Deanship of Graduate Studies
Al-Quds University

Wind Speed Prediction Using Machine Learning Algorithms: A Case Study of Using ANFIS and KNNR

Khalil Shiban Mahmoud Abuayyash

M.Sc. Thesis

Jerusalem – Palestine

1445/2023

Wind Speed Prediction Using Machine Learning Algorithms: A Case Study of Using ANFIS and KNNR

Prepared By:
Khalil Shiban Mahmoud Abuayyash

B.Sc.: Al-Quds University, Palestine

Supervisor:

Dr. Hazem Doufesh, Physics Department, Al-Quds University, Palestine

Co-Supervisor:

Dr. Husain Alsamamra, Physics Department, Al-Quds University, Palestine

A Thesis Submitted in Partial fulfillment of requirements for the degree of Master of Science from the Department of Physics, Faculty of Science and Technology, Al-Quds University.

1445/2023

Al-Quds University
Deanship of Graduate Studies
Physics Department

Thesis Approval

Wind Speed Prediction Using Machine Learning Algorithms: A Case Study of Using ANFIS and KNNR

Prepared By: Khalil Shiban Mahmoud Abuayyash
Registration No: 21610084

Supervisor:
Dr. Hazem Doufesh, Physics Department, Al-Quds University, Palestine.
Co-Supervisor:
Dr. Husain Alsamamra, Physics Department, Al-Quds University, Palestine.

Master thesis submitted and accepted, Date: 26/07/2023.
The names and signatures of the examining committee members are as follows:

1- Head of Committee: Signature:
2- Internal Examiner: Signature:
3- External Examiner: Signature:

Jerusalem – Palestine

1445/2023

Dedication

I dedicate this thesis to my family.

Declaration:

I certify that this thesis submitted for the degree of the master is the result of my research, except where otherwise acknowledged, and that this thesis, neither in whole nor in part, has been previously submitted for any degree to any other university or institution.

Signed: Khalil

Khalil Shiban Mahmoud Abuayyash

Date: 26/07/2023

Acknowledgments

I would like to express my sincere gratitude to all those who have supported and contributed to the completion of this thesis. Without their assistance, guidance, and encouragement, this research would not have been possible.

First and foremost, I am deeply indebted to my supervisors, Dr. Hazem Doufesh and Dr. Husain Alsamamra, for their unwavering support and invaluable guidance throughout this entire journey. Their expertise, patience, and commitment to academic excellence have been instrumental in shaping this thesis and my growth as a researcher.

I am immensely grateful to my colleges in the physics department, especially for Dr. Fadi Shaheen, Mr. Mohammed Abdelkader, for their help. They dedicated time to provide advice and discuss theoretical and technical issues.

My deepest appreciation goes to my friends and family who have supported me unconditionally throughout this academic pursuit. Their love, encouragement, and belief in my abilities have been a constant source of motivation, especially during moments of self-doubt.

Abstract

Wind speed prediction using machine learning algorithms is crucial for various applications, such as wind energy planning and urban development. This paper presents a case study on wind speed prediction in Jerusalem using the Adaptive Neuro-Fuzzy Inference System (ANFIS) and K-Nearest Neighbors Regression (KNNR) algorithms. The study evaluates their performance using multiple metrics, including root mean square (RMSE), bias, and coefficient of determination R2. ANFIS demonstrates good accuracy with lower RMSE (0.196) and minimal bias (0.0003). However, there is room for improvement in capturing overall variability (R2 = 0.15). In contrast, KNNR exhibits a higher R2 (0.4093), indicating a better fit, but with a higher RMSE (1.4209). This study provides insights into the applicability of ANFIS and KNNR in wind speed prediction for Jerusalem and suggests future research directions. The outcomes have practical implications for wind energy planning, urban development, and environmental assessments in similar regions.

Table of Contents

List of Tables

List of Figures

List of abbreviations

AI	Artificial Intelligence
ANFIS	Adaptive Neuro-Fuzzy Inference System
ANN	Artificial Neural Network
AR	AutoRegressive
ARIMA	AutoRegressive Integrated Moving Average
ARMA	AutoRegressive Moving Average
CVNN	Complex-Valued Neural Network
DRPE	Decoupled Recursive prediction Error
EMD	Empirical Mode Decomposition
FFBP	Feedforward Backpropagation
FNN	Feedforward Neural Network
GA	Genetic Algorithm
GRPE	Global Recursive Prediction Error
KNN	K Nearest Neighbors
KNNR	K Nearest Neighbors Regression
LF-DFNN	local feedback dynamic fuzzy neural network
LRRN	Local recurrent neural network
MA	Moving Average
MAE	Mean Absolute Error
MAS	Multiple Architecture System
MBA	Bayesian Model Averaging
MCP	Measure-Correlate-Predict
MF	Membership Function
ML	Machine Learning
MLE	Maximum Likelihood Estimation
MRI	Meteorological Risk Index
MSE	Mean Squared Error
NWP	Numerical Weather Prediction
PSO	Particle Swarm Optimization
R2	Coefficient of determination
RBF	Radial Basis Function
RMSE	Root Mean Square Error

SD	Similar Days technique
SDE	Standard Deviation of Errors
STD	Standard Deviation
SVM	Support Vector Machine
TISM	Improved Time Series Method
WT	Wavelet Transform

Definitions

Term	Definition
Machine learning	A type of artificial intelligence that involves training computer algorithms to learn patterns and relationships in data without being explicitly programmed.
Artificial intelligence	The ability of machines to perform tasks that typically require human intelligence, such as perception, reasoning, learning, and decision-making.
Artificial neural network	A computational model inspired by the structure and function of biological neurons in the brain that is used to learn patterns and relationships in data.
Fuzzy logic	A mathematical framework for dealing with uncertainty and imprecision that is based on degrees of truth and membership in sets.
Fuzzy inference system	A system that uses fuzzy logic to make decisions or predictions based on uncertain or imprecise data.
Regression	A statistical technique used to model the relationship between a dependent variable and one or more independent variables.
K nearest neighbor regressor	A machine learning algorithm that predicts the value of a new data point based on the values of its nearest neighbors in the training data.
Adaptive Neuro-Fuzzy Inference System	A type of fuzzy inference system that uses a neural network to learn the parameters of the fuzzy logic rules.
Renewable energy	Energy that comes from sources that are replenished naturally and rapidly, such as sunlight, wind, water, and geothermal heat.

Chapter 1 Introduction

Meeting the energy needs of the global population is viewed as a significant hurdle, prompting the emergence of new concepts and innovations. The demand for accessible and cost-effective energy is steadily growing. According to Figure (1.1a) based on data from (World Bank, 2013), a positive correlation exists between energy consumption and wealth, with individuals in affluent nations consuming greater amounts of energy compared to those in less developed countries.

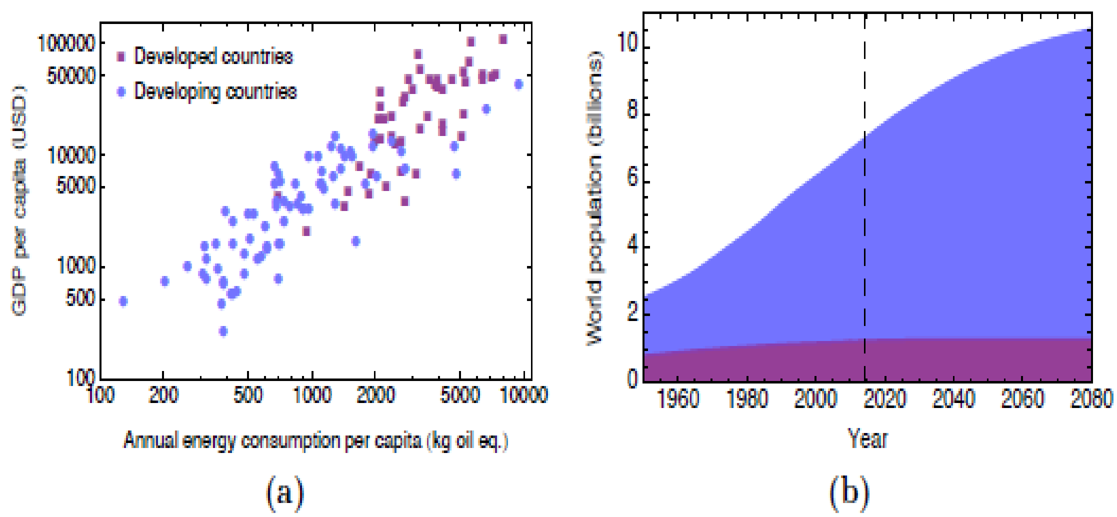

Figure 1.1 : a) GDP per capita vs. annual energy consumption (World Bank, 2013). **b) World population estimates (1950 -2080)** (Until, 2013).

The economic expansion experienced by developing economies directly contributes to a rise in their energy consumption. Consequently, it becomes crucial to ensure a continuous provision of affordable energy to meet the needs of these developing economies. Figure (1.1b), based on data from the United Nations Department of Economic and Social Affairs in 2013, illustrates the projected global population growth until 2080. It is evident that most of this growth is anticipated to occur in developing countries (blue). The combination of a growing population in these nations, along with an increase in energy demand per capita due to their economic growth, will result in a substantial surge in energy demand throughout the 21st century.

The most used energy is non-renewable energy. They are dominant in the electricity generating sector. However, the use of non-renewable energy has some disadvantages (Kabeyi & Olanrewaju, 2022). It is the main source of greenhouse gases, its price is not stable especially in the times of conflicts, and electricity power generation station based on oil tend to be centralized.

Generation of electricity using renewable energy could overcome these issues. Renewable energy sources are attracting more and more attention from electricity generation authorities, and energy companies because they are natural, free, and environmentally friendly. Among various renewable energy sources, wind energy recognized as one of the most significant and potentially valuable energy sources (W. Y. Chang, 2014; Jung & Broadwater, 2014; X. Wang et al., 2011).

In the field of meteorology, wind speed is a measurable parameter that arises from the movement of air between areas of high pressure and low pressure, driven by temperature fluctuations. It is typically quantified in meters per second using an anemometer. Wind speed plays a pivotal role in several areas including weather forecasting, aviation and maritime operations, construction projects, as well as the growth and metabolic processes of numerous plant species.

Wind speed forecasting has an essential role in designing and operation of wind turbines. A reduction in costs and enhancement of the proper utilization of resources could be achieved by accurate wind speed prediction. Accurately predicting wind speed plays a crucial role in effectively scheduling aviation and maritime operations, as well as construction projects. It enables these industries to plan and adjust their activities based on the prevailing wind conditions, ensuring safety and optimal performance.

Moreover, wind speed prediction is valuable for farmers as it assists them in making informed decisions about which plant species to cultivate each year. By considering the expected wind patterns, farmers can select plant varieties that are better adapted to specific wind conditions. This helps optimize crop yields, reduce potential damage from strong winds, and enhance overall agricultural productivity.

1.1 Objectives:

The objectives of this work include:

- Improving wind speed prediction accuracy in Jerusalem using ANFIS and KNNR algorithms.
- Conducting a comparison between ANFIS and KNNR to determine which of both outperforms the other.

1.2 Problem Statement:

Wind energy is acknowledged for its status as a renewable energy source that offers several advantages, including its low cost of electricity generation, abundant availability, high efficiency, and minimal environmental impact (Chang, 2013c). Consequently, numerous countries are increasingly acknowledging the potential of wind energy as a significant opportunity for future electricity generation. As a result, the installed capacity for wind energy is growing at an annual rate of approximately 13 percent (*Growth Rate of Installed Wind Power Capacity Worldwide from 2012 to 2021*, 2023).

Nevertheless, the power output of wind turbines, which are used for converting wind energy into electricity, is influenced by atmospheric meteorology and wind speed (W. Y. Chang, 2013b, 2013a). This dependence on these factors means that unexpected fluctuations in wind turbine power generation can lead to higher operating costs for the electricity system. This is primarily due to an increased demand for primary reserves and can potentially jeopardize the reliability of the electricity supply (Sideratos & Hatziargyriou, 2007).

In an electricity system, it is essential to continuously match the electricity supply with the demand. However, the variability in wind power output poses challenges in maintaining this balance. Enhancing the prediction accuracy of wind speed and power represents a viable solution to address this issue. By improving the precision of wind forecasts, electricity system operators can better anticipate fluctuations in wind power generation and make necessary adjustments to ensure a reliable and balanced electricity supply.

In addition to the balance challenge, the growing integration of wind energy into the power grid has brought about a multitude of challenges that need to be addressed. These challenges encompass several areas, including electricity market clearing, real-time grid operations, ancillary service needs and associated costs, maintaining competitive power quality, as well as ensuring power system stability and reliability (Wu & Hong, 2007).

Enhancing wind forecasting can be regarded as one of ways to address many of these issues. Since forecasting changes in wind power production is necessary for planning spinning reserve capacity and facilitating the management of grid operations (Sideratos & Hatziargyriou, 2007). Accurate prediction of wind speed is required to reduce reserve capacity and increase the share of wind energy (Lei et al., 2009).

In addition, wind power forecasting plays a crucial role in scheduling conventional power plants in advance and facilitating power trading activities in the spot market. (Lange & Focken, 2008). Therefore, improving the performance of wind speed and wind power forecasts has substantial economic and technical implications for the system. Many studies have concluded that wind energy is applicable in Palestine, especially in high areas such as Hebron (wind speed up to 7.5 m/s) and Jerusalem (Albisher & Alsamamra, 2019).

Predicting wind speed is recognized as a crucial yet challenging task in forecasting wind power for electricity generation. While numerous wind speed prediction models have been developed globally, their applicability is limited to specific locations as they are site dependent. Consequently, the implementation of machine learning algorithms utilizing local data becomes imperative to accurately predict wind speed locally in East Jerusalem. By leveraging local data and machine learning techniques, more precise wind speed predictions can be achieved, contributing to improved wind power forecasting for electricity generation in the region.

Chapter 2 Literature Review

Various methods are available for wind speed prediction, classified by time scale or methodology employed. The classification of wind speed prediction methods based on the time scale varies in the different literature sources. One classification includes three categories: immediate short-term forecasting, short-term forecasting (day-ahead), and long-term forecasting (several days ahead) (X. Wang et al., 2011). Another classification includes four categories: Ultra-short-term forecasts (a few minutes to an hour ahead), Short-term forecasts (an hour to several hours ahead), Medium-term forecasts (a few hours to a week ahead), and Long-term forecasts (a week to a year or more ahead) (W. Y. Chang, 2014; Zhao Dongmei et al., 2011; X. Zhao et al., 2011).

In this study, the second classification is used. The classification based on time scale is important because there are different applications for each category. The applications of ultra-short-term forecasting are power market clearing, real-time grid operations, regulation actions. The applications of the short-term are economic load dispatch planning, appropriate load decisions, power market operational security. The applications of the medium-term are unit commitment decisions, reserve requirements decisions, online/offline generator decisions. The applications of the long-term are maintenance planning, operations management, optimal operating cost, feasibility study for design of the wind farm.

The classification of wind speed prediction methods based on the methodology varies in the different literature descriptions. However, making a comparison between many studies (W. Y. Chang, 2014; Jung & Broadwater, 2014; X. Wang et al., 2011), the methods can be divided into seven distinct categories, namely: persistence method, physical methods, statistical methods, spatial methods, regional methods, hybrid methods, and probabilistic methods.

2.1 The Persistence Method

The persistence method is a straightforward approach that assumes the wind speed at a future time will remain the same as the current wind speed at the time of prediction (X. Zhao et al., 2011). In this method, if the actual wind speed at time t is denoted as v(t), then the predicted wind speed at $t + \Delta t$ can be represented as the follows:

$$v(t + \Delta t) = v(t) \tag{2.1}$$

In addition to being the simplest method, the persistence method is also the most cost-effective option available. Given the frequent use of the persistence method by electric utilities for ultra short-term predictions, it is crucial to evaluate any newly developed forecasting method against this classical benchmark. Such a comparison is necessary to assess the potential improvement in forecast accuracy that the new method can offer compared to the forecasts obtained from the persistence method (W. Y. Chang, 2013c).

