AN APPROACH TO PERFORM COMPREHENSIVE WIND TURBINE PERFORMANCE ANALYTICS BY MEANS OF MACHINE LEARNING (SCI-KIT LEARN)

by

MUHAMMAD ZUBAIR

FAISAL IBNE MASUD

MAHAMMAD YEAKUB

BACHELOR OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING

Department of Electrical and Electronic Engineering
INTERNATIONAL ISLAMIC UNIVERSITY CHITTAGONG

JUNE 2022

AN APPROACH TO PERFORM COMPREHENSIVE WIND TURBINE PERFORMANCE ANALYTICS BY MEANS OF MACHINE LEARNING (SCI-KIT LEARN)

by

MUHAMMAD ZUBAIR
FAISAL IBNE MASUD
MAHAMMAD YEAKUB

A thesis
submitted as partial fulfilment of the requirement for the degree of

BACHELOR OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING

Department of Electrical and Electronic Engineering
INTERNATIONAL ISLAMIC UNIVERSITY CHITTAGONG

JUNE 2022

CERTIFICATE OF APPROVAL

This thesis entitled as "**An Approach To Perform Comprehensive Wind Turbine Performance Analytics By Means Of Machine Learning (Sci-Kit Learn)**" submitted by **Muhammad Zubair**, bearing Matric ID. **ET173046**, **Faisal Ibne Masud**, bearing Matric ID. **ET173065,** and **Mahammad Yeakub**, bearing Matric ID. **ET173042** of session **Autumn 2022** to the Department of Electrical and Electronic Engineering, International Islamic University Chittagong, has been accepted as satisfactory in partial fulfilment of the requirements for the degree of Bachelor of Science in Engineering and approved for the examination held on **30th June 2022**.

____

Supervisor
Dr. Md. Aasim Ullah
Assistant Professor
Department of Electrical and Electronic Engineering
International Islamic University Chittagong.

DECLARATION

It is hereby declared that this work has been done by us and no portion of the work contained in this project has been submitted elsewhere for the award of any degree or diploma.

Muhammad Zubair

Faisal Ibne Masud

Mahammad Yeakub

ACKNOWLEDGMENT

All praises and thanks to Almighty Allah (SWT) the Lord of the world, the most beneficent, most merciful for helping us to accomplish this work.

We would also thank our supervisor Dr. Md. Aasim Ullah, Assistant Professor of Department of Electrical and Electronic Engineering, (IIUC) for helping us get started by giving the basic ideas and concepts we required for the project, continuous guidance and support.

Moreover, we are very much thankful to the teachers for their virtuous support and motivation. We are extremely grateful to my parents for their love, prayers, continuous support, and giving me mental strength. We are also thankful to my siblings and friends for their support and inspiration.

Authors

ABSTRACT

With the present day's emphasis on sustainable and secure energy supply, wind power sector is growing rapidly all over the world. Along with the rapid expansion of the wind turbine sector, the wind turbine industry is also growing. Understanding the power response of these systems to the variations in wind velocity is essential for the optimal selection and efficient management of these turbines. This is defined by the power analysis and prediction of wind turbines output. Previous approaches such as mechanical aspects of a turbine are used to work with wind turbine power output. However, these techniques are not capable of analyzing big dataset with hefty number of parameters. In this regard, Machine Learning is a handy tool to analyze a dataset and design a model that can perform prediction related to turbines parameters. Python is one of the Open Source convenient tools to implement ML tasks. In this thesis work, we proposed Scikit Learn based Machine Learning models, which is based on Python for the power analysis. Three different machine learning methods such as XGBoost, LightGBM, Catboost were used for the modeling. The comprehensive dataset based on Western Wind Firms are being used to design the analysis model. The accuracies of these models are validated by estimating the error between the model output and the field observations from these turbines from the comprehensive dataset.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Most of the countries around the world are facing huge environmental impact, and the most promising solution to mitigate these is the use of renewable energy, especially wind power. Though, the use of offshore wind energy is rapidly increasing to meet the elevating electricity demand. The researchers and policymakers have become aware of the importance of providing near accurate prediction of output power. Wind energy is tied to variabilities of weather patterns, especially wind speed, which are irregular in climates with erratic weather conditions. In recent years, the wind energy sector increase in strength. For example, the installed wind energy capacity in the European Union (EU) is 142 GW [1]. In the year 2019 651 GW wind energy was installed in world [2]. The intermittency of wind speed introduces challenges to the prediction of wind power operation during energy integration. This results in challenges associated with planning and regulation capabilities associated with sudden wind speed variations, which impacts on the reliability of power system predictions. Wind power generation and reliability planning rely on fast and robust wind speed prediction and response to system dynamics for better wind power prediction [3]. The global energy report shows that power generation from the wind rose to 54.6 gigawatts (GW) of installed capacity in 2018. China and the USA are leading with installed capacity. Countries like Germany and India are showing a strong appetite for wind energy generation due to useful wind speed prediction capability [4].The role of wind power prediction is becoming increasingly crucial, while the wind penetration rate is continuously growing. Every power system has a reasonable capability to adapt demand changes as the demand estimation has never been entirely accurate. When the penetration rate is relatively low, power systems do not have to pay too much attention to the variance of wind power supply. Machine learning is a field of computer science that focuses on improving the performance of the program by itself with experience. In this technique, the machine is not told how to solve the problem explicitly; instead, the experience is given to the device as different inputs, and the outputs are typically a model that can address future issues of the same kind.

1.2 Thesis Overview

The complete procedure of machine learning includes several steps. First, past experience is usually gathered for the training in the later stage. At that point, the type of a unique target function is resolved, which portrays the relationship between inputs and outputs. From that point onward, an machine learning model is chosen to estimate the target function. At last, a fitting calculation is utilized to assemble the model from the training models. The elimination of variability of the wind is not possible, but by using machine learning, we can make wind power sufficiently more predictable and valuable. This will also help bring greater data rigor to wind farm operations, as these techniques can help wind farm operators make faster, more data-driven, and smarter assessments of how their power output can meet electricity demand. The aim of this research work to predict wind turbine output power using machine learning techniques. Since there is a very limited study has been done on the wind power prediction using machine learning it is greater interest to us how well it performs in this field.

1.3 Thesis Objectives

The objectives of this work is as follows:

- To scrape data for Wind Turbine Power Generation and include as many as data cell possible.
- To analyse the data by using Python Data Science tools thoroughly.
- To use Sci-kit Learn for comprehensive data analysis for calculating power analytics based on the data.

1.4 Report Outline

Five chapters has covered in the course of design and development of this thesis. The chapters and their contents are as follows:

- Chapter 1 is the introductory chapter that gives the overview, motivation and objective of the thesis work.
- Chapter 2 is literature review. Previous work related of this thesis has discussed in this chapter.

- Chapter 3 deals with methodology of the thesis. In this chapter all the tools and required libraries are being discussed.
- Chapter 4 deals with the system design of the thesis. In this chapter Block diagram, Flow chart and Programming of the project has discussed.
- Chapter 4 deals with the system implementation and results, Objective verification and system specification.
- Finally, the summary of this project has discussed in detail in chapter 5. The limitation of the thesis, advantage and future development has discussed on this topic.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In every wind turbine farm the Supervisory control and data acquisition is already installed which collect the data from different sensors. Wind farm consist of several wind turbines to generate power. Then the farm saves these data with respect to several time period. Later these data are used for Machine Learning and other analytics for understanding the problem set and generate a detailed report. This section will review the different studies done so far to predict wind turbine output power.

2.2 Review of Previous Works

Few of the previous work based on wind turbine power analysis has described below.

2.2.1 Forecasting of Wind Turbine Output Power Using Machine learning

In this paper, authors predicted the output power of the wind turbines using the random forest regressor algorithm [5]. The SCADA data is collected for two years from a wind farm located in France. The model is trained using the data from 2017 [6]. The wind direction, wind speed and outdoor temperature are used as input parameters to predict output power. authors test our model for two different capacity factors. The estimated mean absolute errors for the proposed model in this study were 3.6% and 7.3% for and 0.2 capacity factors, respectively [7]. The proposed model in this study offers an efficient method to predict the output power of wind turbine with preferably low error. The data obtain from SCADA is preprocessing in the second step before the data analysis. Thirdly the useful features are selected from the data. In fourth step machine learning model is train using the data. In last step the wind turbine power is predicted. The model accuracy of wind turbine is highly effected by inaccurate SCADA data caused by null entries and irregular operation. Therefore it is really important to pre-process this data which will remove these confusing entries before further analyses. The SCADA data collected have mismatch in date and time which is fixed using Python. The data is further cleaned by applying the cut in and cut out wind speed limit. The data contain some negative output power which is unphysical value which is removed before further analysis. The pitch angle of wind turbine also consist of some irregular value which normally is very low or slightly negative so before further analysis the data is excluded. The features selected for this model are wind turbine generator ambient (external) conditions. The parameters selected for this research are absolute wind direction (Wa), wind speed (Ws) and outdoor temperature (Ot). The average, minimum, maximum and standard deviation values of all four features are consider while analysis. The separation of test and train set is done in continuous ways in order to avoid over-fitting risk. Random Forest Regressor is used to predict the output power of wind turbine. This model have lot of advantages over other existing models in literature including low over-fitting tendency, simple and fast to train. The capacity factor is a design parameter which is used to evaluate the performance of wind farm. The result of this paper is a powerful method for precise and efficient wind power prediction using machine learning. The SCADA data is collected from 2 MW wind turbine in the wind farm in France from 1st January 2017 to 29th January 2018 were used to forecast the output power of the wind turbines [8]. The data is pre-processed before the analysis for better performance. Wind turbine output power for the year 2018 is forecast using the 2017 measurements.

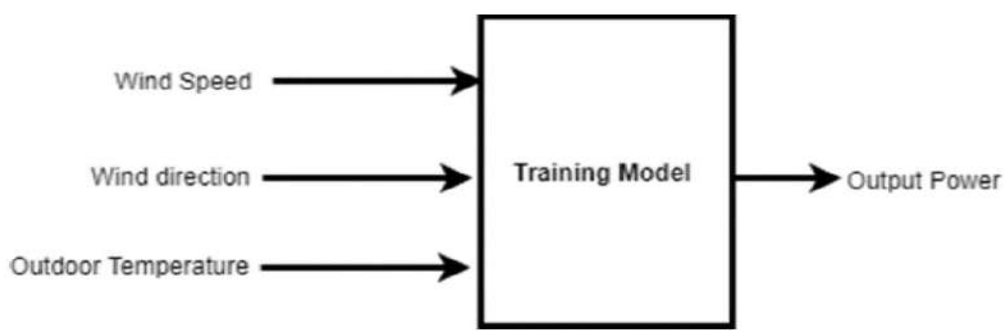

Fig. 2.1 Inputs and Output of the Model Approached

The use of appropriate features helps to improve the prediction. authors used the absolute wind direction, wind speed and outdoor temperature to predict output power . authors estimate the wind turbine performance by the capacity factor for real power output and annual effected power output. Random forest regressor machine learning model to predict the output power. The estimated mean absolute error for our proposed model for the capacity factor 0.4 and 0.2 is 3.6 % and 7.3 % respectively. These results are promising in uncertain and unpredictable wind forecast. The method offers an efficient and comfortable balancing of a preferably low prediction error.

2.2.2 Intelligent Models for the Power Curves of Small Wind Turbines

In this paper, authors propose nonparametric methods for defining the power curves of small wind turbines [9]. For demonstrating the method, two small wind turbines of 50 kW and 2.5 kW rated capacities, installed at 2 different sites, are chosen. Initially, authors compare the generation data of these turbines with the manufacturer's power curves to show the disparities between the manufacturers claim and the real field performance. Further, power curve models for these turbines are developed using artificial intelligence by using machine learning techniques like Artificial Neural Network (ANN), Support Vector Machines (SVM), k-Nearest Neighbors (KNN) and Gradient Boosting Machines (GBM). The accuracies of these models are tested using the field performance data of the turbines and presented in the paper. In this paper, authors propose a nonparametric approach for modeling the power curves. These artificially intelligent models try to learn the velocity-power variations from the manufacture's power curve, thereby developing the capability to estimate the power produced by the wind turbine at any given wind velocities. The performance data of both the turbine were divided into two groups, the first being used for training the model and the second for testing the model. Artificial Neural Network (ANN) function is basically constituted by its input, hidden and output layers. Each layer consists of several interconnected identical neurons, which forms the entire network. The signals from the input layer, after giving a calculated weightage, is passed to a nonlinear function in the hidden layer. Model for T1 turbine has a single input layer and two hidden layers. The first hidden layer consists of three nodes, whereas the second layer has two nodes. In case of turbine T2, there are three layers, which consist of three, two, and four nodes respectively. These layers are activated by tangent hyperbolic transfer function with learning rate of 0.01 and the maximum number of steps involved is 1e6. In both the cases, the optimum number of layers and nodes were chosen through iterations to minimize the errors [10]. In this study, the kernel functions like linear, polynomial, Gaussian, radial and sigmoid were tried with the dataset, and the polynomial kernel in which showed the lowest error was chosen. The K-Nearest Neighbor is one of the simplest nonparametric method which is an instance based model where the operation is locally approximated [11]–[14]. Initially, the instances were introduced in a multidimensional space, followed by dividing the entire dataset into training and testing. Given a test instance, a distance metric is computed between the experiment and from all the training datasets using the Minkowski procedure, by which the k-Nearest Neighbors are chosen from the training experience. It has been reported that KNN models are quite efficient in power forecasting in the wind industry [15]–[18]. The optimal values for k and d have been chosen in such a way that the output errors be minimized. Among the models developed and tested, ANN based approach showed the highest accuracy of 84 per cent and 88 per cent respectively for the turbines. Capability of artificial intelligent modeling techniques in modeling the velocity-power behavior of wind turbines is established under this study.

2.2.3 Condition Monitoring Of Wind Turbines Based On Extreme Learning Machine

This paper adopts extreme learning machine (ELM) algorithms to achieve condition monitoring of wind turbines based on a model-based condition monitoring approach [19]. Compared with the traditional gradient-based training algorithm widely used in the single-hidden layer feed forward neural network, ELM can randomly choose the input weights and hidden biases and need not be tuned in the training process. Therefore, ELM algorithm can dramatically reduce learning time. Models are identified using supervisory control and data acquisition (SCADA) data acquired from an operational wind farm, which contains data of the temperature of gearbox oil sump, gearbox oil exchange and generator winding. The results show that the proposed method can efficiently identify faults of wind turbines. A number of supervised learning methods have been applied in model-based condition monitoring system of wind turbines, such as artificial neural networks (ANNS) [20], [21]. However, ANNS have suffered a drawback in the real-time implementation [22], [23], which engineering applications. limits their Extreme learning machine (ELM) is considered in this paper due to its extremely fast learning speed [24]. A single-hidden layer feed forward neural network (SLFN) has been widely used in many fields such as mode recognition and state prediction, because of its efficient learning skills [25]. However, SLFN always selects a gradient-based neural network algorithm as its training algorithm. The traditional gradient-based training algorithms have some disadvantages such as trapping at local minima, the overtraining, and the high computing burdens, which causes longer training time of the SLFN during the learning process [26]. As a relatively new technique, the ELM can avoid this drawback [8-9]. Compared with the traditional gradient-based training algorithms, ELM randomly chooses the input weights and hidden biases and needs not be tuned in the training process. Thus, ELM algorithm features an extremely faster learning speed than most algorithms such as popular learning back-propagation, and thus dramatically reduce learning time. Furthermore, if the chosen activation function is infinitely differentiable, the ELM can identify distinct samples exactly with zero error under the condition of equal number of hidden-layer neurons and distinct samples processed in the ANN. Supervisory control and data acquisition (SCADA) system is an industrial automation control system, which has been widely used in many industries, including oil and gas, metallurgy, manufacturing, railway system, transportation, energy, and power systems. Modern SCADA systems work relying on multiple hardware and software elements and IT technologies to monitor, gather, and process data.

