DELFT UNIVERSITY OF TECHNOLOGY

MASTER

A machine learning approach to hybrid renewable system power forecasting

Author:
Flip VAN DER WEIJDEN, 5403448

Supervisor:
Prof. Simon WATSON

A thesis submitted in fulfillment of the requirements
for the degree of MSc. Sustainable Energy Technology

in the

Aerospace Engineering
Delft University of Technology

September 29, 2022

FIGURE 1: Caption

"Thanks to my solid academic training, today I can write hundreds of words on virtually any topic without possessing a shred of information, which is how I got a good job in journalism."

Dave Barry

DELFT UNIVERSITY OF TECHNOLOGY

Abstract

Aerospace Engineering
Delft University of Technology

MSc. Sustainable Energy Technology

A machine learning approach to hybrid renewable system power forecasting

by Flip VAN DER WEIJDEN, 5403448

The unprecedented growth of renewable energy has introduced the negative effect of variability in the Dutch grid. The most important risk associated with this variability is an unstable grid due to the difficulty of matching production and consumption. A novel way of mitigating this risk is by accurate power forecasting with the aid of machine learning. This research aimed to create insight into the application of machine learning in the space of renewable power forecasting, by comparing hybrid renewable energy system forecasting models with proposed benchmark models. Improved grid planning and value creation could be two potential benefits of this. In this way, the Dutch TSO, TenneT, and commercial operators can profit from an improved forecasting method.

This study proposes two machine learning techniques for the application of power forecasting. Namely, the tree regressor XGBoost (XGB) and a neural network (NN). Both techniques were used for the models of an individual PV plant and wind farm for the 1-hour and 6-hour ahead forecast horizon. Then, the hybrid models were built with the machine learning techniques and were compared to the individual models and Persistence with respect to the normalized root-mean-squared error (NRMSE). Next to the NRMSE the value creation of the models was studied with respect to Persistence on the 6-hour forecast horizon.

The results show that all proposed models outperform Persistence on the 6-hour forecast horizon. Furthermore, the results show that the hybrid models for both XGB and NN outperform the individual models on the 6-hour forecast horizon. Overall, the XGB models outperform the NN models on both the forecast horizons. Lastly, the proposed models all created positive value with respect to Persistence. However, there was a non-linearity between the NRMSE performance and the cumulative value creation over the test period.

Acknowledgements

First of all, I want to express my gratefulness to everyone at Stecc who gave me the chance to pursue an internship and to work closely with the team on exciting projects. I want to thank all my colleagues for the endless room for development and for all help during my graduation project, it grew me as a person. Secondly, I really want to thank SemperPower and Outsmart for the provision of in-depth knowledge, data acquisition and mental support during the project.

I want to show my appreciation for my TU Delft supervisors prof.dr. Simon J. Watson and Mihir Mehta. First of all for the aid in searching for the right graduation topic. Secondly, for providing all the time and effort into the weekly meetings for guiding me through the graduation process. Lastly, I wish Mihir all the best in finalising his own PhD project, I would love to read the outcomes.

Finally, I want to express my thankfulness to my friends and especially my family. You have all provided me with the necessary mental and financial support over the previous seven years to make this project possible in the first place. I'm glad to finally start my professional career and thereby begin a new chapter in life. I will do my best in using all the experience and knowledge that was given to me during my studies.

Contents

List of Figures

List of Tables

List of Abbreviations

ANN	artificial neural network
AOI	angle of incidence
AR	auto-regressive
ARIMA	auto-regressive integrated moving average
ARMA	auto-regressive moving average
BRP	balance responsible party
BSRN	Baseline Surface Radiation Network
CNN	convolutional neural network
CSP	concentrated solar power
DHI	diffuse horizontal irradiance
DNI	direct normal irradiance
ECMWF	The European Centre for Medium-range Weather Forecasts
ELM	extreme learning machine
EMD	emperical mode decomposition
ESTI	European Solar Test Installation
GHI	global horizontal irradiance
GPU	graphics processing unit
GW	gigawatt
I	integrated
IDE	integrated development environment
IMF	intrinsic mode function
KGP	Koegorspolder
KNN	k nearest neighbours
kW	kilowatt
LSTM	long short-term memory
MACC	Monitoring Atmospheric Composition and Climate
MAE	mean absolute error
MAPE	mean absolute percentage error
MBD	mean bias difference
MLP	multilayer perceptron
MSE	mean squared error
MW	megawatt
NaN	not a number
NN	neural network
NRMSE	normalized root mean squared error
NWP	numerical weather prediction
PACF	partial autocorrelation function
PCC	Pearson correlation coefficient
PSPI	physics-based smart persistence
PTU	programme time unit
PV	photovoltaic
PVGIS	Photovoltaic Geographical Information System

RAM	random access memory
ReLu	rectified linear unit
RMSD	root mean squared distance
RMSE	root mean squared error
RNN	recurrent neural network
SARIMA	seasonal auto-regressive integrated moving average
SHAP	Sapley Additive Explanations
SVM	support vector machine
SVR	support vector regression
TSO	transmission system operator
TU Delft	Delft University of Technology
WT	wavelet transform
XGB	Extreme Gradient Boosting

Chapter 1

Introduction

This chapter first introduces the context and the problem statement that form a foundation for the relevance of this research. Subsequently, the objectives and supplementary research questions are given. Lastly, the outline of this report is presented.

1.1 Context

In the last decade, renewable energy got more attention and is seen as one of the primary solutions to tackle climate change. The most popular renewable energy sources are solar photovoltaic (PV) and wind power. Since the beginning of the year 2000, the European installed renewable capacity has seen a growth of 480 GW. This included over 120 GW of PV and over 150 GW of installed wind power (Errard, Diaz-Alonso, and Goll, 2021).

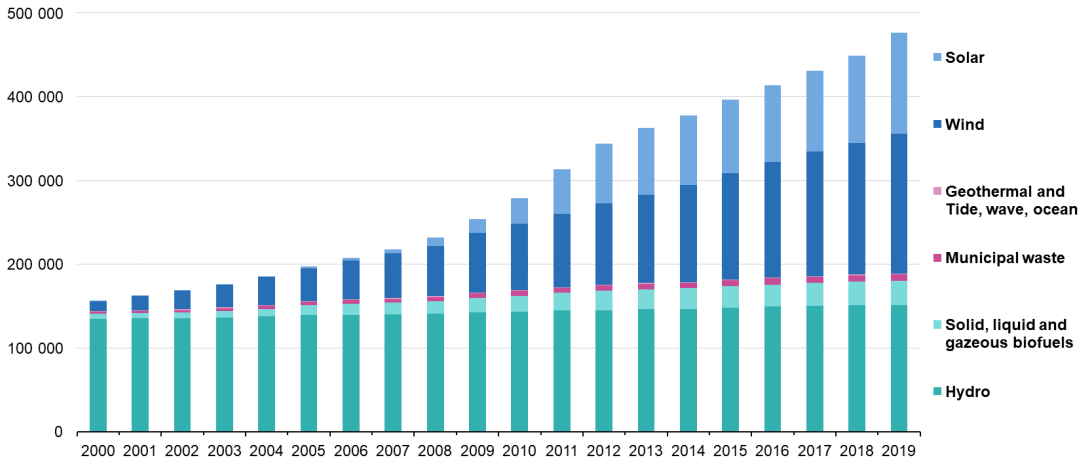


FIGURE 1.1: Installed renewable capacity from 2000 to 2019 of EU27 (Errard, Diaz-Alonso, and Goll, 2021).

The rapid growth of these intermittent technologies has had several significant impacts on the current European and Dutch electricity grid. First, the infrastructure of the electricity grid was and is not ready for connecting all the proposed power installations. This resulted in a physically congested grid, and not everyone could connect because substations were simply full concerning their technical capabilities (Weibelzahl, 2017). Combining PV and

wind in a hybrid renewable system could potentially accelerate the penetration of renewable energy when sharing the same grid connection.
Secondly, the intermittency of renewable power production poses a potential risk of imbalance in the grid, which can negatively impact grid stability. This can even happen on small-scale energy systems, e.g. hybrid PV and wind systems that need to curtail their energy. On larger scales, an unstable grid can obstruct further penetration of renewable capacity and transportation of renewable energy. Nonetheless, several studies have shown that the application of accurate power forecasting, on PV and wind production, can alleviate the problem of intermittency which can help stabilize the entire grid (Dong et al., 2016), (Chaouachi et al., 2009), (Kemmoku et al., 1999), (Bjørndal, Bjørndal, and Rud, 2017).

The Dutch grid operator (TSO), TenneT, is responsible for grid quality and stability and is thus an essential beneficiary of accurate power forecasting. The TSO has many subdivided parties, Balance Responsible Parties (BRPs), who have an obligation to balance their respective portfolio to minimise imbalances in the grid. Accurate planning of actual production is essential for these parties since this minimises their additional costs. Renewable energy producers are obligated by their respective BRPs to communicate their day-ahead power forecast to balance their portfolio if necessary. Accurate power forecasting can play a vital role to minimize costs for these parties.

Current studies on power forecasting generally focus on predicting the output of one single generation type. The aforementioned hybrid energy systems are relatively new to the current Dutch power system, and current literature on power forecasting is scarce (Qadir et al., 2021), (Pang et al., 2021). This study proposes a forecast model for hybrid renewable energy systems, where features and variables of a PV plant and a wind farm are combined. In this way, the opportunity for improved prediction quality is studied for two machine learning techniques. The studied forecast horizons are the 1-hour and 6-hour forecast horizons, so grid stability and commercial profit could be researched.

To conclude, the application of accurate power forecasting on hybrid renewable energy systems could potentially contribute to solving significant risks in the current electricity grid. This could possibly serve a technical and a financial benefit for commercial- and grid operators.

1.2 Objectives and research questions

This research has two primary objectives:

- Create insight into the advantage of hybrid system power forecasting when combining individual data streams of production units into one overarching model.
- Create insight in the forecast value created by the hybrid prediction model.

To realize the above-mentioned objectives, the following supporting research questions have been formulated:

1. What is the current state of the art in the space of short-term renewable power forecasting?
2. What are the most relevant and important input variables for the proposed forecast horizon?

3. How do the individual models' performance compare to the proposed respective benchmark methods?
4. How do the proposed hybrid models perform in comparison with the benchmark method and the sum of the individual models?
5. What is the financial advantage of implementing a hybrid model over the splitting of the predictions into separate models?

1.3 Outline

This research project is structured in the following way:

- Chapter 1: This chapter includes the introduction. Here the context of the project is discussed with the problem statement. Lastly, the objectives and the research questions are explained.
- Chapter 2: Here, the literature review is provided. First, a short introduction on solar and wind energy is given, then the relevant background on forecasting is explained, and lastly, an elaboration on two machine learning techniques is given.
- Chapter 3: This chapter provides the methodology. The process of acquiring data for the modelling and evaluating the results is discussed.
- Chapter 4: Here, the experimental results are presented with the rationale and limitations.
- Chapter 5: This chapter presents the main conclusions on the posed research questions. Furthermore, this chapter reflects on the results and provides recommendations for further research.

Chapter 2

Literature Review

This chapter gives insight into the theoretical background and presents relatable studies in the space of solar & wind power forecasting.
Section 2.1 gives insight into current knowledge on the physics concerning solar and wind power and their relationship. Then, Section 2.3 elaborates on the different techniques of handling data, how to modify data and how to engineer features to enhance model performance. Thereafter, Section 2.4 and 2.5 provide an overview of different forecasting techniques and provide a deep dive into two machine learning models. Lastly, state-of-the-art on current solar and wind power forecasting is provided in Section 2.2 and 2.6.

2.1 Solar & wind energy

The production of solar and wind energy are two well-known means of producing electricity. The first signs of a Photovoltaic (PV) device originated in 1839 when Becquerel demonstrated the PV effect by illuminating Pt electrodes coated with a thin layer of gold. This device had an efficiency of around 1% (Rhodes, 2010). The first signs of harvesting electricity from wind date back to 1887 in Scotland. Prof. James Blyth created a cloth-sailed turbine to power the lighting in his cottage (Price, 2005). Both techniques have endured technological breakthroughs ever since and are anticipated to be the primary sources of electricity generation shortly (Ellabban, Abu-Rub, and Blaabjerg, 2014).

2.1.1 Solar energy

The source of solar energy on earth is the sun. Nuclear interactions in the core where hydrogen is converted into helium induce energy emission in the form of heat and radiation. This radiation travels to earth in the form of electromagnetic waves and are called photons. The irradiance is a measure of the amount of power per unit area or radiant flux received by the earth's surface and is given in Watts per square meter (W/m^2) (Meinel and Meinel, 1977). When photons enter the earth's atmosphere, they are reflected, absorbed, or radiate further to the earth's surface. This results in three different radiation components:

- **Direct Normal Irradiance (DNI)** or direct radiation is the component that has the angle of incidence (AOI) perpendicular to the solar panel.
- **Diffuse Horizontal Irradiance (DHI)** or diffuse radiation is the group of photons incident on the solar panel that are radiated by clouds or reflected by the surroundings, e.g. trees, buildings, ground, etc.

- **Global Horizontal Irradiance (GHI)** or global radiation is the combination of DNI and DHI and represents the total W/m^2 incident on the panel. The mathematical expression is given with Equation 2.1 and a schematic representation is presented in Figure 2.1.

$$GHI = DHI + DNI \cdot cos(AOI) \tag{2.1}$$

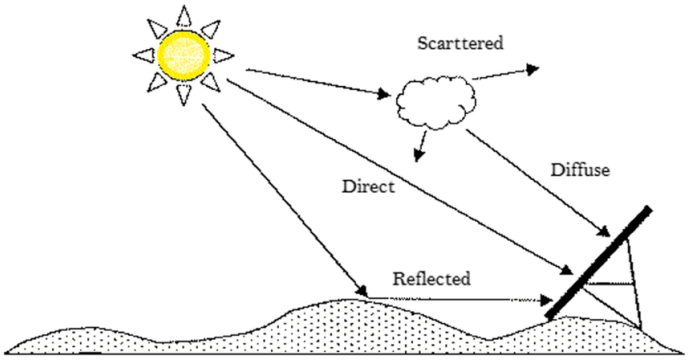


FIGURE 2.1: Schematic representation of the three radiation components in earth's atmosphere on a solar panel. GHI is a combination of direct, diffuse and reflected components (El Mghouchi, 2022).

Electricity generation

Nowadays, two main techniques exist for generating electricity from solar radiation: Photovoltaic technologies and Concentrated Solar Power (CSP). The main principle of PV technology is incoming photons from the sun transferring energy to electrons in the semiconductor material. These electrons jump to a higher energy level and move from the valence band to the conduction band. At the back of the PV cell, a back-electrode exists that closes the circuit enabling the utilisation of electron flow (Rhodes, 2010) (Meinel and Meinel, 1977).
CSP is the technique of transferring the energy from the photons to a solid, liquid or gas in the form of heat. In this way, steam is generated, which eventually drives a steam turbine.

2.1.2 Wind energy

In meteorology, wind speed is an atmospheric variable that originates from air particles that move from high to low-pressure areas. These pressure differences arise from fluctuating temperatures in different areas. Other physical causes for the disturbances in wind speed are given in table 2.1. Wind speed is given in (m/s) and is usually measured with an anemometer.
Wind turbines are used to convert the energy from the wind into usable electrical energy. The blades of the turbine create thrust and rotational speed. A powertrain transfers the rotational speed to the generator where the electrical energy is generated. The theoretical formula of wind power is represented by Equation 2.2. As can be seen $P_w \propto v^3$. Losses and power factors are not taken into account in this formula.

$$P_w = \frac{1}{2}\rho A v^3 \tag{2.2}$$

Here, P_w is the generated power in Watts, ρ is the density of air in (kg/m^3), A is the cross-sectional area of the rotor in (m^2) and v the wind speed in (m/s). This formula is also used for physical forecasting models, as will be discussed in Section 2.4.3.

TABLE 2.1: Frequency domains with physical cause on wind speed (Zhang, Wang, and Wang, 2014)

Frequency domain	Time domain	Physical cause
Extra-low	Months	Climatic changes and human activities
Low	Days	General changes of weather patterns
Medium	Hours	Atmosphere thermal exchange
High	Minutes	Local meteorological effect
Extra-high	Seconds	Turbulence effects of wind speed

2.2 State of the Art

A review on the state of the art was performed in addition to the existing literature review. This study on previous work focused on the use of machine learning techniques for the purpose of forecasting PV, wind and hybrid renewable system power outputs. The hybrid systems were preferred to be a combination of PV and wind generation units. Furthermore, the focus was on the short-term forecast horizon and the use of multiple meteorological features.

(Milligan, Schwartz, and Wan, 2003) proposed a model based on time-series analysis that could improve over relatively simple persistence forecasts. The forecast technique was chosen to be an auto-regressive moving average (ARMA) model to predict both wind speed and wind power. The selection was made because of its simplicity and well-known characteristic as a time-series predictor. The proposed models and methodology beat persistence by 7% in the 1-hour prediction horizon and by 18% in the 6-hour horizon. On the contrary, the 10-minute horizon unperformed with respect to persistence and thus showed that the use of ARMA is heavily dependent on the time period of the forecasts.
The implementation of neural networks (NN) for wind speed and wind power forecasting has been researched by (Panteri and Papathanassiou, 2008), (Kariniotakis, Stavrakakis, and Nogaret, 1996) and (Cadenas and Rivera, 2009). All studies have shown that the NN models outperform persistence on multiple hours ahead predictions. Furthermore, (Cadenas and Rivera, 2009) constructed separate models for each month of the year with seven years of data from measurement systems in La Venta, Mexico. This study concluded that for the short-term predictions a model with two hidden layers had an overall better performance. The model had a mean squared error (MSE) of 0.0016 and thus the model showed very high accuracy for the implementation of local energy security control.
(Dong et al., 2016) proposed a NN prediction model for the day ahead forecast, with an hourly and quarter-hourly resolution, with the use of numerical weather prediction (NWP) data. This study developed three different NN models by clustering the NWP data. First, the historical NWP data is divided into clusters by k-means. Then, the NN was trained on the NWP variables of air pressure, wind speed, temperature and wind direction. The proposed NN prediction model outperformed the persistence model with a normalized root-mean-square error (%) of 12%. This study confirms that the implementation of NWP data on prediction models can have a positive impact on outperforming persistence forecasting models.

In predicting PV power output, (Chaouachi et al., 2009) developed four NN models to predict the 24-hour ahead solar power of a 20 kW PV system, with a resolution of an hour. The proposed models were: multi-layered perceptron, radial basis function and two recurrent neural networks (RNN). Here, (Chaouachi et al., 2009) concluded that the NN ensemble had the highest accuracy because the NN ensembles improved on the generalization and noise tolerance of the original dataset.
In addition to the previous research, (Kemmoku et al., 1999) developed a multi-stage NN to further reduce the mean error of an earlier constructed single-stage NN. With these NN models, the insolation of the next day was predicted. The multi-stage NN was developed with three stages. The first stage predicted the average atmospheric pressure for the next day on the basis of atmospheric pressure data of the past day. The second stage predicted the level of insolation for the next day based on pressure and weather data of the past day. The last stage predicted the insolation of the next day based on the earlier predicted insolation level and weather data of the previous day. The research concluded that the multi-stage NN reduced the mean error by about 10% with respect to the single-stage NN. The persistence forecast was outperformed with about 6% by the multi-stage NN.

(Qadir et al., 2021) studied the performance of several machine learning models in predicting the power of hybrid PV- and wind systems in 2021. The study incorporated seven weather factors with a significant impact on the PV and wind power output. The goal of the study was to identify the best performing and fasted performing machine learning model. The performance of seven different forecast methods was studied: ExtraTrees, AdaBoost, Support Vector Regression (SVR), K-Neighbors, Gaussian Process, Multilayer perceptron (MLP) and Linear regression. The performance of these models was tested over the MSE, MAE and R^2 metrics.
The ExtraTrees regressor had an MSE score of $4.93 * 10^{-5}$, an MAE score of 0.005 and an R^2 score of 0.98. The MLP model had an MSE score of $2.86 * 10^{-5}$, an MAE score of 0.004 and an R^2 score of 0.99. The linear regressor scored the best with an MSE score of $1.041 * 10^{-6}$, an MAE score of 0.00083 and an R^2 score of 0.996.
(Pang et al., 2021) proposed a long short-term memory (LSTM) network to predict the combined power production of hybrid renewable systems including concentrated solar power, PV- and wind generation. The model was trained on historical power and meteorological data for all three generation units. The location of the generation units was chosen to be Northwest China. First, an analysis of correlation was performed. Then, the predictions of the models were tested on RMSE and NRMSE.
The first analysis showed a weak correlation between PV and wind power. (Pang et al., 2021) states that during the day, the DNI is high and wind speed is generally weak. After sunset at night, the wind speed increases in strength because of the surface temperature change.
A benchmark of three individual models was created to test the hybrid model's performance. The group of three individual models had a combined NRMSE of 6.25% whereas the hybrid model, for PV, wind and CSP had an NRMSE performance of 5.50%. The main reasons for the increase in performance, following (Pang et al., 2021), is the aforementioned weak correlation between PV and wind and the strong correlation between DNI, PV and CSP.

