined models.

BNN made use of 14 variables from its Lassocv variable selection model to fit the neural networks. It recorded the lowest RMSE value followed by GAM amongst the individual point forecasting methods. SGB on the other hand recorded the lowest MAE value followed by the BNN while GAM recorded the highest MAPE value followed by the BNN. Hence, accuracy measures using the MAE and the MAPE presents SGB as the best individual point forecasting model followed by the BNN while the GAM performed the least amongst the three models.

Considering the combined forecasting models, the fourth model which is the AQRA performed better than its counterpart fifth model which is the LQRA in all accuracy metrics axis. Model AQRA also outperformed all of the individual point forecasting models as can be seen from Table 4.6. This is in tandem with most literature findings that combining point forecasts presents forecasts whose accuracy measures are lower than the individual forecasting models (Nowotarski and Weron, 2015; Shen et al., 2018; Sun et al., 2017). This finding holds sway for both methods of forecasts combination as

can be seen from their accuracy metrics values which are lower than that of the three individual accuracy metrics' values for the point forecasting models. When we combine forecasting models, the resultant combined forecasts perform better than the individual point forecasting models. Hence, we retain model AQRA as the main forecast combination model, while we drop model LQRA going forward. The next sections are geared towards proffering answers to the last research question which is to quantify the uncertainties in the point forecasts and interval forecasts.

4.7.2 Prediction Interval Width

The analysis of the prediction interval width (PIW) at 95th quantile is presented in this section. Table 4.7 gives the summary statistics of the generated PIW which gives the nature of the PI generated.

Model	**Min**	**Max**	**Mean**	**Median**	**St.Dev**	**Skewness**	**Kurtosis**
BNN	0.3281	0.9217	0.8148	0.8613	0.1016	-0.7914	-0.7346
SGB	0.4378	0.8313	0.7099	0.7487	0.1124	-1.0610	-0.0271
GAM	0.2461	1.1725	0.8762	0.8744	0.1952	-0.2188	-0.7905
AQRA	0.3843	0.7759	0.6277	0.6399	0.1213	-0.5330	-0.9302

Table 4.7: Summary Statistics for PIW at 95th quantile.

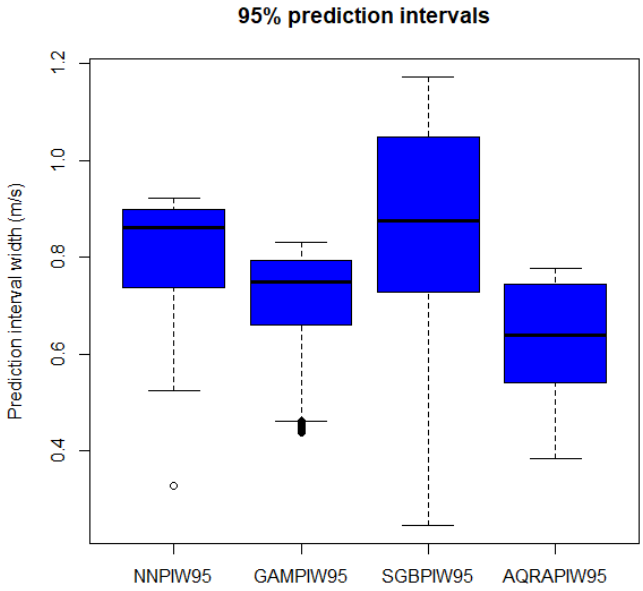

Figure 4.10: PIW of Models BNN-AQRA.

From Table 4.7, model BNN has the least standard deviation followed by model SGB, hence, the PIW from these two models are narrower than those from AQRA and GAM respectively. The skewness measures the distribution of a model and shows from the table that the PIWs are not normally distributed, although close to a normal distribution (except SGB), because a normal distribution has a skewness value of 0. All of the models are also negatively skewed showing that they are left skewed. The Kurtosis value for a distribution is expected to be 3 and all the Kurtosis values for the models shows them to be negative and less than 3 are therefore said to be platykurtic. A visualization for the PIW is presented in Figure 4.10. The figure presents the PIW from Model AQRA as the most symmetrical and best model followed by the Model SGB. Model BNN and GAM are skewed and much narrower than the other two. We can still visualize the PIW using the density plots in Figure 4.11, presenting model

BNN as the most narrow model.

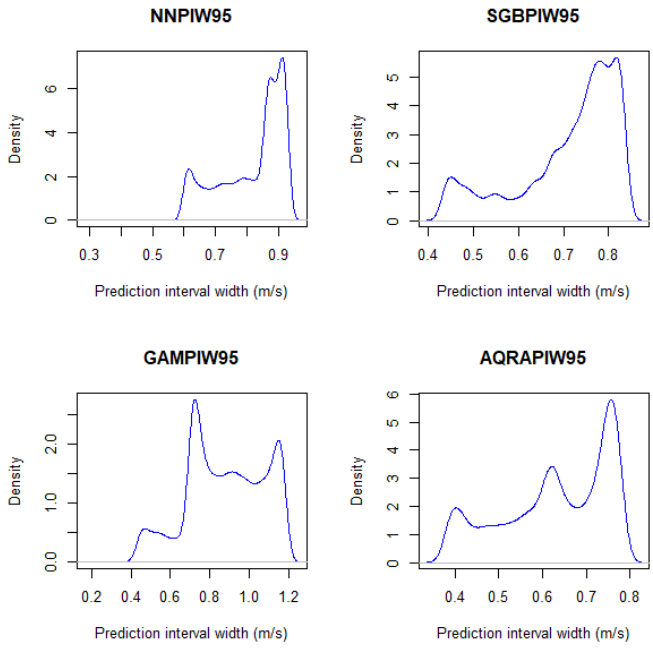

Figure 4.11: Density Plots for PIW of model BNN through AQRA

4.7.3 Prediction Intervals Evaluation

Forecasting using models is known to be filled with much uncertainties given good point forecasts performance evaluation. In this section, we describe the method for quantifying the uncertainties in point forecasting models and also in the combined point forecasting model. In order to measure uncertainties the needed feature to be used is the prediction interval. Along with making use of the AQRA for forecast combination, we also used it to construct prediction intervals through the lower and upper bound estimate (LUBE) (Shen et al., 2018) for the models and for itself at different quantiles. The quantiles considered are the: 90th quantile, the 95th quantile and the 99th quantile respectively.

