SolarCloud: Forecasting Photovoltaic Production

Erdem Başeğmez

Master's thesis

SolarCloud: Forecasting Photovoltaic Production

Erdem Başeğmez

January 29, 2016

Institute for Data Processing
Technische Universität München

Erdem Başeğmez. *SolarCloud: Forecasting Photovoltaic Production.* Master's thesis, Technische Universität München, Munich, Germany, 2016.

Supervised by Prof. Dr.-Ing. K. Diepold and M.Sc. Dominik Meyer; submitted on January 29, 2016 to the Department of Electrical Engineering and Information Technology of the Technische Universität München.

Institute for Data Processing, Technische Universität München, 80290 München, Germany, `http://www.ldv.ei.tum.de`.

Abstract

Predicting the power of Photovoltaic (PV) plants is important for system reliability in modern power systems. In this work, the PV power is analyzed as a time series to find its relationship with its past values and meteorological data. Seasonal Autoregressive Integrated Moving Average (SARIMA), SARIMA with Exogenous inputs (SARIMAX) and artificial neural networks (ANN) are developed to model and forecast the PV power up to 24 hours ahead. After applying the models into a PV station in Germany together with meteorological data collected from several stations, results confirm that the proposed models are effective. Experiments prove that the SARIMAX model reaches higher accuracy at forecasting than the SARIMA model by including forecasted meteorological data, and the ANN model is able to perform better than the SARIMA and SARIMAX models with less effort during training.

Contents

1. Introduction

Renewable energy sources attract attention in the power production sector. Climate change and increases in prices of traditional energy sources trigger the expansion of renewable energy sources, such as solar and wind energy. A higher integration of these renewable sources is expected in the near future. By 2050, it is foreseen that the global electricity would be supplied 11% by the solar energy sources and 12% by the wind energy sources [1]. One of the renewable energy sources, solar energy has a high potential in clean energy.

Local solar plants, e.g. roof-top and solar farms, accounted for over 90% of the solar power market in 2010 and they are expected to dominate the market in the near future [2]. However, there are still some concerns at the operation side about the impact of their connection to the grid [3, 4]. The integration of renewable sources to the power grid is not straightforward, due to the volatility of their power outputs [1]. The performance of a solar plant depends on environmental conditions and the plant's physical parameters. The power output of a solar plant depends highly on the solar irradiance, which varies during the day because of the change of sun's position during a day and throughout the year. Ambient temperature, cell temperature and local climate conditions are among other factors of the solar power output [5]. Therefore, the major problem with the connection to the grid is that variations in the power generation may occur due to exogenous conditions, e.g. decrease in irradiance due to high cloudiness. Weather conditions fluctuate the solar power output and the power output fluctuations may cause frequency fluctuations in the grid. This would have even worse influence during high load times [4].

The energy levels in a power grid should be balanced. The grid operator should maintain the system efficiently. In order to promise that, he should be able to estimate the volume of power produced by unstable sources, e.g. solar plants. A reliable prediction can provide stability in the grid by balancing the stable energy sources accordingly [6]. Accurate forecasts would have important impact on the operation of the power grid and the energy market [7, 8, 9, 1].

Large solar plants can provide sale offers to energy markets. Especially in countries with an operative day-ahead markets, producers can use forecasts of hourly and daily power generation. In such markets, it is required to have highly accurate forecasting systems for optimizing profits and to avoid bad scheduling of the resource allocations [1]. The power output predictions of the solar plants can also be used for off-grid applications where the predictions could serve as input data to the off-grid control applications.

The models of predicting the solar power output can be categorized into: physics-based models and data-based models. Physics-based models are mathematical models of the

power output, which are derived from characteristic parameters, such physical properties of individual components in that particular plant by considering exogenous conditions, such as solar irradiance and cell temperature [10, 11, 12]. Physics-based models are developed to predict the solar power output via evaluating the thermoelectrical performance of the solar plants. Nevertheless, these methods require the knowledge of physical parameters in detail, which are mostly unavailable [4, 5]. Another approach is to predict the power output without the requirement of physical parameters of the solar plant by data-based models with the help of stochastics and machine learning methods [4].

In summary, power output prediction systems are important and necessary to be developed, which can be used for both power grids and local applications [6].

1.1. Objectives

The main objective of this thesis is to implement prediction models, which aim to provide short-term solar power output forecasting. It is intended to reach accurate results in the horizon up to 24 hours. The main problem is to detect the necessary features and find proper approaches, which provide high accuracy. It is aimed to implement time-series models under consideration of exogenous inputs, e.g. climate conditions. This process requires exploration of the historical data and figuring out which data might play the role of a feature in the model.

The aimed prediction models determine their intrinsic parameters from the historical data, which needs to be sufficient and consistent. The available data contains power output data of a few solar plants in given locations. The models, however, are expected to provide better results, if they are also trained with weather data. The difficulty relies on finding the weather data, which is related to the given location. Afterwards the goal is to combine the historical data and select features for various models to minimize prediction errors.