To sum up, the aforementioned studies showed that both statistical models as deep learning models are capable of outperforming the traditional benchmark models. Additionally, all studies in this literature review incorporated multi-variable inputs. These inputs were all related to relevant meteorological variables both historically as predicted by NWP models. What stands out is that both the discussed research papers on forecasting hybrid renewable

system power reported the advantage of the hybrid forecast models over the individual forecast models. Hence, the objectives, research results and conclusions of (Pang et al., 2021) were the basis of this research project.

2.3 Data Analysis

This section presents best practices in the space of data analysis. First, data exploration and visualization methods will be discussed. Then, several techniques on cleaning data are presented. Finally, the last subsection provides an overview of various data transformation methods.

2.3.1 Data exploration

This subsection focuses on the exploration of raw datasets. Methods for visualizing correlations between variables and for the decomposition of individual variables are explained.

Correlation

When first assessing a multivariable dataset a correlation analysis can potentially provide useful insights. Dependencies of different variables are usually given on a scale from -1 to 1. The most common and user-friendly correlation metric is the Pearson correlation coefficient (PCC) or the r-value. It is a linear correlation of two sets of variables. In general, the PCC can be described as the ratio between covariance and the product of the standard deviation, i.e. the normalized covariance as can be concluded from Equation 2.3. Here, n presents the number of samples, x_i the variable x at instance i, y_i the variable y at instance i, $\bar{x}$ and $\bar{y}$ represent the means of both variables. R-values of 1 would represent 100% positive correlation whereas an r-value of 0 has no correlation and an r-value of -1 represents 100% negative correlation (Rodgers and Nicewander, 1988).

$$r_{xy} = \frac{n\sum_i(x_i y_i - n\bar{x}\bar{y})}{\sqrt{\sum_i(x_i^2 - n\bar{x}^2)}\sqrt{\sum_i(y_i^2 - n\bar{y}^2)}} \tag{2.3}$$

With this coefficient, a graph can be generated to provide an overview of the underlying correlations in a dataset. An example of such a graph is the Pearson correlation matrix, as presented in Figure 2.2.

(Hodge and Florita, 2013) conducted research in 2013 where the correlation between wind and solar power was investigated. The forecasts and historical data were obtained from the Western Wind and Solar Integration Study. During this study, the 4-hour ahead and the day ahead prediction errors were analyzed. The research concluded that solar and wind power have a weak correlation of around -0.1 for the day ahead predictions and -0.3 for the 4-hour ahead predictions. As (Hodge and Florita, 2013) expected, the short-term predictions had a higher Pearson correlation than the day-ahead forecast.

Decomposition

Next to analyzing the correlation between variables in a dataset it can also be useful to decompose a single variable for analysis. In this way, cyclical and frequential behaviour can be observed for a better understanding of the individual variables.
When decomposing a time-series dataset, one breaks down the original series into multiple

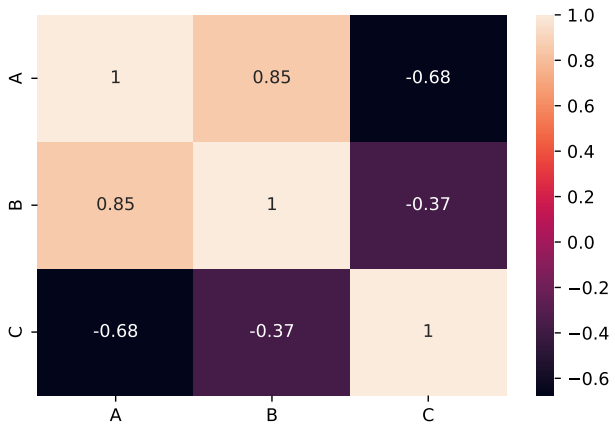


FIGURE 2.2: Example of a Pearson correlation matrix. The numbers inside this matrix are indicative. They represent the Pearson value of the two respective variables.

subsequent series. These series can be systematic and non-systematic, whereas the systematic series contain structure and thus are predictable. For each of the series, a separate forecast can be constructed to improve the overarching forecast value (Dhiman and Deb, 2020). In classical decomposition the original series is broken down into three different components:

- **Trend**, the fluctuating value of the original series that can either increase or decrease
- **Seasonal**, which is the cyclic component of the original series
- **Noise**, which is the random value of the original series

With these components there exist two methods of combining into one overarching forecast value: the additive model and the multiplicative model, as described in equations 2.4 and 2.5 respectively (Nwogu et al., 2019).

$$y(t) = \text{trend} + \text{seasonal} + \text{noise} \tag{2.4}$$

$$y(t) = \text{trend} \cdot \text{seasonal} \cdot \text{noise} \tag{2.5}$$

In both equations, $y(t)$ represents the overarching forecast value.

2.3.2 Input data wrangling

The acquisition of data concerns a large part of any research in the space of data science. Data acquisition can be done by mining algorithms, real-time measurement systems, etc. With every technique, the data can potentially have many irrelevant and missing parts. This might be due to the fact that an algorithm misinterprets a value or by downtime of a measurement installation. To handle missing or incorrect noisy parts, data cleaning is performed to enhance the overall model performance.

Missing data

The occurrence of missing data in acquired datasets is a common understanding in data science. Missing values range from missing dates in a time-series or a NaN value in the

acquired variables (García et al., 2016). Treating missing values can be difficult since the nature of variables vary much. Inappropriate handling of the missing values can easily result in decreasing performance of the model and in drawing the wrong conclusions (García et al., 2016). There are various ways to tackle the problem of missing values in a dataset. The most commonly used methods are erasing the instance or Do Not Impute (DNI), forward and backward filling, K-means clustering, K-means neighbour and moving average imputation (Luengo, García, and Herrera, 2012). Simply erasing instances with missing values can be effective if working with large datasets that are not a time-series (Luengo, García, and Herrera, 2012). This is mainly due to the fact that time-series models require consistent DateTime values to work properly.

Noisy Data

Noisy data is meaningless data that isn't interpretable by machines or computers. It can occur due to messy data mining algorithms, data entry errors, etc. In supervised learning problems, noise can impact the input features, the output values or both. If noise is present in the input variables, one refers to attribute noise (García et al., 2016). A less common instance of noise problems is when noise is present in the output values, which means the performance decreases rapidly. This kind of noise is known as class noise (Zhu and Wu, 2004) (Luengo, García, and Herrera, 2012).
Two common methods of reducing noise are data polishing and the use of noise filters. Data polishing is usually meant for smaller sizes of noise, whereas noise filters can be applied more easily and do not require any modification to the data acquisition technique (Zhu and Wu, 2004).

2.3.3 Pre-processing

This subsection explains the different techniques of transforming cleaned data into usable features to feed into the forecasting models.

Normalization

When feeding numerical data into a statistical or learning model one should take standardizing and normalizing the dataset into account. The main cause for using these methods is that a given dataset can contain data from multiple different sources and thus contain different distributions and ranges. These distributions and ranges are usually not equal which can cause problems with inaccurate results. Varying ranges can maliciously influence a target value due to higher values having more importance over smaller values. So, cleaning the data with standardization, normalization and regularization methods can potentially improve the performance of a data-driven model.

$$x = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{2.6}$$

$$x = \frac{x_i - Q_1(x)}{Q_3(x) - Q_1(x)} \tag{2.7}$$

$$Z = \frac{x - \mu}{\sigma} \tag{2.8}$$

$$\mu = \frac{1}{N}\sum_{i=1}^{N}(x_i) \tag{2.9}$$

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{2.10}$$

When normalizing a data set all ranges will be scaled to a value between [-1,1] or [0,1], e.g. the MinMax scaler as described in equation 2.6. Standardization is usually performed with the robust or Z-standardization equation, represented by Equations 2.7 and 2.8 respectively.

Wavelet transform

Wavelet transform (WT) is the method to transform data into domain and frequency domain features. The time-series is converted to a spectrum where the spectrum will be analyzed as features. The main advantage of the WT method is that outliers in the dataset have a lower effect on the prediction results (Wu et al., 2022).
(Zhang, Yu, and Huang, 2019) proposed a PV power forecasting model using WT combined with an extreme learning machine (ELM). A method was introduced to build a separation prediction model for each instance using historical weather data and PV power output. The weather data were used as input features whereas the PV power output was used as the target to develop a supervised learning model. This method was compared with a model that combined *k*-nearest neighbours (KNN) with a support vector machine (SVM) that was trained on the same data. (Zhang, Yu, and Huang, 2019) showed that the proposed WT model performed much better than the KNN prediction model.

Empirical mode decomposition

Whereas WT is based on domain and frequency, empirical mode decomposition (EMD) is based on time-domain processing. EMD can instantly decompose original time-series into various individual mode functions and residuals. These mode functions are then arranged from high- to low frequency. With this technique, non-linear signals can be decomposed without using any base functions (Wu et al., 2022).
In 2017, (Majumder, Behera, and Nayak, 2017) studied the implementation of EMD for PV forecasting purposes. A PV power forecasting method was proposed that combined EMD with ELM in a hybrid model. Here the signals were further decomposed into different intrinsic mode functions (IMF) to decompose the non-linearity. The results showed that the hybrid EMD-ELM model outperformed the single ELM model. Additionally, the 5-minute ahead forecast models outperformed the 30-minute models with an average of an NRMSE value of 8%.

2.3.4 Feature engineering

After processing and transforming the acquired data, the last step is engineering extra features to improve the model performance. The construction of these new features can either be a modification or combination of existing features or can be a form of time indication. Generally feature engineering consists of the following categories.

Date & Time related features

In time-series data the date and time variables are important. When acquiring data these variables are usually given in DateTime or are given in the respective range for an hour, day, month, etc. Since every data point should have the same importance in the model one

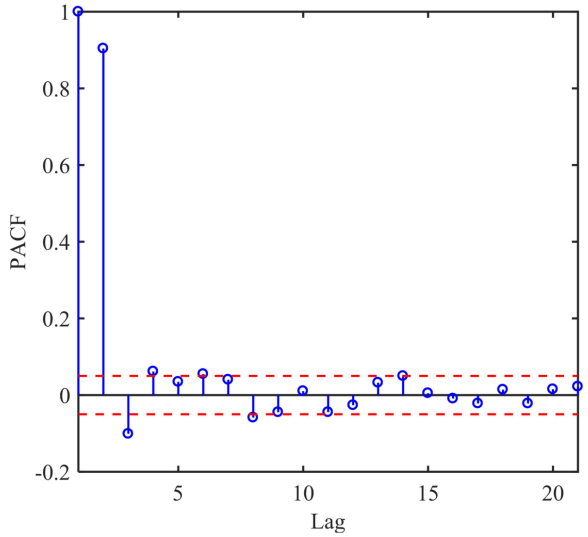


FIGURE 2.3: The PACF of wind speed data (Zhang et al., 2020).

approach is to transform the date and time variables into cyclical features. Engineering a sine and cosine component ranging from -1 to 1 a variable can never be zero and thus have zero influence on the forecast value. This approach of using a time variable is especially useful for solar forecasting because of the cyclical behaviour of the irradiance.

Lag features

Lag features are entries that contain information about past values of time-series variables. This is a useful approach when handling data that has a level of persistence, i.e. that current values are strongly related to previous values. To identify which lag features to use and how many steps back in time a feature needs to be engineered, the partial autocorrelation function (PACF) can be used. This function extracts relationships between past and current values of a time-series. It computes the partial correlation of a stationary dataset with respect to its previous values (Zhang et al., 2020). As can be seen in Figure 2.3 the first and second lag have a strong correlation with the current wind speed value.

Rolling & expanding window

A window feature represents a combination, modification or summary of one or more values of a variable. Window features consist of sliding window features and expanding window features. An example of a sliding window feature is the moving average of x values in the past, which can be seen as a summary of past values.

2.4 Solar & wind power Forecasting

Power forecasting models can be divided into two separate groups. One group is connected to the analysis based on the time series of historical wind or irradiance-related data. The other group is based on values from a numerical weather prediction (NWP) model. However, forecasting of power is typically described by a physical, ordinary statistical or machine learning model approach. A combination of the aforementioned forecast methods is

called a hybrid model (Foley et al., 2012).
Power forecasting can be subdivided into different forecasting horizon groups as shown in Table 2.4. Each forecast horizon has its contribution to a different purpose. Short-term horizons are mainly used for grid planning and dispatch problems. Whereas further horizons are used for strategic decision-making with respect to financial and market optimization purposes. This section explains all commonly used forecasting techniques used for solar- and wind power forecasting.

TABLE 2.2: Forecast horizons for Wind power forecasting (Soman et al., 2010)

Horizon	Timescale	Purpose
Very-short	$\leq$ 30min	Grid planning and market clearing
Short	30min - 6hours	Economic dispatch planning
Medium	6hours - 1day	Day-Ahead market security
Long	1day - 1week	Unit commitment decisions

2.4.1 Persistence

This method is known as the 'Naive predictor'. This approach assumes that the one-step ahead wind speed is equal to the current value of the wind speed by:

$$P(t + \Delta t) = P(t) \tag{2.11}$$

This method has very high accuracy at very-short time scales and because of its simplistic design, it has no need for large computational systems (Soman et al., 2010). Any other forecast method should be tested against this persistence approach to check how to create a relevant impact on forecast accuracy.

2.4.2 Smart persistence

Physics-based Smart Persistence model for intra-hour forecasting of solar radiation (PSPI) that decomposes short-term GHI forecast into the computation of extraterrestrial solar radiation, solar zenith angle and the forecasting of cloud albedo and cloud fraction (Kumler, Xie, and Zhang, 2019). The solar zenith angle is shown in Figure 2.4, this is a well predictable variable that makes smart persistence more extensive than regular persistence. Although the forecasting of cloud albedo and cloud fraction is conducted with an assumption of persistent cloud structures, they have the potential to be improved with more comprehensive time series analysis or machine learning techniques.
Compared to the persistence and smart persistence models, the PSPI does not require additional observations of various atmospheric parameters, but it is customizable that additional observations if available can be ingested to further improve the forecast (Kumler, Xie, and Zhang, 2019).

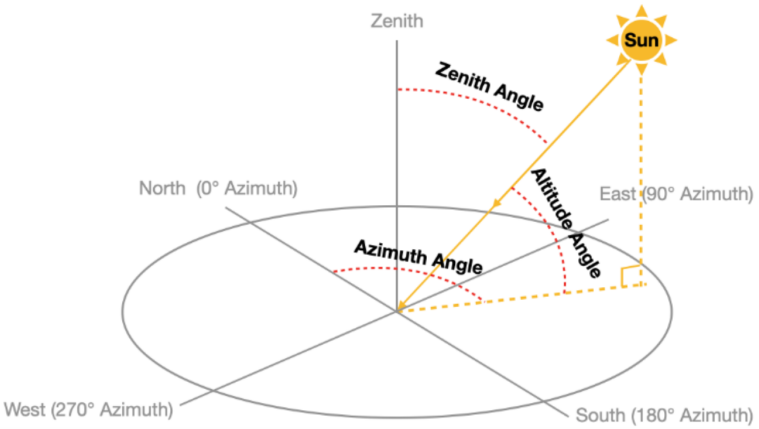


FIGURE 2.4: Schematic representation of the solar zenith angle, altitude angle and azimuth angle (Zhang et al., 2021)

2.4.3 Physical models

Physical methods or deterministic methods are driven by global data of meteorological measurements and atmospheric mesoscale models. These methods usually require large computational systems to operate. A first step to forecast wind power is to acquire detailed meteorological information using mesoscale or linear models that are used to generate wind speed, e.g. NWP data (Artipoli and Schiavone, 2014).
Physical models are usually run 1 to 4 times per day because of the large computational requirements of the operating systems. To make the utilization of these physical models economically feasible they are used for the Medium-term horizon (Soman et al., 2010).

Solar

The physical power calculation approach is based only on the main design parameters of the PV system, and it does not require any historical data (Mayer and Gróf, 2021). Therefore, it is a useful way for PV plant owners who have all the necessary information available in the design documentation of the plant. Regarding its accuracy, evidence can be found for both lower and higher performance of the physical modelling compared to machine learning methods. However, most studies agree that hybrid models, e.g., by adding physically-calculated properties or even just the clear sky irradiance to the inputs of a neural network, outperform either the purely physical or statistical approach (Schmelas et al., 2015).
Physical PV power forecasting models have a high significance in two main applications, in power prediction of new PV installations where no historical production data is available, and hybrid physical and data-driven modelling, which is the most accurate PV power forecasting technique (Mayer and Gróf, 2021).

Wind

With the acquired NWP data, turbine specifics and topography of the surroundings a physical model of a wind farm can be useful. Wind speed data is usually measured and used at a 10-meter height mean wind speed in meters per second. This wind speed data needs to be scaled to the proper hub height of the respective turbines for the power calculation in a physical model. This scaling is done with the use of the logarithmic profile or the power law. The logarithmic profile is used for heights up to 100m and is represented by Equation

2.12 (Aly, Khaled, and Gol-Zaroudi, 2020). whereas the power law is used for even higher purposes and is represented bu Equation 2.13 (Touma, 1977).

$$u(z_2) = u(z_1)\frac{ln((z_2 - d)/z_0)}{ln((z_1 - d)/z_0)} \tag{2.12}$$

Where $u(z_1)$ represents the mean wind speed in (m/s) at the desired height, $u(z_1)$ represents the mean wind speed in (m/s) at the initially measured height, d is the zero plane displacement in (m) and z_0 that represents the surface roughness length in (m).

$$u(z_2) = u(z_1)\left(\frac{z_2}{z_1}\right)^{\alpha} \tag{2.13}$$

Where α is the exponent that is a derived coefficient from the stability of the atmosphere, in neutral conditions, this would be around $\frac{1}{7}$ (Touma, 1977). When the wind speed is scaled properly to hub height, the power generation can be calculated with the use of Equation 2.2, by use of the manufacturer's power curve or by a self-constructed power curve.

Power curve
The power curve is the wind turbine or wind farm-specific function of wind speed and accompanied power output. A power curve can aid in wind power prediction without taking into account technical details such as the generating system components. An example of a power curve is shown in Figure 2.5. As indicated, the cut-in speed (u_c) is the minimum wind speed at which the wind turbine produces relevant power. The rated speed (u_r) is the wind speed at which the turbine produces the maximum power that is rated by the electrical generator. The wind speed at which the turbine produces its maximum power is called the cut-out speed (u_s). This maximum wind speed is usually constrained by the engineering design and safety measures (Lydia et al., 2014).
Power curves can be constructed in several ways: by the manufacturer, by estimation of the power characteristics of the rotor, gearbox ratio and efficiencies and by modelling with the use of historical data wind speed and power data. The latter can be subdivided into deterministic power curve models and probabilistic power curve models. For deterministic power curve models, artificial intelligence is used to approximate the power curve (Wang et al., 2019).
(Pelletier, Masson, and Tahan, 2016) proposed a method to model the power curve of a wind turbine with the use of an artificial neural network (ANN). The set of input features consisted among other things of historical power & meteorological data, operational data, vibration measurements and component temperatures. Other tested models were *k*-nearest neighbours (KNN), manufacturer's power curves and the 5th & 9th order polynomial. Although, the conclusion was made that more input features are required to further decrease the error in the power prediction. (Pelletier, Masson, and Tahan, 2016) showed that the proposed ANN model produced the lowest MAE of all tested models in the study with an MAE value of 15.3%. The KNN model produced an MAE value of 18.8% and the best-performing manufacturer's curve had an MAE value of 18.9%.

An important advantage of physical models is that one doesn't need much, historical, data for the calculation. On the contrary, as mentioned above, acquiring NWP data needs lengthy calculations and it can be economically challenging to make computations on a more frequent basis.

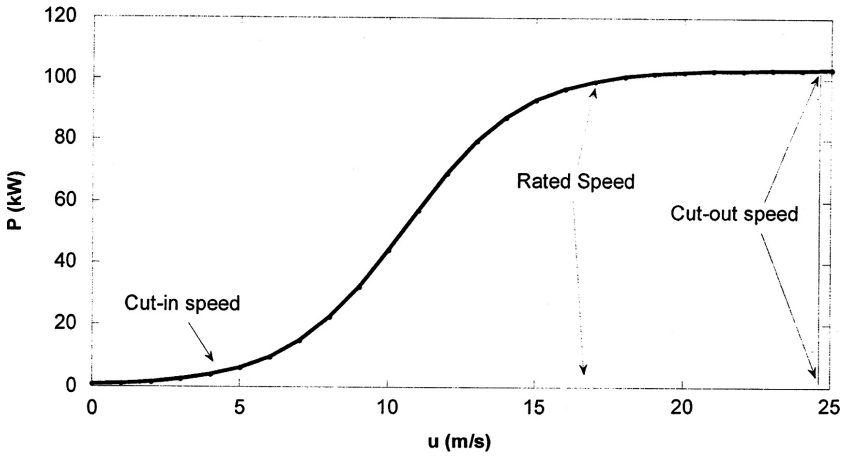


FIGURE 2.5: Example of a typical wind turbine power curve. The cut-in speed, rated speed and cut-out speed are indicated at the respective wind speeds (Lydia et al., 2014).

2.4.4 Statistical models

Statistical methods are based on large data sets of historical power output measurements neglecting relevant meteorological processes. The model determines the relationship between the weather variables and the historical power output to forecast the future power output. Since the impact of meteorological processes is ignored this statistical method can be described as a 'black box' approach (Foley et al., 2012).
Statistical methods can be sub-classified into time-series and learning or artificial intelligence methods.