To quantify uncertainties, five (5) prediction interval evaluation metrics are used

in this study and results summarised in Table 4.8. These metrics provide the quantifiable answer to the third research question by presenting the means to evaluate contextually the uncertainties in the point and interval forecasts. The 5 metrics are the prediction interval nominal coverage (PINC), the prediction interval coverage percentage (PICP), the prediction interval normalised average width (PINAW), the prediction interval normalised average deviation (PINAD) and the prediction interval coverage average normalised width (PICAW). The theoretical formulations of these metrics had been given in Chapter 3 (Equations 3.29, 3.31 through 3.36). We present these uncertainty metrics relating to quantifying the uncertainty in wind speed forecasting using the individual point forecasting models and the AQRA forecasting combination model through the use of AQRA to construct prediction intervals whose values according to these metrics are presented in Table 4.8:

PINC%	**Model**	**%PICP**	**%PINAW**	**%PICAW**	**%PINAD**	**ABLL**	**AAUL**
90	BNN	90.4956	3.7853	8.3489	0.0130	717	727
	SGB	90.6997	3.8336	8.4560	0.1101	701	712
	GAM	90.6334	3.8259	8.4432	0.1109	702	721
	AQRA	90.7194	2.8583	6.3140	0.0650	707	703
95	BNN	95.2807	4.7673	10.5152	0.0388	358	359
	SGB	95.4058	4.1536	9.1076	0.0659	348	341
	GAM	95.3992	5.1263	11.3586	0.0665	345	354
	AQRA	95.3926	3.6725	8.1101	0.0534	342	358
99	BNN	99.0654	7.3491	16.1999	0.0083	72	70
	SGB	99.0983	7.8238	17.9690	0.0298	66	71
	GAM	99.0259	8.8693	19.6512	0.0198	78	70
	AQRA	99.0588	7.1009	15.8021	0.0133	70	73

Table 4.8: Prediction Interval Evaluation.

From Table 4.8, five metrics are used to evaluate PIs from individual and the combined models. Four (4) models in all was evaluated using these metrics. Using the PINC as the predetermined confidence value or the quantile from which the PI is being evaluated. PICP is used to ascertain the reliability of the constructed PIs. Therefore, the more the actual values are covered by the PI, the higher the PICP values (Shen et al., 2018). Also, the PICP value is expected to be greater or equal to the PINC or confidence value else deemed as invalid (Sun et al., 2017). Accurate and satisfactory PI performance is indicative of a higher PICP value and a lower PINAW values (Sun et al., 2017; Shen et al., 2018). Both PICP and PICAW value indicates the quality of the constructed PI. Hence, a high PICP value with small PICAW value is required for a quality PI construction (Shen et al., 2018). Also, given a high PICP, the deviation value of the PIs from the actual value is expressed by the PINAD. Hence, a higher PICP should give a lower PINAD value showing less deviation from the actual value (Mpfumali et al., 2019).

Our discussion on Table 4.8 follows from the foregoing theoretical and literature findings. Using the PICP at the 90% PINC, all forecasts are valid and the Model AQRA has the highest value and presents more coverage than the rest of the models. Model SGB is next and the last is model BNN showing that the PI from model BNN covers less actual values than the rest. Taking the PICP and the PINAW, model AQRA presents both the highest PICP value and the lowest PINAW, thus the most reliable model at this confidence level. This is seconded by Model BNN and the least is model GAM. Although the PI from model BNN covers less actual values, the covered values presents a closely fit weight than SGB and GAM. Taking the PICP and the PICAW, AQRA is the model with the best quality PI since it has the lowest PICAW value and the highest PICP value followed by the BNN and the least is the SGB. The degree of deviation from the actual value is shown by the PINAD and BNN has the least degree of deviation followed by the AQRA, GAM has the highest degree of deviation from the

actual value. The ABLL and AAUL columns represents the number of actual values that are not within the range of the PIW. They represent the number of actual values that are below the lower limit and the actual values that are above the upper limit respectively.

For the 95% confidence value, SGB recordedd the highest PICP value, the GAM is next and the AQRA, while the BNN has the lowest PICP value. At the 99% confidence values, model SGB and BNN gives higher PICP values than the GAM and AQRA. The PICAW and PINAW values presents AQRA as the best model followed by an interchange of SGB and BNN respectively while GAM remains constantly the least considering these two confidence values. Using the PINAD, BNN has the best degree of deviation, secondly the AQRA followed by an interchange between SGB and GAM respectively. The number of target values outside of the lower and upper limit range keep reducing as the confidence level increases. Model AQRA is thus the best model at 90% confidence value, however best models are evaluated based on what these metrics measures such as validity, reliability, quality and degree of deviation at a particular confidence value.

4.7.4 Evaluation of Combined Prediction Intervals

Just as point forecasts can be combined and estimate the evaluation of its accuracy, prediction intervals can also be combined and estimate the accuracy of the resultant combined PI. In this study, two prediction interval combination methods which are the Simple average and the Median method were combined to produce a combined PI. Simple average gets the arithmetic mean of the PIs from individual models using the lower and upper bound estimate. A row wise summation of the lower and upper limits is carried out and averaged by taking each model at the three quantiles considering the total number of models used. The mathematical formulation is given in equation (3.37). The Median method also follows as the simple average method. The median values of all the models considered are at a particular quantile. The mathematical expression

was given in equation (3.38). Given in Table 4.9 is the combined prediction interval evaluation using the same metrics as the individual PI model evaluation.

PINC%	Model	%PICP	%PINAW	%PICAW	%PINAD	ABLL	AAUL
90	SAverage	93.1743	3.5780	7.9661	0.0250	513	524
	Median	92.3978	3.5510	7.8617	0.0250	579	576
95	SAverage	97.1500	4.4299	9.9499	0.0370	198	235
	Median	96.8012	4.2181	9.3775	0.0330	235	251
99	SAverage	98.4137	7.7858	11.2930	0.0110	198	43
	Median	99.3148	7.2292	16.7254	0.0044	52	48

Table 4.9: Combined Prediction Interval Evaluation.

From Table 4.9, the simple average prediction interval combination method gives better PI than the median method at the 90% and 95% confidence values using the PICP. However, it gives an invalid PI in the 99% confidence level recording a value smaller than the confidence level value, while the median method gave a satisfactory coverage value for its PICP. The median method provided reliable and better quality PI than the simple average method, since it records lower values for its PINAW and PICAW at the 90% and 95% confidence levels. The 99% confidence value presents the median method as better than the average method using the PINAW, while the PICAW shows the opposite. The PINAD value for the 90% confidence level records a tie for the two combination methods, while the median method recorded the best value for its degree of deviation using PINAD at the 95% and 99% confidence levels respectively. Lastly, an increment was seen in the number of PI not within the lower and upper limit range for the two methods at the 90% and 95% confidence values. The 99% confidence value shows more number of actuals below the lower limit. These present the median method as the best PI combination method over the simple average method.

4.8 Residual Analysis

The Summary statistics of errors or residuals for the models is seen in Table 4.10. The table shows Model AQRA as having the least standard deviation which means that its error distribution is the narrowest of all the models which can also be seen from the box plot of Figure 4.12. This shows that the best Model is the AQRA in comparison to the others. The minimum values are all negative and the skewness values are far from being normally distributed. The kurtosis value shows erratic patterns, however, since they are all more than 3, they are termed leptokurtic data. The density plot in Figure 4.13 shows similar patterns and no much information can be inferred from these plots except that they are all in between the negative 1 and positive 1 regions in the forecast error axis. This concludes the uncertainty measure for the point and interval forecasts as needed to answer the third research question.

Model	**Min**	**Max**	**Mean**	**Median**	**St.Dev**	**Skewness**	**Kurtosis**
BNN	-1.2156	4.8961	0.0235	-0.0304	0.2077	1.4241	29.9048
SGB	-1.1015	8.4607	-0.0128	-0.0180	0.2550	11.4720	273.8021
GAM	-1.4350	6.4689	0.0028	0.0045	0.2468	4.0097	72.3618
AQRA	-0.9845	4.5838	0.0061	0.0004	0.1887	4.9086	78.1724

Table 4.10: Summary Statistics for the Residuals.