While the persistence method demonstrates relatively higher accuracy than other ultra-short-term wind forecasting methods in certain aspects, its accuracy declines significantly as the time scale of the forecast increases (Wu & Hong, 2007).

2.2 Physical Methods

The physics methods are based on computational fluid dynamics to simulate the behavior of the atmosphere. These methods are particularly valuable for ultra short-term and short-term forecasting as atmospheric dynamics play a significant role during these time scales. Therefore, the utilization of physics methods becomes essential in accurately predicting wind speed and power during these forecast intervals.

The physical methods use weather forecast data and a detailed physical description to model the wind farm site conditions (Kariniotakis et al., 2004; Lange & Focken, 2006). Physics-based methods refine numerical weather prediction (NWP) data by incorporating site-specific conditions through a downscaling approach that focuses on the physics of the lower atmospheric boundary layer. This enables a more detailed and accurate prediction of wind speed and power by considering the unique characteristics of the local environment.

For the downscaling method, a comprehensive physical description of both the wind farm and its surrounding is required, consisting of two parts; wind farm description which includes details such as the layout of the wind farm, the characteristics of individual turbines (including their power curves), and other relevant information specific to the wind farm setup, and terrain description, this part involves providing a detailed account of the terrain surrounding the wind farm, including the roughness of the terrain and the presence of any obstacles that may impact the wind flow patterns.

2.3 Statistical Methods

The statistical methods are easy to model and are a cost-effective alternative to the physical methods. The statistical methods do not require the acquisition of physical data (which is necessary with the physical methods and is one of the major drawbacks of these methods) and instead use historical data. The statistical methods present the relationship between the wind speed prediction and the explanatory variables, including the NWPs and the on-line measured data (Kariniotakis et al., 2004).

The statistical methods use historical data to build a statistical model that predicts the present over the next few hours using the NWP forecast for time t + k and the on-line measurement for time t. The statistical methods can be divided into four categories.

2.3.1 Conventional Statistical Methods

Conventional statistical methods rely on the utilization of time series models to represent the underlying problem and facilitate the prediction of future values. The process of formulating a time series model typically involves four key steps: model identification and estimation, followed by model diagnostics checking, and ultimately, forecasting.

Within this framework, different types of time series models can be applied, including the following:

1. Autoregressive Model (AR): This model considers the past wind speed values to forecast future values, assuming a linear dependence between the current value and its preceding values.

2. Moving Average Model (MA): This model utilizes past forecast errors to predict future wind speeds, assuming a linear relationship between the current value and previous forecast errors.

3. Autoregressive Moving Average Model (ARMA): Combining elements of both autoregressive and moving average models, this approach incorporates past wind speed values and forecast errors to generate future predictions.

4. Autoregressive Integrated Moving Average Model (ARIMA): Extending the ARMA model, ARIMA incorporates differencing to achieve stationarity in the time series. It involves three components: autoregressive (AR), differencing (I), and moving average (MA).

The ARMA model is used in (Milligan et al., 2003) for wind speed prediction in US wind farms. Multiple ARMA models with varying parameters were utilized, with some demonstrating improvements compared to the persistence model. It is worth noting that the performance of these ARMA models is significantly influenced by their specific parameters.

In the study conducted by Liu et al. in 2010, the improved time series method (ITSM) based on ARIMA is utilized to predict the sub-wind speed series obtained from wavelet decomposition (Liu et al., 2010). Similarly, in another study (Peng Lv & Lili Yue, 2011), the Autoregressive Conditional Heteroscedastic (ARCH) model is combined with ARIMA for prediction purposes. In both cases, the results demonstrate that the proposed methods enhance the accuracy of predictions when compared to classical time series models.

In another study conducted by M.-D. Wang et al. in 2012, the ARIMA-ARCH model is employed for directly predicting the wind speed instead of the sub-wind speed series. The performance of this model surpasses that of the ARIMA model, indicating its improved predictive capability (M.-D. Wang et al., 2012).

In (Erdem & Shi, 2011), four distinct approaches based on the ARMA model were introduced and evaluated for predicting the wind velocity vector encompassing both wind speed and direction. These approaches were the component approach, traditional-linked ARMA, vector AR (VAR), and restricted VAR methods. The findings indicated that the component approach exhibited superior performance in predicting wind direction compared to the traditional-linked ARMA method. However, for wind speed prediction, the traditional-linked ARMA approach outperformed the component approach. Moreover, the VAR and restricted VAR approaches demonstrated nearly identical prediction performance in this context.

The AR model in combination with a Bayesian approach is employed in (Miranda & Dunn, 2006). The estimation of the AR model parameters was performed through a Monte Carlo simulation. The simulation results show that the Bayesian approach has great potential for wind speed prediction.

2.3.2 Machine Learning Methods

The architecture of an Artificial Neural Network (ANN) typically consists of an input layer, one or more hidden layers, and an output layer. Within each layer, artificial neurons are interconnected with the neurons from the preceding layer. This structure enables the ANN to effectively capture intricate nonlinear relationships between the input and output layers through a training process. Unlike previously mentioned physical and statistical methods that rely on explicit mathematical expressions, ANN models leverage large sets of historical data to learn and make predictions. Different combinations of the ANN architecture can be achieved by varying the number of hidden layers, the number of neurons in each layer, and the activation function employed. Short-term wind energy forecasting commonly makes use of ANN models due to their flexibility and ability to handle complex patterns. (Kalogirou, 2001).

An ANN method is employed in (Sfetsos, 2002) to predict hourly wind speed data using time series analysis. The suggested method offers an added extra advantage for utilities that use hourly intervals and have substantial wind penetration. In (W. Y. Chang, 2013d), a backpropagation neural network-based model is used for short-term wind predictions, it shows very good accuracy. In (More & Deo, 2003), two methods are used for wind prediction: backpropagation neural network and recurrent neural network. The results indicate that neural network prediction outperforms conventional statistical time series analysis in terms of accuracy.

In (Guo et al., 2012), a multilayer Feed-forward Neural Network based on modified Empirical Mode Decomposition (EMD-based FNN) is used to predict wind speed on a monthly and daily basis in Zhangye city, China. The study findings demonstrate that the proposed method exhibits superior performance compared to both the baseline FNN and the unmodified EMD-based FNN.

In (Li & Shi, 2010), a comprehensive comparative analysis of three different types of ANN approaches is conducted. The approaches include Feed Forward Back-Propagation (FFBP), Radial Basis Function (RBF), and Adaptive Linear Element (ADALINE) methods. The researchers investigate the influence of various factors, such as input variables, learning rates, and model structures, on the accuracy of wind speed predictions. Interestingly, the results indicate that the accuracy of each approach varies depending on these factors. Furthermore, it is observed that even when using the same wind dataset, no single approach consistently outperforms the others.

The research conducted by (Kani & Riahy, 2008) introduces a novel approach to predict the wind speed in the short-term period using ANN in combination with the Markov chain method. The ANN is initially employed to provide a preliminary prediction, while the Markov chain is utilized to refine this prediction based on long-term patterns. The performance is evaluated using data recorded at intervals of 2.5 seconds. The results demonstrate the effectiveness of this approach in enhancing the accuracy of short-term wind speed predictions.

In (Jursa R, 2007), the ANN approach with Particle Swarm Optimization (PSO) for short-term wind speed prediction is presented. PSO is used to choose the input variable from a list of neighboring sites. Compared to ANN, the suggested method reduces the prediction error. Moreover, (Amjady et al., 2011) proposes a prediction approach that combines Modified Hybrid

Neural Network (MHNN) and a new Enhanced Particle Swarm Optimization (EPSO) to achieve high learning capability while avoiding overfitting and falling into local minima. The improved Mutual Information (MI) method is used to pick the most informative input values for the prediction method by filtering irrelevant and redundant input values. The findings support the validity of the presented method.

In (Kitajima & Yasuno, 2010), a Complex-Valued Neural Network (CVNN) is proposed for wind speed prediction. CVNN uses vector quantities (wind speed and direction) as inputs instead of real-valued data. The findings indicate that the accuracy of the predictions surpasses that of the Real-Valued Neural Network (RVNN).

In (Catalão et al., 2011b), a three-layer feed-forward network is utilized, trained using the Levenberg-Marquardt method, to predict subseries of wind speed by employing the Wavelet Transform (WT). The resulting subseries is then reconstructed into the future wind power series through inverse WT. To demonstrate the efficiency of the suggested approach in terms of prediction precision and computational speed, a comparison is made against the reference method.

Due to ANFIS being a pivotal algorithm in machine learning, especially for predicting wind speed in this study, it is crucial to dedicate a subsection within the literature review to comprehensively discuss its significance and relevance.

2.3.2.1 ANFIS:

The fuzzy logic approach employs a nonlinear mapping technique that utilizes linguistic variables (such as low, medium, and high) and a truth variable that ranges from zero to one. This approach is valuable in situations where accurately modeling a system is challenging, but an imprecise model exists. By allowing the utilization of approximations and handling fuzzy data, fuzzy logic helps overcome these difficulties. However, it is important to note that relying solely on fuzzy logic is not entirely satisfactory due to its limited learning capability.

The ANN-fuzzy technique is a hybrid approach that combines the strengths of ANN and fuzzy logic, where ANN is particularly effective in processing fundamental computations using unprocessed data, while fuzzy logic is better suited for complex computations involving advanced reasoning similar to human thinking. By leveraging the advantages of both methods, the approach achieves a promising strategy for various prediction applications, effectively addressing the limitations of each individual method.

In (Sideratos & Hatziargyriou, 2007), a combination of ANN and a fuzzy logic approach is employed to optimize the utilization of NWPs. The process begins with the ANN model providing an initial wind speed forecast utilizing the NWPs. Subsequently, the fuzzy model assesses the accuracy of the predictions generated by the NWPs. Finally, these evaluations are utilized by an ANN model to generate the final predictions. This integration optimizes the utilization of NWPs and enables the generation of accurate predictions for wind speed by leveraging the strengths of both techniques as confirmed by the simulation results.

In their study, (Catalão et al., 2011a) the authors present a hybrid approach combining ANFIS with two methods. The first method is the hybrid wavelet transform, utilized to transform the wind speed series into a collection of consecutive subseries. The second method is particle swarm optimization, employed to adjust the membership function parameters of the ANFIS. The ANFIS independently forecasts the converted subseries, utilizing the benefits of both the wavelet transform and particle swarm optimization techniques. The results obtained with the hybrid WPA method in comparison to the reference approaches were promising, showcasing its potential and effectiveness in addressing the prediction task at hand.

(Hong et al., 2010) propose a method that combines the Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm with ANFIS. The SPSA algorithm is used to train the ANFIS neural network. The experimental results indicate that the proposed method outperforms the reference method, demonstrating its superior performance in the given context.

2.4 Hybrid Methods

The primary concept behind hybrid models is to integrate different approaches to take advantage of each (Wu & Hong, 2007). While the primary objective of combining models is typically to enhance prediction accuracy, it is important to note that this approach does not guarantee superior performance compared to the best individual models in all cases. However, combining models is credible because it optimizes information that is limited in the individual models (Zhao Dongmei et al., 2011). Hybrid models incorporate multiple approaches, such as a mixture of physical and statistical methods or a mixture of short- and medium-term models (Soman et al., 2010).

Various approaches involving the combination of different models have been developed to enhance prediction capabilities. As mentioned earlier, one such combination approach is the ANFIS method. Shi et al in (Shi et al., 2012) present two hybrid models that combine the ARIMA method with different techniques. The first hybrid model integrates ARIMA with ANN, while the second hybrid model combines ARIMA with Support Vector Machines (SVM). The research investigates the applicability of these hybrid models through two time-horizons. In these models, the ARIMA method is employed to capture linear features, while the other methods are utilized to capture nonlinear features. The findings indicate that the hybrid approaches offer feasible alternatives for predicting wind speed time series. However, it is important to note that these hybrid models do not consistently outperform the individual methods across all prediction time horizons examined in the study.

In (Guo et al., 2011), a novel hybrid approach to wind speed prediction was presented. This approach combines a backpropagation neural network with the concept of removing seasonal influences from real wind speed datasets using seasonal exponential fitting. The experimental results demonstrated that the proposed method outperformed a single neural backpropagation network.

The study conducted by (Catalão et al., 2011b), introduces a hybrid model for short-term wind speed forecasting. This model combines ANN with the Wavelet Transform (WT) technique. By utilizing the wavelet transform, the wind speed series is decomposed into more informative constitutive series. The experimental results demonstrate that the proposed hybrid approach is very useful for wind speed forecasting.

In (Sánchez, 2006), a hybrid model of nine time series models is proposed, the ultimate prediction is computed by dynamically adjusting a linear combination of the different predictions, each prediction has its weight based on its performance. The findings indicate that the hybrid model surpasses the performance of the individual time series.

(Bouzgou & Benoudjit, 2011) proposes a Multiple Architecture System (MAS) for the purpose of predicting wind speed. Multilayer Perceptron (MLP) neural networks, multiple linear (MLR)-based regression, SVM regression and Radial Basis Function (RBF) neural networks are examined as suitable models for creating the ensemble forecast. Three combining techniques are investigated using an ANN method: nonlinear fusion, simple average, and weighted average. In comparison to the single prediction, the recommended combination strategies boost performance.

The study conducted by (Qin et al., 2011) investigates the hybrid approach RBF neural networks and persistence methods. According to the experimental results, the persistence method is better for random data with white noise. On the other hand, the RBF method proves to be more suitable for predicting wind speed changes that exhibit a monotonic pattern. The simulation findings show improvements in the hybrid approach.

In the study conducted by (Li et al., 2011) a novel approach known as Bayesian Model Averaging (BMA) is introduced to integrate wind speed forecasts generated by diverse ANN models. Specifically, the research explores three distinct ANN models, namely the RBF network, Back-Propagation (BP) network, and ADALINE network. The BMA methodology assigns weights to individual forecasts based on their corresponding posterior model probabilities, assigning higher weights to superior forecasts and lower weights to inferior ones. The findings provide compelling evidence supporting the efficacy of the proposed hybrid approach, affirming its ability to enhance wind speed prediction accuracy.

In the work conducted by (Haque et al., 2012), a comprehensive analysis of various short-term wind speed prediction methodologies is presented, encompassing the Backpropagation Neural Network (BPNN), Radial Basis Function Neural Network (RBFNN), ANFIS. To enhance the prediction performance of these models, a technique known as similar days (SD) is incorporated, which considers historical weather data corresponding to the predicted day to identify similar wind speed days for further analysis. The experimental findings demonstrate notable improvements in the performance of all the examined models when the SD preprocessing is employed. Particularly, the SD-based ANFIS model exhibits superior performance compared to both single and hybrid models, exhibiting a substantial enhancement in prediction accuracy of up to 48% when contrasted with the individual ANFIS model.

2.5 Spatial Methods

In general, wind speed forecasts are predictions for the future, and the forecast horizon is determined by the needs of the power system. In contrast, spatial correlation forecasting is frequently utilized to describe the wind resources at a site for which there is insufficient information but a nearby monitoring station is accessible. It considers the spatial relationship between wind speeds at different locations. It is a helpful signal when evaluating the potential for wind energy at locations without wind data.

In contrast to conventional wind speed and power prediction, spatial correlation prediction considers the spatial link between wind speed at different locations. Spatial correlation prediction requires wind speed data from multiple spatially correlated locations, and the data often contain time lags, which increases the complexity and cost of spatial correlation prediction. For spatial correlation prediction, either statistical models that consider the spatial correlation information or physical models that consider the terrain information can be used.

There are several approaches to the spatial correlation prediction problem. The Measure-Correlate-Predict (MCP) method is the most often utilized strategy (Thøgersen et al., 2007).

Using wind data from the reference site, the method is utilized to obtain long-term wind data at the target site. Four different MCP methods are presented: Matric MCP, Linear Regression, Wind Index MCP model, and Weibull Scale. They have all been shown to work well, but performance is obviously dependent on the nature of the available data.

In (Kwon, 2010), the long-term wind speed is predicted using the MCP technique, and the probability models are then applied to the target site using Monte Carlo-based numerical simulation. Prior to wind turbine construction, this model is used to evaluate the degree of uncertainty in the potential for all forms of wind energy. The method proved to be an effective means of evaluating the likely annual energy production of the site.