Time-series

Time-series models are convenient to use for modelling purposes and they are computationally inexpensive. It is not based on any pre-determined mathematical model or physical equation and is rather based on patterns. The chance of resulting errors are decreased to a minimum if patterns are in line with historical ones (Abdelaziz et al., 2012). Auto-Regressive Moving Average (ARMA) models are the most commonly used type in the time-series-based approach to forecasting future values of power. Commonly used variations are Auto-Regressive Integrated Moving Average (ARIMA) and seasonal-ARIMA (SARIMA) (Soman et al., 2010).
SARIMA consist of:
Seasonal (S), that stands for the cyclical behaviour of the dataset. With this parameter, one can indicate what the pattern looks like. For example, if there is an annual seasonal dependence on temperature or irradiance.
Auto Regressive (AR), is the autoregression component which means how many instances in the past a current value is still dependent on, i.e. showing the data is regressed. This part refers to the parameter p, the lag order. This lag order can be extracted by the partial auto-correlation function (PACF), as mentioned in 2.3.4.
Integrated (I), this means the data is stationary. Stationary data is obtained by subtracting the current measured values from the previous values. This part refers to the parameter d, the number of subtractions for the data to be stationary.
Moving Average (MA), an indication that the forecast is linearly dependent on its past values. This part refers to the parameter q, the number of forecast errors or window size of the Moving Average. The moving average is an example of a sliding window and is illustrated

by the upper three graphs in Figure 2.12.

2.4.5 Deterministic Forecast

The aforementioned forecasting methods have a deterministic output or point forecast. This means a value for e.g. wind speed, power or multiple variables can be the output. The advantage of this method is that the accuracy is high but the uncertainty of the forecast value is unknown (Jung and Broadwater, 2014).

2.4.6 Probabilistic Forecast

Next to the evaluation of a point forecast, it is important to know the expected uncertainty of the forecast value. In this way, one can make a risk assessment of the result and make decisions on the risk profile. When looking at solar and wind power forecasting, the uncertainty of the atmospheric variables is important to take into account (Jung and Broadwater, 2014). The most commonly used probabilistic methods are statistical scenarios, physical-based ensembles and perturbation-based ensembles. The first two methods compute a probability density function from the deterministic NWP calculations. The last two methods use multiple NWP models, which in their turn are heavy in terms of computational capacity and are more expensive than statistical models (Jung and Broadwater, 2014).

2.5 Machine Learning techniques

This research adopts two machine learning techniques for prediction purposes. The first technique is decision-tree regression and the second is a neural network. This section provides background information on both techniques, describes the different prediction horizons and presents important evaluation metrics.

2.5.1 Decision trees

Decision tree algorithms are classification or regression methods for prediction and data mining purposes. A decision tree is a supervised learning algorithm and consists of a tree structure, that is built with a root node, branches, decision nodes and leaf nodes, see Figure 2.6. Classification trees are models where the supervised target value can take a discrete value. Regression trees are models where the supervised target value can take continuous values, such as power predictions. Decision trees have the advantage of their simple infrastructure and low computational costs (Wu et al., 2007).

Gradient boosting

Gradient boosting is a machine learning algorithm that is also used for classification and regression purposes. Gradient boosting is a predictive model which is an ensemble of multiple other models, which are usually decision trees (Chen and Guestrin, 2016). Since the models in the ensembles are most commonly weak prediction models, the gradient boosting computes a stronger output by iteration and with the use of a loss function. The principle of gradient boosting algorithms is to minimize this loss function or regularized objective. The loss function and the term that penalises the model complexity is Equation 2.14.

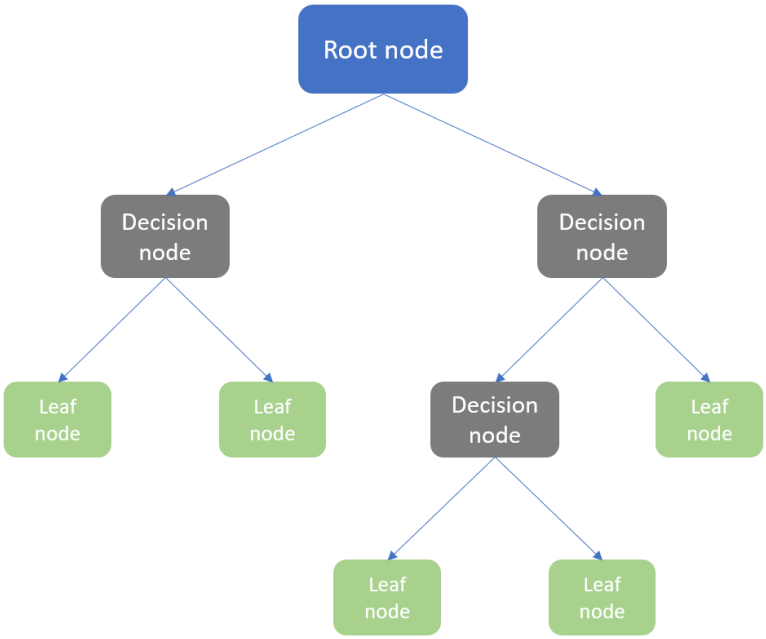


FIGURE 2.6: Schematic representation of a decision tree with a Root node, Decision nodes and Leaf nodes.

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k)$$
$$\text{where } \Omega(f) = \gamma T + \frac{1}{2}\lambda \parallel w \parallel^2 \tag{2.14}$$

Here, l is a differentiable convex loss function that measures the difference between the predicted $\hat{y}_i$ and the target value y_i. The Ω term penalizes the model complexity and includes a regularization term that smooths the learnt weights and avoids over-fitting (Chen and Guestrin, 2016).

Since not all parameters in Equation 2.14 can be optimized within Euclidian space, the equation includes functions instead. For the ensemble function to be optimized an extra term is added to Equation 2.14, resulting in Equation 2.15. This term f_t, for the i-th instance and t-th iteration is added in a greedy way which means that branches and leafs are added iteratively instead of enumerating all possible tree structures (Chen and Guestrin, 2016).

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)) + \Omega(f_t) \tag{2.15}$$

Extreme gradient boosting

Extreme gradient boosting (XGBoost) is a scalable machine learning algorithm for tree boosting developed and published by (Chen and Guestrin, 2016). XGBoost has been utilised in many machine learning competitions and is seen as one of the most used winning algorithms in the Kaggle community (Chen and Guestrin, 2016). XGBoost has several advantages over general Gradient Boosting designs.
XGBoost stands out because of its scalability in all scenarios. The computational speed is ten times faster than other comparable methods and it requires limited memory from the operating system. The scalability is mostly due to the handling of sparse data which is a

quantile sketch method that enables instance weights in tree learning. The combination of the computational speed, the low memory usage and the ability to handle fast amounts of data makes XGBoost an exciting technique for data scientists to process millions of examples on a simple desktop (Chen and Guestrin, 2016).
Another important advantage of using XGBoost is the ability to compute the Shapley additive explanation (SHAP) value of individual features. This means the contribution of the individual features to the predicted value is calculated. With this method, a selection of features can potentially improve model performance. The SHAP value doesn't contain information on the forecast quality.
At the beginning of this year, (Xiong et al., 2022) studied the performance of an XGBoost model compared to an SVM and an LSTM model in predicting wind power. They concluded that the XGBoost model outperformed the SVM and LSTM model with an NRMSE of about 3% and 2% respectively.

2.5.2 Artificial neural network

The neural network (NN) or artificial neural network (ANN) is a relatively new method of forecasting. It has the advantage of easily creating models and forecast complex systems with large amounts of input variables. On top of that, it has the advantage to find nonlinear relationships between features and targets. A multilayer perceptron (MLP) is the feed-forward ANN which refers to the fact that signals are propagated only onward to the neuron in the next layer (Abiodun et al., 2018).
The MLP usually consists out of 3 or more parts, the input layer where all features are fed into the model as inputs. One or more hidden layers which are neither an input nor output layer that contain activation functions to pass on information. Lastly, the MLP has an output layer that contains the desired amount of targets of the model. The input layer has the number of neurons equal to the number of features and the hidden layers can vary in the number of neurons and are an important part of the model optimization (Abiodun et al., 2018).

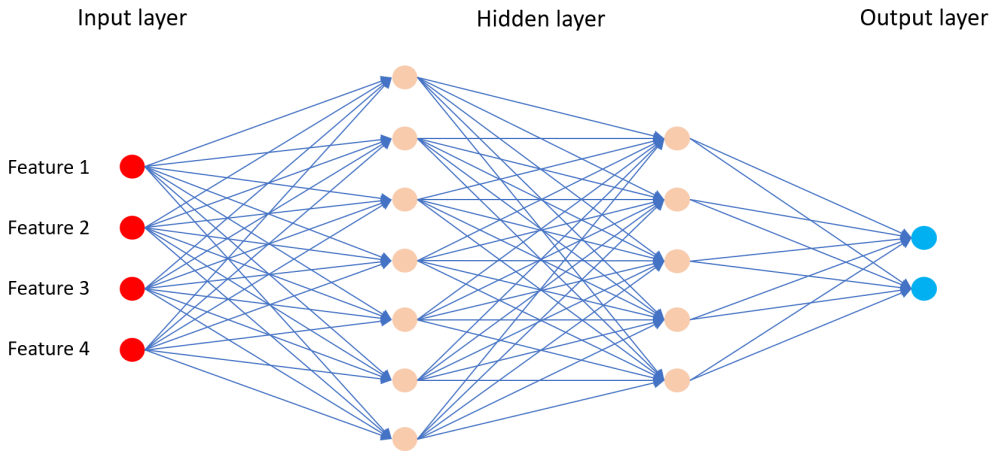


FIGURE 2.7: Schematic representation of a MLP infrastructure with two hidden layers.

Weights & Bias

In the MLP infrastructure in Figure 2.8 information flows from left to right. Information in the form of variable values is inbound via the inputs. If the inputs are x_1, x_2,..., x_n and the

weights are w_1, w_2,..., w_n then the weighted sum is computed with equation 2.16.

$$W_{avg} = \frac{1}{n}\sum(x_1w_1 + x_2w_2 + ... + x_nw_n) \tag{2.16}$$

A bias is then added to be able to shift the input of the activation function. This is useful since only multiplying with the weights has its limitations. The resulting input for the activation function is represented by equation 2.17.

$$z = \frac{1}{n}\sum(x_1w_1 + x_2w_2 + ... + x_nw_n + \text{bias}) \tag{2.17}$$

Activation Function

Activation functions are used to compute the output of neural network nodes or neurons. The various activation functions output a value in a specific range. The functions can be divided into two types: Linear activation functions and non-linear activation functions. The common used activation functions are given in Table 2.3. Here the partial derivative of the activation function $f(x)$ is given. This is used in Machine Learning to update the curve knowing in which direction the slope is changing. Updating the curve and re-iterating the weights is a part of backpropagation.

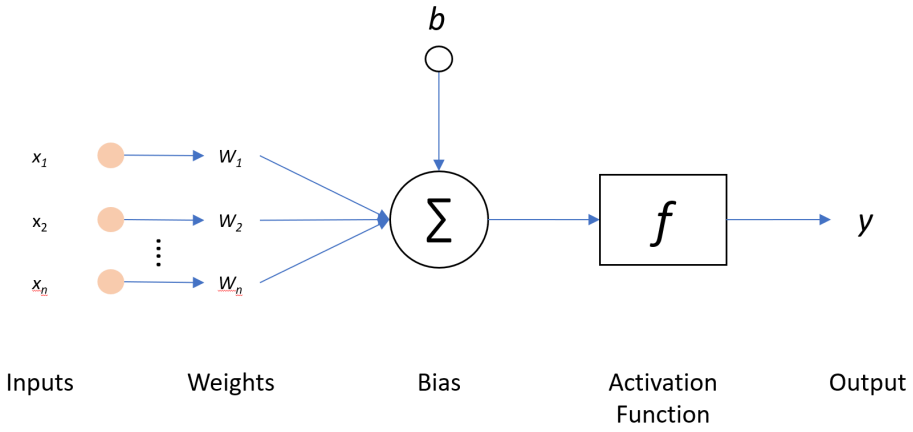


FIGURE 2.8: Schematic representation of the workings of a neuron inside the hidden layer.

Back propagation

Backpropagation is a commonly used algorithm for training a feed-forward neural network. The algorithm calculates the gradient of the respective loss function with respect to the applied weights. This gradient is computed with the use of the chain rule which explains the relevance of the partial derivatives in Table 2.3. When the gradient of the loss is computed, the weights are updated until the loss does not decrease any more. Backpropagation is an algorithm applied in supervised learning.

TABLE 2.3: Most common used neural network activation functions.

Name	Function, $f(z)$	Partial derivative , $f'(z)$	Range
Identity	$f(z) = z$	1	(-∞, ∞)
Binary step	$f(z) = 0$ if $z < 0$	$f'(z) = 0$ if $z \neq 0$	(0,1)
	$f(z) = 1$ if $z \geq 0$	$f'(z)$ = none if $z = 0$	
Sigmoid (logistic)	$f(z) = \frac{1}{1+e^{-z}}$	$f'(z) = f(z)(1 - f(z))$	(0,1)
Hyperbolic tangent	$f(z) = \tanh(z)$	$f'(z) = 1 - f(z)^2$	(-1,1)
Rectified Linear Unit (ReLu)	$f(z) = \max(0, z)$		[0,∞)

Hyperparameters

The algorithm of neural networks has multiple hyperparameters that tune the model for optimal forecast performance. Several important hyperparameters are:

- **Learning rate**: The learning rate is the hyperparameter that controls how quickly a model adapts to a problem. This parameter can be seen as the step size at every iteration whilst moving to a minimum of the proposed loss function. When the learning rate is too small, overfitting can occur. Larger learning rates can regularize the training but can also cause underfitting and diverges the training (Smith, 2018).
- **Batch size**: For NN models to be able to perform backpropagation a gradient error is computed to update weights and enhance performance. This gradient error is calculated for a specific number of samples repeated over the whole validation set in one epoch. This specific number of samples is called batch size. The batch size is usually set at a power 2. The main reason for this is the alignment with the physical processors of the local GPU. The default value is 32. The main advantages of larger batch sizes are the low computational time & cost and the smoothing of datasets. An advantage of lower batch sizes < 32 is the frequent update of the gradient error and thus the weights. This can specifically be useful for noisy features, the frequent update has a regularization effect that can potentially improve the model performance.
- **Dropout**: The dropout term is implemented as an extra layer. The dropout 'chooses' certain neurons at random and lets the model ignore their feedforward information. In this way, good and or bad weight values are deleted at random to create a stochastic element in the model. Dropout is seen as a useful hyperparameter with large datasets. For somewhat smaller datasets a high value of dropout can impact the performance dramatically. This term has to be treated with care since the randomness can result in misleading error performance. When using dropout the outputs should be predicted multiple times to compute an average.

2.5.3 Recurrent neural network

A recurrent neural network (RNN) is a type of NN that uses sequential data or time-series as input. The main difference between RNN and ANN is that RNNs have a 'memory'. RNNs take information from previous inputs and utilize them to influence current inputs and outputs. The outputs of RNNs are thus dependent on previous entries in the sequence

(Mikolov et al., 2010). An example of an RNN is long short-term memory (LSTM) which is explained in the following subsection.

Long short-term memory

Long short-term memory (LSTM) is a version of ANN but differs because of the temporal 'storage' of information from previous entries. An LSTM has the same characteristic as an ANN with respect to input layers, hidden layers and output layers. In addition to these layers, LSTM models have a memory block. An illustration of a memory block in a LSTM is shown in Figure 2.9. This block uses current input values in combination with the previous values that were stored in the memory for feature extraction. This information is stored in C_t and are fed to the next cell. The output data that is based on C_t, called h_t, is also transferred to the next cell.

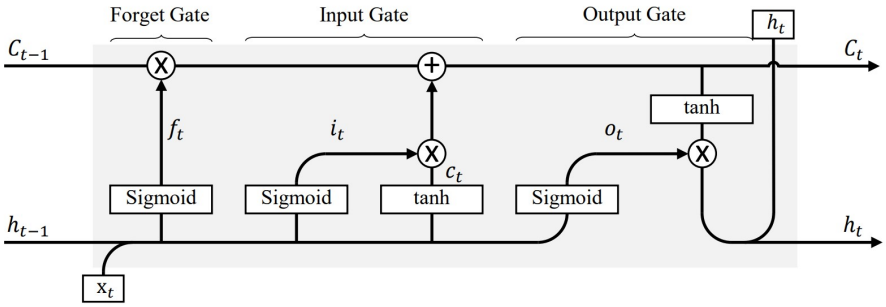


FIGURE 2.9: Schematic representation of one memory block in the LSTM gates architecture (Yu et al., 2017)

As shown in Figure 2.9, the memory block is comprised of three stages: the forget stage, the input stage and the output gate. The input and forget stage govern the flow of new input information, while output information is managed in the output gate. The computation of C_t is comprised of several formulas within the memory block through the three different stages.
As illustrated in Figure 2.9, the forget gate is fed with the current entries and the output of the previous block, x_t and h_{t-1} respectively. These are multiplied with the respective weight matrices W_f and U_f whereafter the bias, b_f, is added (Yu et al., 2017). The computation is explained with Equation 2.18. Here σ corresponds to the sigmoid function as presented in Table 2.3.

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{2.18}$$

Then, the input gate, i_t and the preliminary cell output i_t are computed with Equations 2.19 and 2.20 respectively. Here the weight matrix is represented by W_i and the respective bias is represented by b_i. The next step is the computation of the new cell state C_t with Equation 2.21. Here, the forget state is multiplied, via the Hadamard product, with the previous cell state and is added with the multiplication of the input gate and preliminary cell state (Chung et al., 2014).

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{2.19}$$

$$c_t = \tanh(W_C x_t + U_C h_{t-1} + b_C) \tag{2.20}$$

$$C_t = f_t * C_{t-1} + i_t * c_t \tag{2.21}$$

Finally, the output, h_t, is computed with Equations 2.22 and 2.23 and is transferred to the next cell together with the new cell state, C_t.

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{2.22}$$

$$h_t = o_t * \tanh(C_t) \tag{2.23}$$

2.5.4 Convolutional neural network

Another type of ANN is the convolutional neural network (CNN). CNNs are generally used for recognizing visual patterns from images and is used for object detection and image classification. Like an ANN, CNNs have an architecture of multiple layers. The main difference is that the input for ANNs is a vector, while for CNNs the input is generally a multichannel image. Examples of popular CNN models are ZFNet, VGGNet, GoogleNet and ResNet (Gu et al., 2018). A CNN consists of five main parts: a convolutional layer, a pooling layer and a fully connected layer.
Convolution is the mathematical operation where in the CNN the relationship between neighbouring pixels is preserved by construction of feature maps with neurons. The convoluted feature matrix is constructed with use of a filter matrix or a kernel. This filter moves over the input image matrix and applies element-wise multiplication, as shown in Figure 2.10 (Patterson and Gibson, 2017).
The pooling layer is usually placed in between different convolution layers. Pooling adds

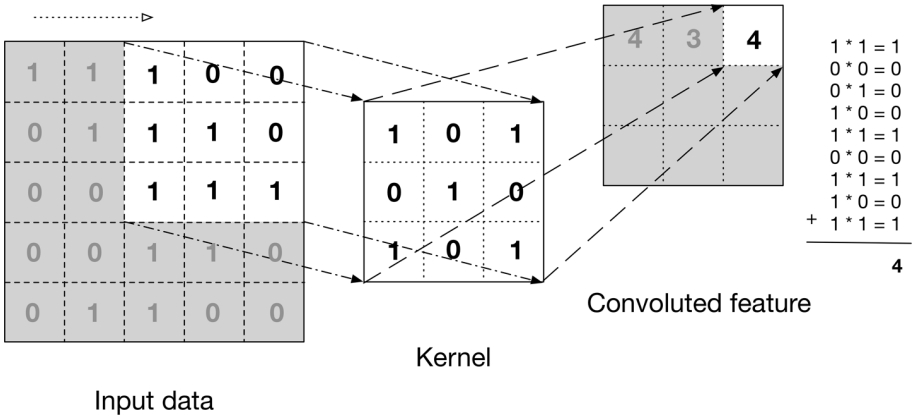


FIGURE 2.10: Illustration of convolution in a CNN, where the kernel moves over the input matrix to compute the convoluted feature matrix (Patterson and Gibson, 2017).

the flexibility of the model to identify patterns, even if the pixels are distorted, e.g. the image is tilted. General pooling layers are average pooling and max pooling. By stacking multiple convolution and pooling layers, the model is able to recognize higher-level feature patterns. Then, a CNN has one or multiple fully connected layers. This layer connects to the last pooling or convolution layer and connects by the same amount of neurons. From this layer, the input is transformed into a vector and the rest of the model represents a general ANN. The output of the ANN part can be optimized by minimizing the respective loss function.

At the beginning of 2022, (Wang, Song, and Cheng, 2022) performed a study where a CNN model was proposed for short-term wind power predictions. The input variables were based on a dataset of a real wind farm in China. The interval of the dataset was 1-minute and fluctuated between 0 and 21 MW. The proposed CNN model was compared with three other machine learning techniques: DeepAR, LSTM and RNN. (Wang, Song, and Cheng, 2022) concluded that the proposed CNN model outperformed all other models on every used evaluation metric with an RMSE of 0.086 compared to 0.43, 1.07 and 0.89 for the DeepAR, LSTM and RNN respectively.