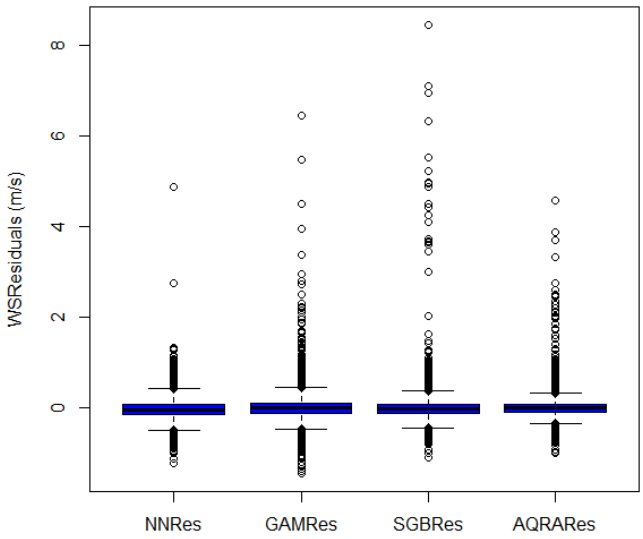

Figure 4.12: Box Plots of Residuals.

4.9 Chapter Summary

This chapter focused majorly on the reporting of various analysis, the results and discussions from the results. It began by giving a brief introduction of the chapter contents. The discussion proceeded by giving the source and description of the research dataset. Exploratory data analysis and variable selection using lasso under which the discussion focused on ANN variable and benchmark variable selection. Benchmark model process analysis was presented under which the SGB and GAM was discussed showing their forecast and density plots. ANN and AQRA forecasts was presented next. Forecast accuracy measures in which point forecasts, interval and its combined forecasts error measures were presented giving a further discussion of each of our findings, thereby proffering answers to the research questions of section 1.2.3. Presented also was the residual analysis and a concluding remarks summarizing the chapter. This

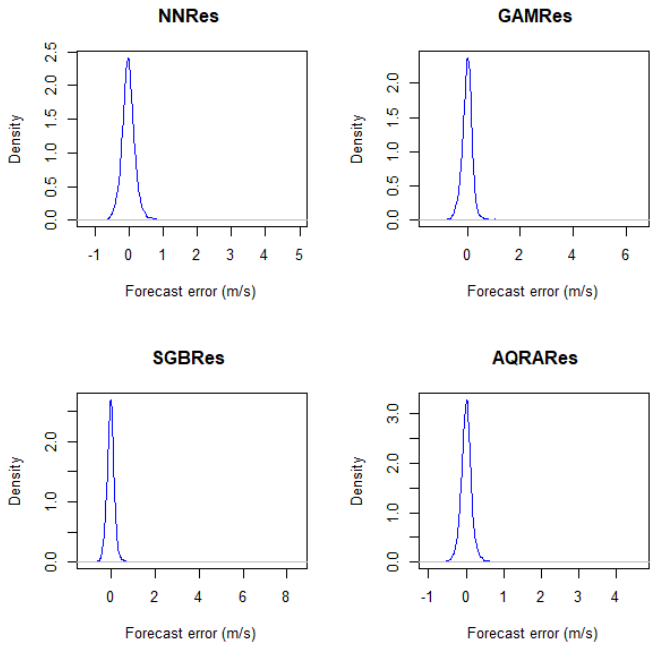

Figure 4.13: Density Plots of Residuals.

report proceeds by giving a summary of each chapter and a conclusion of the findings along with recommendations based on these findings, for future research in the next chapter.

Chapter 5

Summary, Conclusions and Recommendations

5.1 Introduction

This chapter presents the summary of the mini-dissertation from the first chapter to the fourth chapter. It proceeds to give conclusions of the findings after which recommendations based on these conclusions will also be given. The report is concluded by stating areas for future works and research regarding wind speed prediction for power generation.

5.2 Research Summary

The report started with a brief introduction and problem formulation in which the intermittency and variability of the wind makes prediction near an impossible task. The research was focused on carrying out a comparative analysis of Statistical and Machine learning methods along with their combination to wind speed point and interval forecasts for wind power generation. These informs the research questions poised for answers. The research is significant in that wind speed forecasting is needed for incorporating the resulting electrical power into the electricity grid of a country, thereby in-

creasing the overall electrical power generated for distribution. The research scope was to make use of the methods on a South African wind data set obtained from WASA. It is expected that contributions from this research will bring about an equal representation in the exploitation of wind as a renewable energy amongst others such as solar energy. Another envisaged contribution is to present GAM as a viable model for forecasting, comparable to the ARIMA and SARIMAX.

Chapter two gave the theoretical foundations on forecasting regarding the various time scales horizon and their various applications as found in the literature. It explored also, the various relations between the wind speed, the electrical energy and consequently electrical power generation from wind speed forecasting. Various techniques that have been used for various forms of forecasting and forecast combinations along with uncertainty measures were explored. The use of GAM for other forms of forecasting but not for wind speed forecasting was the major identified gap the research fills. Chapter 3 gave many mathematical underpinnings of the models and performance metrics along with uncertainty metrics. Chapter 4 presented the results from various analyses carried out in the implementation using the Python and R for the Lasso variable selection and the models under study. to as certain performance, accuracy measures and uncertainty metrics were used to select best models, proffering answers to the three research questions in section 1.2.3. In most of the discussions, it was found that the model AQRA which is the Additive quantile regression averaging method out performed other models for wind speed forecasting using both the point forecasts accuracy metrics (such as the MAE, RMSE, and MAPE) and the prediction intervals uncertainty measures involving the PINC, PICP, PICAW, PINAW and PINAD.

5.3 Conclusions from Research Findings

This research was geared towards investigating methods useful for forecasting short-term unpredictable and intermittent wind speed for power generation. A method with smaller errors indicates the most desirable method. Two approaches were investi-

gated, viz: individual forecasting models and forecast combination models. Amongst the individual models employed are the Bayesian NN for the ANN compared with two bench mark models involving SGB (ML method) and GAM (Statistical Learning method). The evaluation of these three methods presented ANN as the best using the RMSE while the SGB is the best from the MAE and MAPE forecasts accuracy measures. On the other hand two forecast combination methods, both of which are statistical learning methods were investigated for the forecast combination and the AQRA out performed the LQRA. It also out performed the individual forecasts from the 3 methods (ANN and the two benchmark models) using all the 3 forecast accuracy metrics involving the RMSE, MAE and MAPE. Therefore, we present AQRA from forecasts combination as the best method for point forecasting of wind speed having less error values and the most desirable model.

Furthermore, the prediction interval uncertainty measure, using only the AQRA, the first uncertainty measure employed was the prediction interval width. Using this metric, our analysis at 95% confidence level showed that the model AQRA was the most symmetrical and the best model followed by the SGB even though ANN and the GAM presented narrower prediction intervals than the others. All these models did not give a normal distribution and were all platykurtic. Further investigation on interval forecast involved prediction intervals against the target values.