(Alexiadis et al., 1999) presented a technique for predicting wind speed and the associated power generation from few minutes up to several hours in advance based on cross-correlation of neighboring sites (0.8 to 40 km apart). Based on spatial correlation models, an ANN approach was developed using data of many different sites over a whole year period, which significantly improves the prediction accuracy by 20% compared to the persistence prediction model.

In (Barbounis & Theocharis, 2007), spatial correlation was used to develop a Local Feedback Dynamic Fuzzy Neural Network (LF-DFNN) for wind speed prediction. In this study, LF-DFNN is used to predict the wind speed in multiple steps for the target site using spatial information from two remote reference sites aligned with the target site along the prevailing wind direction. It is shown that LF-DFNN exhibits superior performance over alternative network models in the context of this application.

Based on spatial correlation models, a Takagi, Sugeno, and Kang (TSK) Fuzzy technique is developed in (Damousis et al., 2004) for estimating wind speed and energy at a specific target location. The training task of the model is performed using a genetic algorithm (GA) based learning technique. The evaluation of the performance for different terrain scenarios shows significant progress over the persistent model.

To assess or define the characteristics of the wind resource at a specific location of interest utilizing the information available from a known resource, the Bayesian hierarchical model was constructed (Miranda et al., 2006). The hierarchical model consists of two levels. At the primary level, the wind speed data at the reference site is formulated as the cumulative sum of distinct components, namely the temporal, spatial, and error components. At the subsequent level, a first-order random walk model is employed to capture the temporal aspect, while a multivariate normal distribution is utilized to characterize the spatial component. The ability of Bayesian inference to predict spatial correlations has been demonstrated.

An approach for long-term prediction of wind speed and power using NWPs offered at neighboring wind farm sites is proposed in (Barbounis et al., 2006). Depending on the neural dynamics model, three different forms of local recurrent neural network (LRNN) models are used. In addition, two learning techniques - Global Recursive Prediction Error (GRPE) and Decoupled Recursive Prediction Error (DRPE) - are used to update the weights of the model. The proposed strategies have been shown to outperform static and persistent models.

In (Velázquez et al., 2011), a spatial correlation-based ANN model is developed for estimating wind speed data at a designated station. The neighboring measuring stations' wind speed data are utilized as input signals for the model's input layer. The two most important results of this study are that as the number of reference stations increases, the estimation errors diminish and that there is a tremendous reduction in error when wind direction is included in the input signal. These findings emphasize the importance of incorporating spatial information and wind direction in improving the accuracy of wind speed estimation using the ANN model.

2.6 Regional Methods

The primary objective of regional forecasting is to anticipate the aggregate electricity generation within a defined area, considering a specific number of wind turbines. The necessity for regional forecasting arises as market operators seek to estimate the comprehensive generation of wind power within a particular region. This approach offers a swifter alternative compared to the summation of individual wind farm forecasts, while concurrently enhancing accuracy through the advantageous impacts of spatial smoothing. Numerous methodologies have been proposed to facilitate regional forecasts.

Regional forecasting models predominantly rely on the upscaling approach, which involves scaling up available online measurements and NWPs to cover the entire region. This approach mitigates the challenge of acquiring wind power measurements and NWPs from all wind farms within the region. Although limited research has been conducted on regional forecasting, the growing prominence of wind energy necessitates the development of effective solutions for accurate regional forecasting in the future.

In (Focken et al., 2001), regional forecasts are generated using linear upscaling within the Previento wind power prediction system. The upscaling process involves utilizing the ratio of the measured wind power at a single site to the ensemble of sites. The practical implementation of this approach in Germany is demonstrated, showcasing its effectiveness in regional forecasting.

Three distinct approaches-FNN upscaling, Cascaded model, and FNN cluster model-are offered depending on the availability of NWPs and online SCADA data in (Pinson et al., 2003), which uses an ANN-Fuzzy technique based on the upscaling approach to predict regional wind generation. The outcomes demonstrate an improvement in the overall performance; however, it is observed that surpassing the persistence model's performance during the initial forecast period is challenging. To boost the performance during the initial forecast horizon, SCADA data is required.

In (Murugesan et al., 2012), a finite-state Markov chain model is proposed to predict the aggregate power output of a wind farm. This model considers temporal and spatial dynamics of wind power generation. The temporal dynamics are captured using autoregression analysis. The spatial dynamics, on the other hand, are captured by a rigorous stepwise procedure.

From the information provided above, it can be observed that wind speed prediction models can be categorized in two ways: based on the time interval of prediction and the methodology employed for prediction. In this discussion, various methods have been explored, encompassing their principles, advantages, disadvantages, and practical examples. It is evident that certain models outperform others in specific regions while underperforming in different locations, thus confirming the notion that these models are site dependent. This characteristic forms the basis of the problem under investigation in this study.

Notably, neighboring countries to Palestine have conducted numerous studies related to wind speed prediction. For instance, Khosravi et al. employed three machine learning algorithms (SVR, ANFIS, and FNN) to predict wind speed and other parameters in Iran (Khosravi et al., 2018). The results demonstrated that SVR performed better than the other two models. Similarly, in Nigde, Turkey, researchers utilized five machine-learning algorithms to predict wind power (Demolli et al., 2019), showing that machine-learning algorithms are particularly effective for long-term forecasts.

However, there is a lack of studies specifically focusing on wind speed prediction within Palestine itself. Only one study (Salah et al., 2022) investigated wind speed prediction in the Jerusalem region using machine learning algorithms. The authors employed six different machine learning algorithms and four features for their predictions. The results indicated that SVM outperformed the other algorithms in that study.

Given the unique geographical characteristics and wind patterns of Jerusalem, there is a need for a case study that evaluates the performance of additional machine learning algorithms tailored to this specific context. Therefore, the objective of this study is to address this gap by conducting a comparative analysis of two machine learning algorithms not previously examined in Salah et al.'s study: ANFIS and KNNR, for wind speed prediction in Jerusalem. ANFIS was selected based on its wide utilization and proven effectiveness in the literature across various regions, while KNNR was chosen for its simplicity, small computational time, and limited prior testing in this context.

Chapter 3 Methodology

3.1 Machine Learning

One of the good approaches to understand what machine learning is, is to compare between it and statistics. In statistics, maximum likelihood estimation (MLE) is a systematic tool for statistical inference. However, MLE essentially requires knowledge of the probability distribution from which the data are drawn, up to an unknown parameter of interest. Often the unknown parameter has a physical meaning, and its estimation is key to better understanding certain phenomena. Thus, to enable MLE, one needs to know a lot about the process of data generation, which is called modeling. Modeling may be conditioned by physics or by prior knowledge of the problem, but in any case, it requires some expertise (Bzdok et al., 2018).

However, for some types of data sets, modeling the data they contain is difficult, if not impossible, because the input/output process is not well understood. For such types of data, therefore, a distribution-free approach is required; in other words, we could say that machine learning prefers a black-box approach. Thus, a machine learning model is not explicitly programmed, but is built based on training data to make predictions.

3.1.1 Modeling

This study adopts a conventional machine learning approach that comprises the following steps: obtaining data, processing data, selecting features, constructing a machine learning model, and evaluating the model through testing and validation.

This scientific investigation presents an examination into the prediction of wind speed by employing machine learning algorithms. Specifically, this case study delves into the utilization of two techniques, namely the Adaptive Neuro-Fuzzy Inference System (ANFIS) and the k-Nearest Neighbors Regressor (KNNR), to achieve enhanced wind speed forecasting accuracy.

The ANFIS model represents a hybridized framework that amalgamates the capabilities of fuzzy logic and neural networks. By employing fuzzy logic, the model effectively captures and models the inherent uncertainties present within the system, while the neural network component enables learning and optimization processes. ANFIS develops a fuzzy inference system by adaptively adjusting its parameters (premise, consequent and membership functions parameters) through the utilization of a hybrid learning algorithm.

To accomplish wind speed prediction using ANFIS, the following procedural steps are undertaken:

a) The collection of wind speed data, encompassing historical wind speed records alongside relevant input parameters (e.g., temperature, humidity, pressure).
b) Preprocessing of the data by means of input normalization, and subsequent division into training and testing datasets.
c) Training of the ANFIS model utilizing the designated training dataset. This entails the model's adjustment of its parameters through the employment of forward and backward passes, thereby optimizing both the fuzzy membership functions and the neural network weights.
d) Validation of the trained model via the utilization of the testing dataset, thereby evaluating its performance based on metrics such Root Mean Square Error (RMSE).
e) Once the model has been duly validated, it is primed for wind speed prediction by providing the appropriate input variables.

KNNR is a non-parametric algorithm widely adopted for regression tasks. It leverages the concept of proximity, predicting the output of a novel data point by considering its k nearest neighbors within the training dataset. The predicted value is derived from an average of the target values associated with these nearest neighbors. To accomplish wind speed prediction utilizing KNNR, similar steps to ANFIS with some differences are undertaken.

3.1.2 ANFIS

Fuzzy logic represents an advanced form of logic that extends beyond the binary framework of traditional logic systems. Unlike classical set theory, where variables are confined to possessing a definitive truth value of either 0 or 1, fuzzy logic allows for the assignment of truth values that span the entire range of real numbers between 0 and 1. This pioneering concept was introduced in conjunction with the development of fuzzy set theory, which diverges from classical set theory in its treatment of membership within sets.

Classical set theory operates on the premise that a member either unequivocally belongs (assigned a value of one) or indisputably does not belong (assigned a value of zero) to a set. In stark contrast, fuzzy set theory acknowledges the existence of gradation by permitting the membership degree of an element to assume any value within the inclusive interval of zero to one. This introduces a nuanced approach, where the membership function serves as the bridge to distinguish classical and fuzzy set theories as shown in Figure (3.1).

Within classical set theory, the membership function simply maps the elements of the universe to the discrete values of either zero or one, signifying the absence or presence of membership, respectively. In the context of fuzzy set theory, however, the membership function takes on a broader role by facilitating the mapping of elements in the universe to continuous values encompassing the closed real interval spanning from zero to one. This augmented functionality captures the essence of fuzzy set theory by reflecting the inherent uncertainty and imprecision inherent in many real-world scenarios.

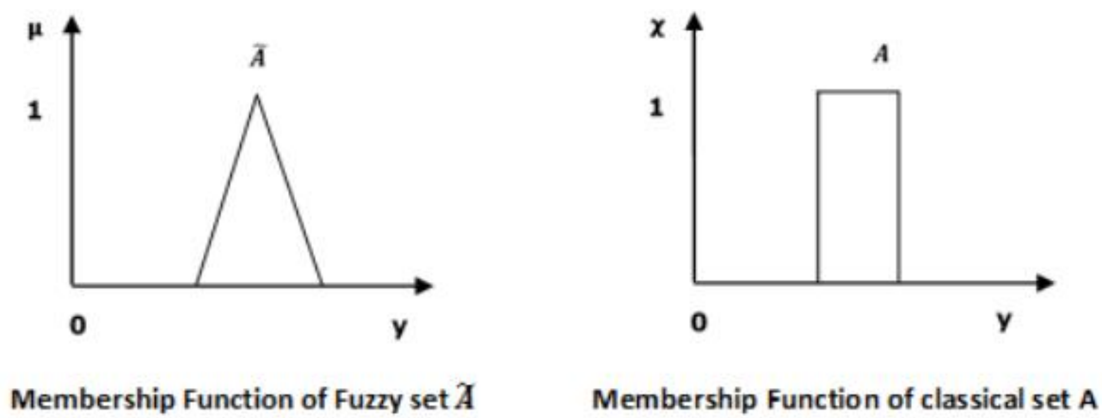

Figure 3.1: A fuzzy set Ã in a universe set U is defined as a set of ordered pairs {(y, $\mu_{\tilde{A}}(y)$) | y ε U}, where $\mu_{\tilde{A}}$ is a membership function ($\mu_{\tilde{A}}$: U → [0,1] and $\mu_{\tilde{A}}(y)$ represents the degree of membership of y in Ã.

A fuzzy if-then rule represents a proposition that adheres to the structure "if A then B," wherein both A and B are fuzzy sets. To exemplify this concept, consider the following scenario: "if the temperature is categorized as low, then the wind speed tends to be high." Here, temperature and wind speed are linguistic variables, while low and high correspond to linguistic values or fuzzy sets.

In instances where only the premise segment of the if-then statement possesses fuzziness, the rule assumes the designation of a Takagi-Sugeno if-then rule. To illustrate, consider the following example: "if the velocity is classified as high, then the friction force can be expressed as k multiplied by the square of the velocity". In this case, the premise, which is the velocity, exhibits fuzziness, while the consequence, which is the calculation of the friction force, follows a deterministic mathematical expression involving a constant (k) and the squared value of the velocity (Jang, 1993).

The incorporation of fuzzy if-then rules, including the Takagi-Sugeno variant, within various scientific domains has proven to be a powerful means of modeling complex systems, enabling the representation of linguistic uncertainty, and facilitating effective decision-making processes.

A fuzzy-rule-based system, also referred to as a fuzzy inference system, comprises a sophisticated framework consisting of five fundamental functional blocks, as expounded by (Jang, 1993):

1. Rule Base: This foundational block encompasses a repository of numerous fuzzy if-then rules that encode expert knowledge or learned patterns. These rules serve as the guiding principles for the system's decision-making processes, delineating relationships between input variables and corresponding output actions.

2. Database: The database component assumes a pivotal role in defining the membership functions associated with the fuzzy sets utilized within the fuzzy rules. It encapsulates

information pertaining to the shape, characteristics, and boundaries of the fuzzy sets, providing the foundation for linguistic representation and subsequent inference operations.

3. Decision-Making Unit: Situated at the heart of the system, the decision-making unit undertakes intricate inference operations based on the rules present in the rule base. By evaluating the fuzzy input variables and applying appropriate reasoning mechanisms, this unit facilitates the determination of the system's output or action.

4. Fuzzification Interface: This interface serves as a crucial intermediary between the crisp input values and the subsequent fuzzy inference processes. By employing various transformation techniques, it converts the crisp inputs into degrees of match or compatibility with the linguistic values defined within the membership functions, thus enabling the system to reason and operate on a linguistic level.

5. Defuzzification Interface: The final block of the fuzzy-rule-based system, the defuzzification interface, assumes the responsibility of transforming the fuzzy outcomes generated by the inference process into a crisp output. Employing diverse defuzzification methods, this interface effectively aggregates and interprets the fuzzy results, ultimately yielding a precise and actionable crisp output.

A comprehensive fuzzy inference system adheres to a meticulously orchestrated series of steps, as outlined by (Jang, 1992), to facilitate intricate fuzzy reasoning processes. These steps encompass:

1. Fuzzification: The initial step entails a comparison of the input variables against the defined membership functions within the premise portion of the fuzzy rules. This comparative analysis aims to ascertain the precise degree of membership for each linguistic label, effectively quantifying the level of association between the input variables and the fuzzy sets.

2. Rule Aggregation: Proceeding from the fuzzification stage, the membership values obtained for each linguistic label are aggregated using a specific T-norm operator, such as multiplication or minimum. This aggregation process serves to determine the firing strength or weight assigned to each rule within the fuzzy inference system, considering the degrees of membership obtained from the previous step.

3. Consequent Generation: Based on the determined firing strengths derived from the rule aggregation phase, the consequent segment of each rule is generated. This qualified consequent can assume a fuzzy or crisp nature, depending on the specific requirements and characteristics of the fuzzy inference system being employed. The consequent represents the anticipated outcome or action associated with the respective rule, serving as a critical component in subsequent reasoning and decision-making processes.

4. Defuzzification: The culminating stage of the fuzzy reasoning process involves the aggregation of the qualified consequents obtained from the previous step. Through a rigorous aggregation process, these consequents are seamlessly combined to yield a single crisp output.

By diligently adhering to these meticulous steps, a sophisticated fuzzy inference system harnesses the power of fuzzy logic and linguistic representation, enabling robust reasoning, decision-making, and inference capabilities in complex real-world scenarios.

Fuzzy inference systems, as per the elucidation provided by (Jang, 1992), are commonly categorized into three distinct types, each predicated on the specific approach employed to generate the system's output. These classifications are as follows:

- Type 1: Within this category, the ultimate output is derived through a meticulous process of weighted averaging. More precisely, it entails calculating the crisp output of each rule and subsequently determining its weighted average. The weighting factor is determined by the firing strength of the respective rule, while the output membership functions are required to adhere to the characteristics of monotonic functions.