2.5.5 Forecast horizon

The aforementioned models form the foundation for predicting future values. This prediction can be one-step ahead or multi-step ahead. One-step ahead predictions have a target where the training values are the $t + 1$ value of this target. Multi-step ahead prediction models have several methods for training a model. The most commonly used methods are discussed in this section (Taieb and Hyndman, 2012).

Direct multi-step

This method makes one prediction at a time and predicts a specific value, $t + n$, into the future. Target values in between are not used for the prediction which can potentially lead to a less accurate model. The main disadvantage of this method is the use of multiple models when predicting multiple timestamps. This increases the computational cost.

Recursive multi-step

Recursive methods predict a one-step ahead timestamp and use this value as a general input feature for the timestamp thereafter, as visualised in Figure 2.11. This process is repeated for more prediction steps into the future.

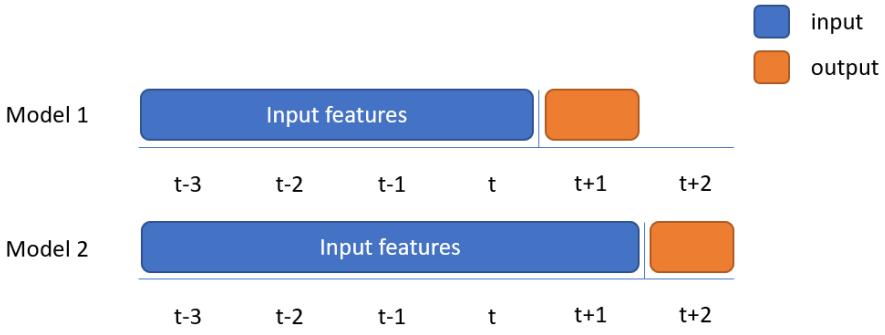


FIGURE 2.11: Visual representation of the recursive prediction method.

Direct recursive hybrid

This method is a combination of the above-mentioned techniques. For example, a second model can be constructed for timestamps to be predicted, in parallel each model is able to use predictions made by models for earlier timestamps as input features.

2.5.6 Evaluation

When studying the performance of predictive models, evaluation of the accuracy of the forecast values is most important. Any output without an elaborate evaluation metric can be classified as useless for further application. Evaluation of a forecasting model can be divided into two parts: the splitting of the data for training, validation and testing and the application of evaluation metrics on the forecast values.

Data separation

The data set can be separated into a training set, a validation set and a testing set. The objective of this method is to train the model on a subset of the total data and evaluate its performance on unseen data.
The training set generally has the most entries and has the purpose to let the model learn patterns in the dataset. In each iteration, the forecasting model is fed with the training data repeatedly until the backpropagation algorithm doesn't improve the loss function.
The validation set is different from the training dataset and has the objective to validate the model performance during training. Furthermore, the validation set is used to optimize the hyperparameters of the model and prevents the model from overfitting.
The testing set is the subset that is used after the model has been trained. With both the forecast value and the test set the evaluation metrics can be used to determine the performance of the trained model.

An important factor in splitting data is the randomness of the split. If the dataset has no time dimension then the order in which the data is split doesn't matter. When handling time-series data the order of splitting is very important. The data must be split taking the temporal order into account in which the values were measured. For time-series data there are two methods of cross-validating the model: a sliding window and forward chaining.
In a **Sliding window** the number of training data points and test data points remain constant per iteration step. With a sliding window, it is not necessary to cover all the data points at once. The train and test sets slide from the first measurement to the last one. A visual representation of both methods is given in Figure 2.12.
Forward chaining is the method where the train and test set expand towards the last data points per iteration step. In this way, no future measurements can be used for training and validating the model.

Metrics

After the separation of the data and training of the model, the performance can be evaluated with the different metric equations plugging in the forecast and test values. In this section, the most commonly used evaluation metrics are shown and discussed by their advantages and pitfalls. Equations 2.24, 2.25, 2.26 and 2.27 represent the mathematical expressions of the evaluation metrics. Where, N is the prediction horizon, p_f is the predicted power and p is the test value (Dhiman and Deb, 2020).

Mean Absolute Error (MAE) describes the mean absolute difference between the predicted values from the trained model and the test values and is represented by Equation 2.24. It is also called the sum of absolute errors. The resulting MAE value is of the same scale as the predicted and test data. The main disadvantage of this is that the comparison with other datasets is difficult because of the different scales.

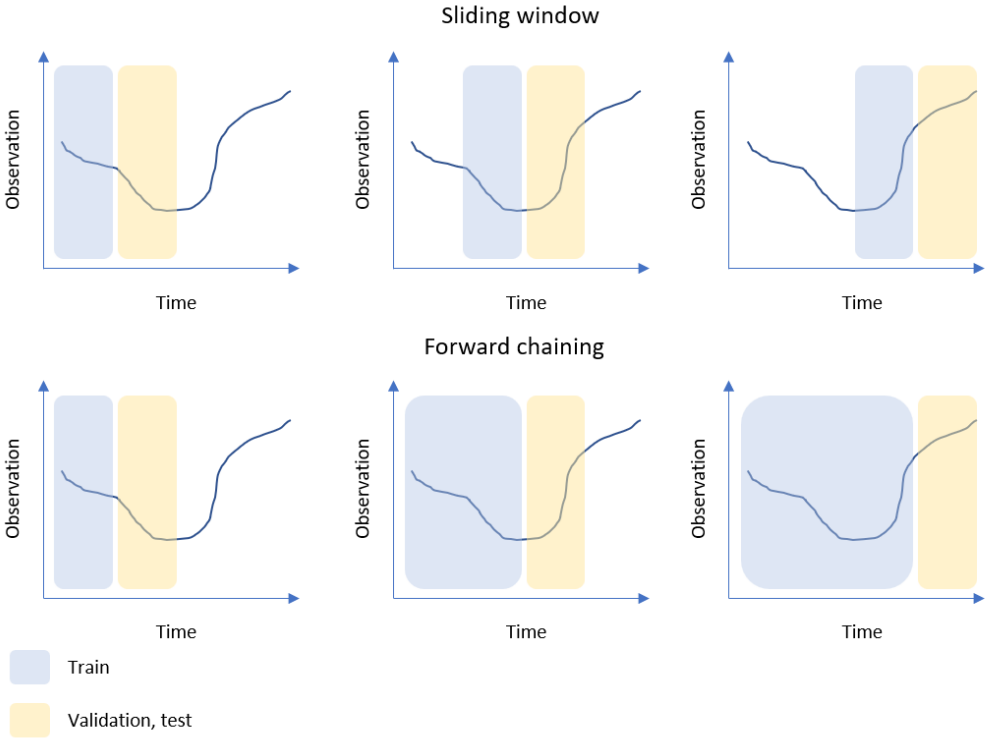


FIGURE 2.12: Schematic representation of two cross-validation methods: the sliding window and forward chaining.

$$MAE = \frac{1}{N}\sum_{t=1}^{N}|p_f - p| \tag{2.24}$$

Mean Absolute Percentage Error (MAPE) describes the mean absolute difference between the predicted values from the trained model and the test values divided by the test value. The factor of 100 indicates the resulting MAPE value will be a percentage which makes it easy to interpret and compare. The difficulty with Equation 2.25 is p_f can be zero or close to zero which causes the resulting MAPE value to blow up or a division by zero.

$$MAPE = \frac{100}{N}\sum_{t=1}^{N}\frac{|p_f - p|}{p_f} \tag{2.25}$$

Mean Squared Error (MSE) describes the average squared difference between the predicted values from the trained model and the test values. The MSE includes both the variance and the bias of the estimator. The MSE value is a measure of the quality of the predictions of a model. Since the difference between the predictions and real values is squared, as can be seen in Equation 2.26. The larger differences are penalized more heavily which makes the metric very sensitive.

$$MSE = \frac{1}{N}\sum_{t=1}^{N}(p_f - p)^2 \tag{2.26}$$

Root Mean Squared Error (RMSE) describes the square root of the MSE and is mathematically represented by Equation 2.27. The RMSE is less sensitive to larger differences between

the predicted value and the test value. The main advantage is that the original unit is preserved which makes it easier to interpret the resulting RMSE value.

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(p_f - p)^2} \tag{2.27}$$

The normalized root mean square error (NRMSE) is calculated with the use of the nominal capacity of the respective generation unit. The NRMSE is then given in percentage and facilitates easy comparability between the performance of different generation units and proposed models.

$$NRMSE = 100\% * \frac{RMSE}{P_{nom}} \tag{2.28}$$

2.6 State of the Art summary

TABLE 2.4: Summary of the state of the art renewable energy forecast models.

Renewable source	Model	Horizon	Benchmark	Improvement
Wind (Milligan, Schwartz, and Wan, 2003)	ARMA	6 h	Persistence	7% (NRMSE, 1 h), 18% (NRMSE, 6 h)
Wind (Cadenas and Rivera, 2009)	NN	6 h	NN	0.0016 (MSE, 6 h)
Wind (Dong et al., 2016)	NN, NWP	1 day	Persistence	12% (NRMSE, 1 day)
Solar (Wang, Song, and Cheng, 2022)	CNN	3 h	DeepAR, LSTM, RNN	0.086, 0.43, 0.89 (RMSE, 3 h)
Solar (Chaouachi et al., 2009)	NN	1 day	NN, MLP ensembles	1% (MAPE, 1 day)
Solar (Kemmoku et al., 1999)	NN	1 day	Persistence	6% (MAE, 1 day)
Solar (Xiong et al., 2022)	XGBoost	1 day	SVM, LSTM	3%, 2% (NRMSE, 1 day)
Hybrid (solar, wind) (Qadir et al., 2021)	ExtraTrees, linear regression, MLP	4 h		0.0000493 (MSE, 4 h) 0.000001 (MSE, 4 h) 0.0000286 (MSE, 4 h)
Hybrid (solar, wind, CSP) (Pang et al., 2021)	LSTM	1 h	Individual models	0.75% (NRMSE, 1 h)

Chapter 3

Methodology

Section 3.1 describes the data analysis process. This includes the acquisition, the exploration & visualisation and the transformation of the relevant data. Then, Section 3.2.1 gives insight into the feature importance methodology and the eventual choice of input features. Section 3.3 describes the infrastructures for both the utilised models based on hyper-parameter optimisation. Finally, Section 3.4 provides the evaluation process, including valuation of the forecast methods.

3.1 Data Analysis

This section first describes the origin and acquisition of the used data. Secondly, the data is visualised and provided in figures to see the dataset's average monthly and hourly data points. Lastly, the process of data transformation and feature engineering is presented. Decisions made in this section were necessary for the overall model performance and were based on examples described in the literature review.

3.1.1 Data origination

For this research, three sources were used to acquire the required data. This included the power and meteorological data of the wind farm Koergorspolder (KGP) in Terneuzen, the power and meteorological data of the web source PVGIS and the numerical weather data of the institute of ECMWF.

Windfarm Koegorspolder

The wind farm used in this research is wind farm Koegorspolder located in Terneuzen, Zeeland, as presented in Figure A.1. The wind farm has an installed capacity of 44 Megawatt (MW) and has 22 wind turbines of 2MW rated capacity each. The turbines are Vestas V80/2000 and have a diameter of 80m. The wind farm was commissioned in 2007. Each turbine has it's own meteorological measurement system that measures: wind speed ($\mathrm{m\,s^{-1}}$), wind direction (°), temperature (°C) and power (MW) on a 10-minute interval. System availability is not measured directly but is indicated with all weather measurements returning NaN values, which will be treated in 3.1.3 as an engineered feature.

PVGIS

For the solar data, there was a missing source with real-life solar panel data. PVGIS was useful for replacing this missing source of measured solar panel data. "PVGIS uses high-quality and high-spatial and temporal resolution data of solar radiation obtained from satellite images, as well as ambient temperature and wind speed from climate reanalysis models." The calculation of the total solar irradiance consists of three parts, as mentioned in the 2.1: the direct irradiance, the reflected irradiance and diffuse irradiance. The different solar irradiance components are calculated using the HELIOSAT method and the SPECMAGIC algorithm (Gracia Amillo, Huld, and Müller, 2014). The calculation of the surface irradiance is performed in two steps. First, the effective cloud albedo is observed using the METEOSAT satellite (Gracia Amillo, Huld, and Müller, 2014). With use of equations 3.1 and 3.2

$$CAL = \frac{\rho - \rho_{cs}}{\rho_{max} - \rho_{cs}} \tag{3.1}$$

$$\rho = \frac{D - D_0}{\cos(\theta) f(\text{distance})} \tag{3.2}$$

In equation 3.2 D is the pixel count of the METEOSAT satellite, D_0 is the dark offset op the measurement, $cos(\theta)$ is the zenith angle and $f(distance)$ is the variation in Sun-Earth distance (Gracia Amillo, Huld, and Müller, 2014). In equation 3.1 ρ_{max} is the calculated reflection per pixel, ρ is the maximum reflection and ρ_{cs} is the clear sky reflection.

The resulting effective albedo is then used to calculate the global irradiance using equation 3.3.

$$G_{global} = (1 - CAL) G_{clear} \tag{3.3}$$

Here, G_{clear} is the clear sky irradiance for cloudless moments. G_{clear} is calculated using the SPECMAGIC model for clear sky irradiance (Müller et al., 2012). The quality of the observed G_{clear} is dependent on the available information on aerosols and water vapour, which are the leading variables for clear sky irradiance (Gracia Amillo, Huld, and Müller, 2014).
The information on water vapour originates from the ECMWF Numerical Weather Prediction model (NWP). The information on aerosols originates from the Monitoring Atmospheric Composition and Climate (MACC). This project was used to calculate the monthly mean of aerosol optical depth. Since the period of the MACC is longer than the satellite data, the average values are used instead of the mean. Validation of the calculated irradiance was performed by calculating the mean bias difference (MBD, W/m^2) and the root-mean-square difference (RMSD, W/m^2) for a selection of ground stations from the Baseline Surface Radiation Network (BSRN) (Gracia Amillo, Huld, and Müller, 2014).

The next step is to estimate the real power output of the PV modules with use of the earlier calculated global irradiance. The actual power output depends on several factors: the angle of incidence, the effect of different solar spectra, the dependence on irradiance, the module temperature, system losses and degradation and effects not taken into account in PVGIS.
The angle of incidence on PV modules is essential because of the resulting reflection of light. This reflection coefficient will increase when the light comes in at a sharp angle. All light is reflected when the incident light is parallel to the PV module. This effect is also known as the angular loss and is responsible for an average loss of 2-4% for a fixed-mount PV module (Martin and Ruiz, 2001).
Different PV module technologies have various sensitivities to incoming light with specific wavelengths. Amorphous-Silicon (a-Si), for example, operates at lower wavelengths,

TABLE 3.1: Validation results of the calculated global horizontal irradiance in comparison with observations from ground stations (Müller et al., 2012) (Gracia Amillo, Huld, and Müller, 2014).

Station	**MBD** (W/m^2)	**RMSD** (W/m^2)
Cabauw (NL)	1.3	86.9
Camborne (UK)	-1.6	74.3
Carpentras (FR)	6.7	56.5
Linderberg (DE)	2.2	70.0
Toraverre (ES)	-1.7	58.6

whereas crystalline-Silicon (c-Si) operates at higher wavelengths. The spectrum of wavelengths varies over the time of day and with meteorological conditions. For example, sunlight is red during sunrise and sunset. The effect of spectral changes is determined using the radiation data from satellites and calculations for the different spectral bands. The loss by spectral variations on the PV module power output for c-Si near the Netherlands is around 2% (Müller et al., 2012).
The efficiency of PV modules depends highly on the intensity of solar irradiance, as seen in Figure 3.1. For all three technologies, the efficiency is close to constant for irradiance values higher than 400 W/m^2. For values lower than 400 W/m^2, the efficiency of the module decreases rapidly. PVGIS takes both the irradiance and temperature into account for the resulting power output with the use of equations 3.4 and 3.5 (Huld et al., 2011):

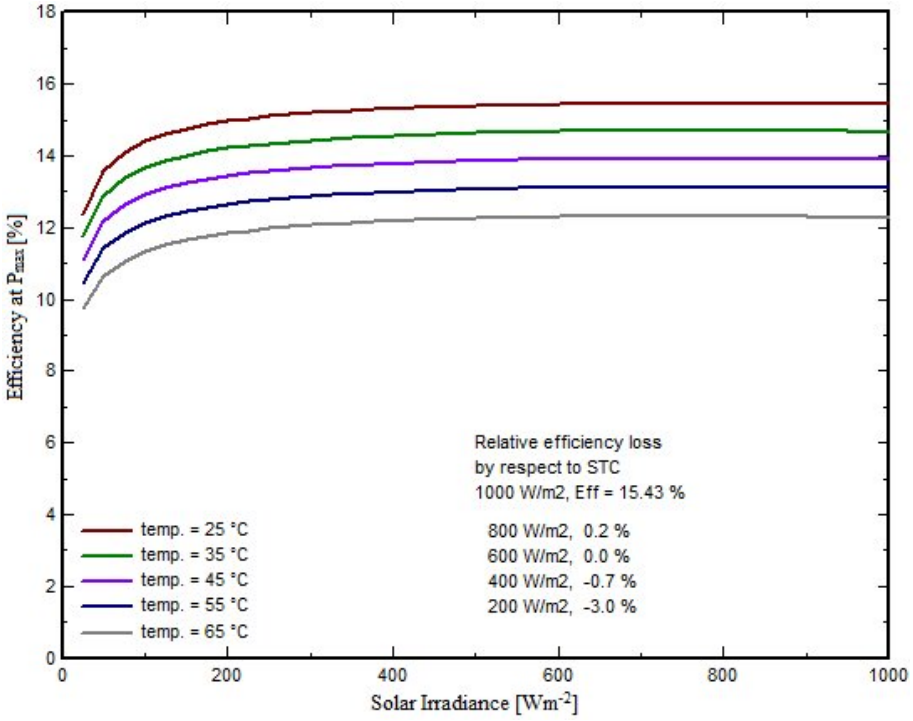


FIGURE 3.1: PV module efficiency curve at different temperatures (° C) and different irradiances (W/m^2) (Razak et al., 2016).

$$P = \frac{G}{1000} * A * \text{eff} = \frac{G}{1000} * A * \text{eff}_{\text{nom}} * \text{eff}_{\text{rel}} \tag{3.4}$$

$$\text{eff}_{\text{rel}} = 1 + k_1 \ln(G') + k_2 \ln(G')^2 + k_3 T'_m + k_4 T'_m \ln(G') + k_5 T'_m \ln(G')^2 + k_6 {T'_m}^2 \tag{3.5}$$

Where:

$$G' = G/G_{STC}$$
$$T'_m = T_m - T_{STC}$$

Here, G is the irradiance (W/m^2) incident on the PV module, eff is the efficiency (%), A is the surface area of the PV module (m^2) and k_1 to k_6 are coefficients that are used by PVGIS and are based on the measurements at the European Solar Test Installation (ESTI), see Table 3.2.

TABLE 3.2: Coefficients used in PVGIS, measured at ESTI for c-Si (Huld et al., 2011)

Coefficient	**c-Si**
k_1	-0.017237
k_2	-0.040465
k_3	-0.004702
k_4	0.000149
k_5	0.000170
k_6	0.000005

The temperature of PV modules is not only dependent on air temperature but is also prone to heating by incoming sunlight and to local wind speeds. Increasing module temperature decreases the overall efficiency, as can be seen in Figure 3.1. PVGIS takes the heating and cooling of the PV module into account by using equation 3.6 (Fairman, 2008):

$$T_m = T_a + \frac{G}{U_0 + U_1 W} \tag{3.6}$$

Here, W is the wind speed (m/s), T_a is the local air temperature (° C), U_0 and U_1 are coefficients used in PVGIS coming from (Koehl et al., 2011).

Numerical Weather Data

Numerical weather predictions (NWP) is a method of predicting future weather based on current conditions by means of mathematical models. Inputs for these models range from weather stations to aeroplanes and weather satellites as shown in Figure 3.2. Various countries employ such models and use them for a day, week or even month ahead predictions. These models are run a few times a day due to the computational intensity of the mathematical equations.
The European Centre for Medium-Range Weather Forecasts (ECMWF) is an institute that

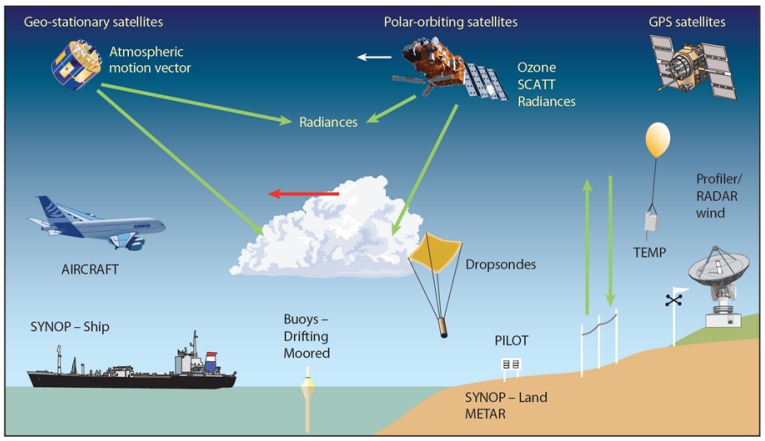


FIGURE 3.2: Representation of different measurement systems, supplying the NWP models with meteorological input data for the mathematical equations (*The ECMWF Ensemble Prediction System* 2012)

provides these models, access to the forecasts and historical data sets. These forecasts who are generated twice a day are used in this research for the 6-hour horizon predictions. Section 3.2.1 provides a detailed description of the variables that were used in this research. Table B.1 provides insight into the usage of NWP for the different models and forecast horizons.