Three confidence levels were considered which are the 90%, 95%, and 99% respectively. All the interval forecasts were checked in the context of their validity, reliability, quality and degree of deviation from the actual values. All the models were valid models since the PICP values were greater than the predetermined confidence levels. AQRA presented more reliable prediction intervals at the 90% having the highest coverage value. Also, this constructed prediction interval at this level of confidence was desirable and having satisfactory performance and quality since the lowest PINAW and PICAW values were recorded by the AQRA followed by the ANN. However, the

PINAD value shows the ANN as the model with the least deviation. Actual values below and above the lower and upper limit kept decreasing showing the AQRA with the least values.

Forecasts from the SGB model have the highest coverage at the 95% confidence level followed by the GAM. Using the PINAW and PICAW for quality, desirable and satisfactory performance evaluation, presented the AQRA as the best model, while ANN and AQRA have the least degree of deviation. A constant decrease was seen with the actual values below and above the lower and upper limits respectively. The 99% confidence level shows the SGB as the most reliable from its highest PICP value followed by the GAM while the quality, accurate and satisfactory performance PI was seen to be the AQRA from its PINAW and PICAW values. The least deviation was seen in the ANN followed by the AQRA. The actual values below and above the lower and upper limit showed no consistency at this confidence level. It also concluded that the best model is the AQRA followed by the SGB.

Analysis from the interval combination method as well was reported. The simple average method and the median method were employed. The simple average gave an invalid PICP at the 99% confidence level. The median method outperformed the simple average method in all prediction interval (accuracy measurement) metrics considered and thus, the best interval combination method. Residual analysis also presents AQRA as the best having the narrowest error non-normal distribution along with the other models.

5.4 Recommendation

From the findings reported in the last section, we hereby recommend short term wind speed prediction to have wind power generation, a rigorous point and interval forecasts along with forecasts combination and intervals combination inorder to have desirable and less prone to error methodology to the electricity generation and distribution company of South Africa (Eskom).

5.4.1 Future Works

Forecasts combination has been seen in this research as a viable means of wind speed forecasting. We recommend more sophisticated methods for forecast combination. Methods such as Deep neural networks and a hybridization of machine learning and statistical learning for combining point forecasts. We also recommend point forecasts using deep learning algorithms such as the Recurrent neural networks-Long and short term memory amongst others. Prediction intervals investigated in this research are the basic forms for constructing PIs, we recommend using Machine/Deep learning based methods for constructing prediction intervals in order to quantify uncertainties in point forecasts.

Appendix A

Model GAM Parameters

	Estimate	**Std. Error**	**t value**	**Pr**$(>\|t\|)$
(Intercept)	7.042363	0.001105	6373	$< 2e-16 * **$

Table A.1: Parameter Coefficients.

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	edf	Ref.df	F	p-value
s(WS_62_max)	7.986	8.544	2438.30	$< 2e-16 ***$
s(WS_60_mean)	9.000	9.000	1598.67	$< 2e-16 ***$
s(WS_60_stdv)	8.792	8.981	612.39	$< 2e-16 ***$
s(WS_40_mean)	8.496	8.921	759.12	$< 2e-16 ***$
s(WD_60_mean)	9.000	9.000	2180.36	$< 2e-16 ***$
s(WD_60_stdv)	8.899	8.997	416.60	$< 2e-16 ***$
s(Tair_mean)	9.000	9.000	24.20	$< 2e-16 ***$
s(Tair_min)	7.731	8.626	24.59	$< 2e-16 ***$

Table A.2: Approximate significance of smooth terms:

— Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.995 Deviance explained = 99.5% GCV = 0.055736 Scale est. = 0.05565 n = 45576

Method: GCV Optimizer: magic Smoothing parameter selection converged after 27 iterations.
The RMS GCV score gradient at convergence was 7.271268e-08 . The Hessian was positive definite. Model rank = 73 / 73

	k‘	edf	k-index	p-value
s(WS_62_max)	9.00	7.99	0.99	0.14
s(WS_60_mean)	9.00	9.00	1.00	0.35
s(WS_60_stdv)	9.00	8.79	0.94	$< 2e-16 **$
s(WS_40_mean)	9.00	8.50	1.02	0.92
s(WD_60_mean)	9.00	9.00	0.74	$< 2e-16 **$
s(WD_60_stdv)	9.00	8.90	0.98	0.08 .
s(Tair_mean)	9.00	9.00	0.98	0.07 .
s(Tair_min)	9.00	7.73	0.99	0.36

Table A.3: Basis dimension (k) checking results.

. Low p-value (k-index¡1) may indicate that k is too low, especially if edf is close to k’. Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Appendix B

Visualizations from Python

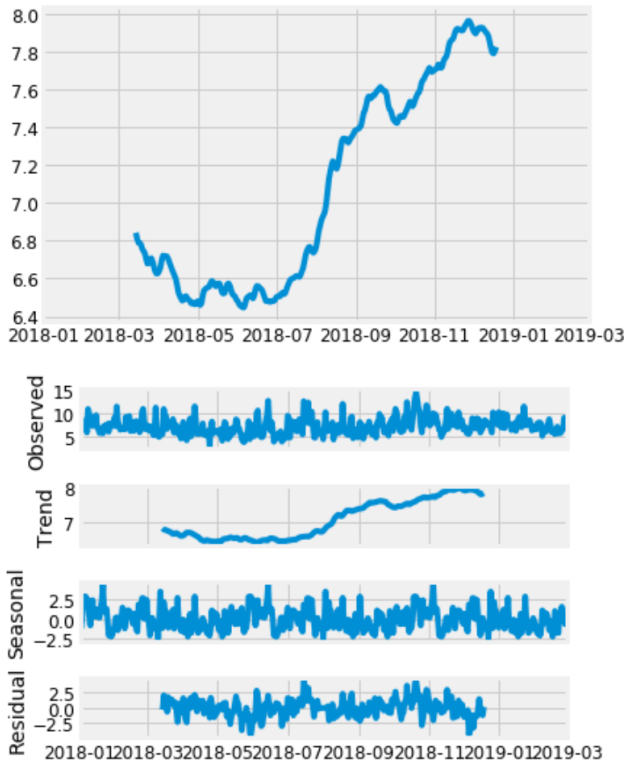

Figure B.1: WindSpeed decomposition.

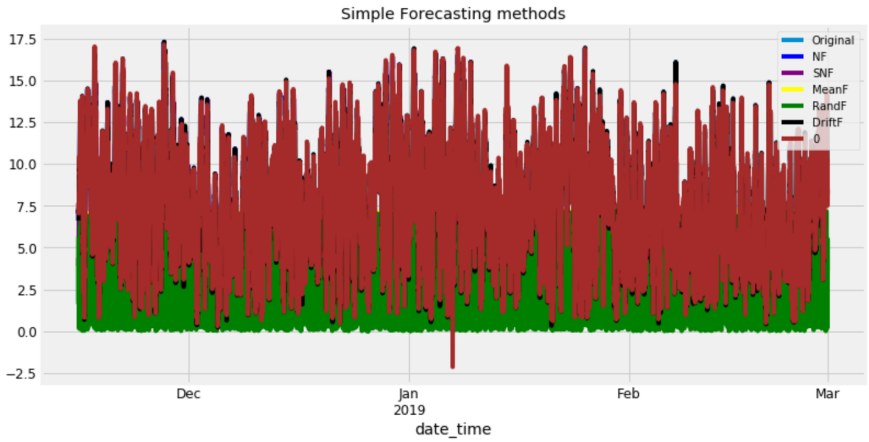

Figure B.2: Plots of the BNN forcasts

	ME	RMSE	MAE	MAPE
Naive Scores	0.000127	0.455594	0.455594	8.452912
SNaive Scores	0.001457	1.296775	1.296775	24.186159
Mean Scores	0.390825	3.115396	3.115396	67.216507
Random Scores	3.953196	4.681748	4.681748	64.534753
Drift Scores	0.000209	0.455596	0.455596	8.452733
MLP Regression Scores	-0.023446	0.209054	0.152607	2.743735

Figure B.3: Scores for the simple forecasts and BNN.