- Type 2: This classification involves the acquisition of a comprehensive fuzzy output by applying the "max" operation to the qualified fuzzy outputs, which correspond to the minimum values between the firing strength and the output membership function of each rule. The resulting fuzzy output can subsequently be transformed into a crisp output using diverse methodologies, such as centroid of area, mean of maximal values, or other suitable schemes.

- Type 3: In the context of Type 3 fuzzy inference systems, Takagi-Sugeno's if-then rules are employed as the cornerstone of the framework. Within this paradigm, the output of each rule assumes a linear combination of the input variables, along with a constant term. The overall output of the system is then determined through the application of a weighted average approach, whereby each rule's output is assigned an appropriate weightage.

Adaptive neuro-fuzzy inference system (ANFIS) is a Takagi-Sugeno fuzzy inference system implemented as an artificial neural network to determine system's properties (membership functions and fuzzy rules). To introduce the architectural essence of ANFIS, we focus on a specific instance involving a fuzzy inference system comprising two inputs, denoted as x and y, along with a single output, referred to as z. Within this scenario, we consider a rule base that encompasses two Takagi-Sugeno's fuzzy if-then rules:

Rule 1: In the presence of input values x characterized by membership in A1 and y characterized by membership in B1, the consequent portion of this rule can be mathematically expressed as:

$$f_1 = p_1 x + q_1 y + r_1 \text{ , where } p, q \text{ and } r \text{ are the consequent parameters.} \qquad (3.1)$$

Rule 2: Similarly, when the input values x exhibit membership in A2 and y exhibits membership in B2, the consequent segment of this rule can be formulated as:

$$f_2 = p_2 x + q_2 y + r_2 \qquad (3.2)$$

ANFIS is composed of five layers (Jang, 1992). for each layer, the node functions are of the same function family and nodes are either fixed (circle nodes) or adaptive (square nodes) as illustrated in Figure (3.2).

Layer 1, the input layer, comprises solely adaptive nodes. These nodes are governed by a unified function denoted as in equation (3.3). The linguistic label (fuzzy set) A_i, and its corresponding membership function μ_{A_i}, ascertain the behavior of this node. Notably, this layer incorporates three premise parameters that significantly impact the node's functionality.

$$O_i^1 = \mu_{A_i}(x) \text{ , where x denotes the input associated with node i.} \qquad (3.3)$$

Layer 2, referred to as the fuzzification layer, encompasses fixed nodes labeled as π. The primary role of these nodes is to multiply the outputs of the previous layer, layer one. Specifically, the node output O_i^2 is computed as the product of the membership functions $\mu_{A_i}(x)$ and $\mu_{B_i}(x)$, where i takes values of 1 and 2, representing the respective nodes within the layer.

Layer 3, known as the normalization layer, comprises fixed nodes labeled as N. These nodes undertake the computation of the ratio between the firing strength of each rule (ω_i) and the sum of all rules' firing strengths. The output of each node within this layer, denoted as O_i^3, is computed as:

$$\overline{\omega_i} = \frac{\omega_i}{\omega_1 + \omega_2} \text{ , with I ranging from 1 to 2.} \tag{3.4}$$

Layer 4, recognized as the defuzzification layer, integrates adaptive nodes governed by a specific node function. The output of each node within this layer, denoted as O_i^4, is obtained through the multiplication of the normalized firing strength $\overline{\omega_i}$ and the consequent function, which assumes the form ($p_i x + q_i y + r_i$). The parameters p, q, and r, known as the consequent parameters, play a critical role in shaping the node's behavior.

Layer 5, designated as the output layer, encompasses a sole fixed node labeled as Σ. This node calculates the overall output by summing the outputs of all nodes within the preceding layer, layer four. The computation can be represented as:

$$O_i^5 = overall\ output = \sum \overline{\omega_i}\, f_i \tag{3.5}$$

Through the intricate interplay of these five layers, ANFIS enables the integration of adaptive neuro-fuzzy inference techniques, facilitating advanced system analysis, decision-making, and pattern recognition in diverse scientific domains.

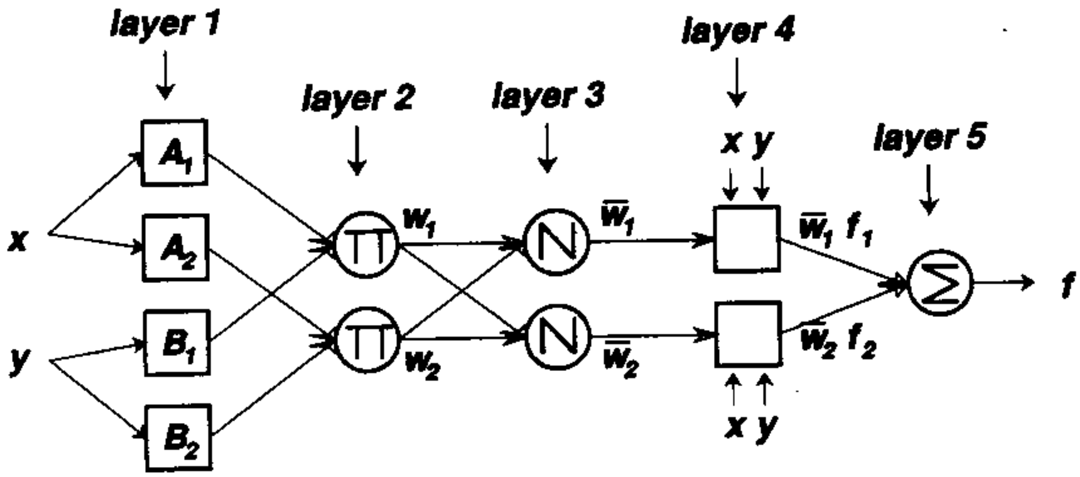

Figure 3.2: ANFIS architecture. Circle nodes are fixed while square nodes are adaptive. Source (Jang, 1993)

3.1.3 KNNR

K-nearest neighbors is a classification algorithm, which is a part of supervised learning. In supervised learning each point in dataset has m features $x_1, x_2, \ldots, x_m$ and a response y. In a classical problem, a set of responses consists of the so-called class labels (a known finite set of elements).

The simple logic behind KNN is to find the nearest neighbor from the dataset to a testing object and check the nearest class label, but a decision based on one neighbor is not accurate, so we take more than one neighbor. The number of considered neighbors is called k (the k in KNN), choosing k is an optimization problem and it depends on the dataset. When k is more than one the predicted class label is the most frequent class label for the k neighbors, for it is possible to have many most frequent class labels, weights are introduced in distance calculations, in this case it is called weighted KNN.

Choosing K is a complicated problem, since when k is small, the classifier is sensitive to outliers although it is almost ideally processes the training data, and when k is increases, the boundary becomes smoother and almost turns into a straight line, classes are split more naturally but more errors occur on the training data. A good approach is to estimate the error rate by splitting the initial dataset into test and training samples (using what is called k-fold cross-validation).

The nearness could be determined based on the common concept of distance i.e., the Euclidean distance (the generalization of Pythagorean law to any finite dimension), but it could be different using another metric.

Definition: Let X be as set, A metric defined on X is a function $d(x, y): X, X \xrightarrow{maps\ to} R$, for any $x, y, z \in X$, which satisfies three conditions:

1. It is non-negative, $d(x, y) \geq 0$ and $d(x, y) = 0$ if and only if $x = y$.
2. It is symmetric $d(x, y) = d(y, x)$.
3. The triangle inequality is satisfied, $d(x, y) \leq d(x, z) + d(z, y)$.

Minkowski distance is a common example of a metric, it is given by (in p dimension):

$$d_q(x, \acute{x}) = \sqrt[q]{|x_1 - \acute{x}_1| + |x_2 - \acute{x}_2| + \cdots + |x_p - \acute{x}_p|}\ where\ q\ in\ an\ integer\ bigger\ than\ 1. \quad (3.6)$$

Euclidean distance is a special case of Minkowski distance where q = 2, Manhattan distance also is a special case where q = 1, when q approaches infinity it is called Chebyshev distance, and it is given by $d_\infty(x, \acute{x}) = \max_{i \in \{1,2,\ldots,p\}} x_i - \acute{x}_i$. Choosing the metric is another optimization problem in KNN.

KNN Classification: Assume that $X = (x_1, \ldots, x_n)$ is a training dataset of size n, where each object of this dataset has p features $x_1 = (x_{i1}, x_{i2}, \ldots, x_{ip}), where\ i \in \{i, \ldots, n\}$ and corresponds to the response $y_i \in Y$, and let d be a metric defined on a p-dimensional set of objects. Thus, a classification algorithm is as follows:

1. For a new object z, calculate $d(x_i, z)$ to each object $x_i, i \in \{1,2, \ldots, n\}$.
2. Arrange the elements of the training dataset in order of non-decreasing distances to z:

$$d\left(z, x_1^{(z)}\right) \leq \ d\left(z, x_2^{(z)}\right) \leq \cdots \ \leq d\left(z, x_n^{(z)}\right), where\ x_i^{(z)} = x_{t_i} \tag{3.7}$$

and t_1 is a solution to the problem:

$$\underset{i \in \{i,\ldots,n\}}{\text{Arg min}}\, d\left(z, x_{t_i}\right) \tag{3.8}$$

and t_2 is a solution to the problem:

$$\underset{i \in \{i,\ldots,n\} / \{i,\ldots,n\}}{\text{Arg min}} \quad d\left(z, x_{t_i}\right) \tag{3.9}$$

3. Renumber the responses according to the instructions in step 2:

$$y_i^{(z)} = y_{t_i}, i \ \in \ \{i, \ldots, n\} \tag{3.10}$$

4. Among K-nearest neighbors, find the most frequent class $y \ \in Y$:

$$a(z, k) = \ \underset{y \in Y}{\text{Arg max}} \,|(y_i^{(z)} = y) \tag{3.11}$$

Where the function |() being the indicator of event A, is equal to one when event A is occurred and zero otherwise.

KNNR (k-nearest neighbors regression) has the same concepts as KNN but differs that for regression we don’t have finite classes' labels as outputs (responses) but the infinite real numbers line. So instead of choosing the most frequent class label, we take the average of the k-nearest neighbors' responses.

3.2 Dataset

The machine-learning algorithms underwent training utilizing wind data sourced from the extensive network of Palestinian meteorological stations. These datasets were meticulously collected over an extensive timeframe spanning 11 years, specifically commencing from January 1, 2008, and concluding on December 31, 2018. To ensure accuracy and representativeness, the wind measurements were meticulously recorded in a continuous manner, employing a cup generator anemometer positioned at a height of 20 meters. Notably, the data acquisition site was

Jabal Al-Mukabber, a village located in East Jerusalem, Palestine, with precise geographic coordinates of Latitude 31.7555 N and Longitude 35.2410 E. This specific region stands at an elevation of 720 meters above sea level, guaranteeing a comprehensive and diverse dataset for the subsequent analyses and model training.

The data set contained four features: timestamp, wind speed, air temperature, and atmospheric pressure. Measurements were taken at 3-hour intervals (8 measurements for each day). The dataset itself consisted of an extensive 32,131 rows of data. Notably, within this dataset, 150 rows were identified to have zero values in the wind speed variable, while an additional 69 rows lacked data, specifically with 36 missing values in the wind speed variable and 33 missing values in the temperature variable.

To address the presence of empty cells and ensure data completeness, a preprocessing step was executed using the pandas imputing function. This essential data manipulation task was accomplished using a Python script, expertly filling the vacant cells with appropriate values. It is worth noting that the existence of zero values in the wind speed variable is deemed acceptable due to the rounding convention associated with wind speed, where values below 0.25 are rounded down to zero.

Table (3.1) provides an illustrative subset of the dataset, offering a glimpse into the intricate interplay of the various variables. Furthermore, Figure (3.3) showcases the comprehensive timelines of all variables encapsulated within the dataset, providing a visual representation of their temporal evolution and patterns.

Table 3.1: A sample of the dataset.

Date	Time	Temp (^oC)	Wind Speed (m/s)	Pressure (mbar)
25-01-2008	8:00	5.4	3.0	922.8
25-01-2008	11:00	7.9	3.5	921.6

25-01-2008	14:00	10.7	2.5	919.7
25-01-2008	17:00	9.0	1.0	919.2
25-01-2008	20:00	8.1	0.0	919.5
25-01-2008	23:00	7.3	1.0	919.1
26-01-2008	2:00	6.5	3.0	918.7

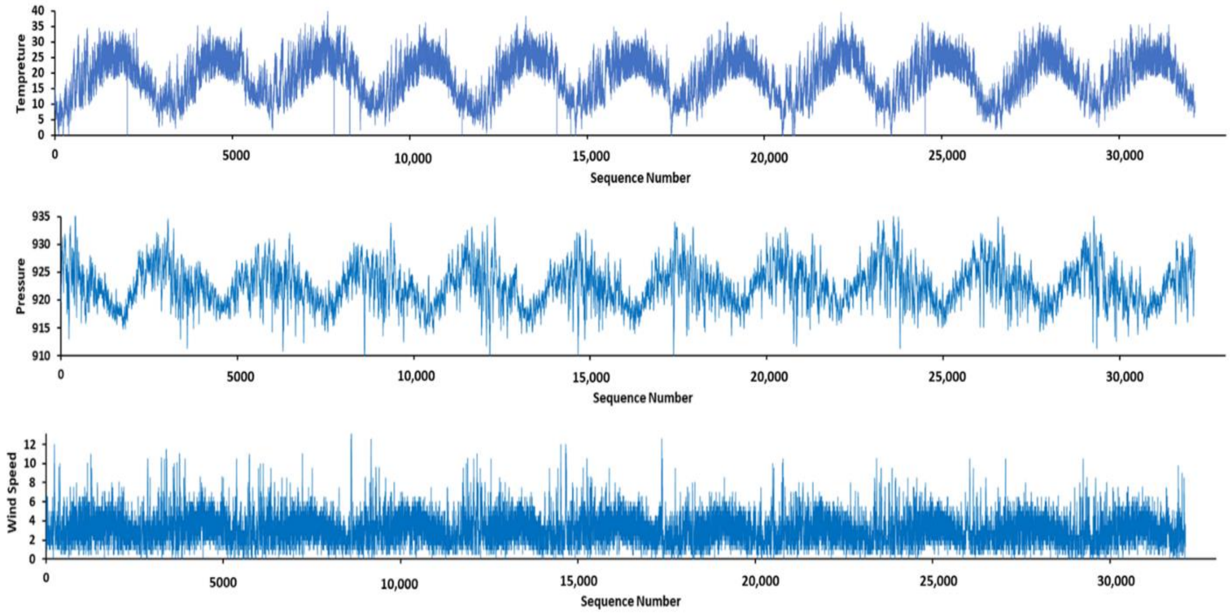

Figure 3.3: The entire timelines for every variable present in the dataset.

A rigorous statistical analysis was conducted on the dataset, yielding insightful findings as presented in Table (3.2). It was observed that the wind speed variable exhibited a minimum value of zero, indicating moments of calm conditions, while the maximum value reached 14.5 m/s, denoting instances of heightened wind intensity. The calculated mean wind speed stood at 3.11 m/s, capturing the central tendency of the dataset, while the corresponding standard deviation was determined to be 1.54 m/s, reflecting the dispersion of values around the mean.

Further analysis revealed that three quarters of the wind speed values were found to be less than or equal to 4 m/s. This threshold holds significance, as wind speeds within this range are deemed suitable for the operation of small-scale turbines. Hence, the dataset offers valuable insights into the prevailing wind conditions, suggesting favorable conditions for harnessing wind energy using compact turbine systems.

Regarding the air temperature, it has a mean of 18.2 ^{o}C and its values range from zero to 39.7 ^{o}C where three quarters of values are less than 23.4 ^{o}C. for the atmospheric pressure, values range from 909 to 939.3 mbar while 75% of values are less than 925mbar.

Table 3.2: The analysis of wind speed data involves calculating various statistical measures such as the minimum, mean, maximum, standard deviation, 25th, 50th, and 75th percentiles of the dataset.