3.1.2 Data Exploration

The power production data of KGP is proportional to the installed capacity of the wind farm. The PV production data on the other hand is initially given for a 1kW system. For this research, a combination of the wind and PV data will be used in a combined model. In this research, the installed capacity is scaled with respect to the annual energy production. This resulted in an installed PV capacity of 63 MW.
Both PV and Wind production data are shown in Figures 3.3 and 3.4 respectively. Both the power units are given in kW and were averaged over each month to observe the annual seasonality of the production outputs. This annual seasonality of power production will be described in Section 3.1.3 as an engineered feature that is used to enhance model performance.
Figure 3.5 represents the average hourly production of the PV and wind production units. Both production data follow the expected respective trends. PV production sees a cyclic pattern between hours 5 and 19, whereas the average wind production profile is more constant over the day and sees a slight increase in the afternoon.

3.1.3 Pre-processing

After the data was acquired and explored, the dataset had to be pre-processed to fill missing values, engineer new features and transform the data to ensure standardization and normalization. This section presents the main choices made to optimize the input features.

Data transformation

Since one of the inputs of the PVGIS tool is the size of the system in kW, the decision was made to scale the PV array to the size where both the total annual energy outputs would be equal. Furthermore, NaN values were handled in this part of the research. As mentioned in

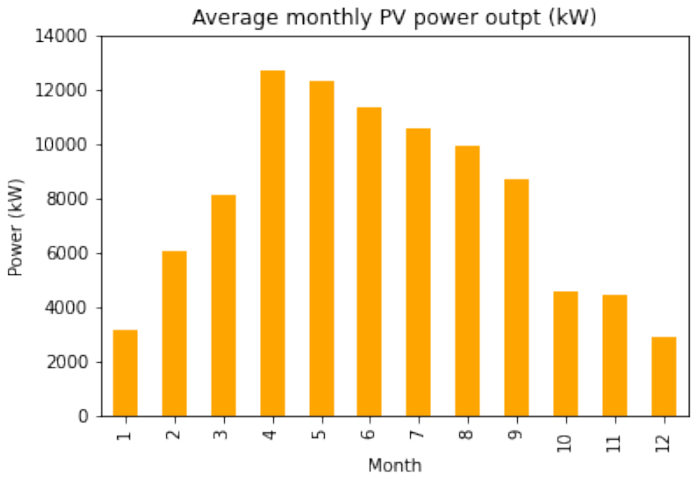


FIGURE 3.3: Average monthly solar power output (kW) over 2019

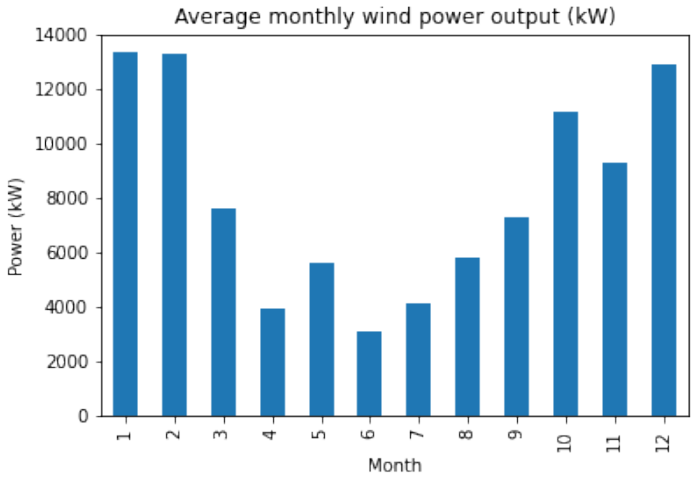


FIGURE 3.4: Average monthly wind power output (kW) over 2019

Section 2.3.2 there are several methods of handling missing data. Missing data in the time-series were handled with care since time-series modelling relies heavily on continuous time variables. In such a case, the missing value was inserted according to the continuity of the series.
Missing values in the meteorological variables were filled by means of forwarding filling because of the high correlation between lag values of meteorological variables, e.g. Figure 2.3.

Feature engineering

During the data preparation for this research 6 new features were constructed. The goal of these new features was to prevent higher values for degrees, days of the year or hours of the day from having a higher impact on the forecast. This is achieved by splitting such variables into an x and a y part by using sine and cosine. In this way, the dependency on higher values will normalize. The engineered features are 2 vectors that are a combination of the wind speed magnitude and wind direction 3.7 and 3.8. There are 4 time-based features

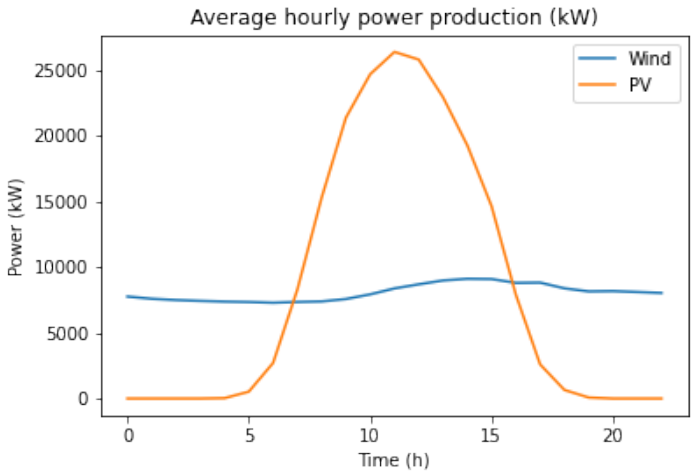


FIGURE 3.5: Average hourly power output (kW) over 2019 and 2020

that should include seasonality over the year, 3.9 and 3.10, and should include the daily cycle of the sun by hourly projections 3.11 and 3.12.

$$v_x = v \cdot \cos(\theta) \tag{3.7}$$

$$v_y = v \cdot \sin(\theta) \tag{3.8}$$

Where, v represents the current wind speed in (m/s) and θ represents the wind direction in degrees $(^\circ)$.

$$T_{\text{yearday, cos}} = \cos\left(\frac{2\pi \cdot t_{\text{day}}}{n_{\text{total, days}}}\right) \tag{3.9}$$

$$T_{\text{yearday, sin}} = \sin\left(\frac{2\pi \cdot t_{\text{day}}}{n_{\text{total, days}}}\right) \tag{3.10}$$

Where, t_{day} represents the current day of the year ranging from 1 to 365 and $n_{total,days}$ represents the total number of days in the year.

$$T_{\text{hourday, cos}} = \cos\left(\frac{2\pi \cdot t_{\text{hour}}}{n_{\text{total, hours}}}\right) \tag{3.11}$$

$$T_{\text{hourday, sin}} = \sin\left(\frac{2\pi \cdot t_{\text{hour}}}{n_{\text{total, hours}}}\right) \tag{3.12}$$

Where, t_{hour} represents the current hour of the year ranging from 1 to 24 and $n_{total,hours}$ represents the total number of hours in a day. A visual representation of the hourly x and y components is shown in Figure 3.6.
As mentioned in Section 3.1.1 the wind turbines have individual measurement systems. These measurement systems are programmed to return NaN values for all measured variables when the turbines aren't operational. This unavailability is not registered as a separate variable so this had to be constructed as a new feature. The unavailability is not a variable with information on future operations.

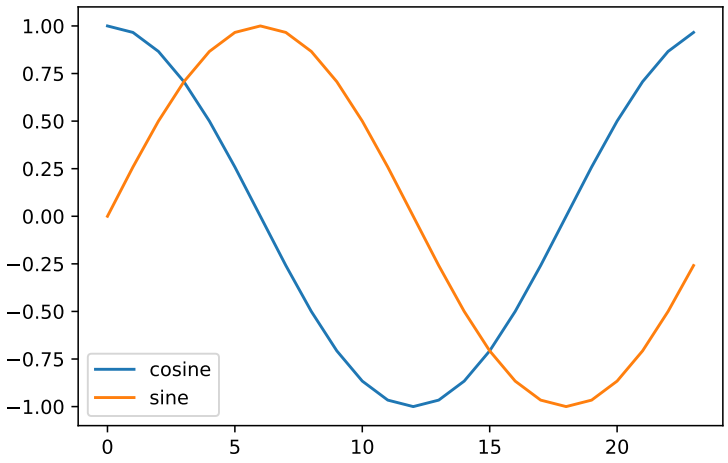


FIGURE 3.6: Plot of the hourly x- and y-component representing Equations 3.11 and 3.12 respectively.

Feature transformation

After scaling the acquired data, handling missing values and constructing new features the data was standardized to even out the dependence on the predictions and to prevent lookahead. The MinMaxScaler from the Scikitlearn package was used to standardize the features before training the models.

3.2 Tree-based regression

This section provides the selection of the most important features to be used in the two proposed models for prediction purposes. This selection of features is based on the PACF of certain features and the feature importance method of computing SHAP values with a tree-bas regression model XGBoost. After the selection of the most relevant features, the XGBoost model was used for prediction purposes. Finally, the input infrastructure, hyperparameters and evaluation of the predictions are presented.

3.2.1 Feature importance

The XGBoost model was used for both the feature importance method for prediction purposes. This feature selection method was used for both the PV and wind models to enhance the model performances and was used for the hybrid model to gain insight into the complementarity of both the power production and meteorological variables. All proposed models were trained and optimized to outperform the benchmark methods. The hyperparameters are given in Table 3.3.
When the models were trained and optimized the SHAP values were computed with the use of the SHAP python package. With this information, a selection of features was made to be used as inputs for the PV and wind models.

3.2.2 XGBoost as prediction model

The main reason a tree regressor was used as a first model type is that it includes the computation of SHAP values that enhances model performance by feature selection and because

TABLE 3.3: A selection of hyperparameter settings for the three proposed XGBoost feature importance models.

Hyperparameter	XGB models
Learning_rate	0.3
min_split_loss	0
max_depth	6
Subsample	0.5
n_estimators	1000

of the relatively simple infrastructure which makes it more convenient to implement any adjustments to the model. This section presents the chosen input variables with respect to their lag values, future information and the targets of the models.

With the use of the SHAP and PACF values for the different features a selection was made for the models. Since for smart persistence, the sun height is used to scale the irradiance prediction this was also used as an input for the XGBoost PV prediction model. The sun height variable contains future information just as the time variables so, the models are fed with the $t+1, t+2, \ldots, t+6$ values of these features.
As can be seen from Figure 3.7, not all features contain the lag $t-3$ value. The reason for this is the low accompanying SHAP value and the lack of correlation when computing the PACF. An example of a feature with low importance for higher lag values is the Availability feature, as presented in Figure B.1. For the hybrid model, all features of the PV and wind model were combined and are presented in Figure 3.8.
The target of all the models is the Power production and throughout the research this feature is predicted for $t+1, t+2, \ldots, t+6$. Predictions beyond the 1-hour horizon were made recursively, as explained in Section 2.5.5. Furthermore, the prediction set for all models had a maximum and minimum value programmed so predictions couldn't be larger than the physical limit, i.e. the installed capacity. The hyperparameters for the models are presented in Table 3.4.

TABLE 3.4: A selection of hyperparameter settings for the three proposed XGBoost models.

Hyperparameter	PV	Wind	Hybrid
Learning_rate	0.3	0.3	0.3
min_split_loss	0	0	0
max_depth	6	6	6
Subsample	0.5	0.5	0.5

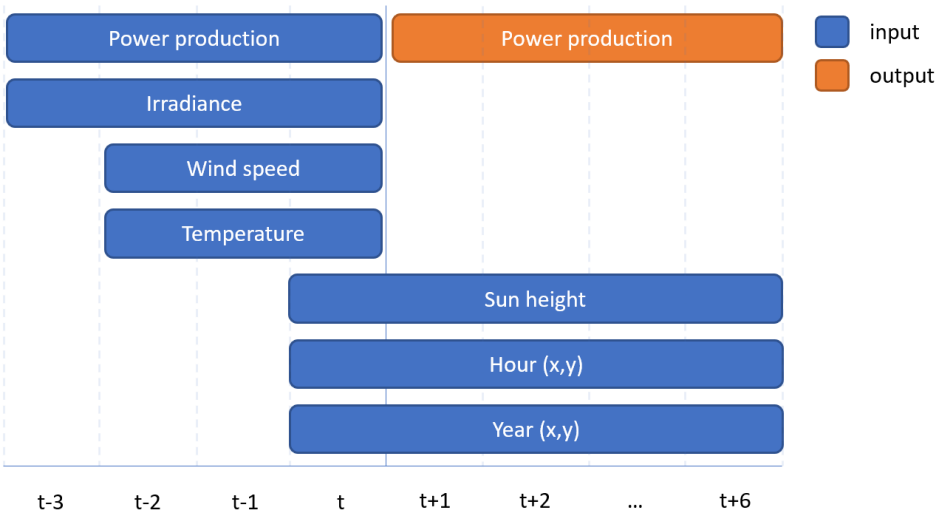


FIGURE 3.7: Visual representation of the input features and output variables for the solar models. This figure represents the 1-hour till 6-hour horizon.

3.3 Neural Network

This research proposes two machine learning models to predict future power production. This section presents the design of the neural network and describes the optimisation process of the hyperparameters.
As discussed in Section 2.5.2 a neural network is built out of a combination of neurons, weights, layers and activation functions. It has the advantage of creating simple models to predict complex datasets and large amounts of input variables. A neural network has low computational costs due to this simplicity and efficient infrastructure and is able to outperform more complex models, e.g. long-short-term memory (LSTM) models. A visual representation of a simple neural network is shown in Figure 2.7.
Three different NN models were constructed in this research for studying the performance of the proposed hybrid model infrastructure. Each model has been optimized individually for its respective hyperparameters and neuron layout. The basic layout of the three models is presented in Table 3.5. Each model has been tuned individually to maximize its performance.
The input features were chosen according to the feature importance and PACF results as de-

TABLE 3.5: Layout of the three proposed NN models.

Layer	Type	Activation	Parameter	Optimization
Input	Dense	-	# of features	-
Hidden 1	Dense	Relu	0-200	steps of 5
-	Dropout	-	0-0.7	steps of 0.01
Hidden 2	Dense	Relu	0-100	steps of 5
Output	Dense	Linear	1	

scribed in Section 3.2.1, a visual representation of the hybrid feature infrastructure is shown in Figure 3.8. The input feature infrastructures for the NN models are identical to the XGBoost input features. The main reason for this is the ability to compare the performance of the models. Each model was optimized for the 1-hour ahead forecast horizon. For calculation of the 6-hour ahead forecast the recursive multi-step technique was used to compute each hour in between, a visual representation is shown in Figure 2.11.

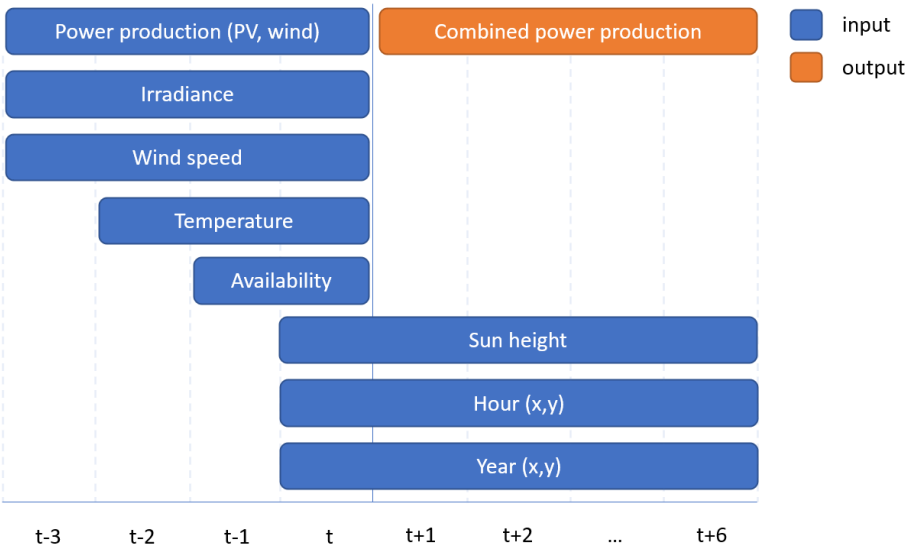


FIGURE 3.8: Visual representation of the input features and output variables for the hybrid models. This figure represents the 1-hour till 6-hour horizon.

As mentioned, each model was trained and optimized individually. A neural network has multiple hyperparameters to be tuned which can cause lengthy optimization processes. Important parameters that were included in the optimization algorithm are the number of neurons in hidden layers 1 & 2, dropout factor, the optimiser (Adam or Stochastic Gradient Descent), learning rate and batch size. The number of neurons increased with steps of 5, the dropout factor with steps of 0.01, the learning rate with steps of factor 10 or logarithmic steps and the batch size increased with steps of factor 2 starting from 1. The optimisers were chosen from literature where Stochastic gradient Descent had an additional decay parameter that varied with a logarithmic step. The optimisers propagate information about the gradients back to the hidden layers where weights and biases are updated to enhance performance. The mean squared error (MSE) was used as the loss function for this process.
All parameters were mutated at random with 100 iterations to find the most optimal set of parameters. The number of epochs was maximized at 500 with the implementation of an early stopping callback with a patience of 5. This callback interrupted training when the validation loss did not improve over five epochs.

3.4 Evaluation

In this research, the main method for measuring the accuracy of the models was the calculation of the root-means-squared error (RMSE). As explained in Section 2.5.6 the RMSE is widely used because of the preservation of the unit. This facilitates for comparing with

other predictions or with other comparable research. Furthermore, the RMSE has a low sensitivity for large differences between the predicted value and the test value.
During the training of the NN, the mean-squared-error (MSE) was used to compute the validation loss and eventually to update the weights and biases via backpropagation.

3.4.1 Valuation

Since one of the research objectives is to create insight into the financial benefit of implementing a hybrid model with respect to the individual models, a valuation method will be introduced in this subsection. During the research, a 6-hour ahead forecast was made recursively for both the XGB and the NN machine learning technique. Since only a 6-hour forecast was implemented the intraday and imbalance markets were used in the valuation assessment. Some important rules for the method are:

- The valuation method is not based on making choices between markets or curtailment strategies.
- Imbalance prices do not change upon bidding of the hybrid system.
- Only hourly bids are placed.
- The 'trader' will have a passive strategy, meaning all differences between forecast and delivery are bought or sold at the imbalance market.

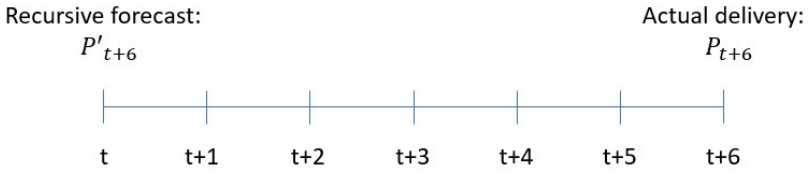


FIGURE 3.9: Illustration of the trading timeline.

To compute an hourly positive or negative impact on the revenue first, the difference between prediction, at t, and actual delivery, at $t+6$, was calculated. As visualized in Figure 3.9, the power forecast is indicated with P'_{t+6} and the actual delivery is indicated by P_{t+6}. For trading in the imbalance market, the regulation state of this market was taken into account. Since the regulation state is a measure in what state the energy throughput of the imbalance market is. For signs 1, -1, 0 and 2 the value of the calculated imbalance delta, as shown in Equation 3.13, is multiplied by either the marginal price of the upward bid, with the marginal price of the downward bid, with both respective marginal prices or with a positive downward regulation and a negative upward regulation respectively (Veen, Abbasy, and Hakvoort, 2010).

$$\Delta P = P'_{t+6} - P_{t+6} \tag{3.13}$$

This calculation was performed for the following models: Persistence, XGB hybrid, XGB summed, NN hybrid and NN summed. The final value creation of all models was set off by the value creation of Persistence.

3.5 Software and hardware

3.5.1 Software

The code for the models and figures was written in the Python 3 language. With the use of PyCharm 2021.3 academic version, the JupiterLab Integrated Development Environment (IDE) was utilised. The following packages were used for modelling purposes:

- keras 2.8.0
- scikit-learn 1.0.2
- tensorflow 2.8.0
- xgboost 1.6.1

3.5.2 Hardware

This research was performed on a desktop Lenovo Thinkpad X380 Yoga. Some key specifications are:
Processor: Intel(R) Core(TM) 8th Gen i7-8550U CPU @ 1.80GHz 1.99GHz.
Installed RAM: 16.0 GB (15.9 GB usable).
System type: 64-bit operating system, x64-based processor.
Windows specifications: Windows 11 Pro, version 21H2.
Progress was saved on a local drive and on the cloud-service of 'Box'.

Chapter 4

Results & Discussion

This section presents the results from this research and discusses its applicability. First, the results of the feature importance method are shown for the individual models in Section 4.1. Then, the optimal hyperparameters are given for the neural network with their respective validation losses in Section 4.2. Subsequently, the prediction results are visualized and evaluated on performance in Section 4.3. Lastly, the results of the valuation study are presented in Section 4.4.