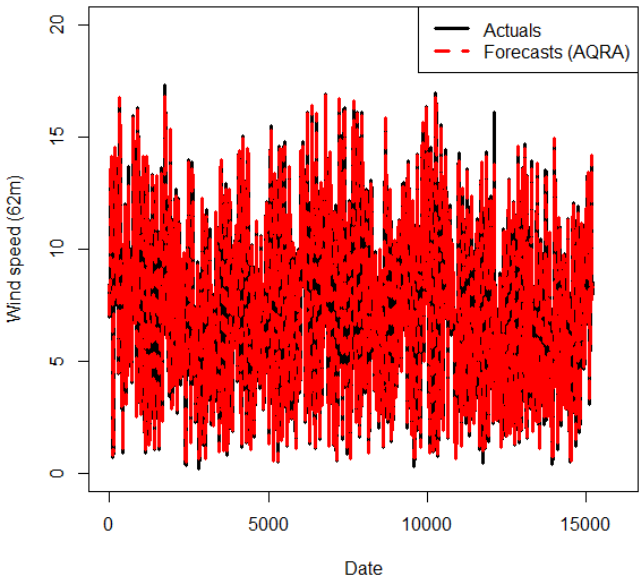

Figure B.4: AQRAM Plots.

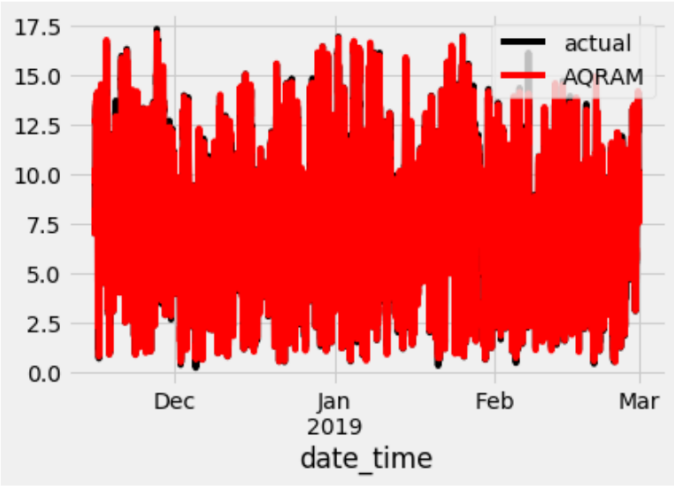

Figure B.5: Dated AQRAM Plots.

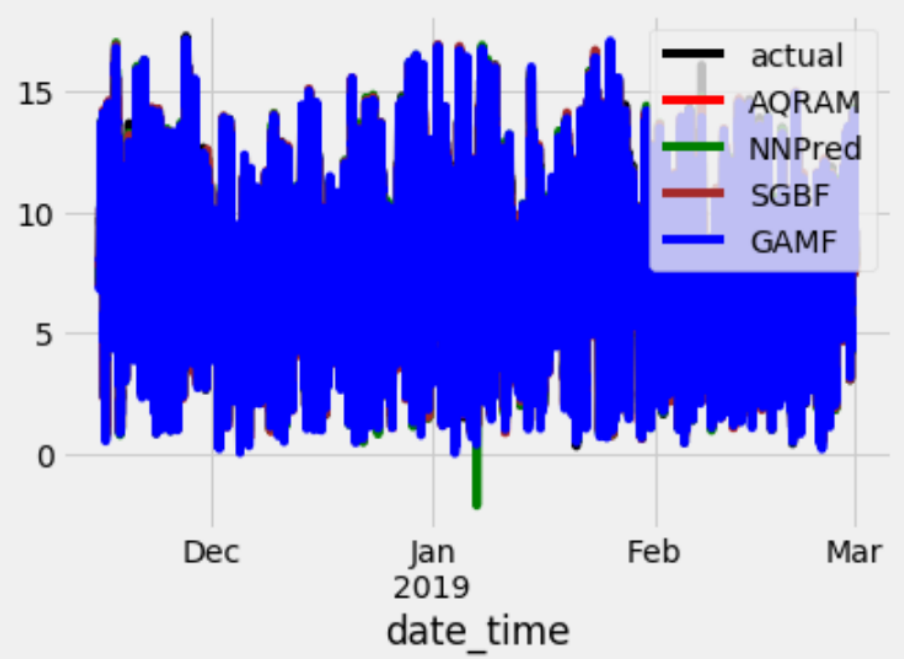

Figure B.6: Dated Plots for all the Models.