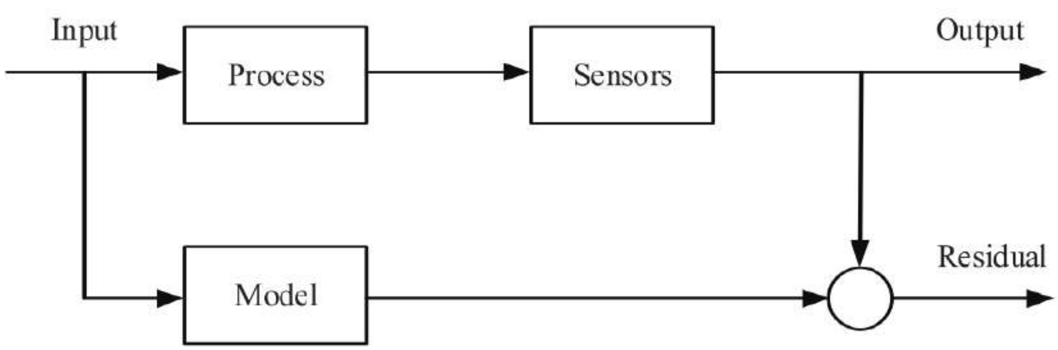

Fig 2.2 Schematic Diagram of Model-Based Condition Monitoring System

In power systems, SCADA is a mature technology and it is used for data collection, device control, parameter adjustment, signal alarm generation and other functions. SCADA data used in this paper were obtained from an operational wind farm. The use of actual operational data of wind turbines is a good method in order to demonstrate the effectiveness of the proposed algorithms. The data cover 12 months' duration and consist of 128 parameters that contain various temperatures, pressures, vibrations, power outputs, wind speed and digital control signals. In order to reduce the amount of data gathered from the operating wind turbines, SCADA data are usually sampled at 10 minute interval. This paper has presented ELM algorithm that has been used in condition monitoring of wind turbines. SCADA data obtained from an operational wind farm, including the temperatures of gearbox oil exchange, gearbox oil sump and generator winding t, are used to verify the effectiveness of the proposed method. Models developed from SCADA data have been used to identify faults in gearbox and generator winding in the turbines. The results have shown that differences between actual output signals and the model predicting signals are caused by a gearbox fault and a generator winding fault. Consequently, the proposed method can provide an early warning of the impending component failure.

CHAPTER 3

METHODOLOGY

3.1 Introduction

Wind energy, being a clean, and sustainable energy resource, has increasing contributions in the modern power systems in the recent years. For example, with an addition of 63GW in 2015, the world's cumulative wind capacity has reached up to 433 GW. With these installations, wind energy today meets 3.7% of the Global electricity demand [27]–[29]. The future projections indicate that, by 2030, the global cumulative installed capacity would reach 2000 GW, contributing 16.7%-18.8% of the World's power demand [30]. Due to the cubic velocity-power relationship, the energy produced from a wind turbine is highly sensitive to the strength of the wind available at the site, where it is installed. As wind is stochastic in nature, its magnitude and direction changes frequently, which in turn result in drastic changes in the turbine output. These frequent and uncertain changes pose significant challenges to wind farm operators and power system managers in integrating the intermittent wind source to the grid. Hence, an understanding on the ramping behaviour of wind turbines, under fluctuating wind conditions, is essential for the efficient and effective management of the wind integrated power grids. For example, to keep the balance between demand and supply, power system operators should have an estimate on the expected contribution from wind on an hourly basis, so that other generating options can be kept ready for ensuring adequate power supply.

3.2 Wind Turbine Parameters

Wind energy is globally becoming important source of electricity. Tower holding the wind turbine is the critical element. The invention is about deciding the particular type of tower for various sites. With the help of innovation the most suitable tower from tubular tower, lattice tower, hybrid tower and guyed tower can be selected. In the set up models of tubular tower, lattice tower, hybrid tower and guyed towers are available. Various wind generators can be tested on these towers for output power and power quality parameters. The parameters that are related to wind turbine power generation are being described below.

3.2.1 Active Power

Active power is the power which is actually consumed or utilized in an AC Circuit is called True power or Active Power or real power. It is measured in kilowatt (kW) or MW. It is the actual outcomes of the electrical system which runs the electric circuits or load. Active power, P, is also commonly referred to as the average power, real power, or true power. It represents useful power expended by loads to perform real work, that is, to convert electric energy to other forms of energy [31]–[35]. Real work performed by an incandescent light bulb is to convert electric energy into light and heat. In electric power, real work is performed for the portion of the current ihat is in phase with the voltage. No real work will result, from the portion where the current is not in phase with the voltage. Real work performed by an incandescent light bulb is to convert electric energy into light and heat. In electric power, real work is performed for the portion of the current ihat is in phase with the voltage. No real work will result, from the portion where the current is not in phase with the voltage. The active power is the rate at which energy is expended, dissipated or consumed by the load, and is measured in units of watts (W). P can be computed by averaging the product of the instantaneous voltage and current.

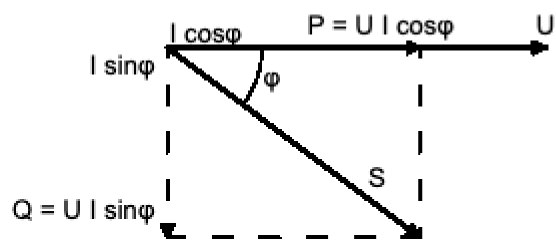

Fig 3.1 Active Power Triangle

The ratio of active power to apparent power in a circuit is called the power factor. For two systems transmitting the same amount of active power, the system with the lower power factor will have higher circulating currents due to energy that returns to the source from energy storage in the load. These higher currents produce higher losses and reduce overall transmission efficiency. A lower power factor circuit will have a higher apparent power and higher losses for the same amount of active power. The power factor is 1.0 when the voltage and current are in phase. It is zero when the current leads or lags the voltage by 90 degrees [36], [37]. When the voltage and current are 180 degrees out of phase, the power factor is negative one, and the load is feeding energy into the source (an example would be a home with solar cells on the roof that feed power into the power grid when the sun is shining). Power factors are usually stated as "leading" or "lagging" to show the sign of the phase angle of current with respect to voltage. Voltage is designated as the base to which current angle is compared, meaning that current is thought of as either "leading" or "lagging" voltage. Where the waveforms are purely sinusoidal, the power factor is the cosine of the phase angle between the current and voltage sinusoidal waveforms [38]–[40]. Equipment data sheets and nameplates will often abbreviate power factor for this reason. n a direct current circuit, the power flowing to the load is proportional to the product of the current through the load and the potential drop across the load. Energy flows in one direction from the source to the load. In AC power, the voltage and current both vary approximately sinusoidally. When there is inductance or capacitance in the circuit, the voltage and current waveforms do not line up perfectly. The power flow has two components – one component flows from source to load and can perform work at the load; the other portion, known as reactive power, is due to the delay between voltage and current, known as phase angle, and cannot do useful work at the load. It can be thought of as current that is arriving at the wrong time. To distinguish reactive power from active power, it is measured in units of "volt-amperes reactive", or var. These units can simplify to watts but are left as var to denote that they represent no actual work output. Energy stored in capacitive or inductive elements of the network gives rise to reactive power flow. Reactive power flow strongly influences the voltage levels across the network. Voltage levels and reactive power flow must be carefully controlled to allow a power system to be operated within acceptable limits. A technique known as reactive compensation is used to reduce apparent power flow to a load by reducing reactive power supplied from transmission lines and providing it locally. For example, to compensate an inductive load, a shunt capacitor is installed close to the load itself. This allows all reactive power needed by the load to be supplied by the capacitor and not have to be transferred over the transmission lines. This practice saves energy because it reduces the amount of energy that is required to be produced by the utility to do the same amount of work. Additionally, it allows for more efficient transmission line designs using smaller conductors or fewer bundled conductors and optimizing the design of transmission towers. Stored energy in the magnetic or electric field of a load device, such as a motor or capacitor, causes an offset between the current and the voltage waveforms. A capacitor is a device that stores energy in the form of an electric field. As current is driven through the capacitor, charge build-up causes an opposing voltage to develop across the capacitor. This voltage increases until some maximum dictated by the capacitor structure. In an AC network, the voltage across a capacitor is constantly changing. The capacitor opposes this change, causing the current to lead the voltage in phase. Capacitors are said to "source" reactive power, and thus to cause a leading power factor. Induction machines are some of the most common types of loads in the electric power system today. These machines use inductors, or large coils of wire to store energy in the form of a magnetic field [41], [42]. When a voltage is initially placed across the coil, the inductor strongly resists this change in a current and magnetic field, which causes a time delay for the current to reach its maximum value. This causes the current to lag behind the voltage in phase. Inductors are said to "sink" reactive power, and thus to cause a lagging power factor. Induction generators can source or sink reactive power, and provide a measure of control to system operators over reactive power flow and thus voltage. Because these devices have opposite effects on the phase angle between voltage and current, they can be used to cancel out each other's effects. This usually takes the form of capacitor banks being used to counteract the lagging power factor caused by induction motors.

3.2.2 Wind Speed

In meteorology, wind speed, or wind flow speed, is a fundamental atmospheric quantity caused by air moving from high to low pressure, usually due to changes in temperature. Wind speed is now commonly measured with an anemometer. Wind speed affects weather forecasting, aviation and maritime operations, construction projects, growth and metabolism rate of many plant species, and has countless other implications [43], [44]. Note that wind direction is usually almost parallel to isobars (and not perpendicular, as one might expect), due to Earth's rotation. Wind speed is affected by a number of factors and situations, operating on varying scales (from micro to macro scales). These include the pressure gradient, Rossby waves and jet streams, and local weather conditions. There are also links to be found between wind speed and wind direction, notably with the pressure gradient and terrain conditions. Pressure gradient is a term to describe the difference in air pressure between two points in the atmosphere or on the surface of the Earth. It is vital to wind speed, because the greater the difference in pressure, the faster the wind flows (from the high to low pressure) to balance out the variation. The pressure gradient, when combined with the Coriolis effect and friction, also influences wind direction. Rossby waves are strong winds in the upper troposphere. These operate on a global scale and move from West to East (hence being known as Westerlies). The Rossby waves are themselves a different wind speed from what we experience in the lower troposphere. Local weather conditions play a key role in influencing wind speed, as the formation of hurricanes, monsoons and cyclones as freak weather conditions can drastically affect the flow velocity of the wind.

3.3.3 Theoretical Power Curve

Power curve of a wind turbine depicts the relationship between output power and hub height wind speed and is an important characteristic of the turbine. Power curve aids in energy assessment, warranty formulations, and performance monitoring of the turbines [45]–[47]. With the growth of wind industry, turbines are being installed in diverse climatic conditions, onshore and offshore, and in complex terrains causing significant departure of these curves from the warranted values. Accurate models of power curves can play an important role in improving the performance of wind energy based systems.

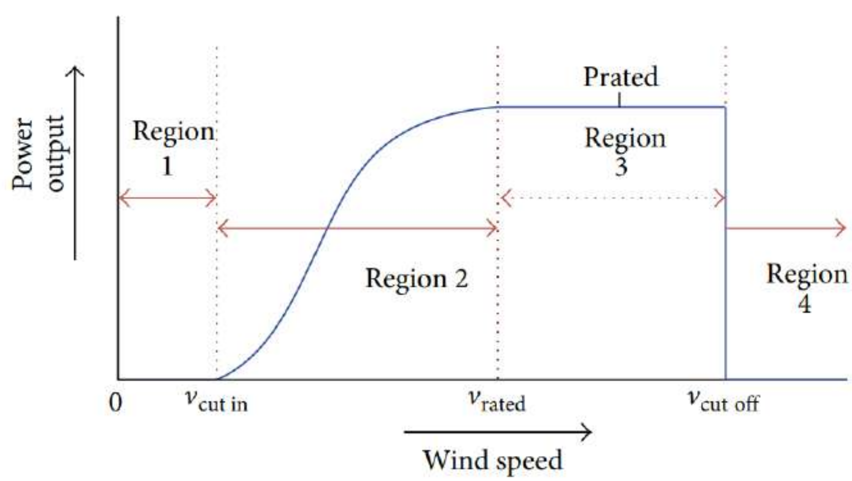

Fig 3.2 Typical power curve of a pitch regulated wind turbine.

Wind energy has emerged as a promising alternative source for overcoming the energy crisis in the world. Wind power based energy is one of the most rapidly growing areas among the renewable energy sources and will continue to do so because of the growing concern about sustainability and emission reduction requirements. The uncertain nature of wind and high penetration of wind energy in power systems are a big challenge to the reliability and stability of these systems. To make wind energy a reliable source, accurate models for predicting the power output and performance monitoring of wind turbines are needed. The power curve of a WT indicates its performance. Accurate models of power curves are important tools for forecasting of power and online monitoring of the turbines. A number of methods have been proposed in various works to model the wind turbine power curve. These methods which use data from manufacturers' specifications and actual data from the wind farms have been utilized by many researchers in various wind power applications [48]–[50]. The literature reviewed reveals that appropriate selection of power curve models can help in improved performance of wind energy based systems. Power curves are supplied by the manufacturers in a tabular or graphical form. However, a generic equation which represents this curve accurately is required in various problems of wind power systems. Derivation of an appropriate function to describe the actual shape of the curve is a very important task. However, the manufacturer's curves are created under standard conditions therefore they may not represent the realistic conditions of the site under consideration. The turbine performance at the wind farms is also not ideal due to wear and tear and aging of turbines. Another method to model the power curves is to derive them using the actual data of wind speed and power measured from the turbines [4]. The data of wind turbines collected by the SCADA (supervisory control and data acquisition) system can be utilized for this purpose. This method can incorporate the actual conditions at the wind farms, thus providing better accuracy in power prediction.

3.2.4 Wind Direction

Wind direction is generally reported by the direction from which it originates. For example, a north or northerly wind blows from the north to the south [51]. The exceptions are onshore winds (blowing onto the shore from the water) and offshore winds (blowing off the shore to the water). Wind direction is usually reported in cardinal (or compass) direction, or in degrees. Consequently, a wind blowing from the north has a wind direction referred to as 0° (360°); a wind blowing from the east has a wind direction referred to as 90°, etc. variety of instruments can be used to measure wind direction, such as the windsock and wind vane. Both of these instruments work by moving to minimize air resistance. The way a weather vane is pointed by prevailing winds indicates the direction from which the wind is blowing. The larger opening of a windsock faces the direction that the wind is blowing from; its tail, with the smaller opening, points in the same direction as the wind is blowing [52], [53]. Modern instruments used to measure wind speed and direction are called anemometers and wind vanes, respectively. These types of instruments are used by the wind energy industry, both for wind resource assessment and turbine control. When a high measurement frequency is needed (such as in research applications), wind can be measured by the propagation speed of ultrasound signals or by the effect of ventilation on the resistance of a heated wire. Another type of anemometer uses pitot tubes that take advantage of the pressure differential between an inner tube and an outer tube that is exposed to the wind to determine the dynamic pressure, which is then used to compute the wind speed.

3.3 Design of the Machine Learning

Machine learning (ML) is a field of inquiry devoted to understanding and building methods that 'learn', that is, methods that leverage data to improve performance on some set of tasks. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so [54]. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks. The ML tools that are being used in this thesis are described below.