4.1 Feature Importance

An XGBoost tree regression model was used to compute the mean SHAP values to determine the impact input features would have on the forecast in this research. Higher SHAP values mean higher importance on the predicted value. The SHAP values are both calculated for the PV and wind input variables as shown in Figures 4.1 and 4.2.
As can be seen from Figure 4.1, the current power (*Power*) has the highest dependency on

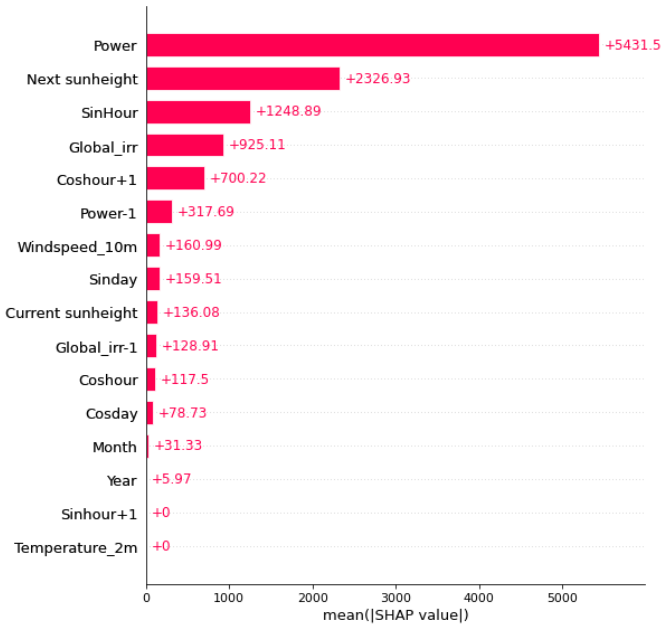


FIGURE 4.1: Mean SHAP values of the PV input variables calculated using the tree regressor XGBoost.

the 1-step ahead power forecast. Furthermore, the hour + 1 (coshour + 1) and the current global irradiance ($Global_{irr}$) have high dependencies on the 1-step ahead forecast. The variables Year and Month were left out for the research for the lower dependencies since their dependency is close to or already zero. These variables were replaced with the engineered features as discussed in Section 3.1.3.

FIGURE 4.2: Mean SHAP values of the wind input variables calculated using the tree regressor XGBoost.

As can be seen from Figure 4.2, the current power (*Power_kgp*) has the highest dependency on the forecast value. Other important variables are current wind speed and temperature. Lagged features are indicated with the variable name, a minus sign, and the respective lag value. The high current power and wind speed dependency were as expected and aligned with the literature. The high dependency on temperature could be linked to the density of air having an impact on the turbine power curve. Variables like wind speed, wind direction vector, and availability seem to perform very poorly. The variable Year was left out for further research since the mean SHAP value was zero. Vectorx and windx represent the engineered features where wind speed is combined with wind direction and where wind direction is split into an x- and a y-component, respectively. The latter was excluded for further research.

4.2 Hyperparameter optimisation

As mentioned in Section 2.5.2, the basic infrastructures of the neural network (NN) models were determined before optimising the hyperparameters. All models had an input layer, two hidden layers, a dropout factor and an output layer, as presented in Table 3.5. The optimisation process performed 100 iterations, and the results are shown in Table 4.1.

TABLE 4.1: Results for the neural network hyperparameter optimisation.

Model	Layer 1	Layer 2	Learning rate	dropout	Batch size	Best val - loss
PV	75	20	0.001	0.05	2	0.0158
Wind	80	20	0.001	0.05	2	0.0074
Hybrid	90	32	0.001	0.1	2	0.0187

First, included hyperparameters in the optimization process were: layer size, learning rate and dropout factor. The batch size was first excluded since, from literature and other comparable studies was concluded that a value of 32, 64, 128 or 256 is common for best practice with the default value of 32 being a reasonably performing value. With this first set-up, the RMSE (%) did not improve over 20% which is a fairly high error for power prediction purposes. To enhance the performance the ranges of optimization were increased to widen the search area for the algorithm. This did not improve the performance. Increasing the layer size, for example, decreased the performance already from about 100 neurons. Then, the batch size was included in the optimization algorithm with a search range of 2 to 256 with steps of factor 2. As can be seen from Figure 4.3, the batch size had an extensive impact on the RMSE (%) of the 1-hour ahead forecast. The RMSE increased rapidly with increasing batch sizes and had a maximum RMSE of 22.7 % for a batch size of 256. The batch size was chosen to be 2 as this value represented the lowest RMSE of 6.48 %, this increased the computational time by almost a factor of 2.
As explained in Section 2.5.2, the batch size is the number of samples used to compute the

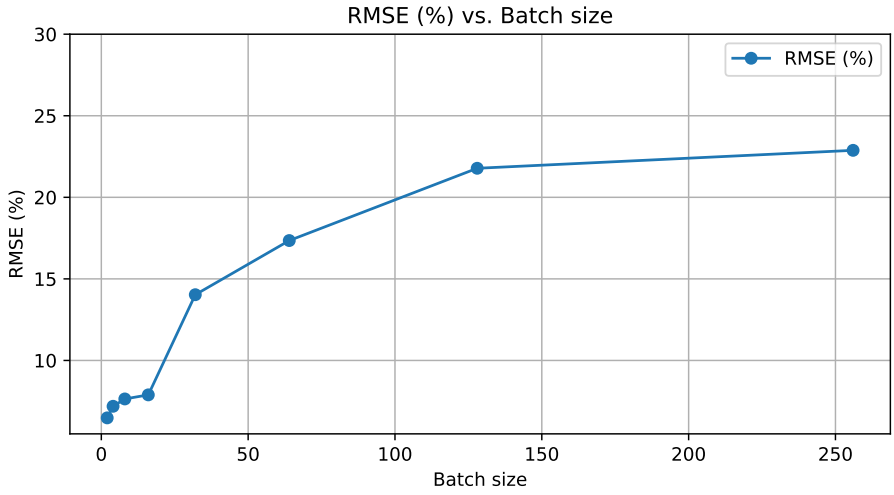


FIGURE 4.3: Accuracy results of the optimization algorithm for varying batch sizes with steps of factor 2.

error gradient for the learning algorithm. Having a larger batch size means the gradient will be estimated over a more extensive set of samples which cancels out noise but induces a higher generalization error. Lower batch sizes, lower than 32 in the case of the results in Figure 4.3, reduce the generalization error. For noisy data, a lower batch size thus has a positive impact on the performance since weights are updated more frequently. Since especially wind data can be classified as noisy data, a lower batch size would be preferable as a hyperparameter for this research.

The result of the optimization of the learning rate is shown in Figure 4.4 where the RMSE (%) is plotted versus the learning rate. The learning rate varied from 0.1 to 0.00001 with a logarithmic step change.
Lastly, the dropout factor was chosen to be 0.05, 0.05 and 0.1 for the PV, wind and hybrid models respectively. For values smaller than 0.1 the dropout layer had a positive impact on the RMSE. Beyond a value of 0.1, the RMSE started to decrease. This might be relatable to the size of the available training data. For larger datasets, a larger dropout factor has a relatively lower impact and vice-versa.

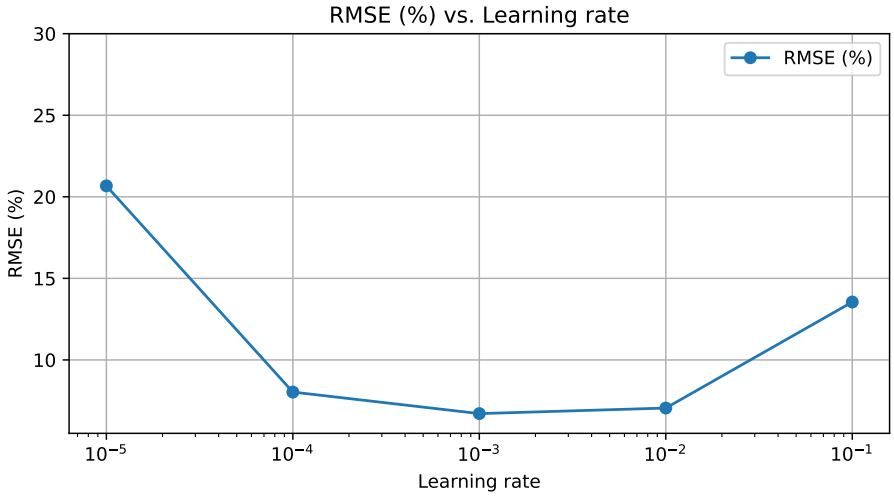


FIGURE 4.4: Accuracy results of the optimisation algorithm for varying learning rates with logarithmic steps.

4.3 Performance evaluation

This section presents the results of the prediction models and compares the proposed models concerning their RMSE performance. First, the 1-hour ahead prediction models are presented. Then, the multi-step ahead prediction models are discussed. Additionally, a comparison between the two proposed machine learning techniques is displayed.

4.3.1 1-hour ahead predictions

The main research question is to compare the performance of a proposed hybrid model performance with the sum of the two individual models. One of the sub-questions to aid in answering the main question is to verify if the individual models would outperform the proposed benchmark methods. This subsection first displays the results of the benchmark verification for both proposed machine learning techniques. Thereafter, the results of the comparison between the sum and the proposed hybrid model are presented in more detail

for the 1-hour ahead predictions.

Table 4.2 displays the RMSE values in percentage for the two proposed machine learning models, wind persistence, smart PV persistence and the different model types. The RMSE is normalized to enable comparison between the different model types. Since the installed capacities differ per model type it makes the comparison with the ordinary RMSE inaccurate. The best NRMSE value per prediction model is displayed in bold.

TABLE 4.2: Results of the 1-hour ahead power predictions for the proposed machine learning techniques and the four power prediction models.

Model	PV NRMSE (%)	Wind NRMSE (%)	Sum NRMSE (%)	Hybrid NRMSE (%)
Persistence	8.78	6.94	6.92	7.38
XGBoost	**8.3**	6.80	**6.36**	**5.95**
NN	8.43	**6.24**	6.70	7.11

Two important findings can be drawn from these results. First, both proposed machine learning techniques for the individual models outperform the respective benchmark methods. This means the individual models produce relevant predictions. Secondly, it can be observed that the XGBoost machine learning technique outperforms the neural network overall with an average of 0.3 % over the four different prediction models. Thirdly, the Persistence of the sum had a better performance than the hybrid Persistence method. The best performing of the two was considered as the benchmark for further comparisons.
Lastly, no clear performance improvement of the hybrid models over the sum models can be observed yet. This can also be seen in Figure 4.5. Here the monthly distribution for the NN summed and hybrid model are presented. What can be seen is the summed model has an overall more accurate performance than the hybrid model. With an average more accurate performance of 0.41%. On the contrary, the XGBoost hybrid model does outperform the XGBoost summed model. Figure 4.6 presents the hourly distribution of the NRMSE results. During all night hours, the hybrid model outperforms the summed model. Whereas, during the morning the summed model has overall better performance. This is possibly related to the better performance of the individual PV model during these hours. Overall the Hybrid model outperforms the summed model with 0.41%.

Detailed result discussion

This subsection provides a more detailed analysis of the results for the 1-hour ahead predictions. The hourly and monthly NRMSE distributions for the individual PV and wind models are, Figure 4.7 and 4.8 are shown in the appendix. What stands out in the PV XGBoost model is the performance during sunrise. On average the persistence method is outperformed by 2.7% till hour 10. During peak hours the performance of XGB and persistence are almost equal. During sunset, the XGB model outperforms persistence again. For the wind XGBoost model, there is no clear pattern of hourly performance.

Figure 4.9a presents the hourly performance, expressed in NRMSE (%), for the hybrid and summed XGBoost model. As is displayed in Table 4.2 and as can be observed from Figure 4.9a the hybrid model outperforms the sum with an average NRMSE of 0.41% over

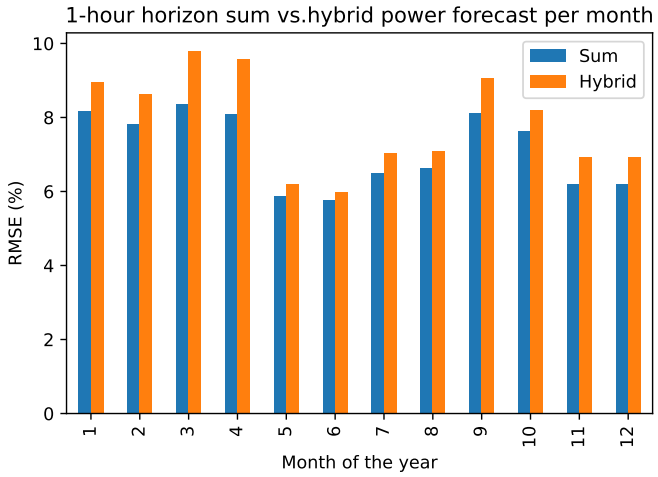


FIGURE 4.5: Monthly NRMSE distribution for summed and hybrid NN models.

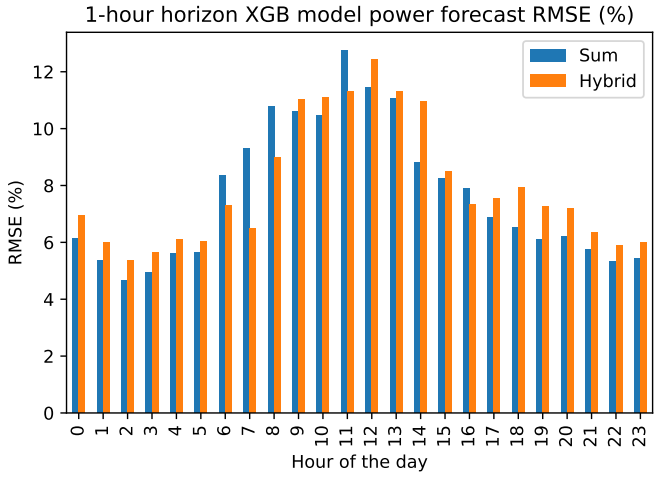


FIGURE 4.6: Hourly NRMSE distribution of summed and hybrid XGBoost models.

the entire test set. Several important findings can be drawn from the hourly NRMSE distribution. During sun hours, between hours 6 and 18, the hybrid model outperforms with respect to the summed model with an average of 0.7%. Whereas between night hours the hybrid model seems to underperform by almost 0.5% on average. This is the first indication that the combination of PV and wind enhances performance over total power production when both units are producing electricity.

Figure 4.9b presents the results of the NN model for the summed and proposed hybrid model. As displayed in Table 4.2, the hybrid model has a higher NRMSE with respect to the summed model over all the hours. This is a disappointing result since the summed model outperforms the hybrid model with an average of 1.6%.

The dependence on seasonality has also been studied and the results for the XGBoost wind power forecast for July and December are presented in Figures C.5 and C.6 respectively. What can be drawn from these results is that the overall performance of the wind

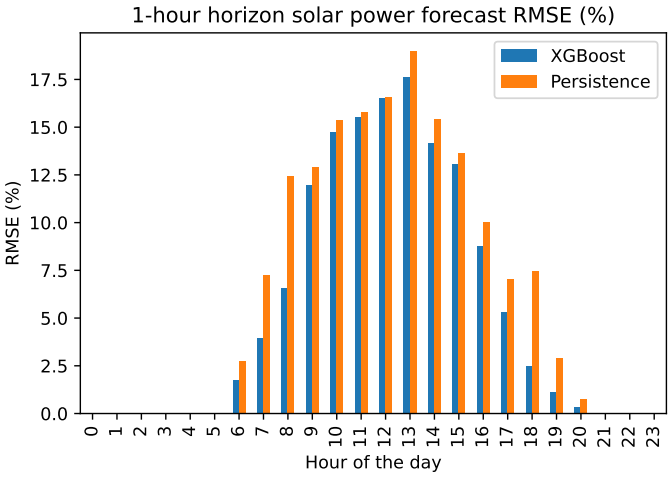


FIGURE 4.7: Hourly NRMSE distribution for PV XGBoost model.

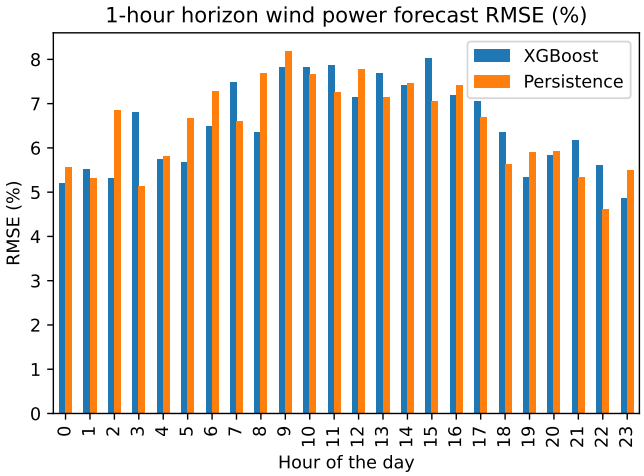
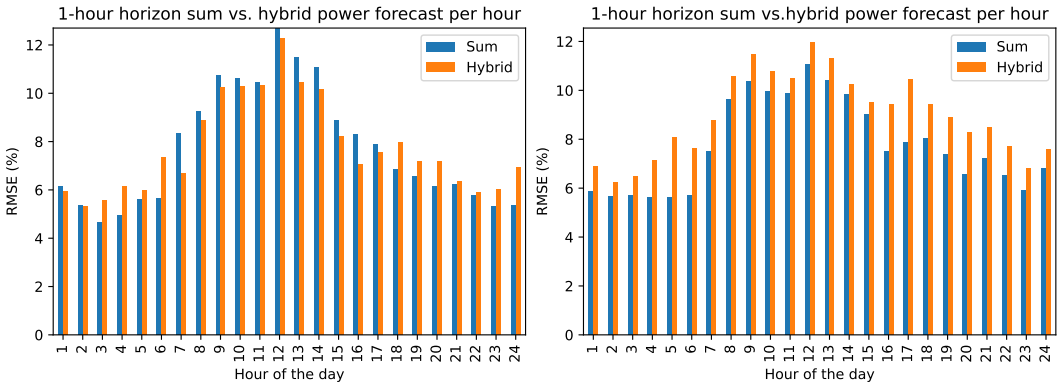


FIGURE 4.8: Hourly NRMSE distribution for wind XGBoost model.



(A) XGBoost RMSE results.

(B) Neural network RMSE results.

FIGURE 4.9: Hourly NRMSE distribution for sum vs. Hybrid models, 1-hour horizon.

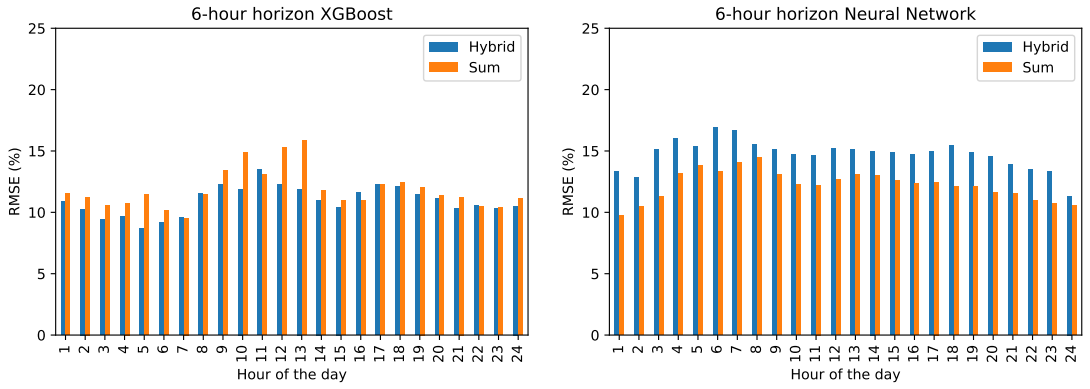
predictions is on average more accurate in July. On the contrary, the wind model only outperforms the persistence method in December. One possible reason could be the more stable

and lower average wind speed during summer, where persistence thrives.

Furthermore, Figure C.4 presents the comparison between the XGBoost hybrid model performance in July and December. What can be seen is that the overall performance in December is lower than in July. Especially during night hours, the model seems to constantly underperform in December. One reason for this difference in performance could be the more intensive and unpredictable cloud cover during the winter months.

4.3.2 6-hour ahead predictions

After computing the performance of the 1-hour horizon models, the 6-hour horizon predictions were made. These predictions were made for PV-, wind-, summed- and hybrid-model. The results for the XGBoost and neural network (NN) predictions performances are presented in Table 4.3 and 4.4 respectively. Figures 4.11a and 4.11b show the graphs related to the results for XGBoost and neural network respectively. The RMSE values are normalized for the respective installed capacities to be able to compare the results.
From Figure 4.11a the following findings were observed. First, clearly, the summed and hybrid model both outperform the persistence NRMSE for all horizons. Secondly, the proposed hybrid model outperforms the summed model over every hour ahead horizon, with an average NRMSE of 1.05%.
From Figure 4.11b the following findings were observed. First, clearly, the summed and hybrid model both outperform the hybrid persistence NRMSE. Secondly, the proposed hybrid model outperforms the summed model beyond the 2nd hour ahead horizon. The hourly distribution of the NRMSE for the 6-hour summed and hybrid XGBoost and NN models are presented in Figures 4.10a and 4.10b respectively. These figures are in line with the finding that both the XGBoost and hybrid models outperform the summed models at the 6-hour horizon.
Furthermore, it can be seen that when comparing Figures 4.11a and 4.11b the XGBoost hybrid model has a more accurate performance than the neural network hybrid model over all proposed prediction horizons.