Appendix C

Sample R code

```
#########################################################################
# Major libraries used in R
library ( ggplot2 )
library ( tseries )
library ( e1071 )
library(glmnet)
library(dplyr)
library(caret)
library(gbm)
library(mgcv)
library(forecast)
################################  Accuracy ########################################
library(forecast)
accuracy(SGBF, actual)
accuracy(GAMF, actual)
accuracy(NNPred, actual)
accuracy(fQRA, actual)
## calculating summary statistics , skewness and kurtosis of GHI ##############################
summary (W)
library ( e1071 )
sd(W)
skewness (W)
kurtosis (W)
# ###############################################################################
# Distributions: time series , qqnorn , density and box plot for Wind Speed
# ###################################################################
win.graph()
par ( mfrow =c(2 ,2))
W <-ts( analyticdata$WS_62_mean )
plot(W, xlab =" Observation number ",ylab =" Wind Speed 62 (m/s ^2) "
     ,main ="(a) Plot of Wind Speed ",col = " blue ")
```

```
plot ( density (W),xlab =" Wind Speed 62 (m/s ^2)
      ",main ="(b) Density plot ",col = " blue ")
qqnorm (W, col = " blue ",main ="(c) Normal QQ plot ")
qqline (W)
boxplot (W, main ="(d) Box plot ",varwidth =TRUE ,
       xlab =" Wind Speed (m/s ^2) ", col = " blue ", horizontal = TRUE )
############################## Density and Box Plots ###################################
PIW95 = c(" NNPIW95 "," GAMPIW95 "," SGBPIW95 "," AQRAPIW95 ")
 win.graph ()
 boxplot ( NNPIW95 ,SGBPIW95 , GAMPIW95 , AQRAPIW95  , names = PIW95 ,
           horizontal = FALSE , main =" 95% prediction intervals ",
            ylab =" Prediction interval width (m/s) ", col = " blue ")

win.graph ()
par ( mfrow =c(2 ,2))
plot ( density ( NNPIW95 ),xlab =" Prediction interval width (m/s) ", col =" blue ",
         main =" NNPIW95 ")
 plot ( density ( SGBPIW95 ),xlab =" Prediction interval width (m/s) ", col =" blue ",
         main =" SGBPIW95 ")
 plot ( density ( GAMPIW95),xlab =" Prediction interval width (m/s) ", col =" blue ",
         main =" GAMPIW95 ")
 plot ( density ( AQRAPIW95 ),xlab =" Prediction interval width (m/s) ",
      col =" blue",main =" AQRAPIW95 ")
# ######################################################################
# ##################Lagging the Wind Speed 62m ######################################
require(dplyr)
analyticdata$lag1 <-lag(analyticdata$WS_62_mean, 1)
analyticdata$lagD <-lag(analyticdata$WS_62_mean, 144)
analyticdata$lagTD <-lag(analyticdata$WS_62_mean, 288)
analyticdata<-slice(analyticdata, 289:n())
################## Filling Nans ##########################################################
```

```
na.zero <- function (x) {
   x[is.na(x)] <- 0
   return(x)
 }
 analticDATA = na.zero(analticDATA)
################## towards Variable selction via LASSO ###################################
ibrary(glmnet)
#attach(site1analyticdata)
y<-analticDATA$WS_62_mean
#s(y)
x<-as.matrix(data.frame(analticDATA[5:50]))
#s(x)
print(x)
summary(x)

LAsso<-glmnet(x, y=y,alpha=1,family="gaussian")
plot(LAsso)
coef(LAsso)
cv.lasso<-cv.glmnet(y=y,x=x,family="gaussian")
cv.lasso
plot(cv.lasso)
coef(cv.lasso,s="lambda.min")
predict.1<-predict(cv.lasso,newx=x)
predict.1
write.table(predict.1,"~/gam1.forecast2.txt",sep="\t")
m.lasso<-mean((y-predict.1)^2)
m.lasso
#########################################################################
#############################################################################################
## GAM models
################################################
```

```
library(mgcv)

#fit2 <-
gam(WS_62_mean~s(WS_62_stdv)+s(WS_60_mean)+s(WS_60_min)+s(WS_60_max)+s(WS_60_stdv)
+ #s(WS_40_mean)+s(WS_40_min)+s(WS_40_max)+s(WS_40_stdv)+s(WS_20_mean)+s(WS_20_min)
# s(WS_20_max)+s(WS_20_stdv)+s(WS_10_mean)+s(WS_10_min)+s(WS_10_max)+s(WS_10_stdv)
#+s(WD_60_mean)+s(WD_60_min)+s(WD_60_max)+s(WD_60_stdv)+s(WD_20_mean)+s(WD_20_mi
n)  # +s(WD_20_max)+s(WD_20_stdv)+s(Tair_mean)+s(Tair_min)+s(Tair_max)+s(Tair_stdv)   #
+s(Tgrad_mean)+s(Tgrad_min)+s(Tgrad_max)+s(Tgrad_stdv)+s(Pbaro_mean)+s(Pbaro_min)   #
+s(Pbaro_max)+s(Pbaro_stdv)+s(RH_mean)+s(RH_min)+s(RH_max)+s(RH_stdv),
#family=gaussian,data = data_train)# insample

fit2 <-
gam(WS_62_mean~s(WS_62_max)+s(WS_60_mean)+s(WS_60_stdv)+s(WS_40_mean)+s(WD_60_m
ean)+s(WD_60_stdv)+s(Tair_min),     family=gaussian,data = data_train)# insample

summary(fit2)

par(mfrow = c(2,2))

gam.check(fit2)

box()

#############################################################################

library(forecast)

fit.fore <- ts(fit.fore)

ws <- ts(data_test$WS_62_mean)

accuracy(fit.fore,data_test$WS_62_mean)

fit.fore <- predict(fit2, newdata=data_test)

fit.fore

fit.fore <- round(fit.fore,4)

write.table(fit.fore,"~/fGAM.txt",sep="\t")

win.graph()

z <- ts(data_test$WS_62_mean)

f1 <- ts(fit.fore)

plot(z,xlab="Date",lwd=3,ylab="Wind speed (62m)")

lines(f1,col="red", lty=2,lwd=3)

legend("topright",col=c("black","red"), lty=1:2,lwd=3,

     legend=c("Actuals", "Forecasts (GAM)"))

#############################################################################
```

```
### Stochastic GRADIENT BOOSTING#########################################################

#install.packages("caret", dependencies = TRUE)#, quiet = TRUE)

#install.packages("gbm", dependencies = TRUE)#, quiet = TRUE)

library(caret)

library(gbm)

#win.graph(width=5,height=7,pointsize=8)

#fit <- train(WS_62_mean~WS_62_stdv+WS_60_mean+WS_60_min+WS_60_max+WS_60_stdv+
#WS_40_mean+WS_40_min+WS_40_max+WS_40_stdv+WS_20_mean+WS_20_min+WS_20_max+
WS_20_stdv+
#WS_10_mean+WS_10_min+WS_10_max+WS_10_stdv+WD_60_mean+WD_60_min+WD_60_max+
WD_60_stdv+
#WD_20_mean+WD_20_min+WD_20_max+WD_20_stdv+Tair_mean+Tair_min+Tair_max+Tair_stdv
+#Tgrad_mean+Tgrad_min+Tgrad_max+Tgrad_stdv+Pbaro_mean+Pbaro_min+Pbaro_max+Pbaro_st
dv+ #RH_mean+RH_min+RH_max+RH_stdv,data = data_train, method = "gbm")# insample

        fit <- train(WS_62_mean~WS_62_max+WS_60_mean+WS_60_stdv+WS_40_mean+
        WD_60_mean+WD_60_stdv+Tair_min,data = data_train, method = "gbm")# insample

fit

summary(fit)

box()

#warnings()

# Calculate the out-of-sample forecasts, based on the available information

fit1.forecast <- predict(fit, newdata = data_test)

fit1.forecast <- round(fit1.forecast,4)

write.table(fit1.forecast,"~/fSGB.txt",sep="\t")

write.csv(fit1.forecast,file = 'Dset4Model.csv')

?write.csv

fit1.for<-as.data.frame(fit1.forecast)

install.packages("writexl")

library(writexl)

FrSGB <- write_xlsx(fit1.for)

#install.packages("forecast", dependencies = TRUE)

view(tempfile)

library(forecast)

accuracy(fit1.forecast,data_test$WS_62_mean)
```

```
#########################################################################
## Linear QUNATILE REGRESSION AVERAGING ##################################
attach(ForecastL2)
head(ForecastL2)
win.graph()
y <- ts(actual)
plot(y, xlab="Observation number", ylab="Wind speed")
library(quantreg)
qr.wind = rq(actual ~ SGBF+GAMF+NNPred,data= ForecastL2, tau=0.5) #tau = 0.025, 0.5, 0.975
#+ Drift+RanF + Mean+ Snaive+NF
summary.rq(qr.wind,se="boot") # can use se = "nid" or se="ker"
lines(qr.wind$fit, col="red")
fQRA =round(fitted(qr.wind),4)
write.table(fQRA,"~/UL0975.txt",sep="\t") #LL0025, QRA05, UL0975
#########################################################################
## OR  Additive QRA#############################
## THIS OPTION TAKES LONG, I SUGGEST YOU USE THE OPTION ABOVE
#########################################################################
library(qgam)
# Calibrate learning rate on a grid
set.seed(5235)
tun <- tuneLearnFast(form=load~s(fGAM, bs="ad")+s(fAQR, bs="ad")+s(fGAMI,bs="ad")+s(FAQRI,
bs="ad")   ,err = 0.05, qu = 0.5, data = forecasts) #qu = 0.5, 0.025=LL, 0.975=UL
tun
fit1 <-qgam(load~s(fGAM, bs="ad")+s(fAQR, bs="ad")+s(fGAMI,bs="ad")+s(FAQRI, bs="ad"),
        err = 0.05, qu = 0.5, lsig = tun$lsig, data = forecasts)# insample
summary(fit1, se="boot") #se =" ker" " nid" "boot"
lines(fit1$fit, col="red")
#plot(fit1, pages=5)
#check.qgam(fit1, pch=19, cex=.3)
```

```
fQRA = fitted(fit1)
write.table(fQRA,"~/fQRA05.txt",sep="\t")
############### Residual Analysis ##########################################
NNRes = ForecastL2 $ actual - ForecastL2 $ NNPred
 GAMRes = ForecastL2 $ actual - ForecastL2 $ GAMF
 SGBRes = ForecastL2 $ actual - ForecastL2 $ SGBF
 AQRARes = ForecastL2 $ actual - ForecastL2 $ AQRAM
 ## Summary Statistics for forecast errors
 ## ResLSTM , ResSVR , ResFFNN , ResConvex , ResQRA
 library ( e1071 )
 summary ( AQRARes)
 sd( AQRARes )
 skewness ( AQRARes)
 kurtosis ( AQRARes )
# Residual box - pot and density plot
Boxres = c(" NNRes "," GAMRes "," SGBRes "," AQRARes ")
win.graph ()
 boxplot ( NNRes , GAMRes , SGBRes , AQRARes , names = Boxres ,
         horizontal= FALSE , main ="",ylab =" WSResiduals (m/s) ",
         col = " blue ")
win.graph ()
par ( mfrow =c(2 ,2))
plot ( density ( NNRes ),xlab =" Forecast error (m/s) ", col =" blue ",
      main ="NNRes ")
plot ( density ( GAMRes ),xlab =" Forecast error (m/s) ", col =" blue ",
      main =" GAMRes")
plot ( density ( SGBRes ),xlab =" Forecast error (m/s) ", col =" blue ",
      main ="SGBRes ")
plot ( density ( AQRARes ),xlab =" Forecast error (m/s) ", col =" blue ",
      main =" AQRARes")
##############################################################################
```