3.4 Machine Learning Toolbox

Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support-vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy. Scikit-learn is a NumFOCUS fiscally sponsored project. The scikit-learn project started as scikits.learn, a Google Summer of Code project by French data scientist David Cournapeau. Its name stems from the notion that it is a "SciKit" (SciPy Toolkit), a separately-developed and distributed third-party extension to SciPy. The original codebase was later rewritten by other developers. In 2010 Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort and Vincent Michel, all from the French Institute for Research in Computer Science and Automation in Rocquencourt, France, took leadership of the project and made the first public release on February the 1st 2010. Of the various scikits, scikit-learn as well as scikit-image were described as "well-maintained and popular" in November 2012. Scikit-learn is one of the most popular machine learning libraries on GitHub. Scikit-learn is largely written in Python, and uses NumPy extensively for high-performance linear algebra and array operations. Furthermore, some core algorithms are written in Cython to improve performance [71]. Support vector machines are implemented by a Cython wrapper around LIBSVM; logistic regression and linear support vector machines by a similar wrapper around LIBLINEAR. In such cases, extending these methods with Python may not be possible.

3.5 Data Pooling Technique

When the images are excessive large, pooling layers segment would reduce the number of boundaries. In spatial pooling, subsampling means decreasing the dimensionality of each guide which holds the significant data. [13].

- There are different kinds of pooling:
- Max Pooling: The greatest pixel worth of the group is chosen.
- Min Pooling: The base pixel worth of the cluster is chosen.
- Average Pooling: The normal worth of the relative multitude of pixels in the group is chosen.
- Sum Pooling: The aggregate worth of the multitude of pixels in the group is chosen.
- Stochastic pooling: The pooled map reaction by inspecting from a multinomial conveyance framed from the enactments of each pooling district is chosen.

3.6 Data Rectification using RELU

RELU (A unit utilizing the rectifiers is likewise called a redressed direct unit RELU) gives 0 output if the input is 0 or less than 0, and draw output in any case. If the input is more prominent than 0, the output is equivalent to the input [12]. It has become the default initiation job for various types of neural organizations, because a model that uses it is easier to prepare and typically achieves superior execution. Unlike the sigmoid capacity RELU is non-straight and has the advantage of having no backpropagation errors, which is useful for large neural organizations. It allows models to learn quicker and perform better by conquering the evaporating slope. The RELU is the most utilized actuation work on the planet at this moment. The activity of RELU is nearer to the manner in which our natural neurons work.

3.7 Data Batch Normalization

Batch normalization is a strategy intended to consequently normalize the contributions to a layer in a profound learning neural organization. Fundamentally, to standardize the contributions of each layer in order to solve the inner covariate shift issue, batch normalization is used. To build the soundness of a neural organization, batch normalization standardizes the output of a past enactment layer by taking away the group mean and separating by the clump standard deviation. Nonetheless, after this shift/size of initiation outputs by some arbitrarily introduced boundaries, the loads in the following layer are as of now not ideal. Cluster standardization increases the standardized output by a "standard deviation" parameter (gamma), as well as adding a "signify" parameter to each layer (beta). The the regularization is being used. Dropout is a regularization method. We arbitrarily shut down certain neurons (units) on each layer and don't utilize those neurons in both forward spread and backpropagation, on every emphasis [16]. The learning calculation will have no clue about which neurons will be closed down on each cycle, since the units that will be exited on every emphasis will be arbitrary. Exiting can be viewed as briefly deactivating or overlooking neurons of the organization. This procedure is applied in the preparation stage to lessen over fitting impacts. Overfitting is a mistake which happen when an organization is excessively firmly fit to a restricted arrangement of info tests. During preparing dropout tests from a dramatic number of various diminished organizations. It is not difficult to inexact the impact of averaging the forecast of this load of diminished organizations by basically utilizing a solitary un-diminished organization that has more modest loads at test time.

3.8 Loss Calculation

Training loss is an important parameter for understanding models performance. In the event that we utilize this misfortune, we will prepare a ML model to yield a likelihood over the C classes for each picture. It is utilized for multiclass arrangement. The difference between what model believes the yield appropriation should be and really what is the first dissemination, is shown by cross entropy. [17]. A popular option of squared error is cross entropy, as it is characterized. It is used when node initiations can be interpreted as addressing the possibility that every theory is correct, for example, when the yield is a likelihood circulation. As a result, it is used as a blunder task in neural organizations with softmax actuations in the output layer. So just the positive class Cp keeps its term in the misfortune, in the particular (and regular) instance of Multi-Class grouping the marks are one-hot. There is just a single component of the Target vector t which isn't zero ti = tp. So disposing of the components of the summation which are zero because of target marks.

3.9 Extreme Gradient Boosting

XGBoost[2] (eXtreme Gradient Boosting) is an open-source software library which provides a regularizing gradient boosting framework for C++, Java, Python. It works on Linux, Windows, and macOS. From the project description, it aims to provide a "Scalable, Portable and Distributed Gradient Boosting (GBM, GBRT, GBDT) Library". It runs on a single machine, as well as the distributed processing frameworks Apache Hadoop, Apache Spark, Apache Flink, and Dask. XGBoost initially started as a research project by Tianqi Chen as part of the Distributed (Deep) Machine Learning Community (DMLC) group. Initially, it began as a terminal application which could be configured using a libsvm configuration file. It became well known in the ML competition circles after its use in the winning solution of the Higgs Machine Learning Challenge. Soon after, the Python and R packages were built, and XGBoost now has package implementations for Java, Scala, Julia, Perl, and other languages. This brought the library to more developers and contributed to its popularity among the Kaggle community, where it has been used for a large number of competitions. It was soon integrated with a number of other packages making it easier to use in their respective communities. It has now been integrated with scikit-learn for Python users and with the caret package for R users.

It can also be integrated into Data Flow frameworks like Apache Spark, Apache Hadoop, and Apache Flink using the abstracted Rabit and XGBoost4J. XGBoost is also available on OpenCL for FPGAs. An efficient, scalable implementation of XGBoost has been published by Tianqi Chen and Carlos Guestrin. While XGBoost model often achieves higher accuracy than a single decision tree, it sacrifices the intrinsic interpretability of decision trees.

Salient features of XGBoost which make it different from other gradient boosting algorithms include:

- Clever penalization of trees
- A proportional shrinking of leaf nodes
- Newton Boosting
- Extra randomization parameter
- Implementation on single, distributed systems and out-of-core computation
- Automatic Feature selection

3.10 LightGBM

LightGBM, short for Light Gradient Boosting Machine, is a free and open source distributed gradient boosting framework for machine learning originally developed by Microsoft. It is based on decision tree algorithms and used for ranking, classification and other machine learning tasks. The development focus is on performance and scalability. The LightGBM framework supports different algorithms including GBT, GBDT, GBRT, GBM, MART and RF [72]–[75]. LightGBM has many of XGBoost's advantages, including sparse optimization, parallel training, multiple loss functions, regularization, bagging, and early stopping.

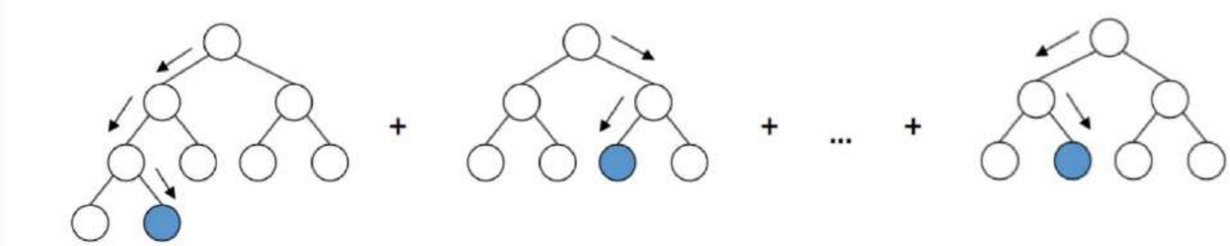

Fig 3.3 LightGBM Algorithm Decision Trees

A major difference between the two lies in the construction of trees. LightGBM does not grow a tree level-wise row by row as most other implementations do Instead it grows trees leaf-wise. It chooses the leaf it believes will yield the largest decrease in loss. Besides, LightGBM does not use the widely-used sorted-based decision tree learning algorithm, which searches the best split point on sorted feature values, as XGBoost or other implementations do. Instead, LightGBM implements a highly optimized histogram-based decision tree learning algorithm, which yields great advantages on both efficiency and memory consumption. The LightGBM algorithm utilizes two novel techniques called Gradient-Based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) which allow the algorithm to run faster while maintaining a high level of accuracy. Gradient-Based One-Side Sampling (GOSS) is a method that leverages the fact that there is no native weight for data instance in GBDT. Since data instances with different gradients play different roles in the computation of information gain, the instances with larger gradients will contribute more to the information gain. Thus, in order to retain the accuracy of the information, GOSS keeps the instances with large gradients and randomly drops the instances with small gradients. Exclusive Feature Bundling (EFB) is a near-lossless method to reduce the number of effective features. In a sparse feature space many features are nearly exclusive, implying they rarely take nonzero values simultaneously. One-hot encoded features are a perfect example of exclusive features. EFB bundles these features, reducing dimensionality to improve efficiency while maintaining a high level of accuracy. The bundle of exclusive features into a single feature is called an exclusive feature bundle.

3.11 Catboost

CatBoost[6] is an open-source software library developed by Yandex. It provides a gradient boosting framework which among other features attempts to solve for Categorical features using a permutation driven alternative compared to the classical algorithm.

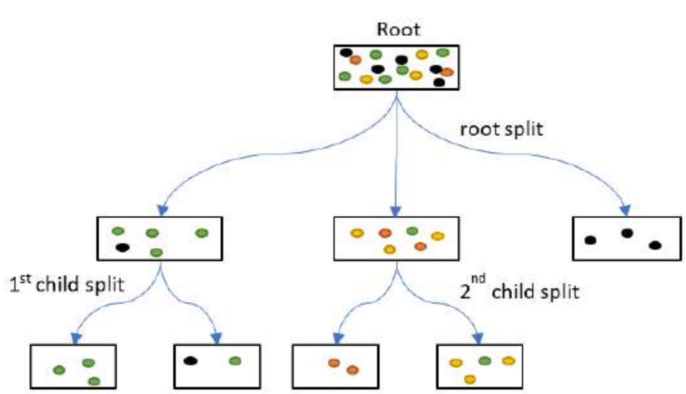

Fig 3.4 Catboost Algorithm Decision Trees

It works on Linux, Windows, macOS, and is available in Python, R, and models built using catboost can be used for predictions in C++, Java,[10] C#, Rust, Core ML, ONNX, and PMML. The source code is licensed under Apache License and available on GitHub CatBoost has gained popularity compared to other gradient boosting algorithms primarily due to the following features.

- Native handling for categorical features
- Fast GPU training
- Visualizations and tools for model and feature analysis
- Using Oblivious Trees or Symmetric Trees for faster execution
- Ordered Boosting to overcome overfitting

CHAPTER 4

SYSTEM DESIGN

4.1 Introduction

The use of wind energy worldwide has overgrown in recent years to reduce greenhouse gas emissions. Wind power is free, but the installation and maintenance of wind turbines remain very costly. The size of the installation of the wind turbine is not only determined by wind statistics at a given location, but also by turbine infrastructure and maintenance. Therefore it is necessary to have proper power analysis tools. The rise in global temperature and severe climate change worldwide has increased environmental concerns. Nowadays, more than 90% of the world's electricity comes from fossil fuels (World-Bank 2015), and that energy production plays a vital role in global warming. Any changes in this field can have a significant impact on the environment. To solve these problems, the approached system designed is being discussed in this section.

4.2 Methodology

This implementation technique is separated into the sections below; the overall methodology is shown in Figure 4.1.

- Dataset Categorization
- Algorithmic survey
- Implementation

4.3 Dataset

A dataset is a grouping of features. In tabular features, a record refers to a database table or several database tables, each column in the chart represents a specific variable, and every row corresponds to a specific record within that record. A data set lists the value of each variable, such as b. the size or for each member of the data collection, the weight of an object. Each value is referred to as a date. A record can also contain a series of files or archives. In the field of open data, datasets are units of information published in publicly available open databases;

the European Open Data Portal combines more than 500,000 datasets. In this context, other definitions have been proposed, but there is currently no formal definition. Other issues, such as real-time data sources and non-relational datasets, are also difficult to agree on. Several characteristics determine the formation and properties of records. These cover the quantity and type of elements and volatiles, and the different statistical indicators applied to them, for example, excellence deviation and kurtosis. Values can be real or integer numbers, such as the height of a B. person in centimeters, or nominal data, such as the racial representation of a. person, where B. is not composed of numbers.

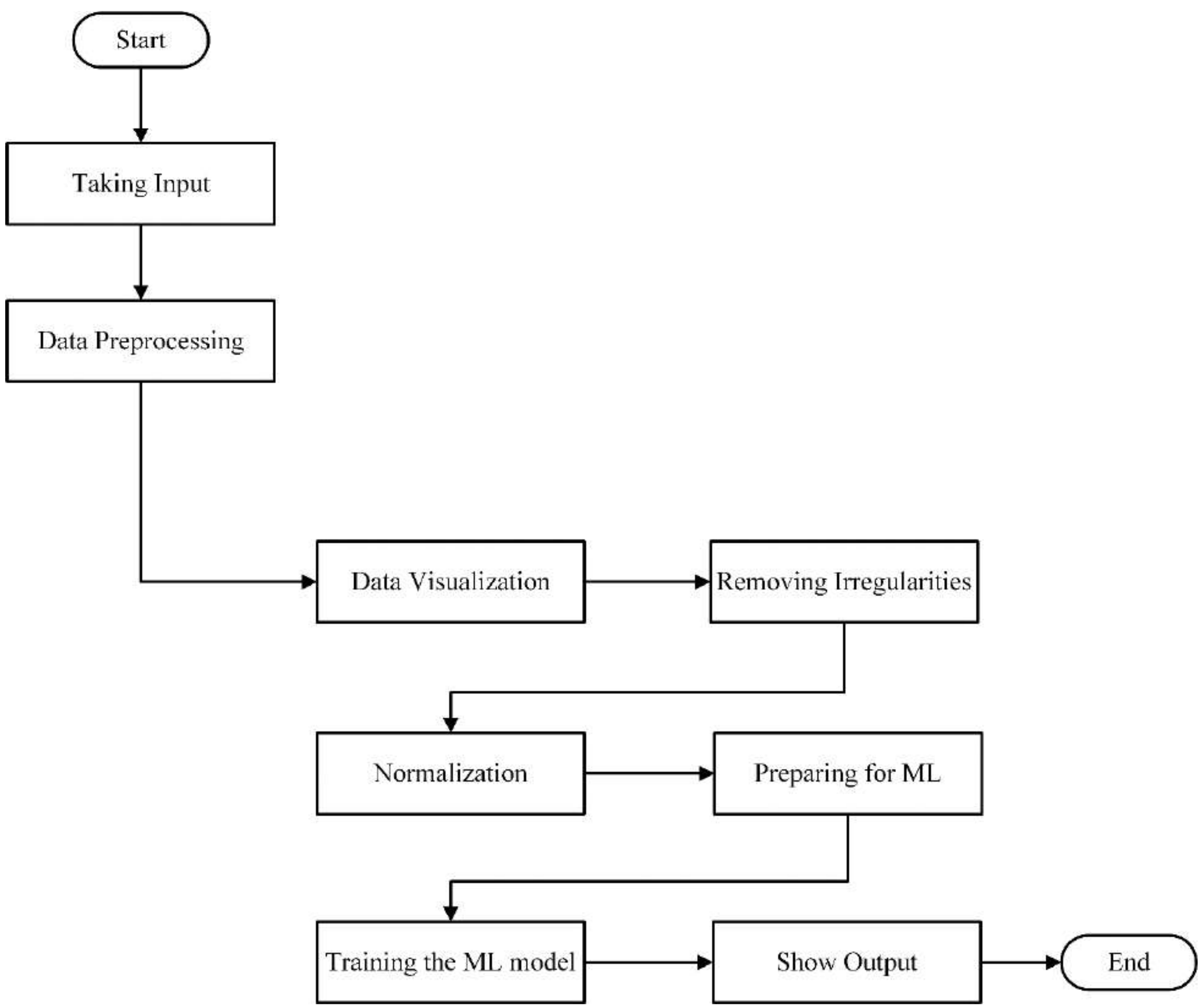

Fig 4.1 Methodology of the proposed work

More generally, any value described as a measurement level may be used. The values for each variable are usually the same However, there may be some values that are missing that need to be specified in some method in statistics, a feature set is normally derived from real observations from a sample of a statistical population, with each row corresponding to an observation of a member of that population. Datasets are sometimes created by algorithms to test certain types of software. Statistical analysis software of the present day, such as SPSS, still uses the classic dataset method to display data. If data are missing or suspect, adaptive methods can be used to finish the data set.