(A) XGBoost NRMSE results.

(B) NN NRMSE results.

FIGURE 4.10: Hourly RMSE distribution for sum vs. Hybrid models, 6-hour horizon.

In Figures C.1 and C.2 the results for the PV and wind models are presented. What can be seen is that in both the XGBoost and the NN models the PV predictions are performing better in the 1 and 2-hour ahead horizon than the wind models. From the 3rd prediction hour

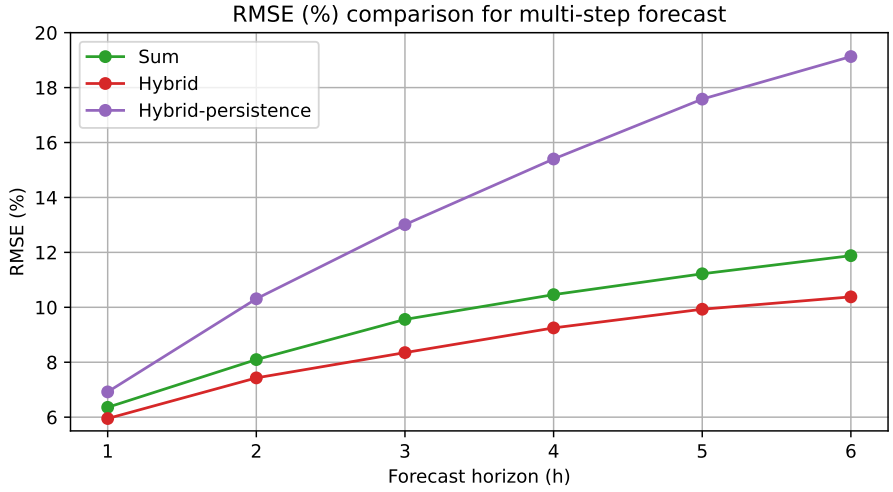
ahead the XGBoost wind model starts outperforming the PV model and at the 6-hour ahead the wind model outperforms the PV model by 2.34%. For the NN models, the wind model outperforms the PV model by 1.46% at the 6-hour prediction.

TABLE 4.3: NRMSE performance results of the 6-hour ahead XGBoost models.

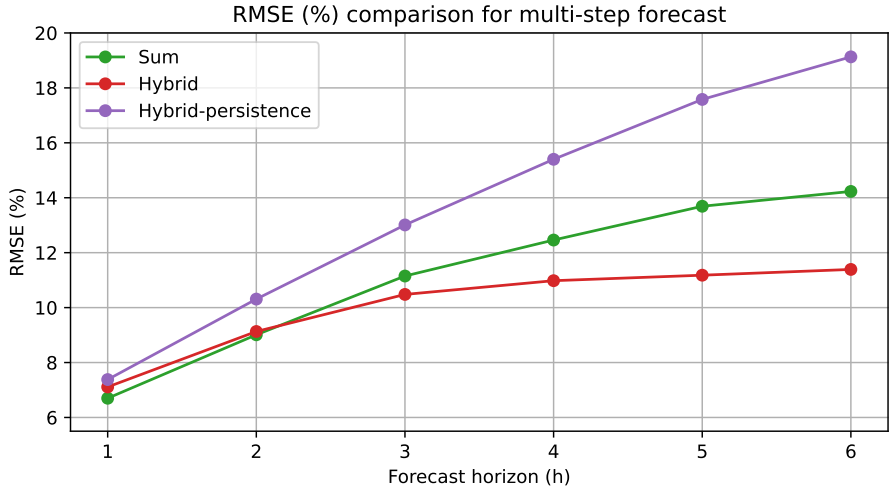
Hour	**1**	**2**	**3**	**4**	**5**	**6**
PV	8.31	10.02	11.41	12.35	12.55	13.56
Wind	6.80	9.80	12.30	13.60	15.10	15.90
Combined - persistence	6.92	10.31	13.01	15.4	17.58	19.13
Summed	6.36	8.10	9.56	10.46	11.22	11.88
Hybrid	5.95	7.43	8.35	9.25	9.93	10.38

TABLE 4.4: NRMSE performance results of the 6-hour ahead NN models.

Hour	**1**	**2**	**3**	**4**	**5**	**6**
PV	8.43	10.83	11.94	12.79	13.14	13.4
Wind	6.24	9.29	11.24	12.67	13.97	14.86
Combined - persistence	7.38	10.31	13.01	15.40	17.58	19.13
Summed	6.70	9.01	11.15	12.46	13.69	14.23
Hybrid	7.11	9.62	10.36	10.84	11.12	11.43



(A) XGBoost NRMSE results.



(B) Neural network NRMSE results.

FIGURE 4.11: 6-hour ahead NRMSE performance results.

Detailed result discussion

This subsection provides a more detailed analysis of the results for the 6-hour ahead predictions. When comparing the 6-hour ahead hourly NRMSE distribution of the neural network in Figure 4.10b, with the 1-hour ahead hourly distribution of the neural network for the wind predictions in Figure 4.9b. One finding is the 6-hour ahead prediction performance is more constantly distributed over the hours of the day. This could be related to the different profiles of PV and wind production over the whole day. Where PV is dominant between hours 10 and 16, the wind is generally more dominant in the afternoon and at night. When doing a 6-hour ahead forecast the overall average production profile could potentially have a soothing effect on the error distribution.
Figures 4.5 and 4.12 present the monthly NRMSE distribution for the summed and hybrid NN models. What can be seen is that the summed and hybrid follow the same seasonality pattern with an average higher error during winter and a more accurate performance during summer. The difference in performance near the 6-hour horizon is clearly increasing which is in line with the results from 4.11b.

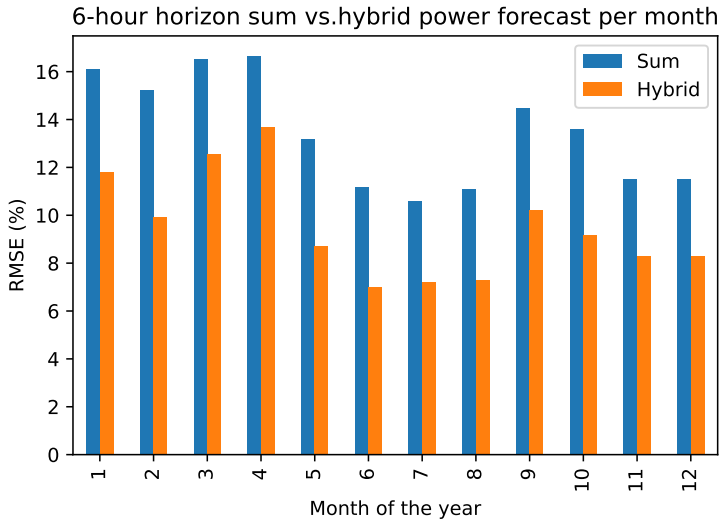


FIGURE 4.12: Monthly NRMSE distribution for summed and hybrid NN models.

4.4 Valuation

This section presents the results of the valuation performance of the five proposed models. Table 4.5 shows the cumulative value creation for the XGB hybrid & summed model, the NN hybrid & summed model and the persistence benchmark per remaining month. As can be seen, both the hybrid models were the best-performing models. The NN summed model and the persistence method have the worst cumulative performance. These differences are mostly related to the imbalance delta, the difference between forecast and actual delivery, ΔP.

TABLE 4.5: Cumulative value creation per month of the 4 proposed models. The highest cumulative value is shown in bold.

Month	**8**	**9**	**10**	**11**	**12**
XGB hybrid	**8375**	**14918**	20028	31360	**43084**
XGB summed	6031	14747	21894	31353	38839
Persistence	0	0	0	0	0
NN summed	7221	12939	19236	26699	35951
NN hybrid	7083	14462	**24742**	**34839**	42983

The best-performing model is found to be the XGB hybrid model. However, the difference between the XGB and NN hybrid models is small at the end of the month. During the period the NN hybrid model even outperforms the XGB hybrid model in the months of October and November. This might be counterintuitive since the XGB model had an overall more accurate performance. The reason for this difference in accuracy and value creation can be related to the regulation state factor of the imbalance market. This brings a luck factor

into the cumulative value creation since a larger imbalance delta doesn't directly result in negative revenue.

Detailed result discussion

To explain this counter-intuitive result a more detailed description is presented in this subsection. In Figure 4.13 the cumulative value creation is presented for three days: July 6th, 7th and 8th. Two events are indicated with blue and red respectively. At the blue event, a clear division in value creation between the four different models in PTU 1 can be observed. The following happens during this PTU.
The regulation state of the imbalance market was 2. This means the imbalance delta is multiplied with a positive downward price or a negative upward price depending on the surplus of the imbalance delta. What can be seen is that the XGB summed and NN hybrid do not have an increase or decrease in the cumulative value creation. This is due to the fact that both these models and Persistence produced a forecast that resulted in a shortage, negative value, of the imbalance delta. When the regulation state is 2, the upward price is higher than the mid price and the respective settlement price is positive the BRP has to pay TenneT.
Since the XGB hybrid and the NN summed had a surplus in the imbalance delta, the cumulative value creation spiked at this event. Both because the other models had to pay for their imbalance and because the price difference was a factor 15, as can be seen in Table 4.6. This event indicates that the forecast error is not only important from a percentage point of view but also if the error creates a surplus or a shortage. This could explain why in October and November the NN hybrid model, which had a higher NRMSE, created more value than the XGB hybrid model.

TABLE 4.6: Respective prices and regulation states of the two presented events.

Event	Up	Down	Regulation state
Blue	218.02	15.11	2
Red	22.88	22.88	-1

The PTU in the red event is the case with a regulation state of -1. This means the imbalance delta is multiplied with the marginal downward price, 22.88 in this case. In this PTU all models produced a forecast with a resulting shortage in the imbalance delta and performed better than Persistence. The difference in value creation here is due to the size of the resulting imbalance delta. For the XGB hybrid and NN summed model, this delta was much smaller and resulted in a lower cost and thus a higher value creation with respect to Persistence and the other models.

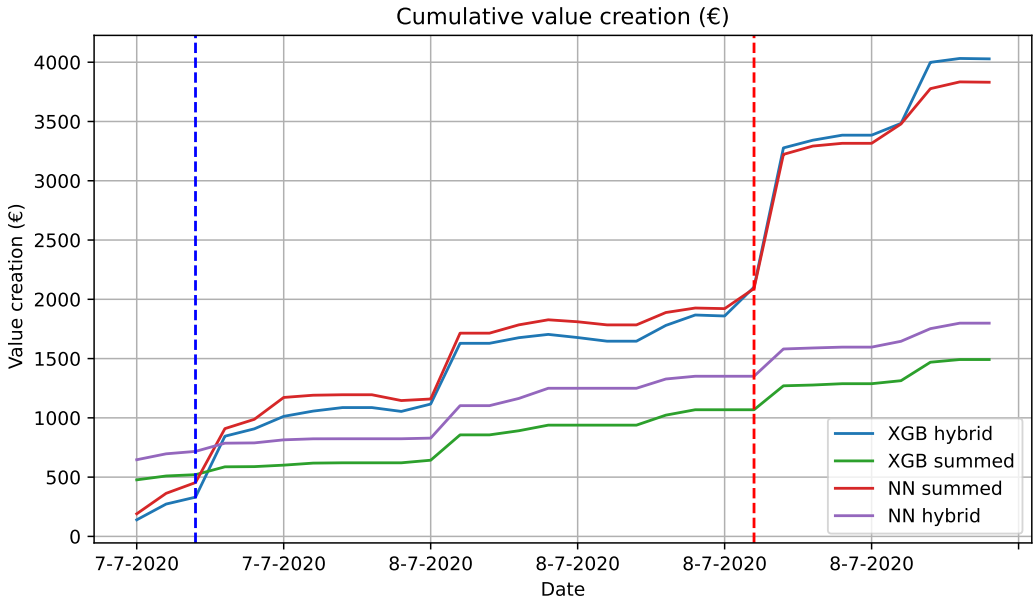


FIGURE 4.13: Cumulative value creation of the four proposed models for the 6th, 7th and 8th of July in 2020. The blue and red vertical line represent two specific events.

Chapter 5

Conclusion & recommendations

5.1 Conclusions

The fast growth of installed renewable capacity has brought several challenges to the dutch grid. First, the grid became full in terms of physical connection capacity. Secondly, grid security is currently at risk due to the higher relative intermittency of the energy producers. These two main challenges were the reason for this research on power forecasting of PV and wind hybrid systems.

To best tackle the above-mentioned challenges in the Dutch grid, three topics were introduced to create relevant insight and answer the main research goal. First, the individual PV and wind models were built for the two proposed machine learning models. These models were optimized with respect to their hyperparameters to minimize the RMSE metric. These models were compared to the proposed smart PV persistence and wind persistence to ensure the relevancy of further research. Then, the proposed hybrid model was built and was again optimized to reduce the RMSE metric. This optimisation process was repeated for the 1-hour ahead forecast till the 6-hour ahead horizon. The 6-hour ahead model was used to evaluate the model performance with respect to the value creation in the energy market. The final research goal was to compare the performance of the hybrid model with respect to the individual models on NRMSE and value creation.

The individual models and the hybrid models all had their own set of input features and hyperparameters. The input features for the PV and wind models were chosen with the use of a feature importance method based on a tree-based regression model. The machine learning model was chosen to be the XGBoost (XGB) method. Then, the 1-hour horizon models were constructed for the two proposed machine learning techniques: XGB and a neural network. The hyperparameters were optimized with the use of a random search optimization algorithm to reduce the RMSE metric and were first compared with the proposed persistence benchmark methods.
The same process was repeated for the 6-hour horizon where four recursive models were built with numerical weather prediction (NWP) data as extra input features. Finally, a comparison was made with respect to the prediction performance and the value creation of the individual and hybrid models.

From the results of the above-mentioned research topics, several conclusions were drawn. First, all proposed models outperformed the respective persistence benchmarks on the 1st till 6th hour ahead prediction. Secondly, there was no clear increased performance for the

hybrid model over the summed model for the 1-hour predictions. On the contrary, the XGBoost hybrid model did outperform the summed model for all prediction horizons beyond the first hour. In addition, the NN hybrid model outperformed the NN summed model significantly for the 6-hour ahead predictions. Thirdly, all the XGBoost models outperformed the NN models for both the 1-hour and the 6-hour horizons. Lastly, the proposed summed and hybrid models created more value than the persistence method. There is no clear relation between the performance on the NRMSE metric and created value for the 6-hour ahead horizon. Since the hybrid NN model created more value in October and November while the hybrid XGBoost had the most accurate performance. All proposed models created positive value with respect to Persistence over the studied period.

5.2 Limitations

During the research project, the following limitations were identified that had a potentially negative effect on the results:

- **Data origination.** The data that was used for the PV models, were variables generated by a separate online model PVGIS. These generated variables are based on NWP data and satellite measurements. On the contrary, the input variables for the wind models were real-time measurements for power and several meteorological variables. Since the models were meant for local power prediction the PVGIS variables could have potentially ignored local weather conditions such as cloud formation or shading.
- **Data quality.** Since the PV features were generated by the PVGIS tool, the data was relatively clean and complete. The wind variables on the contrary had more impurities and required pre-processing measures.
- **Training, test and validation length.** The length of the dataset was spread over 2 years, 2019 and 2020. This data was split over a training, test and validation set to optimize and evaluate the models properly. This resulted in a test set that didn't span a whole year. In this way, not every month and every season could be evaluated.
- **Spatial dependence.** The location of the project was chosen to be in Terneuzen since data of the wind farm KGP was made available. The results of this research are now bound to this specific location and no conclusion can be drawn about hybrid systems in general.

5.3 Recommendations

Based on the research, the following recommendations for future projects are proposed:

- In this research two machine learning techniques were proposed to construct both the individual and hybrid models. Future research could implement an ensemble model to find complementarities between different machine learning techniques.
- Next to the creation of ensemble models, novel machine learning techniques could be introduced to this topic next to the proposed XGBoost and neural network models. As presented in the literature review, multiple models have been found to be useful in power forecasting.
- This research project is meant to study the prediction performance of hybrid system modelling with real-time data. Since real-time PV data was lacking, the PVGIS tool

was used. Future research could search for local hybrid systems with real-time data of both PV and wind power production and meteorological measurements. In this way, local weather dependence could be studied in addition to larger-scale weather impact.

- The value creation, in this research, was based on the 6-hour ahead forecast. Future research could increase the prediction horizon and improve the trading strategy by including the day-ahead market. Future research could investigate the similarities between production profiles and the imbalance markets. In this way sizing of hybrid systems could be optimized to reduce imbalance costs.
- To further extend the value creation algorithm, a battery could potentially be added to the hybrid system so possible imbalance delta's can be corrected near the actual delivery.
- For the feature importance model, the 4-hour lagged variables were used to study the dependence on the predictions. Future research could increase the amount of lagged values to study if the daily cycle of PV and or wind power increases the forecast performance.
- For this research three NWP variables were used for the multi-step ahead predictions. Future research could increase the number of features and maybe use more advanced tools to decrease the grid size.

Appendix A

KGP location

FIGURE A.1: Screenshot of the Netherlands with the location of wind farm Koegorspolder in Terneuzen (Lat: 51.284, Long: 3.849).

Appendix B

Feature input infrastructure

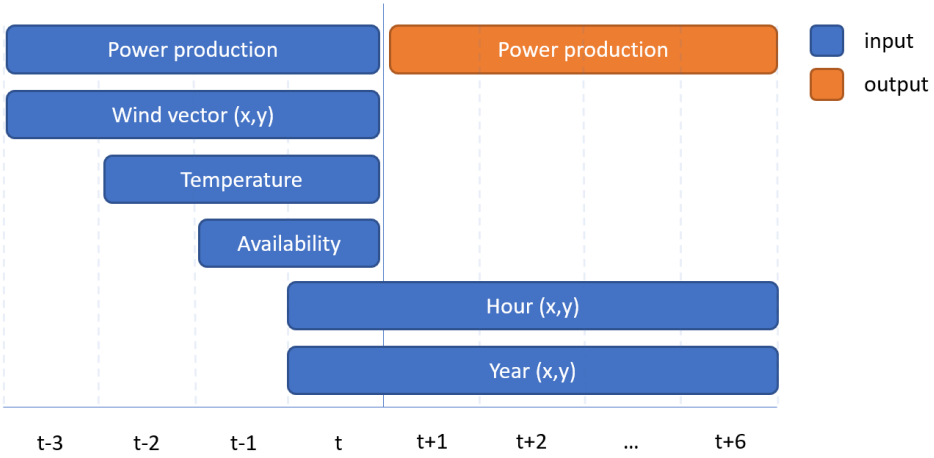


FIGURE B.1: Visual representation of the input features and output variables for the wind models. This figure represents the 1-hour to 6-hour horizon.

TABLE B.1: Utilization of meteorological and Power variables of wind farm Koegorspolder (KGP) and the PVGIS source.

Variable	**Wind 1h**	**Wind 6h**	**PV 1h**	**PV 6h**	**Hybrid 1h**	**Hybrid 6h**
Wind speed (KGP)	X	X	-	-	X	X
Wind speed (NWP)	-	X	-	X	-	X
Wind direction (KGP)	X	X	-	-	X	X
Wind direction (NWP)	-	X	-	-	-	X
Power (KGP)	X	X	-	-	X	X
Temperature (KGP, PVGIS)	X	X	X	X	X	X
Temperature (NWP)	-	X	-	X	-	X
Irradiance (PVGIS)	-	-	X	X	X	X
Power (PVGIS)	-	-	X	X	X	X
Sun height (PVGIS)	-	-	X	X	X	X

Appendix C

Forecast performance evaluation

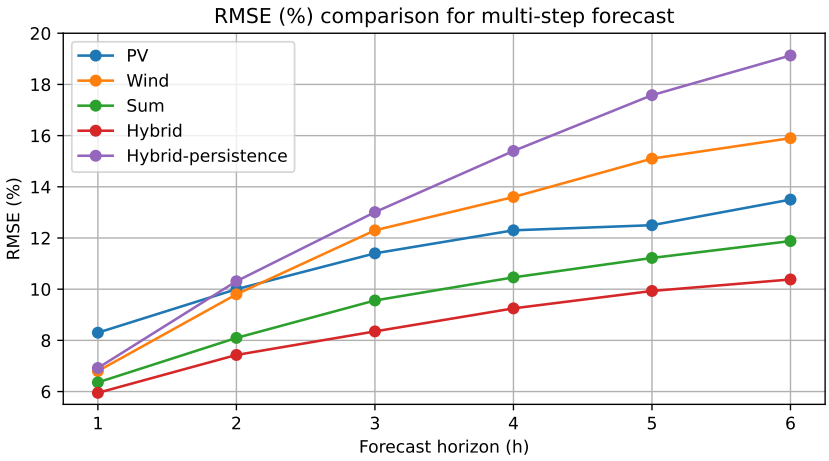


FIGURE C.1: 6-hour ahead predictions for the XGBoost models.