Centralized statistical quantities	Temp (^{o}C)	Pressure (mbar)	Speed (m/s)
Mean	18.22	922.55	3.11
SD	6.98	3.67	1.54
Min	0	909.0	0
25%	12.6	919.9	2.0
50%	18.5	922.3	3.0
75%	23.4	924.9	4.0
Max	39.7	927.3	14.5

To analyze the relationships between the variables, the correlations were calculated using the Pearson coefficient. The calculation was done with the Python libraries NumPy and Pandas, the correlation matrix is shown in Table (3.3), a heatmap representing the correlation matrix was created with the libraries Matplotlib and Seaborn, it is shown in Figure (3.4). In addition, pair plots (Figure 3.5) between all variables (the timestamp is expressed in nanoseconds, the notation “1e18” signifies ten to the power of 18, the notation “@20 m” denotes the wind speed measured at an elevation of 20 meters, and the pressure is expressed in millibars) were created using the latter two libraries.

Table 3.3: Correlation matrix: depicting the correlation coefficients between dataset variables.

	Time	Temperature	Pressure	Speed
Time	1	0.023	0.029	-0.083
Tempreature	0.023	1	-0.415	0.009
Pressure	0.029	-0.415	1	-0.35
Speed	-0.083	0.009	-0.35	1

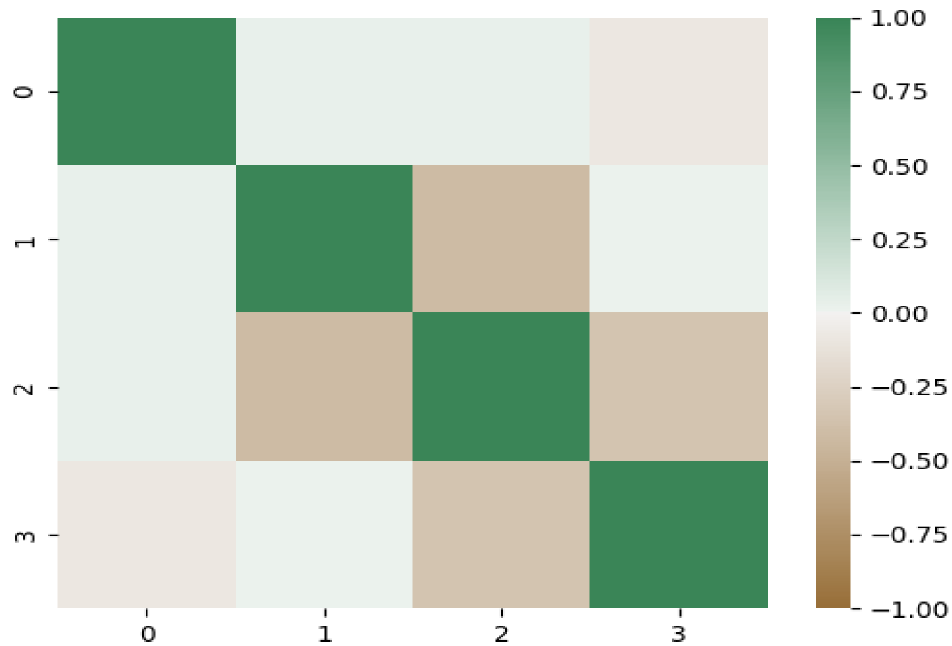

Figure 3.4: A heatmap representing the correlation matrix.

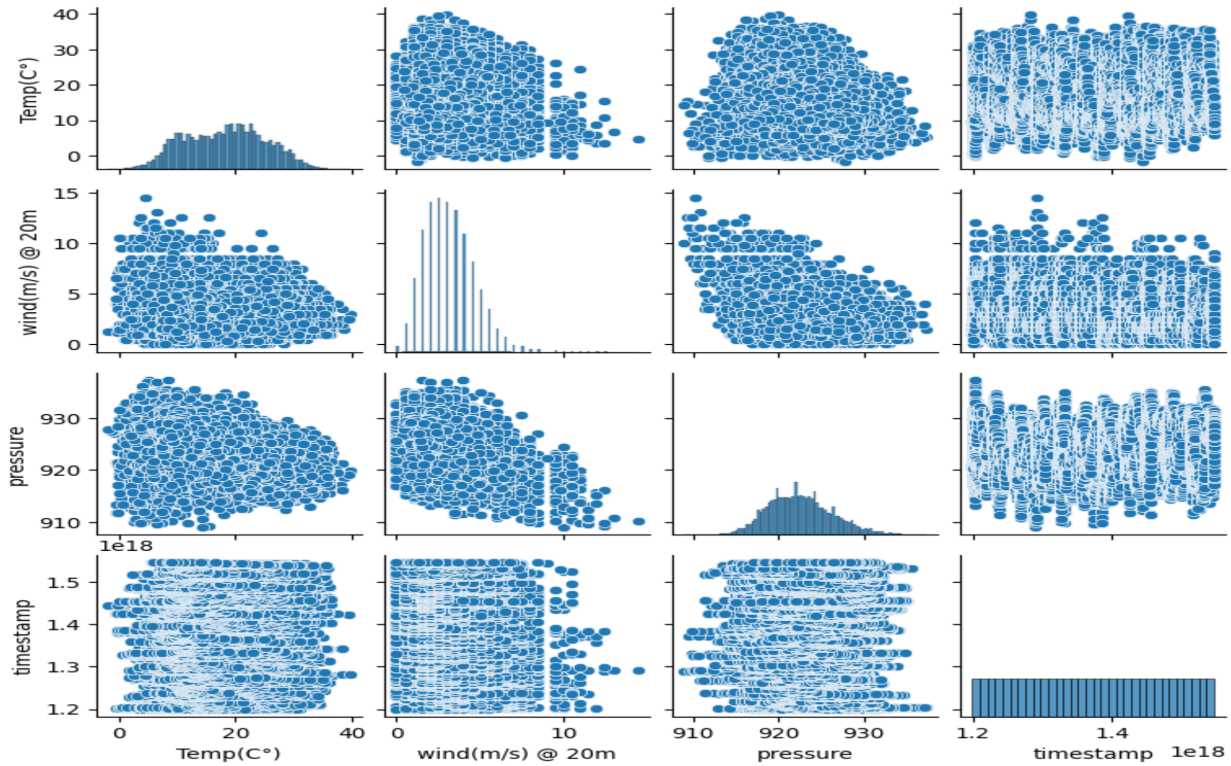

Figure 3.5: Pair plots displaying pairwise relationships among all variables.

3.3 Evaluation Metrics

The prediction of wind speed inherently encompasses a characteristic element of uncertainty, rendering exactness unattainable. Consequently, it assumes paramount significance to diligently evaluate the accuracy of wind speed predictions. Crucially, the evaluation process necessitates meticulous scrutiny of error measurements utilizing data independent of those employed for model construction or parameter tuning.

The adoption of such a rigorous evaluation methodology ensures the robustness and generalizability of the wind speed prediction model. By judiciously employing unseen data, unaffected by the model's development process, the assessment not only gauges the model's efficacy but also safeguards against potential overfitting or biased performance estimation. This stringent evaluation practice provides critical insights into the model's predictive capabilities, enabling researchers and practitioners to ascertain its reliability and make informed decisions based on the outcomes.

Consequently, by prioritizing comprehensive and unbiased evaluation methodologies, the accuracy and credibility of wind speed predictions can be appropriately assessed, further enhancing the reliability and utility of the predictive models deployed in this domain.

The prediction error is the numerical deviation between the actual measurement and the corresponding prediction, and is mathematically expressed as follows:

$$e_{t+k|t} = v_{t+k} - \hat{v}_{t+k|t} \tag{3.12}$$

Where v_{t+k} represents the actual wind speed (measured value) at a specific moment 't+k', while $\hat{v}_{t+k|t}$ denotes the predicted wind speed calculated at time ‘t’ for the projected future time ‘t+k’.

It is crucial to assess the accuracy of a predictive model on data that was not utilized in its construction or parameter tuning. Various evaluation metrics, such as Bias Eq (3.13), Mean Absolute Error (MAE) Eq (3.14), Mean Square Error (MSE) Eq (3.15), Root Mean Square Error (RMSE) Eq (3.16), Standard Deviation of Errors (SDE) Eq (3.17), and coefficient of determination (R2) Eq (3.18) are employed to determine the effectiveness of the model (Allison et al., 2020; Madsen et al., 2005; X. Zhao et al., 2011).

$$Bias_k = avg(e_{t+k|t}) = \bar{e}_k = \frac{1}{N}\sum_{t=1}^{N} e_{t+k|t} \tag{3.13}$$

$$MAE_k = \frac{1}{N}\sum_{t=1}^{N} \left|e_{t+k|t}\right| \tag{3.14}$$

$$MSE_k = \frac{1}{N}\sum_{t=1}^{N} \left(e_{t+k|t}\right)^2 \tag{3.15}$$

$$RMSE_k = \sqrt{MSE_k} = \sqrt{\frac{1}{N}\sum_{t=1}^{N} \left(e_{t+k|t}\right)^2} \tag{3.16}$$

$$SDE_k = Sdt(e_{t+k|t}) = \sqrt{\frac{1}{N}\sum_{t=1}^{N} \left(e_{t+k|t} - \bar{e}_k\right)^2} \tag{3.17}$$

$$R2 = 1 - \frac{\sum_{t=1}^{N} \left(v_{t+k} - \hat{v}_{t+k|t}\right)^2}{\sum_{t=1}^{N} \left(v_{t+k} - \bar{v}_{t+k}\right)^2} \tag{3.18}$$

Where 'N' denotes the size of the testing sample set, representing the total count of data instances specifically allocated for evaluation and testing purposes within the dataset.

Bias serves as a metric to assess the disparity between the average forecasted wind speed and the actual observed values, indicating overestimation (bias > 0) or underestimation (bias < 0) of the method. However, it only shows systematic errors and lacks information about the forecasting method's accuracy alone (Y. Zhao et al., 2016).

In contrast, Mean Absolute Error (MAE) employs the original data's units, providing a more precise analysis of both random and systematic errors when compared across different models (Zhang et al., 2019). It is a non-negative real-valued value, and lower values indicate better accuracy.

Mean Squared Error (MSE) quantifies the average squared disparity between the observed and predicted values, quantifying the model's error. It would be zero in an ideal scenario with 100% accuracy. RMSE considers both random and systematic errors, where larger values indicate greater deviations and smaller values indicate more precise predictions. Significant discrepancies between MAE and RMSE values suggest a wider spread of predicted values in comparison to the measured data (Y. Zhao et al., 2016).

R2 is a coefficient of determination indicating the amount of variance explained by the prediction model, with values close to 1 indicating an optimal model and negative values indicating a poor prediction.

When conducting model comparisons, it is crucial to quantitatively measure the advancements achieved by the advanced model in relation to the benchmark model. To this end, the improvement over the benchmark model can be mathematically expressed as follows (Madsen et al., 2005):

$$Imp_{EC} = \frac{EC_{ref} - EC_{adv}}{EC_{ref}} \tag{3.19}$$

In this context, the symbol EC denotes the evaluation criteria used for assessing the performance of the models, including established metrics such as R2, MAE, etc.

3.3.1 References Models

A variety of reference models have been established as benchmark models to serve as a basis for preliminary assessments of the accuracy of novel wind prediction models. This subsection delves into an examination of notable reference models currently in existence.

1. Persistence model

The persistence model, a widely employed benchmark model, posits that the forthcoming wind speed remains identical to the present wind speed, as mentioned previously.

2. The weighted summation of persistence and the mean power production.

The persistence model exhibits suitability for very short-term forecasts but proves inadequate for longer time horizons. To address this limitation, a novel reference model has been proposed (Nielsen et al., 1998). It is a weighted summation that combines the persistence component with the mean component as follows:

$$v_{t+k|t} = a_k \, v_t + (1 - a_k)\bar{v} \tag{8}$$

where v_t represents the measured value at time t, $\bar{v}$ denotes the estimated mean, and a_k signifies the correlation coefficient between v_t and v_{t+k}.

3.3.2 Forecast Accuracy:

Prediction accuracy decreases with increasing prediction duration. For short-term forecasts, the MAE is generally between 5% and 15%, and errors increase rapidly as the forecast horizon becomes longer. For example, MAE is usually between 13 and 21% for 1-2 days ahead and increases to 20-25% for 3 days (Zack, 2003). In (Marti et al., 2006) it is shown how the performance of the models exhibits a correlation with the topographic intricacy of the terrain. As the complexity of the terrain increases, MAE increases significantly, i.e., in a flat terrain MAE is much lower than in a complex terrain.

Model performance is correlated with seasonal variability in (Lange M & Focken U, 2006). Since wind speeds are higher and uncertainties are larger for summer storms, the prediction errors are more pronounced in summer compared to winter. In (Pinson & Kariniotakis, 2004), the Meteorological Risk Index (MRI), which evaluates weather stability, is introduced to compare forecast errors. As the MRI increases, the forecast error increases linearly. The forecast uncertainty in (Lange & Heinemann, 2002) also depends on air pressure rather than wind speed. At low pressure, the prediction error is larger than at high pressure.

Chapter 4 Results and Discussions

Two machine-learning algorithms were applied on the dataset to predict wind speed, which are adaptive (ANFIS), and k nearest neighbors regression (KNNR). The simulation testbed used Intel(R) Core (TM) i7-1265H CPU running @ 2.3 GHz, with 16 GB memory, 64-bit MS Windows 10 Home with x64 processor architecture, The Python environment setup consisted of Python (3.8.11) and common ML libraries, mainly scikit-learn (0.24.2), SKFuzzy and NumPy, among other libraries for data extraction and visualization, such as seaborn and matplotlib. The dataset was split into training and testing sets. This section demonstrates and discusses the results of each model, then A comparison between the two models is conducted. Finally, a comparison between this study and other studies is conducted.

4.1 ANFIS:

Several ANFIS experiments were conducted by varying different parameters to search for the optimal model. The experiments involved testing different numbers of membership functions (one, two, and three) for each feature, as well as different epochs, number of populations, and membership functions. Results were analyzed and discussed for each variation.

The evaluation metrics showed that there wasn't a significant difference between using one membership function and three membership functions (RMSE is 0.198 for both). However, using two membership functions resulted in even better results than both one and three, so it was selected as the optimal number of membership functions.

When two membership functions are used for each feature, it usually involves creating two fuzzy sets, with one corresponding to low values and the other to high values. The shapes of these fuzzy sets can vary and may be triangular, trapezoidal, or Gaussian, among others, depending on the data and the specific problem being analyzed. For this problem, the generalized bell function was used for its simplicity and performance.

Four variations of rules were used in this scenario, where the form of the rule is "if [feature1] is low (high) and [feature2] is low (high), then [output] is equal to [coefficients]". For example, one rule might be "if temperature is low and pressure is low, then wind speed is equal to $p_1 * temp + q_1 * pressure + r_1$.

The dataset, which contains 32,131 samples, was divided into two sets: 30% for testing (9,640 samples) and 70% for training (22,491 samples). The ANFIS model used four premise membership functions, resulting in 24 parameters. The RMSE for the training set was 0.193, while the RMSE for the testing set was 0.196.

Figures (4.1) and (4.2) show the membership functions for the temperature feature before (with mu = -2.9 and std = 3.08) and after (with mu = -2.0 and std = 3.09) modeling, respectively. Similarly, Figures (4.3) and (4.4) depict the membership functions for the pressure feature before (with mu = -4.24 and std = 3.89) and after (with mu = -4.7 and std = 3.8) modeling, respectively, demonstrating changes in the statistical properties of the features due to the modeling process.

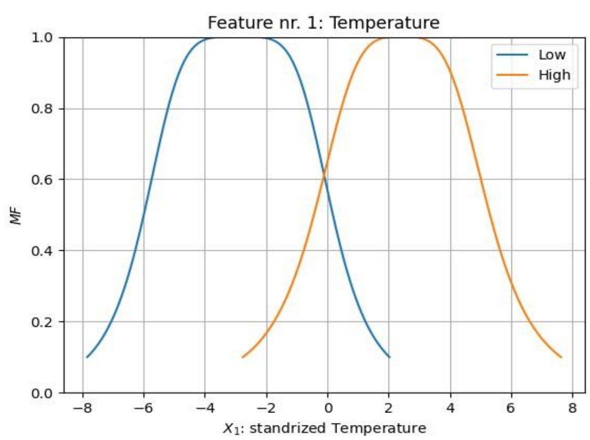

Figure 4.1: The membership functions for the temperature before modeling.