4.3.1 Training Set

Figure 4.1 illustrates the process of algorithm learning. A common challenge in machine learning is the study and development of algorithms that learn from and make projections about the feature. Such algorithms work by establishing mathematical models based on input data to make data-driven predictions and conclusions. The feature used to create the type, the finishing model is derived from many datasets. In special, it is common to use three datasets at different phases of the model building process. The model is primarily fitted to a training data set. The training dataset is a set of samples matching the parameters, for example, the weights of the relation between neurons in the artificial neural network of the B. model. The model is trained on the training dataset using a simple Bayesian classifier from a neural network or supervised learning strategy, for example, by employing optimization techniques such as incline descent or stochastic gradient descent.

In exercise, training typically is made up of a pair of input vectors and a keep in touch output vector, where the reply keys are usually labeled as targets or tags. The current model is driven by a training dataset, and for each input vector in the training dataset, a comparison with the target is generated. Based on the comparison results, the parameters of the model are tuned using a specific learning algorithm. Fitting the model includes variable choosing and parameter estimation. The fitted framework is often used to predict the observed responses on a second dataset (called the validation dataset). The validation dataset unbiasedly assesses how well the model fits the training dataset when adjusting the number of layers of secret units and the layer width of the hyperparameter of the neural network model. The validation dataset can be regularized by stopping training since large errors in the validation dataset are an indication of overfitting of the training dataset. This simple approach is complex in practice because the errors in the validation dataset may vary during training, which may lead to multiple local minima. Due to this complexity, special rules have been developed to determine the onset of overfitting.

4.3.2 Test Set

A test data set is a sort of data set that is used to objectively assess the final model's compliance with the training data set. If the test data set's data is not at all used in training, such as in cross-validation, it is also called a retention data set. A test feature set is a set that is unrelated to the training data set but has the same possibility distribution. A model fitted to the training feature set

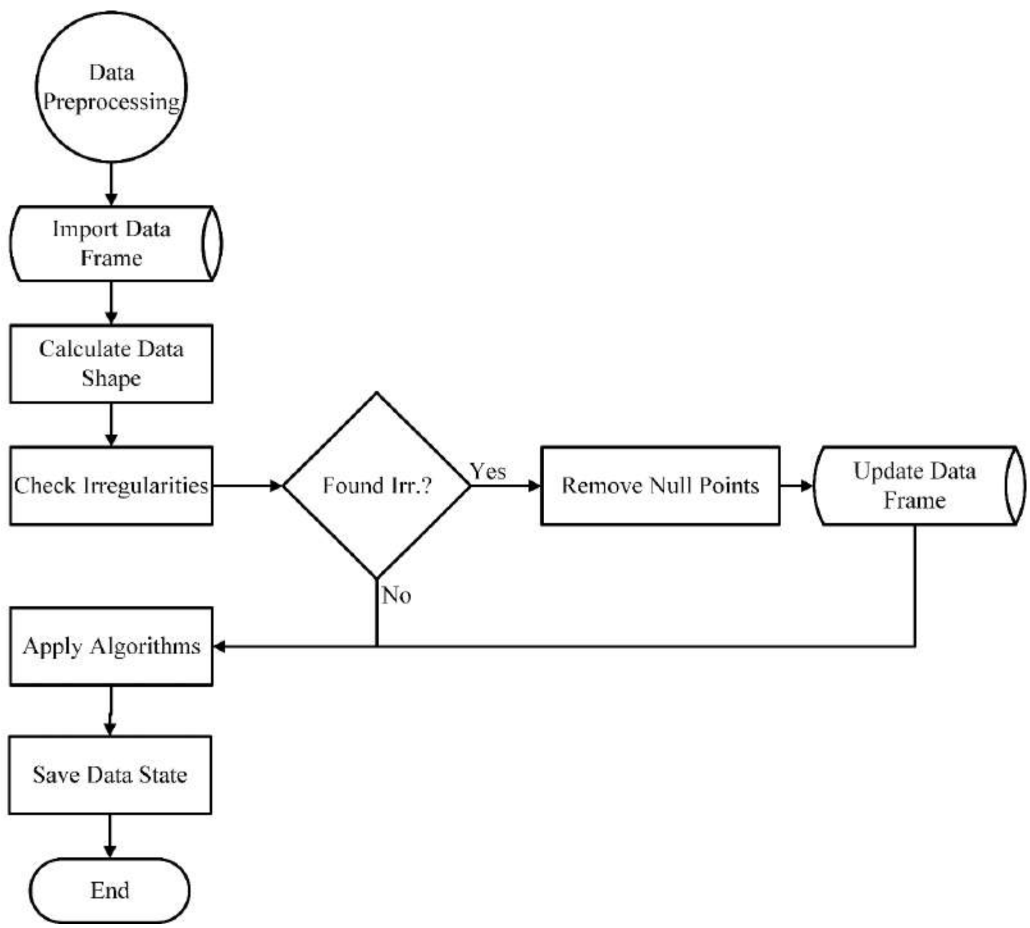

Fig 4.2 Steps of Data Preprocessing

will have little or no overfitting if it fits well to the test data set. If the training data set is a better fit than the test data set, it usually means that overfitting has occurred. Therefore, the test set is a collection of illustrations and is used only as a generalization to assess the validity of a very specific classification. The data set can be iteratively given away into a training data set and a confirmation data set. This procedure is called cross-validation. These recurring splitting can be done in a variety of ways, for example, by splitting the data into two identical data sets, using them as training/validation, and then selecting a random subset as the validation/training or repeated validation data set the following is an example of a case in point. Additional test data sets not included in cross-validation may be used to validate model performance.

4.4 Data Cleansing

Data cleansing is the identification, repair, or removal of corrupted or incorrect documentation from a data file, table or sequence. It is also the process of identifying missing data, inexact, or irrelevant portions of data, and putting back, modifying, or removing them. Grimy or fat information. Data cleaning can be done interactively as a batch process using a data processing tool or script. After cleaning, records should be compatible with other familiar records in the method. The unpredictable identified or deleted may be primarily due to user data entry errors, shipping or storage damage or dissimilar data dictionary definitions for comparable entities in various stores. Data cleaning differs from data confirmation, which almost always takes place at the time of data entry, meaning that data is rejected by the system at the time of entry, not in batches. The actual data cleaning process of fixing values against a list of known entities may entail deleting typos or correcting values against a list of known entities. Legitimacy can be severe, such as rejecting b. valid zip codes or obscure addresses. Some data cleaning solutions clean the data by associating it with a valid data set. A common type of data cleaning is data enrichment, which adds additional information to the data. For example, an address and phone number associated with that address are added. Data cleaning also includes data matching and normalization. This is the aggregation of data from different file formats, names, and columns into a coherent data set. An easy example is the dissemination of short tables.

- A set of high-quality data standards must be met. These include. Validity. Indicators conform to defined business rules and constraints. Ensuring validity is relatively easy when designing data collection systems using modern database technologies. Invalid data are mainly found in legacy situations where there are no application software constraints or where inappropriate data collection techniques are used, such as spreadsheets that do not use cell validity. Once implemented, it is difficult to restrict user access to the room for this purpose. The categories of data restrictions are as follows:
- The column data type constraint value should be of a specific data type, logical, numeric, integer or real, data, etc.
- Range restriction. Numbers and dates should, in general, be inside a specified orbit. That is, they are limited to a minimum and maximum value.
- Uniqueness restriction. Within a record, a field or mixture of fields must be unique. For the sample, two people can't have the same social security number.
- Setting subscription limits. Column values are derived from a separate set of estimates or codes. A person's gender, for example, can be female, male, or unknown.
- Foreign Key Restriction. This is one of the collection's most common members. A column's set of values is defined by a column in another table that contains a unique value. In the US taxpayer database, for example, the state column must be one of the defined US states or territories. A separate state table keeps track of the allowed states/territories. The term foreign key is taken from comparative database terminology. Orderly Expression Patterns: Sometimes text fields need to be verified this way. A phone number, for example, may be required to have a pattern.
- Confirmation across fields: when multiple fields are used, some conditions must be met. For example, in lab medicine, the components of the white blood cell differential are percentages, so their sum must be 100. In hospital databases, the date of discharge must not be before time than the date of admission.
- Accuracy: the degree of agreement with measured or actual values. Increased accuracy in data cleaning requires access to external data sources where true value exists, which can be difficult as such data with gold value are often not available. In cleansing, especially of customer contact data, accuracy is improved by using outmost databases

to match postal codes with geographical locations and to check whether the postal code contains an address.

- Completeness: the level at which all necessary activities are recorded. Incomplete data can rarely be corrected by cleaning the data. No one can assess information that was not recorded at the time the data were recorded. In some cases, it is possible to trace back the source of the interview data and correct any omissions.
- Consistency. The degree of equivalence between a set of activities in a system. Inconsistency occurs when two sets of recorded data are inconsistent.
- Consistency: the degree to which a set of data measures is reported with the same unit of measurement in all systems. In data sets collected in different locations, weight may be recorded in pounds or kilograms and the units of measurement must be converted using mathematics.

4.5 Algorithmic Analysis

Supervised learning is a machine learning feature learning task that draws inputs to outputs based on sample input-output pairs, as shown in Figure 4.3. Features are obtained from labeled training data containing training examples. In supervised learning, each example consists of a pair of input objects (usually a vector) and desired output values (also called monitoring signals). The supervised learning algorithm examines the training data and produces an approximation function that may be used to map fresh samples. The best-case scenario is that the algorithm accurately determines the class labels of the invisible instances. This necessitates that the learning algorithm generalizes instances that are not visible in the training data in a reasonable manner. To solve a particular auxiliary learning problem, the following steps should be taken

- Determine the type of instances for learning. Before that, it is necessary to determine what type of feature will be used as the training set. For example, for handwriting analysis, it is possible to use a single handwritten nature, a complete handwritten phrase, or an entire handwritten line.
- Collect training sets. The training set should be representative of how the feature will be used in the real world. Thus, a set of input articles should be collected, and the

communication outputs should also be collected from human experts and measurements.

- Set the appearance of the input properties of the learning function. The precision of the learning function depends on how the input objects are presented. Typically, the input object is transformed into a vector of attributes with many properties describing the object. Due to the curse of dimensionality, the number of attributes cannot be too huge. However, enough information is needed to accurately predict the output.
- Define the structure of the learning function and the corresponding learning algorithm. For example, an engineer might use a support vector machine or a decision tree.
- Finalize the design. Use the acquired training set to run the learning algorithm. A few supervised learning algorithms require the user to specify certain control restrictions. These boundaries can be set by optimizing performance on a subset of the training set (called the validation set) or by cross-validation.
- Evaluating the precision of the training task. Once the parameters have been determined and trained, the resulting function's performance must be judged on a separate test set from the training set.

Unsupervised learning is a learning algorithm that draws inferences from input data containing labeled, unanswered questions. The most common form of unsupervised learning is bundle analysis, which is used to analyze data to find secret patterns or clustering in the data. Clustering is modeled using matrices with defined similarity measures, such as Euclidean distance or latent distance. Unsupervised learning is a sort of machine learning that looks for previously unnoticed patterns in a dataset without using existing labels and with little human supervision. Unlike supervised learning, which uses human-labeled data, unsupervised learning, also called self-organization, allows latent density models to be constructed from the input. Together with supervised learning and reinforcement learning, it forms one of the three principal branches of machine learning. In semi-random learning, interconnected variants, supervised and unsupervised methods are used. The two main supervised learning methods are principal component analysis and cluster analysis. Cluster analysis is used in supervised learning to cluster or partition data sets with common characteristics to evaluate algorithmic relationships. Collection analysis is a branch of machine learning that classifies unlabeled, classified, and unclassified data. Instead of responding to clusters, cluster analysis identifies generalizations in the data and responds to their presence or absence in each new data item. This method helps to identify anomalous data points that do not fit into any cluster.

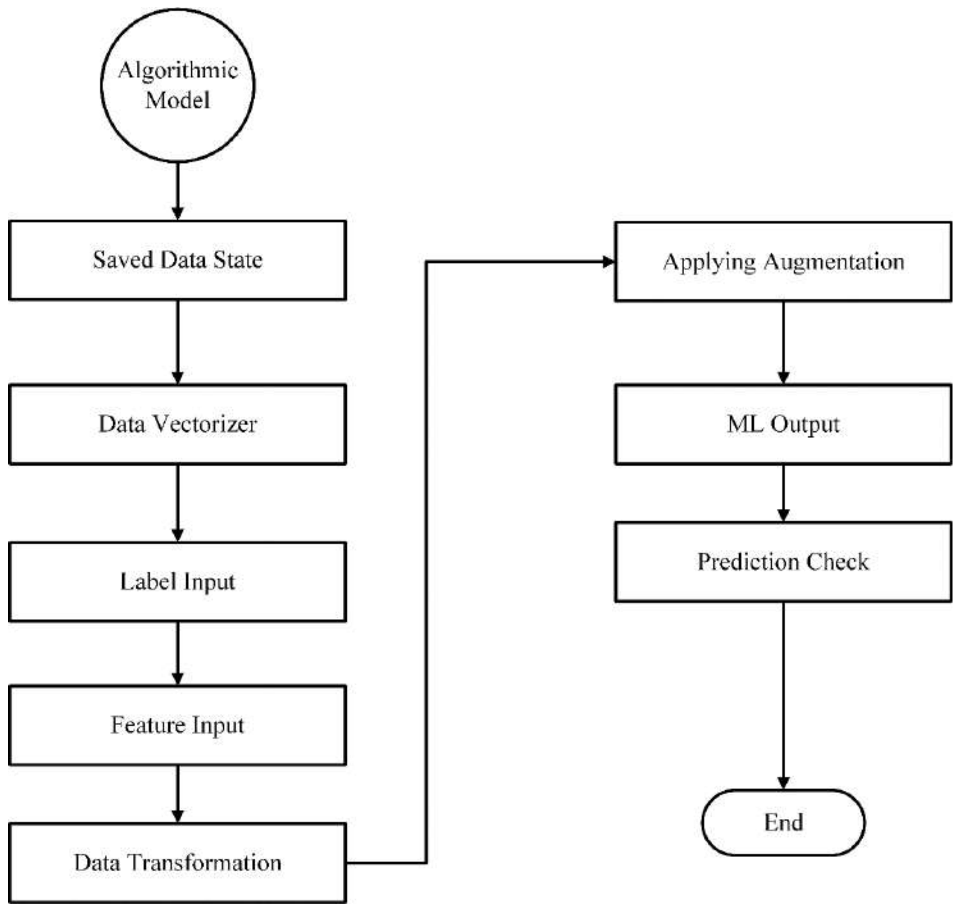

Fig 4.3 Analysis of different machine learning algorithms

An unconstrained learning strategy can only describe the need to learn a new feature space by maximizing the objective function or minimizing the loss function that reflects the properties of the original space. Thus, constructing a covariance matrix is not unconstrained learning, but requires factorization of the covariance matrix, since linear algebra maximizes the diversity of eigenvalue subtraction operations; this is called principal component analysis. Similarly, logarithmic transformation of a dataset is not a meaningless learning, but rather a reduction of the distance function between the generated data by passing the input data through multiple sigmoid functions, called an auto-encoder. While the core application of supervised learning is statistical density estimation, supervised learning also covers many other areas of summarization and interpretation of data elements. Unlike supervised learning, which can be argued to attempt to assume potential conditional distributions, generative adverbial networks can be applied to both supervised learning and supervised and reinforcement strategies.