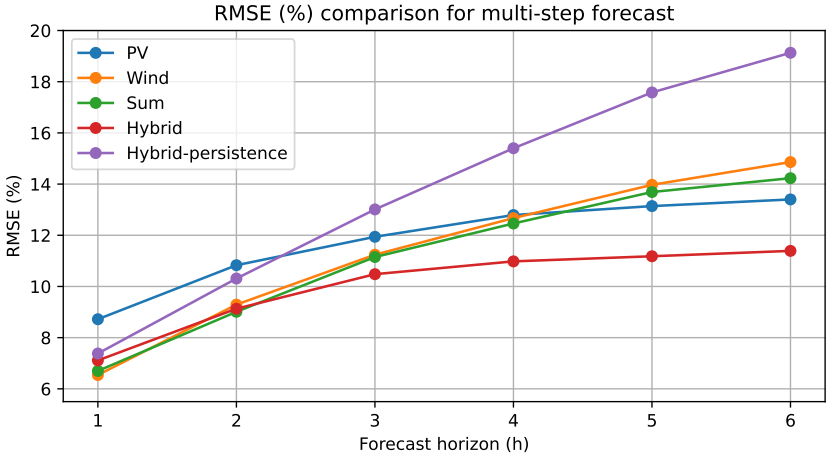


FIGURE C.2: 6-hour ahead predictions for the Neural Network models.

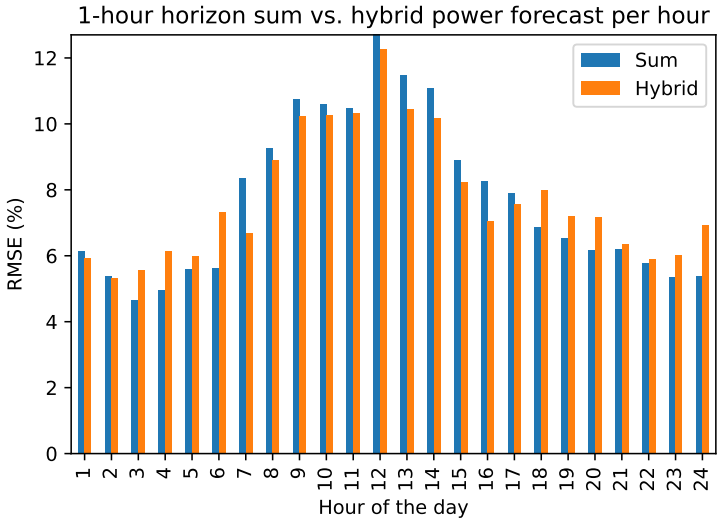


FIGURE C.3: Hourly NRMSE distribution for hybrid XGBoost model.

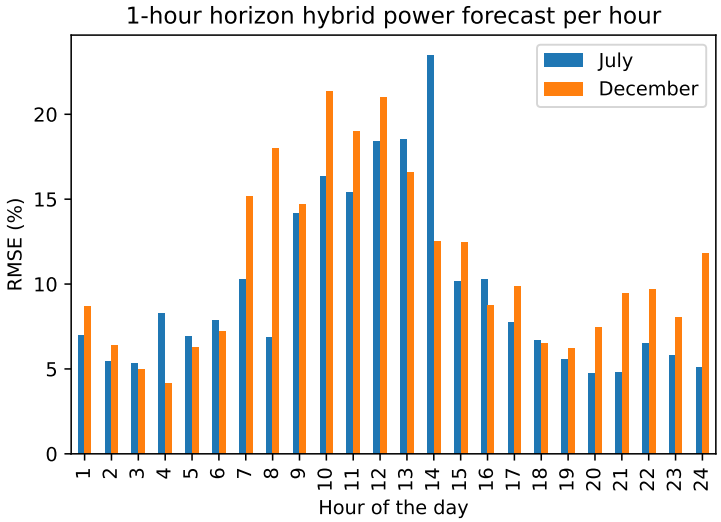


FIGURE C.4: Hourly NRMSE distribution for the hybrid XGBoost model.

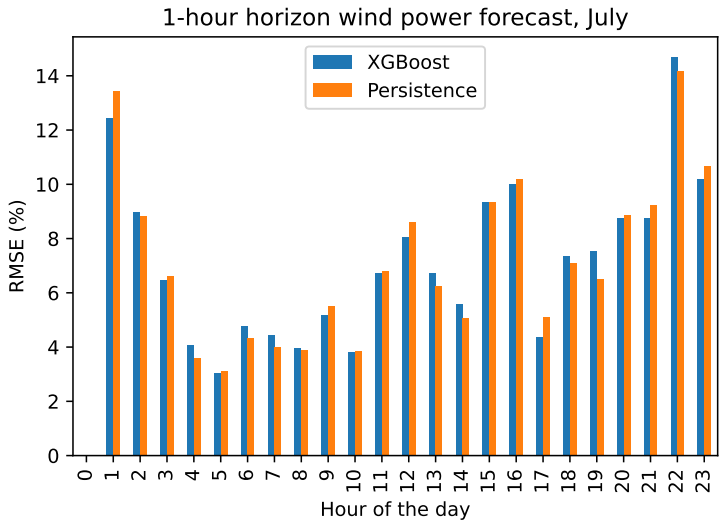


FIGURE C.5: Hourly NRMSE distribution for the wind XGBoost model.

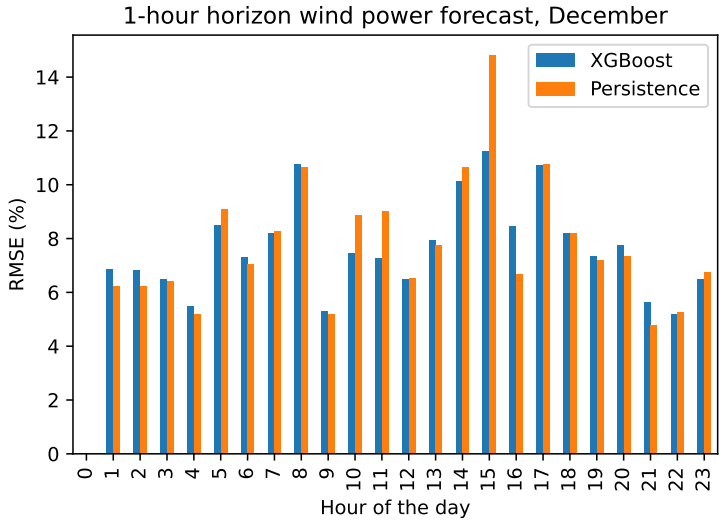


FIGURE C.6: Hourly NRMSE distribution for the wind XGBoost model.

Bibliography

Abdelaziz, Almoataz et al. (Dec. 2012). "Short Term Wind Power Forecasting Using Autoregressive Integrated Moving Average Modeling". In.

Abiodun, Oludare Isaac et al. (2018). "State-of-the-art in artificial neural network applications: A survey". In: *Heliyon* 4.11, e00938. ISSN: 2405-8440. DOI: https://doi.org/10.1016/j.heliyon.2018.e00938. URL: https://www.sciencedirect.com/science/article/pii/S2405844018332067.

Aly, Aly Mousaad, Faiaz Khaled, and Hamzeh Gol-Zaroudi (2020). "Aerodynamics of Low-Rise Buildings: Challenges and Recent Advances in Experimental and Computational Methods". In: *Aerodynamics*. Ed. by Mofid Gorji-Bandpy and Aly-Mousaad Aly. Rijeka: IntechOpen. Chap. 2. DOI: 10.5772/intechopen.92794. URL: https://doi.org/10.5772/intechopen.92794.

Artipoli, Gianluca and Francesco Schiavone (Feb. 2014). "Physical Modeling in Wind Energy Forecasting". In: *DEWI Magazin* 44, p. 10.

Bjørndal, Endre, Mette Bjørndal, and Linda Rud (2017). "Market Power Under Nodal and Zonal Congestion Management Techniques". In: *Norwegian School of Economics*.

Cadenas, Erasmo and Wilfrido Rivera (Jan. 2009). "Short term wind speed forecasting in La Venta, Oaxaca, México, using artificial neural networks". In: *Renewable Energy* 34, pp. 274–278. DOI: 10.1016/j.renene.2008.03.014.

Chaouachi, Aymen et al. (2009). "Neural network ensemble-based solar power generation short-term forecasting". In: *World Academy of Science, Engineering and Technology* 54, pp. 54–59.

Chen, Tianqi and Carlos Guestrin (2016). "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, California, USA: Association for Computing Machinery, 785–794. ISBN: 9781450342322. DOI: 10.1145/2939672.2939785. URL: https://doi.org/10.1145/2939672.2939785.

Chung, Junyoung et al. (2014). "Empirical evaluation of gated recurrent neural networks on sequence modeling". In: *arXiv preprint arXiv:1412.3555*.

Dhiman, Harsh and Dipankar Deb (Sept. 2020). "A Review of Wind Speed and Wind Power Forecasting Techniques". In.

Dong, Lei et al. (2016). "Wind power day-ahead prediction with cluster analysis of NWP". In: *Renewable and Sustainable Energy Reviews* 60, pp. 1206–1212. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2016.01.106. URL: https://www.sciencedirect.com/science/article/pii/S1364032116001362.

El Mghouchi, Youness (July 2022). "Best Combinations of Inputs for ANN-Based Solar Radiation Forecasting in Morocco". In: *Technology and Economics of Smart Grids and Sustainable Energy* 7. DOI: 10.1007/s40866-022-00152-z.

Ellabban, Omar, Haitham Abu-Rub, and Frede Blaabjerg (2014). "Renewable energy resources: Current status, future prospects and their enabling technology". In: *Renewable and Sustainable Energy Reviews* 39, pp. 748–764. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2014.07.113. URL: https://www.sciencedirect.com/science/article/pii/S1364032114005656.

Errard, Audrey, Fernando Diaz-Alonso, and Michael Goll (2021). *Electrical capacity for wind and solar photovoltaic power - statistics*. URL: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Electrical_capacity_for_wind_and_solar_photovoltaic_power_-_statistics#Evolution_of_electricity_production_capacity_by_main_fuel_groups.

Fairman, D (Feb. 2008). "Assessing the outdoor operating temperature of photovoltaic modules". In: *Prog. Photovolt. Res.* 16, pp. 307–315. DOI: 10.1002/pip.813.

Foley, Aoife M. et al. (2012). "Current methods and advances in forecasting of wind power generation". In: *Renewable Energy* 37.1, pp. 1–8. ISSN: 0960-1481. DOI: https://doi.org/10.1016/j.renene.2011.05.033. URL: https://www.sciencedirect.com/science/article/pii/S0960148111002850.

García, Salvador et al. (2016). "Big data preprocessing: methods and prospects". In: *Big Data Analytics* 1.1, pp. 1–22.

Gracia Amillo, Ana, Thomas Huld, and Richard Müller (Sept. 2014). "A New Database of Global and Direct Solar Radiation Using the Eastern Meteosat Satellite, Models and Validation". In: *Remote Sensing* 6, pp. 8165–8189. DOI: 10.3390/rs6098165.

Gu, Jiuxiang et al. (2018). "Recent advances in convolutional neural networks". In: *Pattern Recognition* 77, pp. 354–377. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2017.10.013. URL: https://www.sciencedirect.com/science/article/pii/S0031320317304120.

Hodge, Bri-Mathias and Anthony Florita (July 2013). "Investigating the Correlation Between Wind and Solar Power Forecast Errors in the Western Interconnection". In: DOI: 10.1115/ES2013-18423.

Huld, Thomas et al. (Dec. 2011). "A power-rating model for crystalline silicon PV modules". In: *Solar Energy Materials and Solar Cells - SOLAR ENERG MATER SOLAR CELLS* 95, pp. 3359–3369. DOI: 10.1016/j.solmat.2011.07.026.

Jung, Jaesung and Robert P. Broadwater (2014). "Current status and future advances for wind speed and power forecasting". In: *Renewable and Sustainable Energy Reviews* 31, pp. 762–777. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2013.12.054. URL: https://www.sciencedirect.com/science/article/pii/S1364032114000094.

Kariniotakis, George, G. Stavrakakis, and E.F. Nogaret (Dec. 1996). "Wind power forecasting using advanced neural networks models". In: *Energy Conversion, IEEE Transactions on* 11, pp. 762 –767. DOI: 10.1109/60.556376.

Kemmoku, Y et al. (1999). "Daily insolation forecasting using a multi-stage neural network". In: *Solar Energy* 66.3, pp. 193–199.

Koehl, Michael et al. (2011). "Modeling of the nominal operating cell temperature based on outdoor weathering". In: *Solar Energy Materials and Solar Cells* 95, pp. 1638–1646.

Kumler, Andrew, Yu Xie, and Yingchen Zhang (2019). "A Physics-based Smart Persistence model for Intra-hour forecasting of solar radiation (PSPI) using GHI measurements and a cloud retrieval technique". In: *Solar Energy* 177, pp. 494–500. ISSN: 0038-092X. DOI: https://doi.org/10.1016/j.solener.2018.11.046. URL: https://www.sciencedirect.com/science/article/pii/S0038092X18311460.

Luengo, Julián, Salvador García, and Francisco Herrera (2012). "On the choice of the best imputation methods for missing values considering three groups of classification methods". In: *Knowledge and information systems* 32.1, pp. 77–108.

Lydia, M. et al. (2014). "A comprehensive review on wind turbine power curve modeling techniques". In: *Renewable and Sustainable Energy Reviews* 30, pp. 452–460. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2013.10.030. URL: https://www.sciencedirect.com/science/article/pii/S1364032113007296.

Majumder, Irani, Manoja Behera, and Niranjan Nayak (Apr. 2017). "Solar power forecasting using a hybrid EMD-ELM method". In: pp. 1–6. DOI: 10.1109/ICCPCT.2017.8074179.

Martin, N. and J.M. Ruiz (2001). "Calculation of the PV modules angular losses under field conditions by means of an analytical model". In: *Solar Energy Materials and Solar Cells* 70.1, pp. 25–38. ISSN: 0927-0248. DOI: https://doi.org/10.1016/S0927-0248(00)00408-6. URL: https://www.sciencedirect.com/science/article/pii/S0927024800004086.

Mayer, Martin János and Gyula Gróf (2021). "Extensive comparison of physical models for photovoltaic power forecasting". In: *Applied Energy* 283, p. 116239. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2020.116239. URL: https://www.sciencedirect.com/science/article/pii/S0306261920316330.

Meinel, Aden B and Marjorie P Meinel (1977). "Applied solar energy: an introduction". In: *NASA STI/Recon Technical Report A* 77, p. 33445.

Mikolov, Tomas et al. (2010). "Recurrent neural network based language model." In: *Interspeech*. Vol. 2. 3. Makuhari, pp. 1045–1048.

Milligan, Michael, M. Schwartz, and Yih-Huei Wan (Jan. 2003). "Statistical Wind Power Forecasting Models: Results for U.S. Wind Farms; Preprint". In.

Müller, Richard et al. (Mar. 2012). "A New Algorithm for the Satellite-Based Retrieval of Solar Surface Irradiance in Spectral Bands". In: *Remote Sensing, vol. 4, issue 3, pp. 622-647* 4, pp. 622–647. DOI: 10.3390/rs4030622.

Nwogu, Eleazar C et al. (2019). "Choice between mixed and multiplicative models in time series decomposition". In: *Int J Stat Appl Math* 9.5, pp. 153–159.

Pang, Simian et al. (2021). "Hybrid Forecasting Methodology for Wind Power-Photovoltaic-Concentrating Solar Power Generation Clustered Renewable Energy Systems". In: *Sustainability* 13.12. ISSN: 2071-1050. DOI: 10.3390/su13126681. URL: https://www.mdpi.com/2071-1050/13/12/6681.

Panteri, E. and Stavros Papathanassiou (Jan. 2008). "Evaluation of two simple wind power forecasting models". In.

Patterson, Josh and Adam Gibson (2017). *Deep learning: A practitioner's approach*. " O'Reilly Media, Inc."

Pelletier, Francis, Christian Masson, and Antoine Tahan (2016). "Wind turbine power curve modelling using artificial neural network". In: *Renewable Energy* 89, pp. 207–214. ISSN: 0960-1481. DOI: https://doi.org/10.1016/j.renene.2015.11.065. URL: https://www.sciencedirect.com/science/article/pii/S096014811530481X.

Price, Trevor J. (2005). "James Blyth — Britain's First Modern Wind Power Pioneer". In: *Wind Engineering* 29.3, pp. 191–200. DOI: 10.1260/030952405774354921. eprint: https://doi.org/10.1260/030952405774354921. URL: https://doi.org/10.1260/030952405774354921.

Qadir, Zakria et al. (2021). "Predicting the energy output of hybrid PV–wind renewable energy system using feature selection technique for smart grids". In: *Energy Reports* 7, pp. 8465–8475. ISSN: 2352-4847. DOI: https://doi.org/10.1016/j.egyr.2021.01.018. URL: https://www.sciencedirect.com/science/article/pii/S2352484721000196.

Razak, Amelia et al. (Oct. 2016). "Investigation of the Effect Temperature on Photovoltaic (PV) Panel Output Performance". In: *International Journal on Advanced Science, Engineering and Information Technology* 6, p. 682. DOI: 10.18517/ijaseit.6.5.938.

Rhodes, Christopher (Mar. 2010). "Solar Energy: Principles and Possibilities". In: *Science progress* 93, pp. 37–112. DOI: 10.3184/003685010X12626410325807.

Rodgers, Joseph Lee and W. Alan Nicewander (1988). "Thirteen Ways to Look at the Correlation Coefficient". In: *The American Statistician* 42.1, pp. 59–66. ISSN: 00031305. URL: http://www.jstor.org/stable/2685263 (visited on 08/26/2022).

Schmelas, Martin et al. (2015). "Photovoltaics energy prediction under complex conditions for a predictive energy management system". In: *Journal of solar energy engineering* 137.3.

Smith, Leslie N (2018). "A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay". In: *arXiv preprint arXiv:1803.09820*.

Soman, Saurabh et al. (Oct. 2010). "A review of wind power and wind speed forecasting methods with different time horizons". In: pp. 1 –8. DOI: 10.1109/NAPS.2010.5619586.

Taieb, Souhaib Ben and Rob J. Hyndman (2012). "Recursive and direct multi-step forecasting: the best of both worlds". In.

The ECMWF Ensemble Prediction System (2012). URL: https://www.ecmwf.int/sites/default/files/elibrary/2012/14557-ecmwf-ensemble-prediction-system.pdf.

Touma, Jawad S. (1977). "Dependence of the Wind Profile Power Law on Stability for Various Locations". In: *Journal of the Air Pollution Control Association* 27.9, pp. 863–866. DOI: 10.1080/00022470.1977.10470503. eprint: https://doi.org/10.1080/00022470.1977.10470503. URL: https://doi.org/10.1080/00022470.1977.10470503.

Veen, Reinier AC Van der, Alireza Abbasy, and Rudi A Hakvoort (2010). "A comparison of imbalance settlement designs and results of Germany and the Netherlands". In: *Young Energy Engineers & Economists Seminar (YEEES), 8-9 April 2010, Cambridge, UK*. Citeseer.

Wang, Hai-Kun, Ke Song, and Yi Cheng (Jan. 2022). "A Hybrid Forecasting Model Based on CNN and Informer for Short-Term Wind Power". In: *Frontiers in Energy Research* 9. DOI: 10.3389/fenrg.2021.788320.

Wang, Yun et al. (2019). "Approaches to wind power curve modeling: A review and discussion". In: *Renewable and Sustainable Energy Reviews* 116, p. 109422. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2019.109422. URL: https://www.sciencedirect.com/science/article/pii/S1364032119306306.

Weibelzahl, Martin (2017). "Nodal, zonal, or uniform electricity pricing: how to deal with network congestion". In: *Frontiers in Energy* 11, pp. 210–232.

Wu, Xindong et al. (Dec. 2007). "Top 10 algorithms in data mining". In: *Knowledge and Information Systems* 14. DOI: 10.1007/s10115-007-0114-2.

Wu, Yuan-Kang et al. (2022). "Completed Review of Various Solar Power Forecasting Techniques Considering Different Viewpoints". In: *Energies* 15.9, p. 3320.

Xiong, Xiong et al. (2022). "A Short-Term Wind Power Forecast Method via XGBoost Hyper-Parameters Optimization". In: *Frontiers in Energy Research*, p. 574.

Yu, James J.Q. et al. (Aug. 2017). "Delay Aware Intelligent Transient Stability Assessment System". In: *IEEE Access* PP, pp. 1–1. DOI: 10.1109/ACCESS.2017.2746093.

Zhang, Dandan, Yuanlong Yu, and Zhiyong Huang (Jan. 2019). "Forecasting Solar Power Using Wavelet Transform Framework Based on ELM". In: pp. 186–202. ISBN: 978-3-030-01519-0. DOI: 10.1007/978-3-030-01520-6_17.

Zhang, Lei et al. (Nov. 2020). "Probability Density Forecasting of Wind Speed Based on Quantile Regression and Kernel Density Estimation". In: *Energies* 13, p. 6125. DOI: 10.3390/en13226125.

Zhang, Yao, Jianxue Wang, and Xifan Wang (2014). "Review on probabilistic forecasting of wind power generation". In: *Renewable and Sustainable Energy Reviews* 32, pp. 255–270. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2014.01.033. URL: https://www.sciencedirect.com/science/article/pii/S1364032114000446.

Zhang, Yichao et al. (Sept. 2021). "Machine Learning for Light Sensor Calibration". In: *Sensors* 21, p. 6259. DOI: 10.3390/s21186259.

Zhu, Xingquan and Xindong Wu (2004). "Class noise vs. attribute noise: A quantitative study". In: *Artificial intelligence review* 22.3, pp. 177–210.