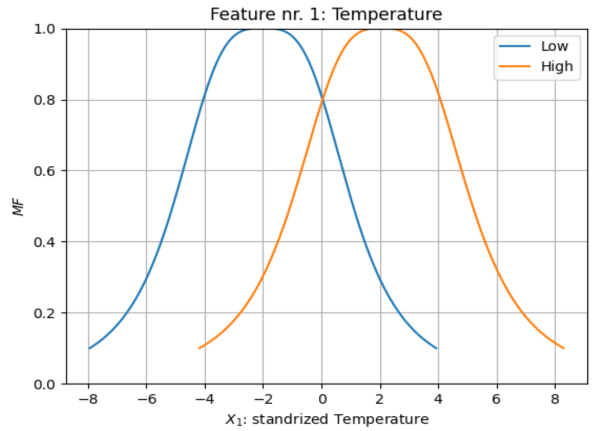

Figure 4.2: The membership functions for the temperature after modeling.

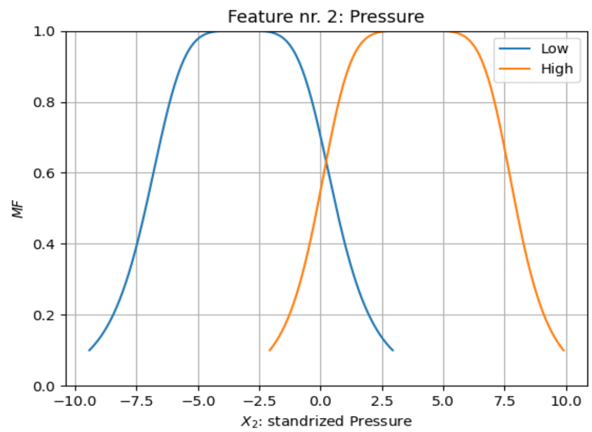

Figure 4.3: The membership functions for the pressure before modeling.

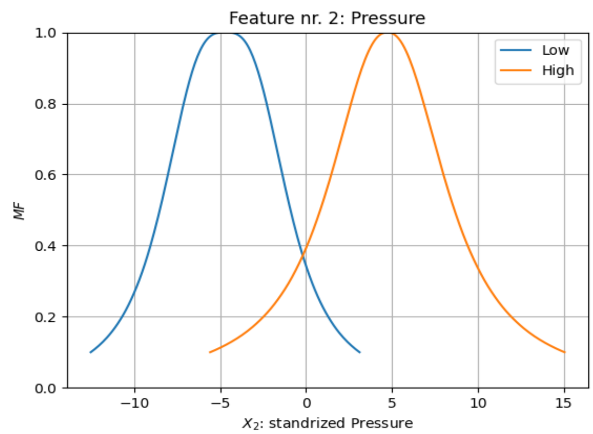

Figure 4.4: The membership functions for the pressure after modeling.

The ANFIS algorithm's prediction visualization is depicted in Figure (4.5), where it compares the predicted and actual wind speed values. The visualization revealed a discernible pattern in the ANFIS algorithm's predictions, supporting the recently mentioned accuracy measure (RMSE 0.193), although a few small outliers were visible. According to Figure (4.6), RMSE of the ANFIS model decreases significantly before 60 epochs, after which it approaches a horizontal asymptote of approximately 0.12.

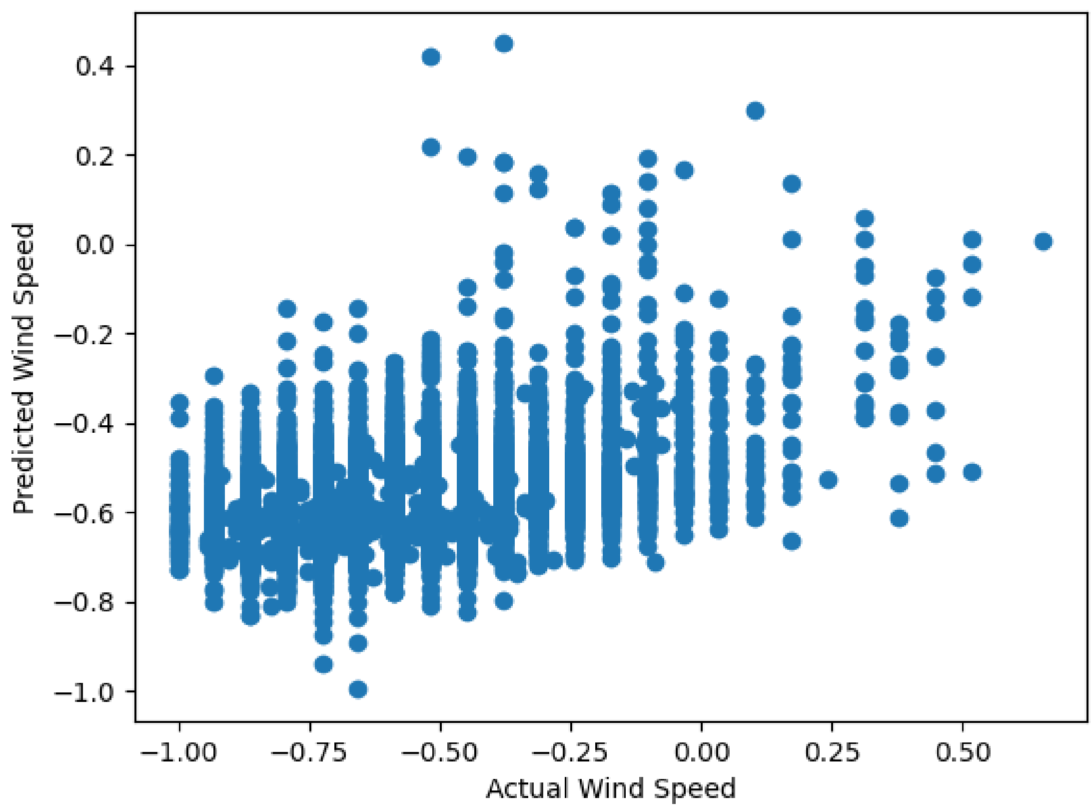

Figure 4.5: The ANFIS algorithm's prediction visualization (predicted wind speed (standarized) vs. actual (standarized)).

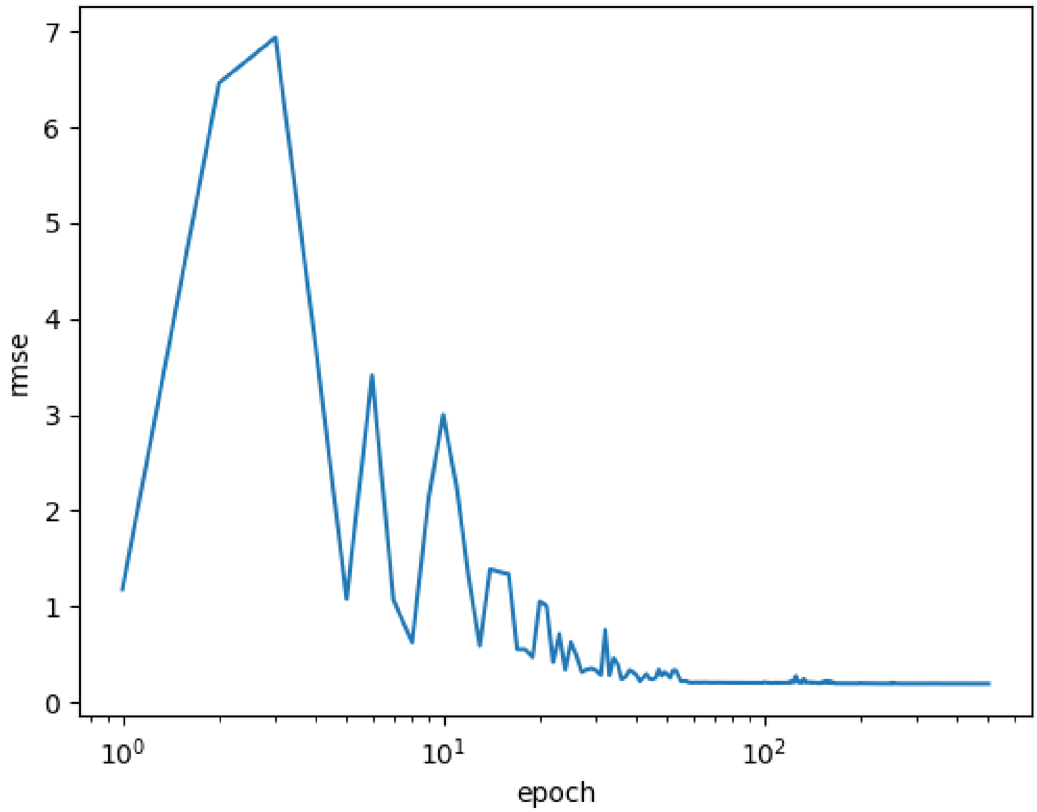

Figure 4.6: The RMSE of the ANFIS model per epoch.

Bias and R2 values were computed, yielding a bias value of 0.0003 and an R2 value of 0.15. A bias of 0.0003 implies a close correspondence between the predicted and observed wind speed values on average. A negligible bias suggests the absence of systematic overestimation or underestimation of wind speed by the model.

When considering these evaluation metrics together, a bias close to zero and a low RMSE suggest that the model is performing well in terms of predicting wind speed. However, the low R2 value of 0.15 implies that the predicted values account for only a small fraction of the observed wind speed data's variability. This suggests the potential presence of unaccounted factors or variables influencing wind speed.

4.2 KNNR:

Several experiments were performed to find the optimal model for KNNR, with different parameters used for each experiment. The variations included changing the number of neighbors, the type of metric used, and the weight function used in prediction. The results for each variation were analyzed and discussed.

In Figures (4.7, 4.8), the correlation between RMSE and the number of neighbors (k) is illustrated. Specifically, Figure (4.7) demonstrates this relationship when the weight function used is "distance", and the metric utilized is the Minkowskian metric with a power of two (also known as the Euclidean metric). On the other hand, Figure (4.8) depicts the same relationship but with the "uniform" weight function replacing the "distance" function. In both figures, RMSE initially begins at a high value of approximately two, then gradually decreases until it approaches a horizontal asymptote. In Figure (4.7), the RMSE approaches an asymptote of approximately 1.66, while in figure (4.8), it approaches an asymptote of approximately 1.42.

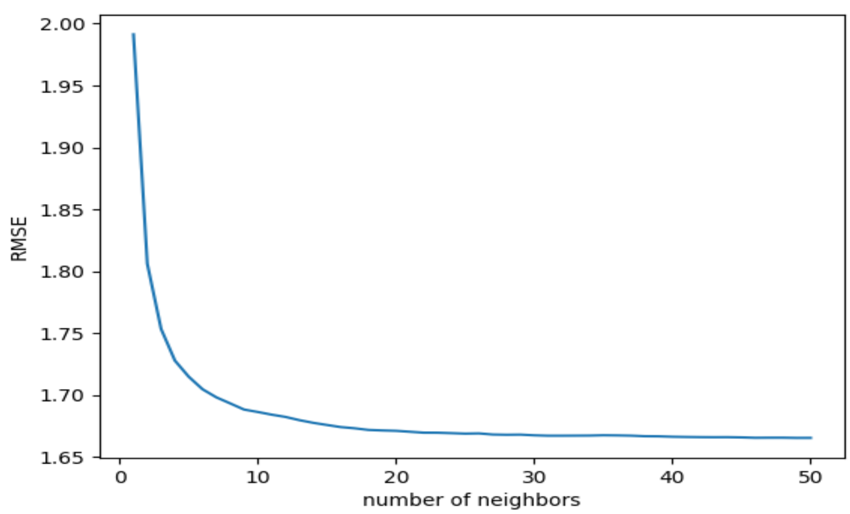

Figure 4.7: The RMSE of the KNNR model vs. the number of neighbors (“distance” weight function, p equals 2).

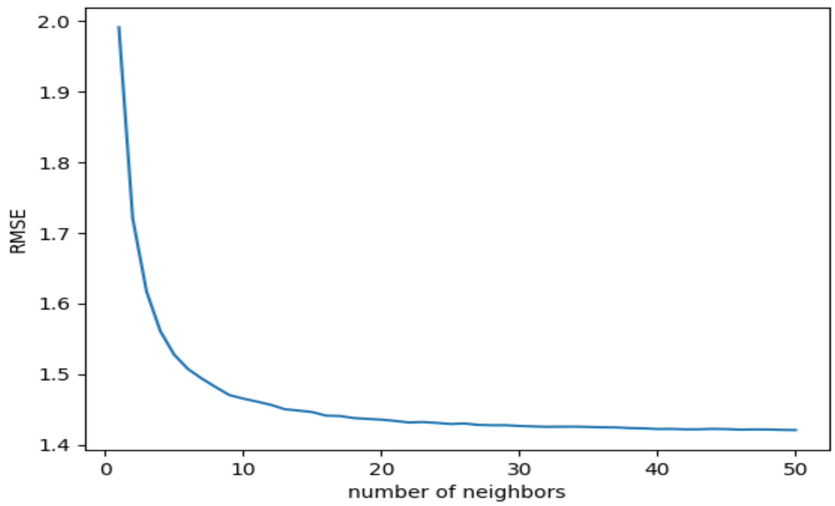

Figure 4.8: The RMSE of the KNNR model vs. the number of neighbors ("uniform" weight function, p equals 2).

Figures (4.9, 4.10) illustrates the relationship between the RMSE and the Power parameter for the Minkowski metric (p). Figure (4.9) shows this relationship when the weight function used is "distance," and the number of neighbors (k) is 5. Figure (4.10) displays the same relationship but with the "uniform" weight function replacing the "distance" function.

In Figure (4.9), the RMSE oscillates between p=5 to p=38. Generally, the RMSE increases with an increase in power. Therefore, the best values are at small powers, particularly at p=1 and p=2. On the other hand, in Figure (4.10), the RMSE generally decreases as the power increases, except for the interval between p=20 to p=40.

Overall, there is a small difference between the RMSE values in both figures, indicating that choosing the Power parameter for the Minkowski metric (p) is not a significant issue.

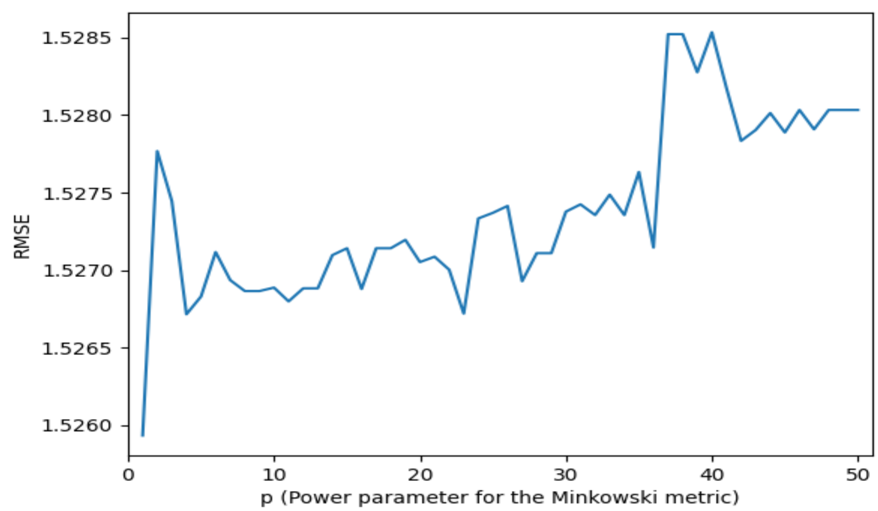

Figure 4.9: The RMSE of the KNNR model vs. the power parameter for the Minkowski metric ("distance" weight function, k equals 5).