CHAPTER 5

SYSTEM IMPLEMENTATION AND RESULT

5.1 Introduction

In this chapter, the complete implementation and objective justification have discussed with proper demonstration. All the obtained result from the data analysis is being mentioned in this chapter.

5.2 Dataset Overview

The dataset is the fundamental part of this thesis work. The overview of the dataset is being demonstrated on the Fig 5.1.

date/time	lv activepower (kw)	wind speed (m/s)	theoretical_power_curve (kwh)	wind direction (°)
01 01 2018 00:00	380.047790527343	5.31133604049682	416.328907824861	259.994903564453
01 01 2018 00:10	453.76919555664	5.67216682434082	519.917511061494	268.64111328125
01 01 2018 00:20	306.376586914062	5.21603679656982	390.900015810951	272.564788818359
01 01 2018 00:30	419.645904541015	5.65967416763305	516.127568975674	271.258087158203
01 01 2018 00:40	380.650695800781	5.57794094085693	491.702971953588	265.674285888671

Fig 5.1 Overall Dataset Overview

date/time	lv activepower (kw)	wind speed (m/s)	theoretical_power_curve (kwh)	wind direction (°)	month	hour
01 01 2018 00:00	380.047790527343	5.31133604049682	416.328907824861	259.994903564453	1	0
01 01 2018 00:10	453.76919555664	5.67216682434082	519.917511061494	268.64111328125	1	0
01 01 2018 00:20	306.376586914062	5.21603679656982	390.900015810951	272.564788818359	1	0
01 01 2018 00:30	419.645904541015	5.65967416763305	516.127568975674	271.258087158203	1	0
01 01 2018 00:40	380.650695800781	5.57794094085693	491.702971953588	265.674285888671	1	0

Fig 5.2 Month and Hour Variable Datatype Conversion

	wind speed (m/s)	theoretical_power_curve (kwh)	lv activepower (kw)
count	50530.00	50530.00	50530.00
mean	7.56	1492.18	1307.68
std	4.23	1368.02	1312.46
min	0.00	0.00	-2.47
25%	4.20	161.33	50.68
50%	7.10	1063.78	825.84
75%	10.30	2964.97	2482.51
max	25.21	3600.00	3618.73

Fig 5.3 Description of the Dataset

Later, the datatype of the month and hour is being converted into integer from string for demonstration purpose. After the conversion it is being stored in a Pandas array which is demonstrated in the Fig 5.2. The overall description of the dataset is being shown in the Fig 5.3.

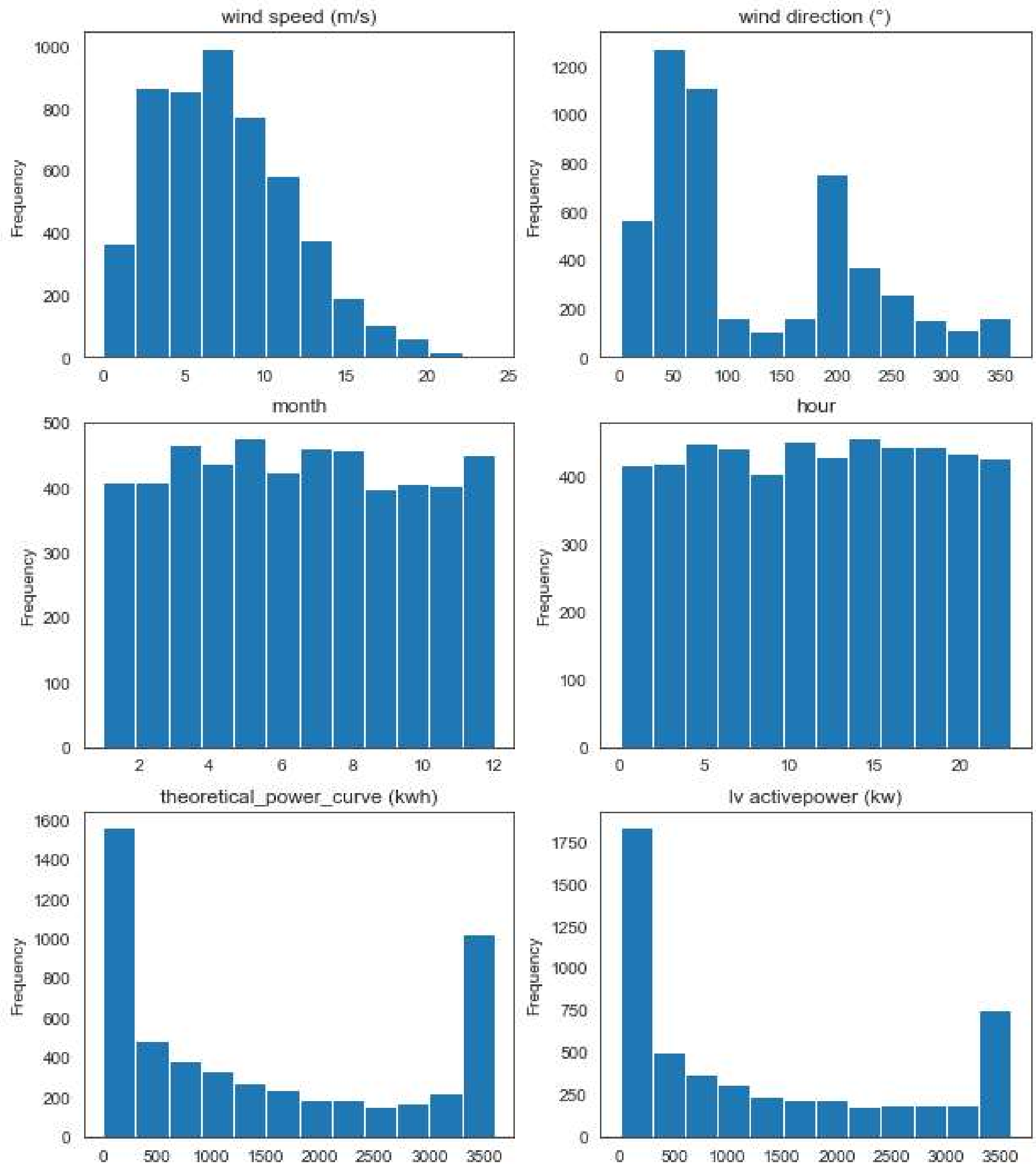

Fig 5.4 Data Distribution Histogram

The data distribution has shown in Fig 5.4. This graph is very crucial to understand the overall the data density in the specific numeric category.

After the data distribution, it is important to understand the correlation between the parameters in the dataset. The correlation depends on the parametric values in the respective rows and columns.

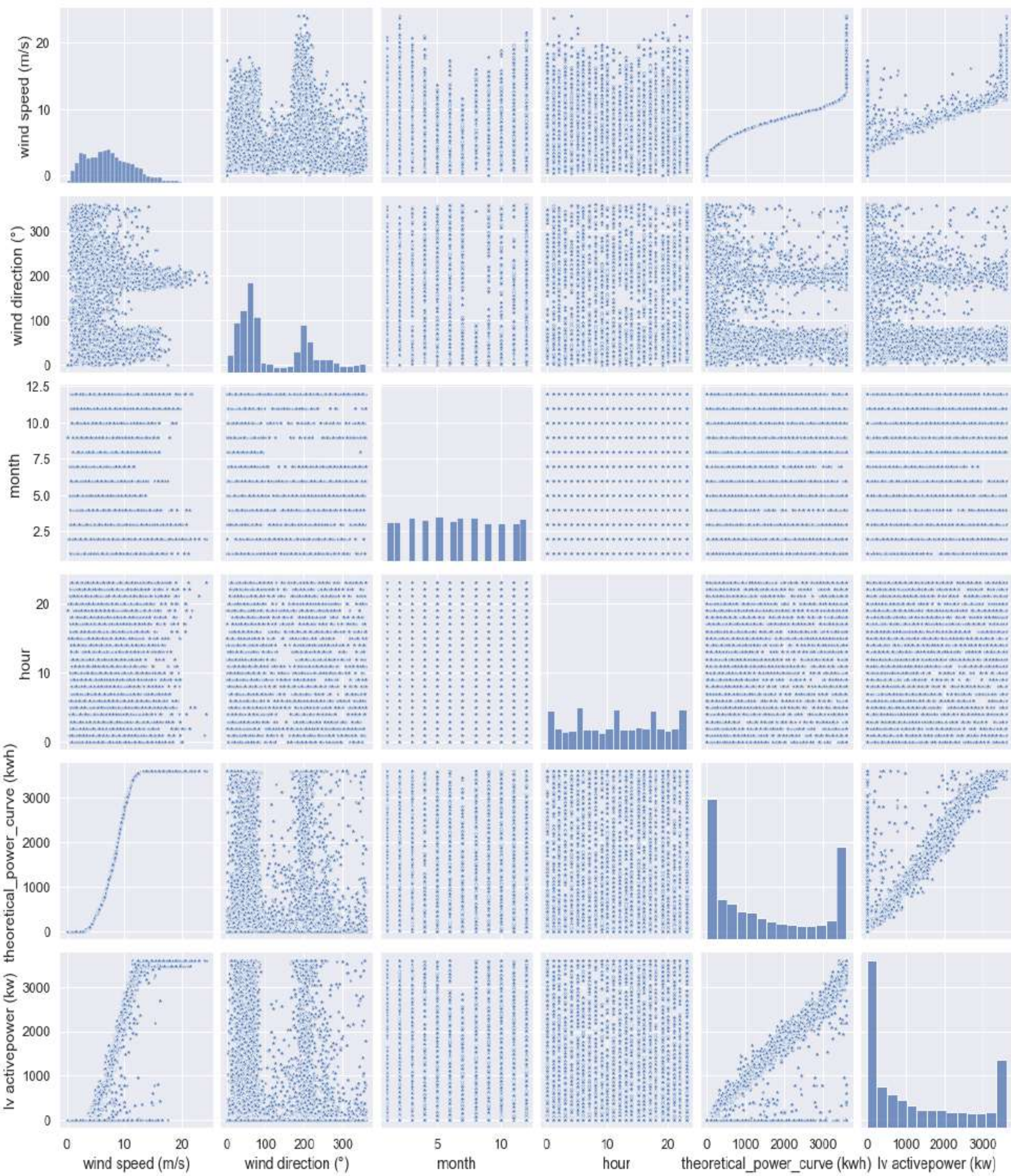

Fig 5.5 Correlation between All Parameters

The possible correlation between all the parameters in the dataset is being shown in the Fig 5.5.

5.3 Power Production Analysis

Power generation is the process of generating electric power from sources of primary energy. For utilities in the electric power industry, it is the stage prior to its delivery such as transmission, distribution, etc. to end users or its storage. The power production analysis is being demonstrated in the following figures.

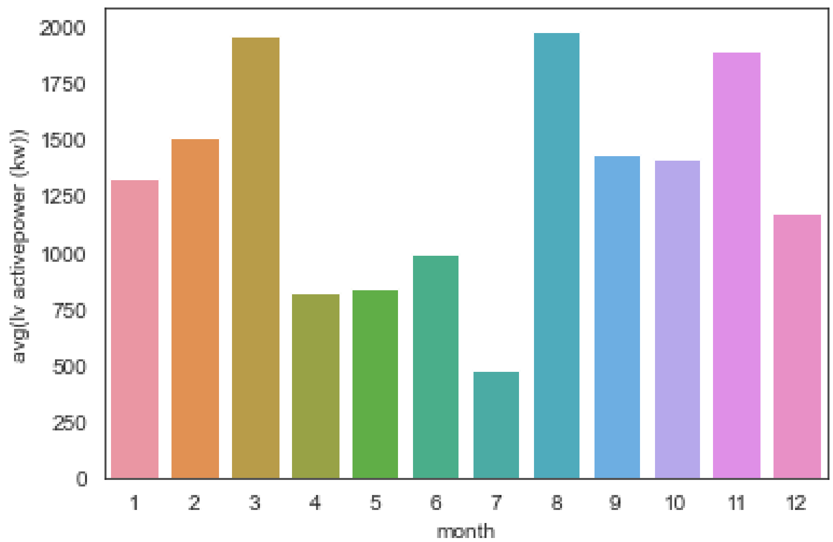

Fig 5.6 Average Power Production by Month

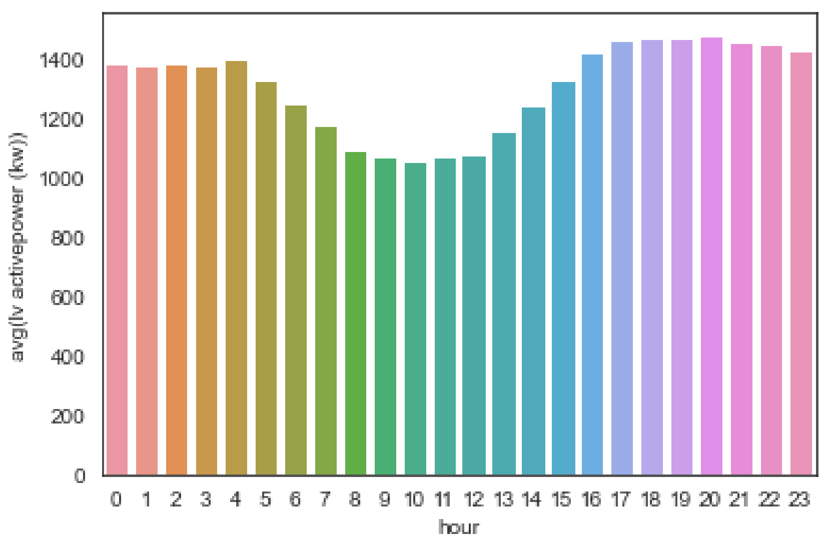

Fig 5.7 Average Power Production by Hour

The average monthly power production is being demonstrated on the Fig 5.6 and the hourly power production is being demonstrated in the Fig 5.7.

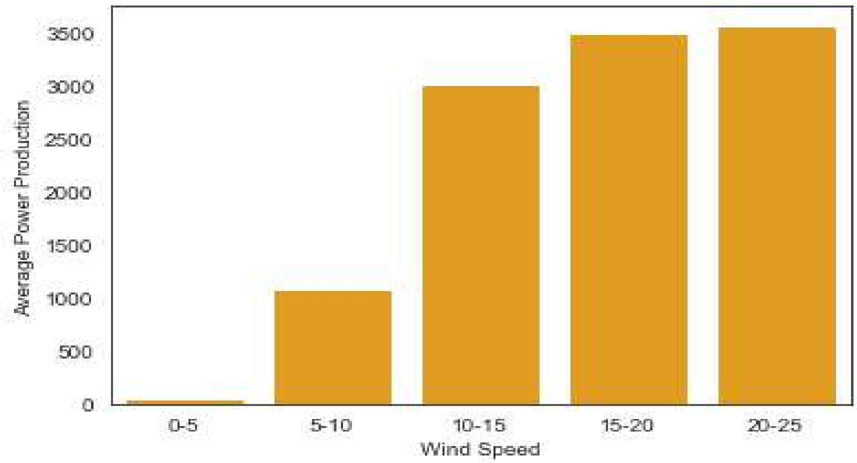

Fig 5.8 Average Power Production for 5 m/s Wind Speed Increments

The production varies with respect to the increment of wind speed which is being demonstrated in the Fig 5.8.