References

Abhinav, R., Pindoriya, N. M., Wu, J. and Long, C. (2017), 'Short-term wind power forecasting using wavelet-based neural network', *Energy Procedia* **142**, 455–460.

Abuella, M. and Chowdhury, B. (2017), Hourly probabilistic forecasting of solar power, *in* '2017 North American Power Symposium (NAPS)', IEEE, pp. 1–5.

Baghirli, O. (2015), 'Comparison of lavenberg-marquardt, scaled conjugate gradient and bayesian regularization backpropagation algorithms for multistep ahead wind speed forecasting using multilayer perceptron feedforward neural network', *Independent thesis Advanced level (degree of Master), Uppsala University, Disciplinary Domain of Science and Technology, Earth Sciences, Department of Earth Sciences, Sweden* .

Barbosa de Alencar, D., de Mattos Affonso, C., Limão de Oliveira, R., Moya Rodríguez, J., Leite, J. and Reston Filho, J. (2017), 'Different models for forecasting wind power generation: Case study', *Energies* **10**(12), 1976.

Bates, J. M. and Granger, C. W. (1969), 'The combination of forecasts', *Journal of the Operational Research Society* **20**(4), 451–468.

Brown, B. G., Katz, R. W. and Murphy, A. H. (1984), 'Time series models to simulate and forecast wind speed and wind power', *Journal of climate and applied meteorology* **23**(8), 1184–1195.

Buhari, M., Adamu, S. S. et al. (2012), Short-term load forecasting using artificial neural network, *in* 'International Multi Conference of Engineers and Computer Scientists Hongkong', pp. 14–16.

Chen, Q. and Folly, K. A. (2018), Comparison of three methods for short-term wind power forecasting, *in* '2018 International Joint Conference on Neural Networks (IJCNN)', IEEE, pp. 1–8.

Cho, H., Goude, Y., Brossat, X. and Yao, Q. (2013), 'Modeling and forecasting daily electricity load curves: a hybrid approach', *Journal of the American Statistical Association* **108**(501), 7–21.

Crate, S. A. and Nuttall, M. (2016), Introduction: Anthropology and climate change, *in* 'Anthropology and climate change', Routledge, pp. 9–36.

Craven, P. and Wahba, G. (1978), 'Smoothing noisy data with spline functions', *Numerische mathematik* **31**(4), 377–403.

Fasiolo, M., Goude, Y., Nedellec, R. and Wood, S. N. (2017), 'Fast calibrated additive quantile regression', *arXiv preprint arXiv:1707.03307* .

Ferreira, M., Santos, A. and Lucio, P. (2019), 'Short-term forecast of wind speed through mathematical models', *Energy Reports* **5**, 1172–1184.

Friedman, J. H. (2001), 'Greedy function approximation: a gradient boosting machine', *Annals of statistics* pp. 1189–1232.

Friedman, J. H. (2002), 'Stochastic gradient boosting', *Computational statistics & data analysis* **38**(4), 367–378.

Friedman, J., Hastie, T. and Tibshirani, R. (2001), *The elements of statistical learning*, Vol. 1, Springer series in statistics New York.

Gaillard, P., Goude, Y. and Nedellec, R. (2016), 'Additive models and robust aggregation for gefcom2014 probabilistic electric load and electricity price forecasting', *International Journal of forecasting* **32**(3), 1038–1050.

Giebel, G. and Kariniotakis, G. (2017), Wind power forecasting—a review of the state of the art, *in* 'Renewable Energy Forecasting', Elsevier, pp. 59–109.

Goude, Y., Nedellec, R. and Kong, N. (2014), 'Local short and middle term electricity load forecasting with semi-parametric additive models', *IEEE transactions on smart grid* **5**(1), 440–446.

Hastie, T. and Tibshirani, R. (1990), 'Generalized additive models, volume 43 crc press'.

Hastie, T., Tibshirani, R., Friedman, J. and Franklin, J. (2005), 'The elements of statistical learning: data mining, inference and prediction', *The Mathematical Intelligencer* **27**(2), 83–85.

Hastie, T., Tibshirani, R. and Wainwright, M. (2015), *Statistical learning with sparsity: the lasso and generalizations*, Chapman and Hall/CRC.

Haykin, S. (1994), *Neural networks: a comprehensive foundation*, Prentice Hall PTR.

Heaton, J. (2008), *Introduction to neural networks with Java*, Heaton Research, Inc.