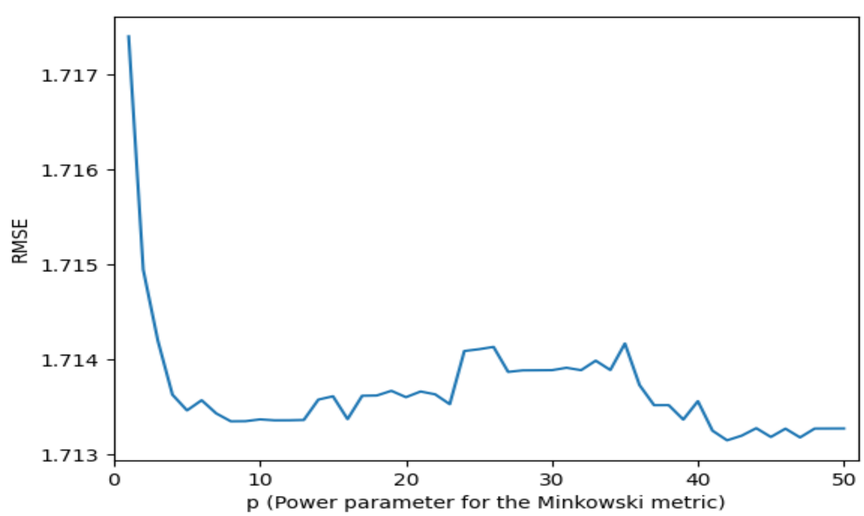

Figure 4.10: The RMSE of the KNNR model vs. the Power parameter for the Minkowski metric ("uniform" weight function, k equals 5).

In Figure (4.11), the prediction visualization of the KNNR algorithm compares the actual and predicted wind speed values. The results showed that the algorithm struggled to predict high wind speeds since they were infrequent and represented only a small percentage of neighbors for each data point. This finding supports the high RMSE accuracy measure of 1.42 mentioned earlier.

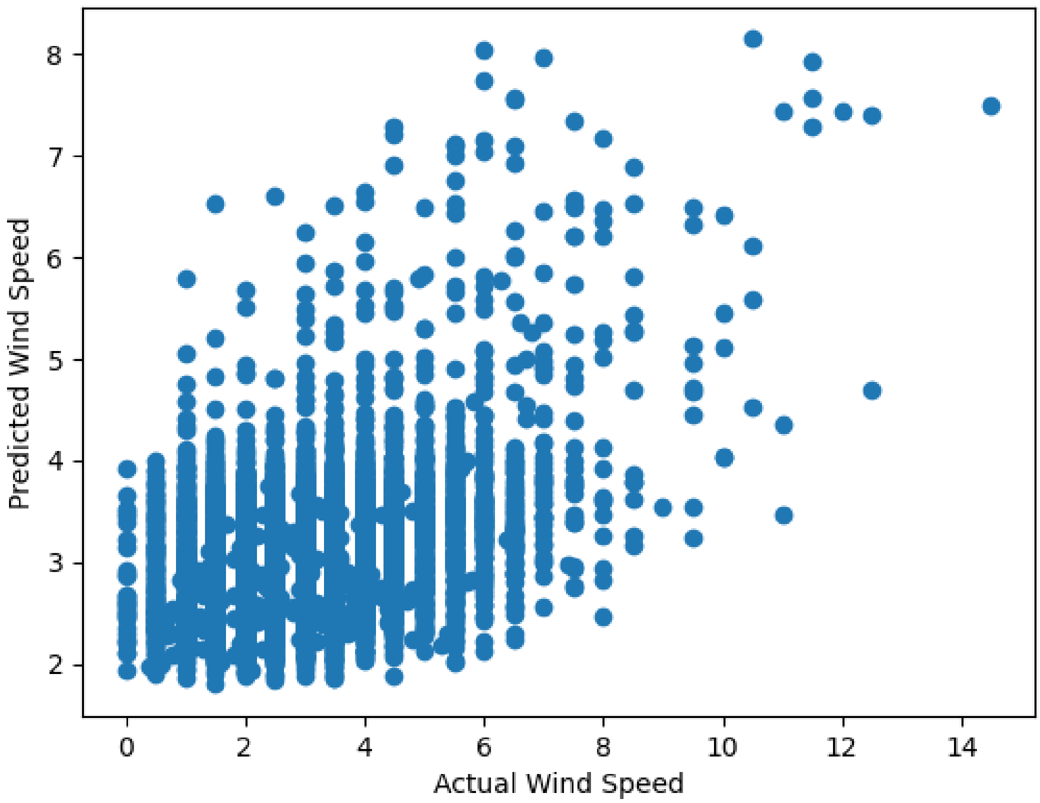

Figure 4.11: The KNNR algorithm's prediction visualization (predicted wind speed vs. actual).

The RMSE value of 1.421 indicates a relatively higher average prediction error compared to the ANFIS model. A higher RMSE suggests that the KNNR model's predictions have larger deviations from the actual observed wind speed values. The R2 value of 0.41 suggests that approximately 40.93% of the variance in the observed wind speed values is explained by the predicted values. Although it is an improvement compared to the ANFIS model, the R2 value still indicates that a significant portion of the variability in the wind speed data remains unexplained by the KNNR model. The bias value of 0.037 indicates a slight overall tendency of the KNNR model to slightly overestimate the wind speed values on average. However, the bias is relatively small and close to zero, suggesting that the model's overall tendency to overestimate or underestimate the wind speed is minimal.

When comparing these metrics to the ANFIS model, the KNNR model exhibits a higher RMSE, indicating larger prediction errors. However, the KNNR model has a higher R2 value, suggesting a better fit to the observed wind speed data compared to the ANFIS model. The bias for the KNNR model is slightly higher than that of the ANFIS model but remains relatively small. Additionally, the ANFIS model generated a denser prediction compared to the KNNR model.

In contrast to the article authored by (Salah et al., 2022), which employed six different machine learning models on the identical dataset, the ANFIS model demonstrated superior performance with lower RMSE values (0.193) compared to all six models. The respective RMSE values for these models were as follows: MLR (1.37), Ridge (1.38), Lasso (1.37), Random Forest (1.16), SVR (1.38), and LSTM (1.21).

Chapter 5 Conclusions and Future Works

5.1 Conclusions

This study investigated the use of machine learning algorithms for wind speed prediction using a dataset collected from a Palestinian meteorological station. The study focused on two popular algorithms: ANFIS and KNNR.

The results of this study demonstrated the potential of machine learning algorithms in wind speed prediction, which can help in optimizing wind energy generation and reducing the cost of energy production. The ANFIS model achieved an RMSE of 0.193 m/s, while the KNNR model achieved an RMSE of 1.42 m/s. The study also found that ANFIS outperformed KNNR in terms of accuracy, but KNNR had a faster computation time.

5.2 Future Works

There are several potential areas for future research in wind speed prediction using machine learning algorithms. One possible direction is to investigate the use of deep learning models, such as Convolutional Neural Networks (CNNs) which has shown promise in other time-series prediction tasks. Another potential area for future research is to explore the use of ensemble methods, which combine multiple models to improve accuracy and robustness. Additionally, it may be useful to investigate the impact of different preprocessing techniques and feature selection methods on the performance of machine learning models for wind speed prediction. Finally, the results of this study can be further validated using additional datasets from different locations and climates to assess the generalizability of the proposed approach.

References

Albisher, H., & Alsamamra, H. (2019). An Overview of Wind Energy Potentials in Palestine. *Journal of Energy and Natural Resources*, *8*(3), 98. https://doi.org/10.11648/j.jenr.20190803.11

Alexiadis, M. C., Dokopoulos, P. S., & Sahsamanoglou, H. S. (1999). Wind speed and power forecasting based on spatial correlation models. *IEEE Transactions on Energy Conversion*, *14*(3), 836–842. https://doi.org/10.1109/60.790962

Allison, S., Bai, H., & Jayaraman, B. (2020). Wind estimation using quadcopter motion: A machine learning approach. *Aerospace Science and Technology*, *98*, 105699. https://doi.org/10.1016/j.ast.2020.105699

Amjady, N., Keynia, F., & Zareipour, H. (2011). Wind Power Prediction by a New Forecast Engine Composed of Modified Hybrid Neural Network and Enhanced Particle Swarm Optimization. *IEEE Transactions on Sustainable Energy*, *2*(3), 265–276. https://doi.org/10.1109/TSTE.2011.2114680

Barbounis, T. G., & Theocharis, J. B. (2007). A locally recurrent fuzzy neural network with application to the wind speed prediction using spatial correlation. *Neurocomputing*, *70*(7–9), 1525–1542. https://doi.org/10.1016/j.neucom.2006.01.032

Barbounis, T. G., Theocharis, J. B., Alexiadis, M. C., & Dokopoulos, P. S. (2006). Long-term wind speed and power forecasting using local recurrent neural network models. *IEEE Transactions on Energy Conversion*, *21*(1), 273–284. https://doi.org/10.1109/TEC.2005.847954

Bouzgou, H., & Benoudjit, N. (2011). Multiple architecture system for wind speed prediction. *Applied Energy*, *88*(7), 2463–2471. https://doi.org/10.1016/j.apenergy.2011.01.037

Bzdok, D., Altman, N., & Krzywinski, M. (2018). Statistics versus machine learning. *Nature Methods*, *15*(4), 233–234. https://doi.org/10.1038/nmeth.4642

Catalão, J. P. S., Pousinho, H. M. I., & Mendes, V. M. F. (2011a). Hybrid Wavelet-PSO-ANFIS Approach for Short-Term Wind Power Forecasting in Portugal. *IEEE*

Transactions on Sustainable Energy, *2*(1), 50–59. https://doi.org/10.1109/TSTE.2010.2076359

Catalão, J. P. S., Pousinho, H. M. I., & Mendes, V. M. F. (2011b). Short-term wind power forecasting in Portugal by neural networks and wavelet transform. *Renewable Energy*, *36*(4), 1245–1251. https://doi.org/10.1016/j.renene.2010.09.016

Chang, W. Y. (2013a). An RBF Neural Network Combined with OLS Algorithm and Genetic Algorithm for Short-Term Wind Power Forecasting. *Journal of Applied Mathematics*, *2013*, 1–9. https://doi.org/10.1155/2013/971389

Chang, W. Y. (2013b). Comparison of Three Short Term Wind Power Forecasting Methods. *Advanced Materials Research*, *684*, 671–675. https://doi.org/10.4028/www.scientific.net/AMR.684.671

Chang, W. Y. (2013c). Short-Term Wind Power Forecasting Using the Enhanced Particle Swarm Optimization Based Hybrid Method. *Energies*, *6*(9), 4879–4896. https://doi.org/10.3390/en6094879

Chang, W. Y. (2014). A Literature Review of Wind Forecasting Methods. *Journal of Power and Energy Engineering*, *02*(04), 161–168. https://doi.org/10.4236/jpee.2014.24023

Chang, W. Y. (2013d). Application of Back Propagation Neural Network for Wind Power Generation Forecasting. *International Journal of Digital Content Technology and Its Applications*, *7*(4), 502–509. https://doi.org/10.4156/jdcta.vol7.issue4.61

Damousis, I. G., Alexiadis, M. C., Theocharis, J. B., & Dokopoulos, P. S. (2004). A fuzzy model for wind speed prediction and power generation in wind parks using spatial correlation. *IEEE Transactions on Energy Conversion*, *19*(2), 352–361. https://doi.org/10.1109/TEC.2003.821865

Demolli, H., Dokuz, A. S., Ecemis, A., & Gokcek, M. (2019). Wind power forecasting based on daily wind speed data using machine learning algorithms. *Energy Conversion and Management*, *198*, 111823. https://doi.org/10.1016/j.enconman.2019.111823

Erdem, E., & Shi, J. (2011). ARMA based approaches for forecasting the tuple of wind speed and direction. *Applied Energy*, *88*(4), 1405–1414. https://doi.org/10.1016/j.apenergy.2010.10.031

Focken, U., Lange, M., & Waldl, H.-P. I. (2001). *Previento-A Wind Power Prediction System with an Innovative Upscaling Algorithm*.

Growth rate of installed wind power capacity worldwide from 2012 to 2021. (2023, May 12). https://www.statista.com/statistics/268386/increase-in-wind-power-capacity/#:~:text=From%202020%20to%202021%2C%20the%20global%20installed%20wind,capacity%20worldwide%20from%202012%20to%202021%20%28in%20percent%29

Guo, Z., Wu, J., Lu, H., & Wang, J. (2011). A case study on a hybrid wind speed forecasting method using BP neural network. *Knowledge-Based Systems*, *24*(7), 1048–1056. https://doi.org/10.1016/j.knosys.2011.04.019

Guo, Z., Zhao, W., Lu, H., & Wang, J. (2012). Multi-step forecasting for wind speed using a modified EMD-based artificial neural network model. *Renewable Energy*, *37*(1), 241–249. https://doi.org/10.1016/j.renene.2011.06.023

Haque, A. U., Mandal, P., Kaye, M. E., Meng, J., Chang, L., & Senjyu, T. (2012). A new strategy for predicting short-term wind speed using soft computing models. *Renewable and Sustainable Energy Reviews*, *16*(7), 4563–4573. https://doi.org/10.1016/j.rser.2012.05.042

Hong, Y. Y., Chang, H. L., & Chiu, C. S. (2010). Hour-ahead wind power and speed forecasting using simultaneous perturbation stochastic approximation (SPSA) algorithm and neural network with fuzzy inputs. *Energy*, *35*(9), 3870–3876. https://doi.org/10.1016/j.energy.2010.05.041

Jang, J. S. R. (1992). Self-learning fuzzy controllers based on temporal backpropagation. *IEEE Transactions on Neural Networks*, *3*(5), 714–723. https://doi.org/10.1109/72.159060

Jang, J. S. R. (1993). ANFIS: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, *23*(3), 665–685. https://doi.org/10.1109/21.256541

Jung, J., & Broadwater, R. P. (2014). Current status and future advances for wind speed and power forecasting. *Renewable and Sustainable Energy Reviews*, *31*, 762–777. https://doi.org/10.1016/j.rser.2013.12.054

Jursa R. (2007). Variable selection for wind power prediction using particle swarm optimization. *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*, 2059–2065.

Kabeyi, M. J. B., & Olanrewaju, O. A. (2022). Sustainable Energy Transition for Renewable and Low Carbon Grid Electricity Generation and Supply. *Frontiers in Energy Research*, *9*. https://doi.org/10.3389/fenrg.2021.743114

Kalogirou, S. A. (2001). Artificial neural networks in renewable energy systems applications: a review. *Renewable and Sustainable Energy Reviews*, *5*(4), 373–401. https://doi.org/10.1016/S1364-0321(01)00006-5

Kani, S. A. P., & Riahy, G. H. (2008). A new ANN-based methodology for very short-term wind speed prediction using Markov chain approach. *2008 IEEE Canada Electric Power Conference*, 1–6. https://doi.org/10.1109/EPC.2008.4763386

Kariniotakis, G., Pinson, P., Siebert, N., Giebel, G., & Barthelmie, R. (2004). The state of the art in short term prediction of wind power-from an offshore perspective. *SeaTech Week-Ocean Energy Conference ADEME-IFREMER*.

Khosravi, A., Koury, R. N. N., Machado, L., & Pabon, J. J. G. (2018). Prediction of wind speed and wind direction using artificial neural network, support vector regression and adaptive neuro-fuzzy inference system. *Sustainable Energy Technologies and Assessments*, *25*, 146–160. https://doi.org/10.1016/j.seta.2018.01.001

Kitajima, T., & Yasuno, T. (2010). Output prediction of wind power generation system using complex-valued neural network. *Proceedings of SICE Annual Conference 2010*, 3610–3613.

Kwon, S.-D. (2010). Uncertainty analysis of wind energy potential assessment. *Applied Energy*, *87*(3), 856–865. https://doi.org/10.1016/j.apenergy.2009.08.038

Lange M, & Focken U. (2006). State-of-the-art in wind power prediction in Germany and international developments. *Proceedings of the Second Workshop of International Feed-in Cooperation*.

Lange, M., & Focken, U. (2006). *Physical approach to short-term wind power prediction* (Vol. 208). Springer.