5.4 Power Production Analysis With Respect To Wind Speed

Wind speed plays a vital role in the power production from wind turbines. In Figure 5.9 the available data in wind speed category is being demonstrated and the amount of power production is also being visualized.

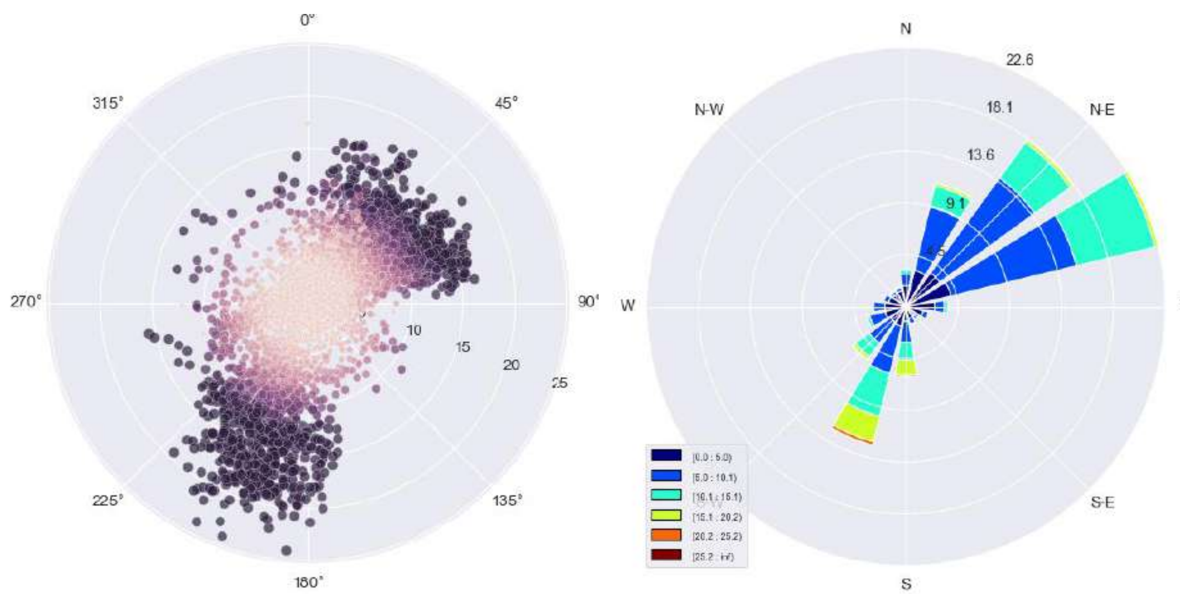

Fig 5.9 Power Production for WRTo Wind Speed and Direction

In Figure 5.10, the power production with respect to specific wind speed is being demonstrated. It is clear from the graph that the power generation increases with the wind speed of a certain place.

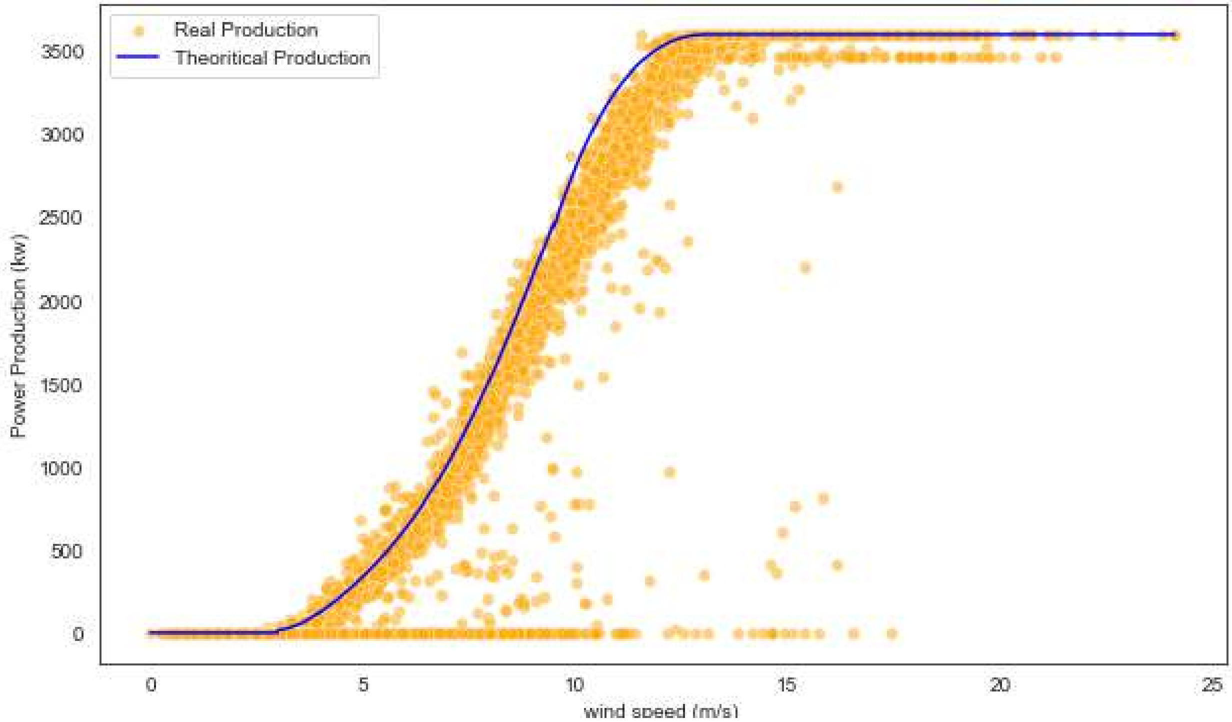

Fig 5.10 Wind Speed and Power Production Chart

5.5 Zero Power Calculation by Data Parameters

The zero power indicates the production and generation equivalence in a plant or turbine. The relation between Zero power with wind speed is being demonstrated in Fig 5.11.

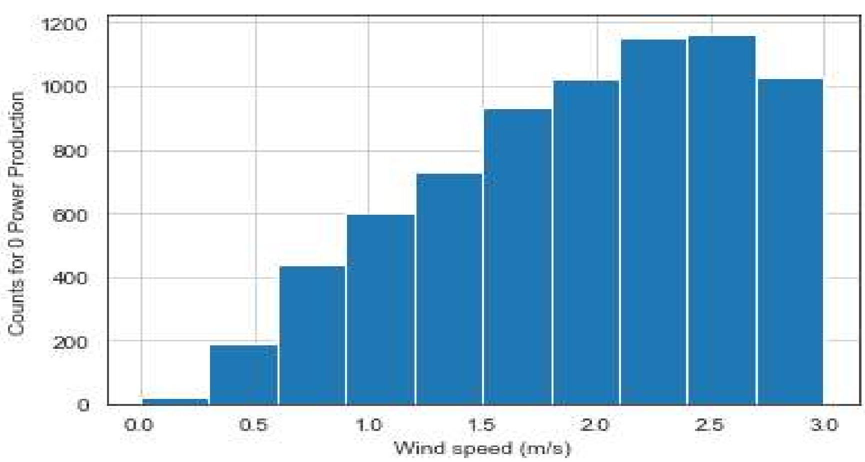

Fig 5.11 Wind Speed Distribution for zero Power Production

The data columns that are responsible for Zero power, is being demonstrated on the Fig 5.12.

	wind speed (m/s)	theoretical_power_curve (kwh)	lv activepower (kw)
2525	2.78	0.00	0.00
1534	1.09	0.00	0.00
6886	1.86	0.00	0.00
6021	2.83	0.00	0.00
5653	0.71	0.00	0.00

Fig 5.12 Filtration of Data Where the Real and Theoretical Power Productions are Equal to zero

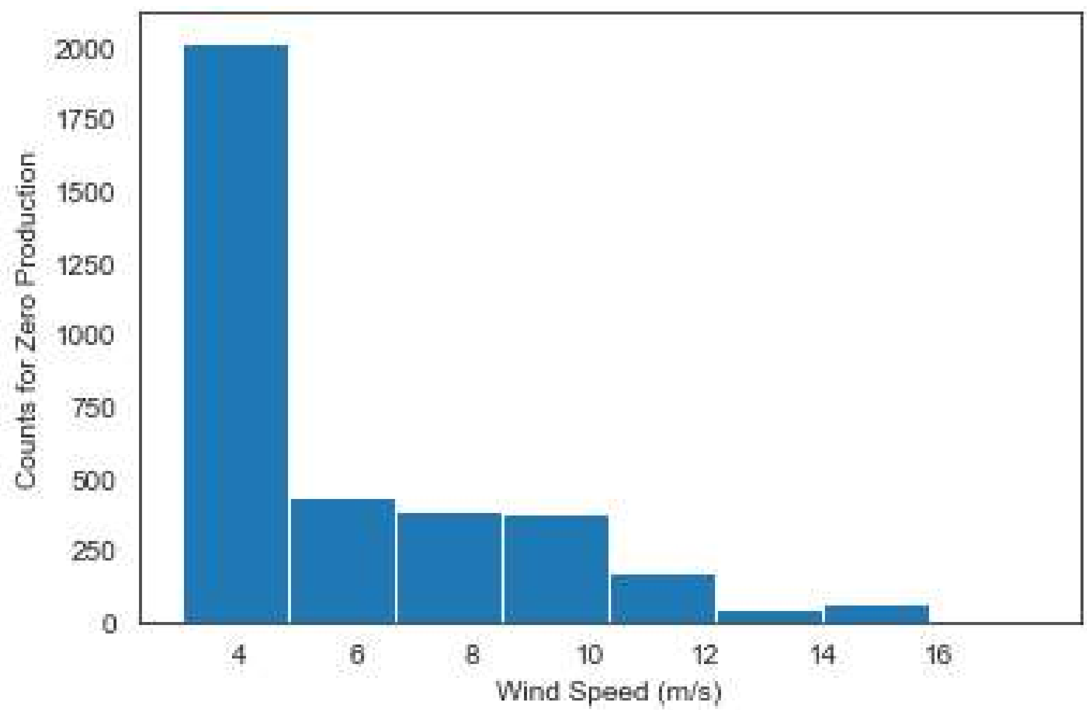

Fig 5.13 Wind Speed Counts for Zero Power Production

In figure 5.13, the total number of values that are being categorized as Zero power is being grouped with respect to the wind speed category they belong to. Later, the zero power production by month is being demonstrated on the Fig 5.14.

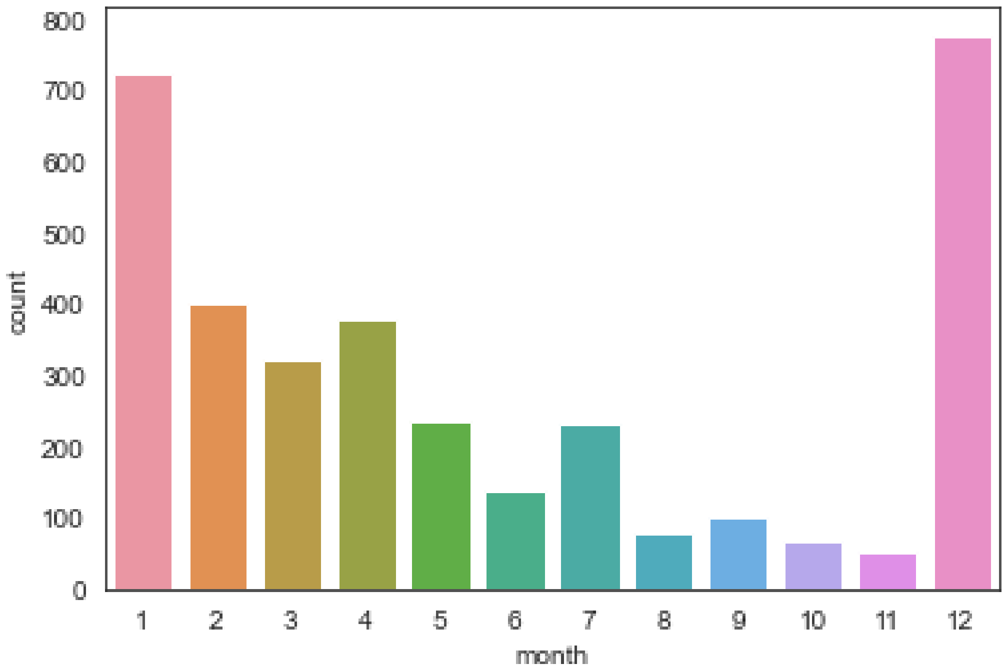

Fig 5.14 Zero Power Count by Monthly

5.6 Energy Production Calculation

The total energy production is being calculated and later to the Machine Learning algorithms.

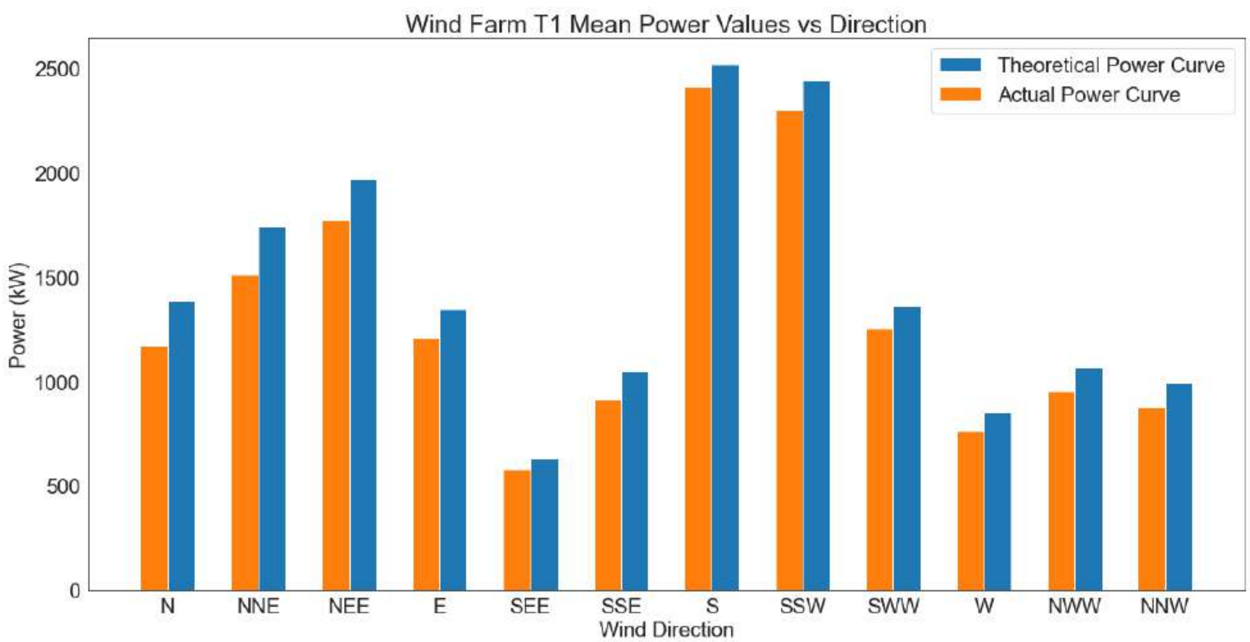

Fig 5.15 Wind Farm Mean Power Values vs Direction

The energy production is calculated by using the Theoretical Power Curve and Actual Power Curve which is being demonstrated in Fig 5.15. The wind farm energy generation with respect to wind direction is demonstrated on Fig 5.16. Later the losses are also calculated and demonstrated on Fig 5.17.