Hoeting, J. A., Madigan, D., Raftery, A. E. and Volinsky, C. T. (1999), 'Bayesian model averaging: a tutorial', *Statistical science* pp. 382–401.

Hunga, E. (2018), 'Short-term hourly load forecasting in south africa using neural networks', *MSc. Dissertation, School of Statistics and Actuarial Science University of Witwatersrand, Johannesburg, SA* .

Jones, K. and Wrigley, N. (1995), 'Generalized additive models, graphical diagnostics, and logistic regression', *Geographical Analysis* **27**(1), 1–18.

Landberg, L. (1999), 'Short-term prediction of the power production from wind farms', *Journal of Wind Engineering and Industrial Aerodynamics* **80**(1-2), 207–220.

Lim, M. and Hastie, T. (2015), 'Learning interactions via hierarchical group-lasso regularization', *Journal of Computational and Graphical Statistics* **24**(3), 627–654.

Lin, T. C. (2007), 'Application of artificial neural network and genetic algorithm to forecasting of wind power output'.

Liu, B., Nowotarski, J., Hong, T. and Weron, R. (2015), 'Probabilistic load forecasting via quantile regression averaging on sister forecasts', *IEEE Transactions on Smart Grid* **8**(2), 730–737.

Liu, H. (2008), 'Generalized additive model', *Department of Mathematics and Statistics University of Minnesota Duluth: Duluth, MN, USA* .

Maciejowska, K., Nowotarski, J. and Weron, R. (2016), 'Probabilistic forecasting of electricity spot prices using factor quantile regression averaging', *International Journal of Forecasting* **32**(3), 957–965.

Mbuvha, R. (2017), 'Bayesian neural networks for short term wind power forecasting', *Independent thesis Advanced level (degree of Master), KTH, School of Computer Science and Communication (CSC), Sweden* .

Morina, M., Grimaccia, F., Leva, S. and Mussetta, M. (2016), Hybrid weather-based ann for forecasting the production of a real wind power plant, *in* '2016 International Joint Conference on Neural Networks (IJCNN)', IEEE, pp. 4999–5005.

Mpfumali, P. (2019), 'Probabilistic solar power forecasting using partially linear additive quantile regression models: an application to south african data', *MSc. Dissertation, Department of Statistics School of Mathematical and Natural Sciences University of Venda, Thohoyandou SA* .

Mpfumali, P., Sigauke, C., Bere, A. and Mulaudzi, S. (2019), 'Day ahead hourly global horizontal irradiance forecasting—application to south african data', *Energies* **12**(18), 3569.

Nowotarski, J. and Weron, R. (2015), 'Computing electricity spot price prediction intervals using quantile regression and forecast averaging', *Computational Statistics* **30**(3), 791–803.

Odhiambo, N. M. (2009), 'Electricity consumption and economic growth in south africa: A trivariate causality test', *Energy Economics* **31**(5), 635–640.

Olaofe, Z. O. (2013), Wind energy generation and forecasts: a case study of Darling and Vredenburg sites, PhD thesis, University of Cape Town.

Pierrot, A. and Goude, Y. (2011), 'Short-term electricity load forecasting with generalized additive models', *Proceedings of ISAP power* **2011**.

Pinson, P. et al. (2013), 'Wind energy: Forecasting challenges for its operational management', *Statistical Science* **28**(4), 564–585.

Pinto, M. d. O. et al. (2013), 'Fundamentos de energia eólica', *Rio de Janeiro: LTC* .

Plan, Y. and Vershynin, R. (2016), 'The generalized lasso with non-linear observations', *IEEE Transactions on information theory* **62**(3), 1528–1537.

Pya, N. and Wood, S. N. (2016), 'A note on basis dimension selection in generalized additive modelling', *arXiv preprint arXiv:1602.06696* .

Schmidhuber, J. (2015), 'Deep learning in neural networks: An overview', *Neural networks* **61**, 85–117.

Sebitosi, A. B. and Pillay, P. (2008), 'Grappling with a half-hearted policy: The case of renewable energy and the environment in south africa', *Energy policy* **36**(7), 2513–2516.

Shadish, W. R., Zuur, A. F. and Sullivan, K. J. (2014), 'Using generalized additive (mixed) models to analyze single case designs', *Journal of school psychology* **52**(2), 149–178.

Shen, Y., Wang, X. and Chen, J. (2018), 'Wind power forecasting using multi-objective evolutionary algorithms for wavelet neural network-optimized prediction intervals', *Applied Sciences* **8**(2), 185.

Sigauke, C. (2017), 'Forecasting medium-term electricity demand in a south african electric power supply system', *Journal of Energy in Southern Africa* **28**(4), 54–67.

Sigauke, C., Nemukula, M. and Maposa, D. (2018), 'Probabilistic hourly load forecasting using additive quantile regression models', *Energies* **11**(9), 2208.

Sun, X., Wang, Z. and Hu, J. (2017), 'Prediction interval construction for byproduct gas flow forecasting using optimized twin extreme learning machine', *Mathematical Problems in Engineering* **2017**.

Taieb, S. B., Huser, R., Hyndman, R. J. and Genton, M. G. (2016), 'Forecasting uncertainty in electricity smart meter data by boosting additive quantile regression', *IEEE Transactions on Smart Grid* **7**(5), 2448–2455.

Thomas, T. and Cheriyan, E. P. (2012), Wind energy system for a laboratory scale microgrid, *in* '2012 IEEE Students' Conference on Electrical, Electronics and Computer Science', IEEE, pp. 1–5.

Verma, S. M., Reddy, V., Verma, K. and Kumar, R. (2018), Markov models based short term forecasting of wind speed for estimating day-ahead wind power, *in*

'2018 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS)', IEEE, pp. 31–35.

Wang, L., Wang, Z., Qu, H. and Liu, S. (2018), 'Optimal forecast combination based on neural networks for time series forecasting', *Applied Soft Computing* **66**, 1–17.

Wang, Y., Zhang, N., Tan, Y., Hong, T., Kirschen, D. S. and Kang, C. (2018), 'Combining probabilistic load forecasts', *IEEE Transactions on Smart Grid* .

Wood, S. N. (2001), 'mgcv: Gams and generalized ridge regression for r', *R news* **1**(2), 20–25.

Yan, J., Liu, Y., Han, S., Gu, C. and Li, F. (2014), 'A robust probabilistic wind power forecasting method considering wind scenarios'.

Ying, C., Yuyu, D., Jie, D., Zhibao, C., Rongfu, S. and Hai, Z. (2012), 'A statistical approach of wind power forecasting for grid scale', *AASRI Procedia* **2**, 121–126.

Zhang, G., Patuwo, B. E. and Hu, M. Y. (1998), 'Forecasting with artificial neural networks:: The state of the art', *International journal of forecasting* **14**(1), 35–62.

Zhang, Y., Wang, J. and Wang, X. (2014), 'Review on probabilistic forecasting of wind power generation', *Renewable and Sustainable Energy Reviews* **32**, 255–270.

Zhu, B., Chen, M.-y., Wade, N. and Ran, L. (2012), 'A prediction model for wind farm power generation based on fuzzy modeling', *Procedia Environmental Sciences* **12**, 122–129.