Lange, M., & Focken, U. (2008). New developments in wind energy forecasting. *2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, 1–8. https://doi.org/10.1109/PES.2008.4596135

Lange, M., & Heinemann, D. (2002). Accuracy of short term wind power predictions depending on meteorological conditions. *CD-Proc. of the 2002 Global Windpower Conference, Paris, France*.

Lei, M., Shiyan, L., Chuanwen, J., Hongling, L., & Yan, Z. (2009). A review on the forecasting of wind speed and generated power. *Renewable and Sustainable Energy Reviews*, *13*(4), 915–920. https://doi.org/10.1016/j.rser.2008.02.002

Li, G., & Shi, J. (2010). On comparing three artificial neural networks for wind speed forecasting. *Applied Energy*, *87*(7), 2313–2320. https://doi.org/10.1016/j.apenergy.2009.12.013

Li, G., Shi, J., & Zhou, J. (2011). Bayesian adaptive combination of short-term wind speed forecasts from neural network models. *Renewable Energy*, *36*(1), 352–359. https://doi.org/10.1016/j.renene.2010.06.049

Liu, H., Tian, H.-Q., Chen, C., & Li, Y. (2010). A hybrid statistical method to predict wind speed and wind power. *Renewable Energy*, *35*(8), 1857–1861. https://doi.org/10.1016/j.renene.2009.12.011

Madsen, H., Pinson, P., Kariniotakis, G., Nielsen, H. Aa., & Nielsen, T. S. (2005). Standardizing the Performance Evaluation of Short-Term Wind Power Prediction Models. *Wind Engineering*, *29*(6), 475–489. https://doi.org/10.1260/030952405776234599

Marti, I., Kariniotakis, G., Pinson, P., Sanchez, I., Nielsen, T. S., Madsen, H., Giebel, G., Usaola, J., Palomares, A. M., & Brownsword, R. (2006). Evaluation of advanced wind power forecasting models–results of the anemos project. *European Wind Energy Conference, EWEC 2006*, 9-pages.

Milligan, M., Schwartz, M., & Wan, Y. (2003). *Statistical wind power forecasting models: Results for US wind farms*. National Renewable Energy Lab.(NREL), Golden, CO (United States).

Miranda, M. S., & Dunn, R. W. (2006). One-hour-ahead wind speed prediction using a Bayesian methodology. *2006 IEEE Power Engineering Society General Meeting*, 6-pp.

Miranda, M. S., Shaddick, G., Dunn, R. W., Li, F., & Bell, K. R. W. (2006). Bayesian Inferencing for Wind Resource Characterisation. *2006 International Conference on Probabilistic Methods Applied to Power Systems*, 1–6. https://doi.org/10.1109/PMAPS.2006.360207

More, A., & Deo, M. C. (2003). Forecasting wind with neural networks. *Marine Structures*, *16*(1), 35–49. https://doi.org/10.1016/S0951-8339(02)00053-9

Murugesan, S., Zhang, J., & Vittal, V. (2012). Finite state Markov chain model for wind generation forecast: A data-driven spatiotemporal approach. *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*, 1–8. https://doi.org/10.1109/ISGT.2012.6175764

Nielsen, T. S., Joensen, A., Madsen, H., Landberg, L., & Giebel, G. (1998). A new reference for wind power forecasting. *Wind Energy*, *1*(1), 29–34. https://doi.org/10.1002/(SICI)1099-1824(199809)1:1<29::AID-WE10>3.0.CO;2-B

Peng Lv, & Lili Yue. (2011). Short-term wind speed forecasting based on non-stationary time series analysis and ARCH model. *2011 International Conference on Multimedia Technology*, 2549–2553. https://doi.org/10.1109/ICMT.2011.6002447

Pinson, P., & Kariniotakis, G. (2004). On-line assessment of prediction risk for wind power production forecasts. *Wind Energy*, *7*(2), 119–132. https://doi.org/10.1002/we.114

Pinson, P., Siebert, N., & Kariniotakis, G. (2003). Forecasting of regional wind generation by a dynamic fuzzy-neural networks based upscaling approach. *EWEC 2003 (European Wind Energy and Conference)*, 5-pages.

Qin, X., Jiang, C., & Wang, J. (2011). Online clustering for wind speed forecasting based on combination of RBF neural network and persistence method. *2011 Chinese Control and Decision Conference (CCDC)*, 2798–2802.

Salah, S., Alsamamra, H. R., & Shoqeir, J. H. (2022). Exploring Wind Speed for Energy Considerations in Eastern Jerusalem-Palestine Using Machine-Learning Algorithms. *Energies*, *15*(7), 2602. https://doi.org/10.3390/en15072602

Sánchez, I. (2006). Short-term prediction of wind energy production. *International Journal of Forecasting*, *22*(1), 43–56. https://doi.org/10.1016/j.ijforecast.2005.05.003

Sfetsos, A. (2002). A novel approach for the forecasting of mean hourly wind speed time series. *Renewable Energy*, *27*(2), 163–174. https://doi.org/10.1016/S0960-1481(01)00193-8

Shi, J., Guo, J., & Zheng, S. (2012). Evaluation of hybrid forecasting approaches for wind speed and power generation time series. *Renewable and Sustainable Energy Reviews*, *16*(5), 3471–3480. https://doi.org/10.1016/j.rser.2012.02.044

Sideratos, G., & Hatziargyriou, N. D. (2007). An Advanced Statistical Method for Wind Power Forecasting. *IEEE Transactions on Power Systems*, *22*(1), 258–265. https://doi.org/10.1109/TPWRS.2006.889078

Soman, S. S., Zareipour, H., Malik, O., & Mandal, P. (2010). A review of wind power and wind speed forecasting methods with different time horizons. *North American Power Symposium 2010*, 1–8. https://doi.org/10.1109/NAPS.2010.5619586

Thøgersen, M. L., Motta, M., Sørensen, T., & Nielsen, P. (2007). Measure-correlate-predict methods: case studies and software implementation. *European Wind Energy Conference & Exhibition*, 10.

Until, E. (2013). World Population Prospects: 2012 Revision. *United Nations Department of Economic and Social Affairs: New York, NY, USA.*

Velázquez, S., Carta, J. A., & Matías, J. M. (2011). Influence of the input layer signals of ANNs on wind power estimation for a target site: A case study. *Renewable and Sustainable Energy Reviews*, *15*(3), 1556–1566. https://doi.org/10.1016/j.rser.2010.11.036

Wang, M.-D., Qiu, Q.-R., & Cui, B.-W. (2012). Short-term wind speed forecasting combined time series method and arch model. *2012 International Conference on Machine Learning and Cybernetics*, *3*, 924–927. https://doi.org/10.1109/ICMLC.2012.6359477

Wang, X., Guo, P., & Huang, X. (2011). A Review of Wind Power Forecasting Models. *Energy Procedia*, *12*, 770–778. https://doi.org/10.1016/J.EGYPRO.2011.10.103

World Bank. (2013). *World Development Indicators 2013.* . The World Bank.

Wu, Y.-K., & Hong, J.-S. (2007). A literature review of wind forecasting technology in the world. *2007 IEEE Lausanne Power Tech*, 504–509.

Zack, J. (2003). Overview of wind energy generation forecasting. *Draft Report for NY State Energy Research and Development Authority and for NY ISO, True Wind Solutions LLC, Albany, NY, USA*, *17*.

Zhang, J., Yan, J., Infield, D., Liu, Y., & Lien, F. (2019). Short-term forecasting and uncertainty analysis of wind turbine power based on long short-term memory network and Gaussian mixture model. *Applied Energy*, *241*, 229–244. https://doi.org/10.1016/j.apenergy.2019.03.044

Zhao Dongmei, Zhu Yuchen, & Zhang Xu. (2011). Research on wind power forecasting in wind farms. *2011 IEEE Power Engineering and Automation Conference*, 175–178. https://doi.org/10.1109/PEAM.2011.6134829

Zhao, X., Wang, S., & Li, T. (2011). Review of Evaluation Criteria and Main Methods of Wind Power Forecasting. *Energy Procedia*, *12*, 761–769. https://doi.org/10.1016/j.egypro.2011.10.102

Zhao, Y., Ye, L., Li, Z., Song, X., Lang, Y., & Su, J. (2016). A novel bidirectional mechanism based on time series model for wind power forecasting. *Applied Energy*, *177*, 793–803. https://doi.org/10.1016/j.apenergy.2016.03.096

Appendix A

A.1 ANFIS Code

```
“ “ “
import numpy as np

import skfuzzy as fuzz

from skfuzzy import control as ctrl

from sklearn.model_selection import train_test_split


# import data

data = np.loadtxt('final.csv', delimiter=',')

temp = data[:,0]

pressure = data[:,1]

wind_speed_actual = data[:,2]


# Split the data into training and testing sets

temp_train, temp_test, pressure_train, pressure_test, wind_speed_actual_train,
wind_speed_actual_test \

= train_test_split(temp, pressure, wind_speed_actual, test_size=0.2, random_state=42)


# find maximum, minimum and average of data columns

temp_max = np.max(temp)

temp_min = np.min(temp)

temp_mean = np.mean(temp)

s = temp_max + temp_min

m = s / 2
```

```
temp_middle = int(m)


pressure_max = np.max(pressure)
pressure_min = np.min(pressure)
pressure_mean = np.mean(pressure)
s = pressure_max + pressure_min
m = s / 2
pressure_middle = round(m,2)


wind_speed_actual_max = np.max(wind_speed_actual)
wind_speed_actual_min = np.min(wind_speed_actual)
wind_speed_actual_mean = np.mean(wind_speed_actual)
s = wind_speed_actual_max + wind_speed_actual_min
m = s / 2
wind_speed_actual_middle = round(m,2)


# Define the antecedent and consequent membership functions
temp_range = np.linspace(temp_min, temp_max, num=100)
temp_low = fuzz.trimf(temp_range, [temp_min, temp_min, temp_middle])
temp_high = fuzz.trimf(temp_range, [temp_middle, temp_max, temp_max])


pressure_range = np.linspace(pressure_min, pressure_max, num=100)
pressure_low = fuzz.trimf(pressure_range, [pressure_min, pressure_min, pressure_middle])
pressure_high = fuzz.trimf(pressure_range, [pressure_middle, pressure_max, pressure_max])
```

```
wind_speed_range = np.linspace(wind_speed_actual_min, wind_speed_actual_max, num=100)

wind_speed_low = fuzz.trimf(wind_speed_range, [wind_speed_actual_min, wind_speed_actual_min, wind_speed_actual_middle])

wind_speed_medium = fuzz.trimf(wind_speed_range, [5, wind_speed_actual_middle, 9.5])

wind_speed_high = fuzz.trimf(wind_speed_range, [wind_speed_actual_middle, wind_speed_actual_max, wind_speed_actual_max])


# Define the fuzzy variables

temp_input = ctrl.Antecedent(temp_range, 'temperature')

pressure_input = ctrl.Antecedent(pressure_range, 'pressure')

wind_speed_output = ctrl.Consequent(wind_speed_range, 'wind_speed')


# Add the membership functions to the fuzzy variables

temp_input['low'] = temp_low

temp_input['high'] = temp_high


pressure_input['low'] = pressure_low

pressure_input['high'] = pressure_high


wind_speed_output['low'] = wind_speed_low

wind_speed_output['medium'] = wind_speed_medium

wind_speed_output['high'] = wind_speed_high
```

```
# Define the fuzzy rules
rule1 = ctrl.Rule(temp_input['low'] & pressure_input['low'], wind_speed_output['low'])
rule2 = ctrl.Rule(temp_input['high'] & pressure_input['high'], wind_speed_output['high'])
rule3 = ctrl.Rule(temp_input['low'] & pressure_input['high'], wind_speed_output['medium'])
rule4 = ctrl.Rule(temp_input['high'] & pressure_input['low'], wind_speed_output['medium'])

# Define the control system and add the rules
wind_speed_ctrl = ctrl.ControlSystem([rule1, rule2, rule3, rule4])


# Create a simulation to evaluate the control system
wind_speed_sim = ctrl.ControlSystemSimulation(wind_speed_ctrl)


wind_speed_predicted_test = []
for i in range(len(temp_test)):
    wind_speed_sim.input['temperature'] = temp_test[i]
    wind_speed_sim.input['pressure'] = pressure_test[i]
    wind_speed_sim.compute()
    wind_speed_predicted_test.append(wind_speed_sim.output['wind_speed'])


rmse_test = np.sqrt(np.mean((wind_speed_actual_test - wind_speed_predicted_test)**2))
print("RMSE on testing set:", rmse_test)
```
“ “ “

A.2 KNNR Code

```
import numpy as np
```

```
import pandas as pd

import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split

from sklearn.neighbors import KNeighborsRegressor

from sklearn.metrics import mean_squared_error

from sklearn.metrics import r2_score


# Load wind speed dataset

wind_data = pd.read_csv("final.csv")


# Separate features and target variable

X = wind_data.iloc[:,0:2]

y = wind_data.iloc[:,2]


# Split the data into training and test sets

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


weights = 'distance'  # Weight points by inverse of their distance

metric = 'minkowski'  # Distance metric

k = 50  # Number of neighbors

p = 1

rmse_list = []

ks = []

# for k in range(1,51) :
```

```
#     # Instantiate KNN regression model and fit the training data
#     knnr = KNeighborsRegressor(n_neighbors=k, weights=weights, metric=metric, p=p)
#     knnr.fit(X_train, y_train)

#     # Make wind speed predictions using the test data
#     y_pred = knnr.predict(X_test)
#     rmse = np.sqrt(mean_squared_error(y_test, y_pred))
#     ks.append(k)
#     rmse_list.append(rmse)
ps = []
# for p in range(1,51) :
#     # Instantiate KNN regression model and fit the training data
#     knnr = KNeighborsRegressor(n_neighbors=5, weights=weights, metric=metric, p=p)
#     knnr.fit(X_train, y_train)
#     # Make wind speed predictions using the test data
#     y_pred = knnr.predict(X_test)
#     rmse = np.sqrt(mean_squared_error(y_test, y_pred))
#     ps.append(p)
#     rmse_list.append(rmse)

# Instantiate KNN regression model and fit the training data
knnr = KNeighborsRegressor(n_neighbors=5, weights="uniform", metric=metric, p=2)
knnr.fit(X_train, y_train)
```

```
# Make wind speed predictions using the test data
y_pred = knnr.predict(X_test)
# Calculate the RMSE
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
print("RMSE: ", rmse)
r2 = np.sqrt(r2_score(y_test, y_pred))
print("R2: ", r2)
bias = (y_test-y_pred).sum() / len(y_test)
print("Bias: ", bias)
```

التنبؤ بسرعة الرياح باستخدام خوارزميات تعلم الآلة: دراسة خاصة باستخدام ANFISو KNNR

إعداد:

خليل شيبان محمود أبوعياش

إشراف:

د. حازم دوفش، د. حسين السمامرة

ملخص

يعد التنبؤ بسرعة الرياح باستخدام خوارزميات تعلم الآلة أمرًا مهمًا للعديدِ من التطبيقات، مثل تخطيط لمحطات طاقة الرياح والتنمية الحضرية. يقدم هذا البحثُ دراسة حالة حول توقع سرعة الرياح في القدس باستخدام خوارزمية ANFIS وخوارزمية KNNR. تم تقييم أداء الخوارزميات باستخدام عدة مقاييس، بما في ذلك المتوسط التربيعي الجذري للخطأ (RMSE) والانحياز (bias)، ومعامل التحديد (R2). تُظهر خوارزمية ANFIS دقة جيدة حيث كانت قيمة RMSE (0.196) وقيمة الانحياز ($3*10^{-3}$). ومع ذلك، هناك مجال للتحسين في التقاط التغير العام (R2 = 0.15). بالمقابل، تُظهر خوارزمية KNNR R2 أعلى (0.41)، مما يشير إلى تطابق أفضل، ولكن بـ RMSE أعلى (1.421). يقدم هذا البحث رؤية حول قابلية تطبيق ANFIS و KNNR في توقع سرعة الرياح في مدينة القدس ويقترح اتجاهات بحثية مستقبلية. تحمل النتائج آثارًا عملية على تخطيط محطات طاقة الرياح والتنمية الحضرية وتقييمات البيئة في المناطق المماثلة.