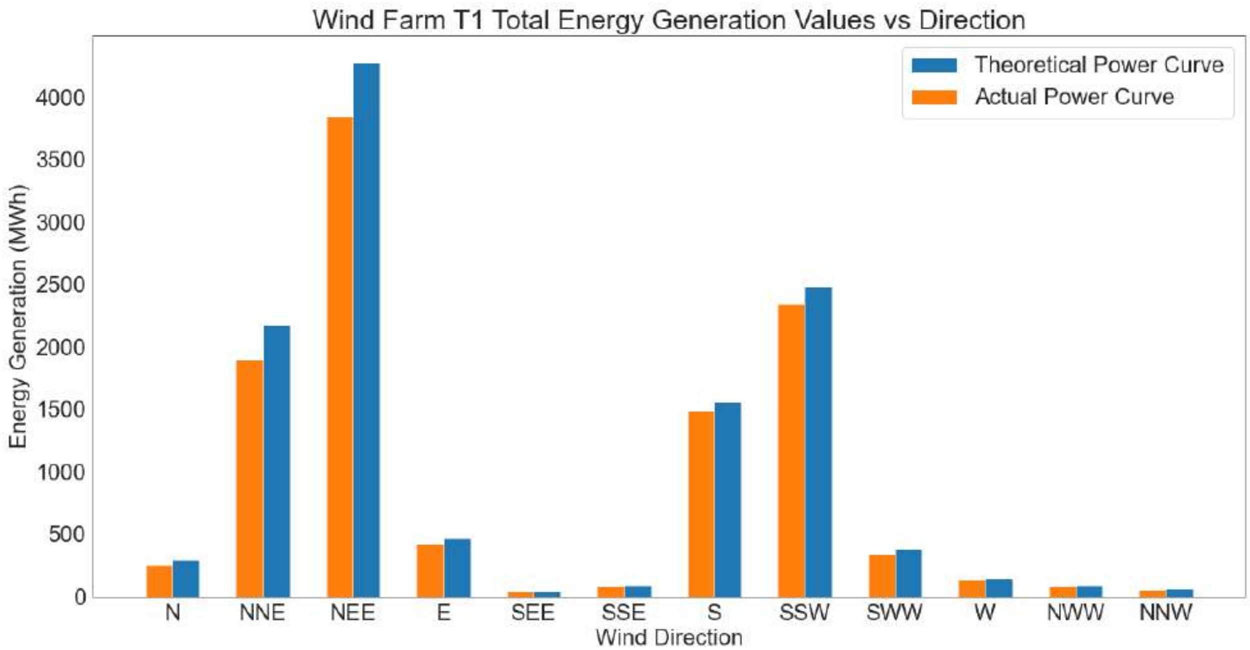

Fig 5.16 Wind Farm Total Energy Generation Values vs Direction

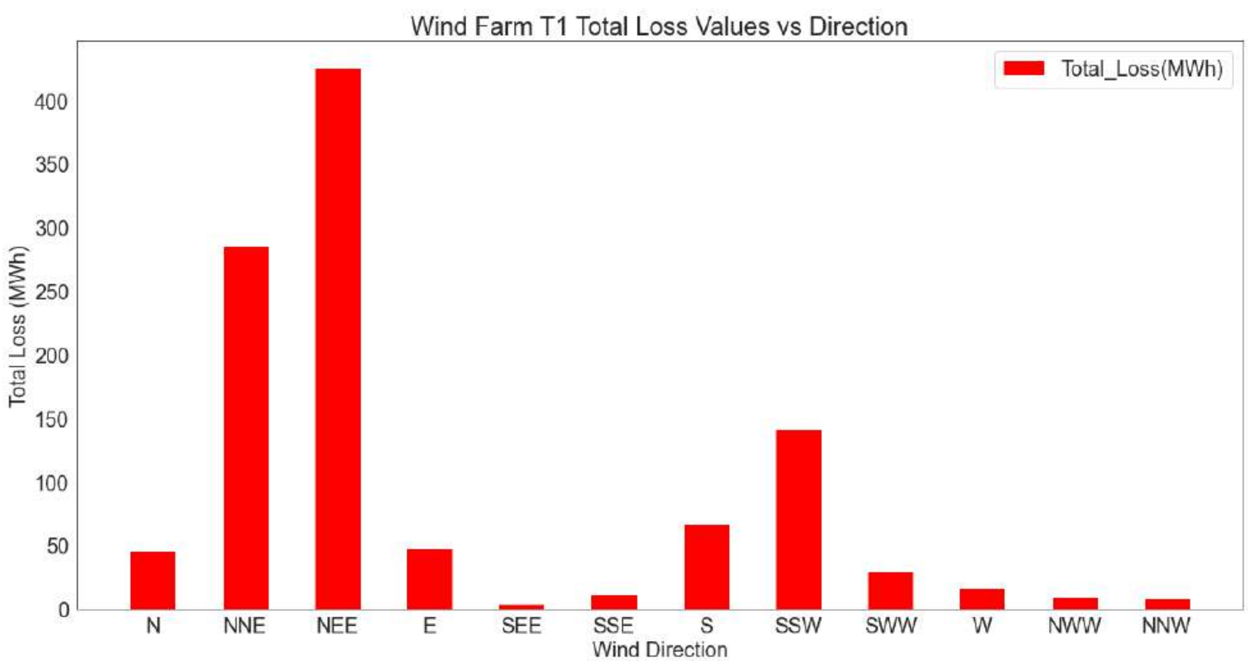

Fig 5.17 Total Loss Values vs Direction

5.7 Machine Learning Implication and Accuracy Scores

To preform power generation prediction from the data we have analyzed, several Machine Learning algorithms are being applied which re being demonstrated below. The XGBRegressor algorithm has applied to perform dataset training and testing process. In the end, we achieved 0.88 accuracy score from the algorithm.

```
XGBRegressor(base_score=0.5, booster='gbtree', callbacks=None,
             colsample_bylevel=1, colsample_bynode=1, colsample_bytree=1,
             early_stopping_rounds=None, enable_categorical=False,
             eval_metric=None, gamma=0, gpu_id=-1, grow_policy='depthwise',
             importance_type=None, interaction_constraints='',
             learning_rate=0.02, max_bin=256, max_cat_to_onehot=4,
             max_delta_step=0, max_depth=4, max_leaves=0, min_child_weight=1.1,
             missing=nan, monotone_constraints='()', n_estimators=100,
             n_jobs=-1, nthread=-1, num_parallel_tree=1, predictor='auto',
             random_state=0, reg_alpha=0.3, ...)
```

```
preds=model_xgb1.predict(x_train)
score=mean_squared_error(y_train,preds)
score**0.5
```

```
450.24259358157246
```

```
r2_score(y_train,preds)
```

```
0.8823147743453111
```

```
LGBMRegressor(learning_rate=0.07, n_estimators=80, num_leaves=2,
              objective='regression', reg_alpha=0.3, reg_lambda=0.7)
```

```
preds=model_lgb1.predict(x_train)
score=mean_squared_error(y_train,preds)
score**0.5
```

```
391.9123625549006
```

```
r2_score(y_train,preds)
```

```
0.910832474267455
```

However, the GBMRegressor algorithm has achieved better accuracy score with compared to the XGBoost. The obtained accuracy score was 0.91. These algorithms are being implemented by using Sci-kit learn and the dataset are being distributed by using Pandas data frame operation.

CHAPTER 6

CONCLUSIONS

6.1 Conclusions

In this paper, a comprehensive analysis for wind turbine power generation and prediction is being proposed. The models are being used by using open source python tools like Numpy, Pandas, Matplotlib, Seaborn and Sci-kit learn. Wind energy, being a clean, and sustainable energy resource, has increasing contributions in the modern power systems in the recent years. Therefore, it is very necessary to understand the power generation, consumption as well as prediction.

6.2 Findings of this Work

The findings of this work has discussed below.

- The production of power is highly dependent on the wind speed which is highly available of March, August and November.
- The dataset is susceptible to outliers. Therefore it has considered during feeding the ML algorithm.

6.3 Future Improvement

Some future work can be implemented later by further research has proposed below.

- The results achieved from the thesis is theoretical.
- Applying the data analysis and ML output will be a challenging future work.

REFERENCES

[1] M. Liao, L. Dong, L. Jin, and S. Wang, "Study on rotational speed feedback torque control for wind turbine generator system," *2009 Int. Conf. Energy Environ. Technol. ICEET 2009*, vol. 1, pp. 853–856, 2009, doi: 10.1109/ICEET.2009.211.

[2] J. Taghinezhad, E. Mahmoodi, M. Masdari, and R. Alimardani, "Spectral Analyses of an Optimized Ducted Wind Turbine using Hot-Wire Anemometry," *7th Iran Wind Energy Conf. IWEC 2021*, May 2021, doi: 10.1109/IWEC52400.2021.9467033.

[3] Y. Yin, M. Liao, and P. Lyu, "The dynamic stability analysis of wind turbines under different control strategies," *Proc. 5th IEEE Int. Conf. Electr. Util. Deregulation, Restruct. Power Technol. DRPT 2015*, pp. 2581–2586, Mar. 2016, doi: 10.1109/DRPT.2015.7432683.

[4] G. Shi, J. Zhang, X. Cai, and M. Zhu, "Decoupling control of series-connected DC wind turbines with energy storage system for offshore DC wind farm," *2016 IEEE 7th Int. Symp. Power Electron. Distrib. Gener. Syst. PEDG 2016*, Jul. 2016, doi: 10.1109/PEDG.2016.7527064.

[5] H. Rashid, W. Haider, and C. Batunlu, "Forecasting of Wind Turbine Output Power Using Machine learning," *2020 10th Int. Conf. Adv. Comput. Inf. Technol. ACIT 2020 - Proc.*, pp. 396–399, Sep. 2020, doi: 10.1109/ACIT49673.2020.9208852.

[6] D. Jiang, Q. Huang, and L. Hong, "Theoretical and experimental study on wind wheel unbalance for a wind turbine," *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 351–355, 2009, doi: 10.1109/WNWEC.2009.5335787.

[7] P. Koetket, T. Chaichana, Y. Auttawaitkul, W. Chiracharit, S. Khunkhet, and J. Waewsak, "Increasing efficiency in wind energy electricity generating by signal processing from wind measuring equipment on wind turbine for the determination of wind direction," *2018 Int. Work. Adv. Image Technol. IWAIT 2018*, pp. 1–4, May 2018, doi: 10.1109/IWAIT.2018.8369735.

[8] K. Naoi and M. Shiono, "Relationship between Incident Angle of Wind on Rotor Blade and Output of a Drag-type Multi-blade Vertical-Axis Wind Turbine with Stationary Multi-vanes," *2021 11th Int. Conf. Power, Energy Electr. Eng. CPEEE 2021*, pp. 147–152, Feb. 2021, doi: 10.1109/CPEEE51686.2021.9383349.

[9] R. Veena, V. Femin, S. Mathew, I. Petra, and J. Hazra, "Intelligent models for the power curves of small wind turbines," *Proc. 2016 Int. Conf. Cogener. Small Power Plants Dist. Energy, ICUE 2016*, Nov. 2016, doi: 10.1109/COGEN.2016.7728965.

[10] B. Ioan, B. Horia, and O. P. T. Susana, "Determination of the power generated by a wind turbine in constant wind and variable wind," *2015 9th Int. Symp. Adv. Top. Electr. Eng. ATEE 2015*, pp. 769–773, Jun. 2015, doi: 10.1109/ATEE.2015.7133919.

[11] C. Ostia, J. M. Martinez, K. T. Coderes, G. J. Estacio, A. R. Ramchahi, and J. V. Pulido, "Development of a dual wind turbines using the savonius and Archimedes spiral principle," *2018 IEEE 10th Int. Conf. Humanoid, Nanotechnology, Inf. Technol. Commun. Control. Environ. Manag. HNICEM 2018*, Mar. 2019, doi: 10.1109/HNICEM.2018.8666238.

[12] M. Mehrshad, R. Effatnejad, and A. Mohammadpour, "Transient simulation of fixed-speed wind turbine with grid fault variety in real wind farm," *IEEE Int. Symp. Ind. Electron.*, pp. 248–253, 2014, doi: 10.1109/ISIE.2014.6864619.

[13] M. Q. Duong and G. N. Sava, "Coordinated reactive power control of DFIG to Improve LVRT characteristics of FSIG in wind turbine generation," *2017 11th Int. Conf. Electromechanical Power Syst. SIELMEN 2017 - Proc.*, vol. 2017-January, pp. 256–260, Nov. 2017, doi: 10.1109/SIELMEN.2017.8123328.

[14] Y. Dai, J. Wang, X. Liu, and C. Wen, "Optimal design of characteristic wind speed for small horizontal axis wind turbine and experimental research," *2011 Int. Conf. Electr. Inf. Control Eng. ICEICE 2011 - Proc.*, pp. 5150–5153, 2011, doi: 10.1109/ICEICE.2011.5777005.

[15] J. Wang, M. Qin, A. Cai, and D. Zhang, "Analysis of bearing capacity behavior of single bucket foundation for offshore wind turbines under eccentric horizontal loading in soft clay," *Proc. 2010 World Non-Grid-Connected Wind Power Energy Conf. WNWEC 2010*, pp. 159–164, 2010, doi: 10.1109/WNWEC.2010.5673131.

[16] L. Dong, M. Liao, Y. Li, X. Song, and K. Xu, "Study on aerodynamic design of horizontal axis wind turbine generator system," *2009 Int. Conf. Energy Environ. Technol. ICEET 2009*, vol. 1, pp. 841–844, 2009, doi: 10.1109/ICEET.2009.208.

[17] J. P. Lyons, M. C. Robinson, P. Veers, and R. W. Thresher, "Wind turbine technology - the path to 20% US electrical energy," *IEEE Power Energy Soc. 2008 Gen. Meet. Convers. Deliv. Electr. Energy 21st Century, PES*, 2008, doi: 10.1109/PES.2008.4596953.

[18] C. Zhang and J. Hu, "The UAE wind turbine performance prediction using wind tunnel airfoil data," *Proc. 2010 World Non-Grid-Connected Wind Power Energy Conf. WNWEC 2010*, pp. 79–83, 2010, doi: 10.1109/WNWEC.2010.5673181.

[19] P. Qian, X. Ma, and Y. Wang, "Condition monitoring of wind turbines based on extreme learning machine," *2015 21st Int. Conf. Autom. Comput. Autom. Comput. Manuf. New Econ. Growth, ICAC 2015*, Oct. 2015, doi: 10.1109/ICONAC.2015.7313974.

[20] N. Kodama and T. Matsuzaka, "Web-based data acquisition system of wind conditions and its application to power output variation analysis for wind turbine generation," *2006 SICE-ICASE Int. Jt. Conf.*, pp. 3747–3750, 2006, doi: 10.1109/SICE.2006.315027.

[21] D. Jiang, Q. Huang, and L. Hong, "Test system of a small wind turbine under laboratory conditions,"

WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf., pp. 283–286, 2009, doi: 10.1109/WNWEC.2009.5335789.

[22] M. Y. W. Teow, H. K. Chiu, and R. H. G. Tan, "A computational modelling of wind turbine mechanical power and its improve factor determination," *IET Conf. Publ.*, vol. 2016, no. CP688, 2016, doi: 10.1049/CP.2016.1358.

[23] Y. Li, Y. Zheng, N. Zhu, and F. Zhao, "Wind turbine kinetic energy accumulation and release regulation for wind farm optimization," *Proc. - 2019 4th Int. Conf. Mech. Control Comput. Eng. ICMCCE 2019*, pp. 231–235, Oct. 2019, doi: 10.1109/ICMCCE48743.2019.00060.

[24] M. Sateesh Kumar, Y. V. Pavan Kumar, D. John Pradeep, and C. P. Reddy, "Analysis on the Effectiveness of Vertical Axis Wind Turbine for Domestic Consumers," *2020 Int. Symp. Adv. Electr. Commun. Technol. ISAECT 2020*, Nov. 2020, doi: 10.1109/ISAECT50560.2020.9523698.

[25] Y. Xingjia, G. Changchun, X. Zuoxia, L. Yan, and L. Shu, "Variable speed wind turbine maximum power extraction based on fuzzy logic control," *2009 Int. Conf. Intell. Human-Machine Syst. Cybern. IHMSC 2009*, vol. 2, pp. 202–205, 2009, doi: 10.1109/IHMSC.2009.174.

[26] M. S. Davis, A. Jafarian, F. Ferdowsi, and M. R. Madani, "Wind energy harvesting capability of a novel cascaded dual-rotor horizontal-axis wind turbine," *Int. Conf. Electr. Comput. Commun. Mechatronics Eng. ICECCME 2021*, Oct. 2021, doi: 10.1109/ICECCME52200.2021.9590963.

[27] B. K. Saxena and K. V. S. Rao, "Wind turbine failure analysis for wind farm at Devgarh in Rajasthan," *Proc. - 2013 Int. Conf. Renew. Energy Sustain. Energy, ICRESE 2013*, pp. 196–199, Oct. 2014, doi: 10.1109/ICRESE.2013.6927814.

[28] M. H. Zamani, G. H. Riahy, and R. Z. Foroushani, "Introduction of a new index for evaluating the effect of wind dynamics on the power of variable speed wind turbines," *Transm. Distrib. Expo. Conf. 2008 IEEE PES Powering Towar. Futur. PIMS 2008*, 2008, doi: 10.1109/TDC.2008.4517174.

[29] X. Yao, Y. Liu, J. Bao, and Z. Xing, "Research and simulation of direct drive wind turbine VSCF characteristic," *Proc. IEEE Int. Conf. Autom. Logist. ICAL 2008*, pp. 1683–1687, 2008, doi: 10.1109/ICAL.2008.4636425.

[30] J. Fu, H. L. Zhang, B. He, and Q. S. Fan, "The effect of wind shear and pitch control delaying on wind turbine blade loading," *2011 Int. Conf. Electr. Inf. Control Eng. ICEICE 2011 - Proc.*, pp. 5485–5489, 2011, doi: 10.1109/ICEICE.2011.5776941.

[31] R. I. Putri, S. Adhisuwignjo, and M. Rifari, "Design of simple power converter for small scale wind turbine system for battery charger," *Proc. - 2018 3rd Int. Conf. Inf. Technol. Inf. Syst. Electr. Eng. ICITISEE 2018*, pp. 169–173, Jul. 2018, doi: 10.1109/ICITISEE.2018.8721018.

[32] X. Yao, J. Liu, Z. Xing, Y. Jing, and G. Li, "Study on power control of doubly fed wind turbine," *Chinese Control Decis. Conf. 2008, CCDC 2008*, pp. 3630–3635, 2008, doi: 10.1109/CCDC.2008.4598007.

[33] A. Hesami and A. H. Nikseresht, “Performance Enhancements of Savonius Wind Turbine using a Hybrid Augmentation System,” *7th Iran Wind Energy Conf. IWEC 2021*, May 2021, doi: 10.1109/IWEC52400.2021.9466964.

[34] M. Dehghani, “Conceptual Design and Initial Analysis of an Oscillating Wind Turbine,” *7th Iran Wind Energy Conf. IWEC 2021*, May 2021, doi: 10.1109/IWEC52400.2021.9466989.

[35] M. F. Ahmed Fayeem, A. H. Galib, and P. Saha, “Micro wind turbine as an alternative power source in Bangladesh,” *2019 Int. Conf. Sustain. Technol. Ind. 4.0, STI 2019*, Dec. 2019, doi: 10.1109/STI47673.2019.9067980.

[36] S. Chetan, C. Festila, E. H. Dulf, and R. Both-Rusu, “Analysis of a new horizontal axes wind turbine with 6/3 blades,” *2018 IEEE Int. Conf. Autom. Qual. Testing, Robot. AQTR 2018 - THETA 21st Ed. Proc.*, pp. 1–4, Jul. 2018, doi: 10.1109/AQTR.2018.8402770.

[37] Mohsen Ghalei Monfared Zanjani, Hosein Kazemi Kargar, and Mina Ghalei Monfared Zanjani, “High impedance fault detection of distribution network by phasor measurement units - IEEE Conference Publication,” *2012 Proc. 17th Conf. Electr. Power Distrib.*, 2012, Accessed: Jun. 28, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/6254586.

[38] F. Wang, D. Liu, and L. Zeng, “Study on computational grids in placement of wind turbines using genetic algorithm,” *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 369–372, 2009, doi: 10.1109/WNWEC.2009.5335776.

[39] P. Balakrishnan and V. Chenniappan, “Container wind energy for wind turbines,” *2017 Int. Conf. Comput. Commun. Informatics, ICCCI 2017*, Nov. 2017, doi: 10.1109/ICCCI.2017.8117804.

[40] M. Yin, Y. Xu, C. Shen, J. Liu, Z. Y. Dong, and Y. Zou, “Turbine Stability-Constrained Available Wind Power of Variable Speed Wind Turbines for Active Power Control,” *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 2487–2488, May 2017, doi: 10.1109/TPWRS.2016.2605012.

[41] J. Li, B. Zhou, and H. Guo, “Hardware-in-loop simulation of wind turbine based on BLDCM,” *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 125–129, 2009, doi: 10.1109/WNWEC.2009.5335825.

[42] S. Sina, “Aeroelastic Stability of Horizontal Axis Wind Turbine Blades,” *7th Iran Wind Energy Conf. IWEC 2021*, May 2021, doi: 10.1109/IWEC52400.2021.9466973.

[43] S. Shayestehnezhad, S. Kargar, and A. L. Nichkoohi, “A Numerical Study of INVELOX Wind Turbine Considering the Inlet Shape Design,” *7th Iran Wind Energy Conf. IWEC 2021*, May 2021, doi: 10.1109/IWEC52400.2021.9466962.

[44] J. Chen and D. Jiang, “Study on modeling and simulation of non-grid-connected wind turbine,” *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 292–296, 2009, doi: 10.1109/WNWEC.2009.5335791.

[45] S. K. Haghighian, H. G. Yeh, and T. Rahimi, “Analysis of Wind Generator Contribution on load Frequency Control Considering Control Signals Delays,” *7th Iran Wind Energy Conf. IWEC 2021*, May 2021, doi: 10.1109/IWEC52400.2021.9466980.

[46] Y. Dai, Y. Lu, Y. Sun, and Y. He, “Wind energy resource assessment and derivation of characteristic wind speed in wind field of Inner Mongolia,” *2011 Int. Conf. Consum. Electron. Commun. Networks, CECNet 2011 - Proc.*, pp. 3787–3790, 2011, doi: 10.1109/CECNET.2011.5769431.

[47] B. G. Kumaraswamy, B. K. Keshavan, and Y. T. Ravikiran, “Analysis of seasonal wind speed and wind power density distribution in Aimangala wind form at Chitradurga Karnataka using two parameter weibull distribution function,” *IEEE Power Energy Soc. Gen. Meet.*, 2011, doi: 10.1109/PES.2011.6039587.

[48] V. Shirsath and R. Agrawal, “A Review of Wind Station Data Modeling for Wind Turbine Reliability Enhancement to Optimize Wind Energy Considering Turbine Design,” *2020 5th IEEE Int. Conf. Recent Adv. Innov. Eng. ICRAIE 2020 - Proceeding*, Dec. 2020, doi: 10.1109/ICRAIE51050.2020.9358335.

[49] Y. Ma, Z. Hu, J. Wang, R. Lai, and Y. Xing, “Research on fixed-pitch wind turbine running in deep stall region,” *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 267–272, 2009, doi: 10.1109/WNWEC.2009.5335802.

[50] C. Zhao, J. Luo, S. Xie, and H. Li, “Experiment validation of vertical axis wind turbine control system based on wind energy utilization coefficient characteristics,” *2015 IEEE Int. Conf. Cyber Technol. Autom. Control Intell. Syst. IEEE-CYBER 2015*, pp. 1640–1644, Oct. 2015, doi: 10.1109/CYBER.2015.7288191.

[51] B. Glushakow and H. Qin, “Wind Turbine Lightning Protection of Blades, Electronics and Humans,” *2019 15th Int. Symp. Light. Prot. SIPDA 2019*, Sep. 2019, doi: 10.1109/SIPDA47030.2019.8951561.

[52] O. Beik and A. Al-Adsani, “Wind turbine productivity and wind energy assessment: An ontario case study,” *2020 IEEE Electr. Power Energy Conf. EPEC 2020*, Nov. 2020, doi: 10.1109/EPEC48502.2020.9320110.

[53] C. Wang, Z. Fu, W. Huang, Y. Liu, and S. Zhang, “Doubly-fed wind turbine mathematical model and simulation,” *Proc. - 2014 Int. Symp. Comput. Consum. Control. IS3C 2014*, pp. 793–795, 2014, doi: 10.1109/IS3C.2014.210.

[54] W. Cui, F. Yu, X. Liu, and J. Whitty, “Analysis of the passive yaw mechanism of small horizontal-axis wind turbines,” *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 105–109, 2009, doi: 10.1109/WNWEC.2009.5335837.

[55] Y. Zhu, R. Zhao, and J. Zhao, “Output power smoothing control for the PMSG based wind farm by using the allocation of the wind turbines,” *2017 20th Int. Conf. Electr. Mach. Syst. ICEMS 2017*, Oct. 2017, doi: 10.1109/ICEMS.2017.8055952.

[56] M. Altin, O. Goksu, A. D. Hansen, and P. E. Sorensen, “Aggregated wind power plant models consisting

of IEC wind turbine models," *2015 IEEE Eindhoven PowerTech, PowerTech 2015*, Aug. 2015, doi: 10.1109/PTC.2015.7232807.

[57] H. T. Ma, P. B. Brogan, K. H. Jensen, and R. J. Nelson, "Sub-synchronous control interaction studies between full-converter wind turbines and series-compensated ac transmission lines," *IEEE Power Energy Soc. Gen. Meet.*, 2012, doi: 10.1109/PESGM.2012.6345523.

[58] M. Hanafi, H. Akram, D. El Messoudi, and T. Chafik, "Study of the Wind Effect on the Quality and Lubrication System of Wind Turbines," *Proc. 2017 Int. Renew. Sustain. Energy Conf. IRSEC 2017*, Sep. 2018, doi: 10.1109/IRSEC.2017.8477293.

[59] F. Wang, D. Liu, and L. Zeng, "Modeling and simulation of optimal wind turbine configurations in wind farms," *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 416–420, 2009, doi: 10.1109/WNWEC.2009.5335756.

[60] A. Omidi and M. Kalantar, "Improved Fault Ride through Strategy of Doubly Fed Induction Generator based Wind Turbine Using Model Predictive Control," *7th Iran Wind Energy Conf. IWEC 2021*, May 2021, doi: 10.1109/IWEC52400.2021.9467000.

[61] R. Chen, J. Chen, Z. Chen, C. Gong, and Y. Yan, "An aerodynamic characteristic measurement method for fixed-pitch wind turbine," *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 246–250, 2009, doi: 10.1109/WNWEC.2009.5335797.

[62] J. Zhang, Z. Zhou, and Y. Lei, "Design and research of high-performance low-speed wind turbine blades," *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 161–165, 2009, doi: 10.1109/WNWEC.2009.5335818.

[63] J. Kazda, M. Mirzaei, and N. A. Cutululis, "On the Architecture of Wind Turbine Control Required for Induction-based Optimal Wind Farm Control," *Proc. Am. Control Conf.*, vol. 2018-June, pp. 3074–3079, Aug. 2018, doi: 10.23919/ACC.2018.8431663.

[64] M. Qiu, Z. Zhou, and J. Zhang, "Three-dimensional flow simulation for horizontal axis wind turbine," *WNWEC 2009 - 2009 World Non-Grid-Connected Wind Power Energy Conf.*, pp. 171–175, 2009, doi: 10.1109/WNWEC.2009.5335820.

[65] F. Meng, J. Wenske, and A. Gambier, "Wind turbine loads reduction using feedforward feedback collective pitch control based on the estimated effective wind speed," *Proc. Am. Control Conf.*, vol. 2016-July, pp. 2289–2294, Jul. 2016, doi: 10.1109/ACC.2016.7525259.

[66] X. Wang, Z. Jiang, H. Lu, X. Wang, Y. Meng, and S. Li, "Independent Pitch Control Strategy and Simulation for Reducing Unbalanced Load of Wind Turbine," *Proc. 32nd Chinese Control Decis. Conf. CCDC 2020*, pp. 5535–5539, Aug. 2020, doi: 10.1109/CCDC49329.2020.9164293.

[67] L. Arnedo, S. Dwari, S. Motapon, and V. Blasko, "System level wind turbine controls with seamless transitions between standalone and grid connected mode," *PEMWA 2012 - 2012 IEEE Power Electron.*

Mach. Wind Appl., 2012, doi: 10.1109/PEMWA.2012.6316389.

[68] M. S. Genc and K. Sekhoune Ozden, “Flow Physics Analysis of a Vertical Axis Wind Turbine Using FloEFD,” *7th Iran Wind Energy Conf. IWEC 2021*, May 2021, doi: 10.1109/IWEC52400.2021.9467011.

[69] P. Sorensen, B. Andresen, J. Fortmann, and P. Pourbeik, “Modular structure of wind turbine models in IEC 61400-27-1,” *IEEE Power Energy Soc. Gen. Meet.*, 2013, doi: 10.1109/PESMG.2013.6672279.

[70] Z. Mingqiu, H. Xiangsheng, and L. Zhi, “Study on tower load measurements of the offshore wind turbine,” *2019 4th Int. Conf. Power Renew. Energy, ICPRE 2019*, pp. 123–126, Sep. 2019, doi: 10.1109/ICPRE48497.2019.9034840.

[71] L. M. Fernandez, C. A. Garcia, F. Jurado, and J. R. Saenz, “Aggregation of doubly fed induction generators wind turbines under different incoming wind speeds,” *2005 IEEE Russ. Power Tech, PowerTech*, 2005, doi: 10.1109/PTC.2005.4524685.

[72] A. Martyanov, N. Martyanov, and E. Sirotkin, “State Observer for Variable Speed Wind Turbine,” *Proc. - 2018 Int. Ural Conf. Green Energy, Ural. 2018*, pp. 97–100, Nov. 2018, doi: 10.1109/URALCON.2018.8544344.

[73] S. Khan, “Conflicts in Betz Limit and An Alternative Approach for Wind Turbines,” *2020 IEEE Reg. 10 Symp. TENSYMP 2020*, pp. 1438–1443, Jun. 2020, doi: 10.1109/TENSYMP50017.2020.9230876.

[74] E. Rebello, D. Watson, and M. Rodgers, “Practical aspects and results - Providing secondary frequency regulation from a single type 4 wind turbine and a battery storage system,” *IEEE Green Technol. Conf.*, vol. 2021-April, pp. 105–111, Apr. 2021, doi: 10.1109/GREENTECH48523.2021.00027.

[75] I. Alkalbani and F. M. Guangul, “Economical Evaluation of Energy Harvesting Using Vertical Axis Wind Turbine from a Wind Turbulence Created by Moving Cars,” *2021 Int. Conf. Smart City Green Energy, ICSCGE 2021*, pp. 26–29, 2021, doi: 10.1109/ICSCGE53744.2021.9654414.