1.2. Related Work

There are several research papers, which have similar interests as this thesis. These papers use different data-driven models based on various inputs. In [13] a hybrid model is built with Support Vector Machine (SVM) and Seasonal Autoregressive Integrated Moving Average (SARIMA) based on small-scale grid-connected Photovoltaic (PV) plans in order to provide short-term power forecasting. The authors of [14] build an Autoregressive Moving Average with Exogenous Inputs (ARMAX) model and takes temperature, precipitation amount, insolation duration and humidity as exogenous inputs. In [15] artificial neural network-based model estimates the power of a PV module and its performance is compared for different seasons. Similarly, [4] predicts power output of PV panels using artificial neural networks by including exogenous inputs, such as air temperature, cell temperature, solar irradiance and wind speed. In [16] the insolation is predicted with fuzzy theory and

based on the insolation forecasting, 24 hours ahead predictions of PV power output is provided by artificial neural networks. The authors of [2] investigate Autoregressive Integrated Moving Average (ARIMA), Radial Basis Function Neural Networks (RBFNN) and Least Squares Support Vector Machine (LS-SVM) in order to forecast PV power production at the aggregated system level. In [1] short-term power forecasting is provided for PV plans based on historical similarity with a model called Historical Similar Mining (HISIMI) and optimizes the model with a genetic algorithm.

1.3. Outline

The thesis is organized as follows: In Chapter 2, the theory is explained, the background in statistics is given and time series analysis is presented. In addition, the mathematical background of the time series models are introduced, which help to predict PV power. Chapter 3 presents the experimental setup and the analysis of the available PV power data together with meteorological data. In Chapter 4, the modeling of the PV power data is made and results are shown by visualizations. Chapter 5 includes discussions and Chapter 6 presents the conclusions.

2. Theory

The observations of the PV power are data points in time, of which the adjacent ones are observably not independent of each other. The approach, which concerns systematically finding the time correlations between the data points, is referred as time series analysis. Time series analysis can be categorized into two approaches, time domain approach and frequency domain approach. The time domain approach is explained by the dependence of the current value on the past values and thus presumes a correlation between adjacent points in time. This approach concentrates on predicting future values of a time series as a function of the current and past values. For instance, linear regression of the current value on the past values of a time series can be used as a forecasting model. By introducing autoregressive integrated moving average models, linear regression methods are improved systematically [17, 18].

The frequency domain approach makes the assumption that the time series analysis relates to periodic variations found in data. This type of analysis evaluates the variance associated with the seasonality in interest and forms a variance profile to describe the time series. Which approach should be preferred for a time series analysis depends on the application. Even though time and frequency domain approaches may produce similar results, time domain approach performs better for the prediction of short horizons. Since the aim of the project is to predict the next few hours of the PV power based on the last few hours, it is opted for the time domain approach in this thesis. The seasonality concern in PV power can be handled by extending the ARIMA models accordingly [17].

In addition to the ARIMA model, artificial neural networks in the domain of time series modeling are studied. Artificial neural networks are powerful at capturing nonlinear features and therefore they can be useful at modeling PV power with exogenous inputs, where nonlinear relationships are present.

2.1. Background in Statistics

A time series can have different types of structural inclinations. For an analysis, statistical questions should be asked to the experimental time series data.

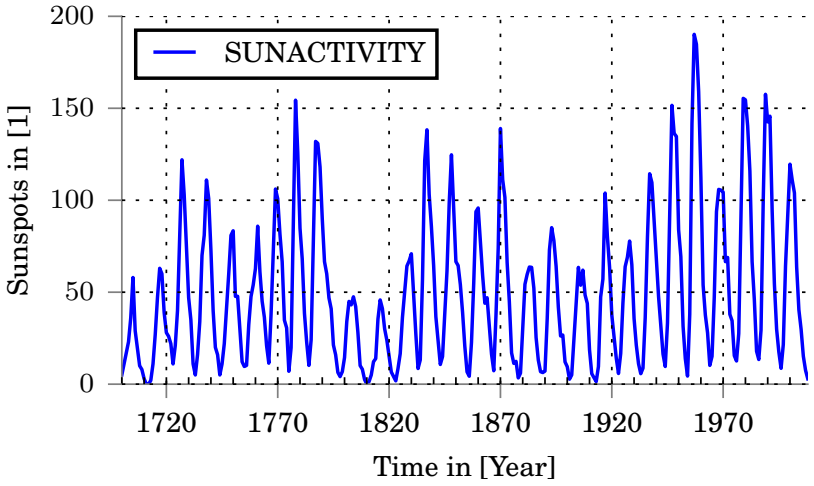

Figure 2.1.: Example time series plot of sunspots per year

The main goal of time series analysis is to develop mathematical models to describe the time series data. A time series can be defined as a sequence of random variables, which is ordered by the timestamps in the data. Assuming $\{x_t\}$ is a collection of random variables, where t denotes the integer sample number and $t \geq 1$, x_1 would denote the first value in the series and x_2 the second, and so on. The experimental data is measured discretely due to the methods of collection. In order to conduct an analysis, the adjacent samples in the discrete experimental data should be equally spaced from each other, meaning the sampling interval must be constant throughout the data.

A time series can be distinguished by observing the repeating patterns and smoothness in the plotting of values versus time. The smoothness is an indication of a correlation of the time series x at time t, which has a value of x_t, and previous values such as $x_{t-1}, x_{t-2}, ...$

In order to understand the fundamentals of random variables and their statistical properties, white noise is studied below.

2.1.1. Dependence

The correlation is a fundamental feature of time series analysis. The dependence can be examined via correlation and useful descriptions of a time series can be made in terms of covariance and correlation functions.

Mean function

The mean function provides the expected value of a random process. The mean function can be denoted with $\mu_x t$ for a given time series x_t and is defined as

$$\mu_{xt} = E[x_t] = \int_{-\infty}^{\infty} x f_t(x) dx. \tag{2.1}$$

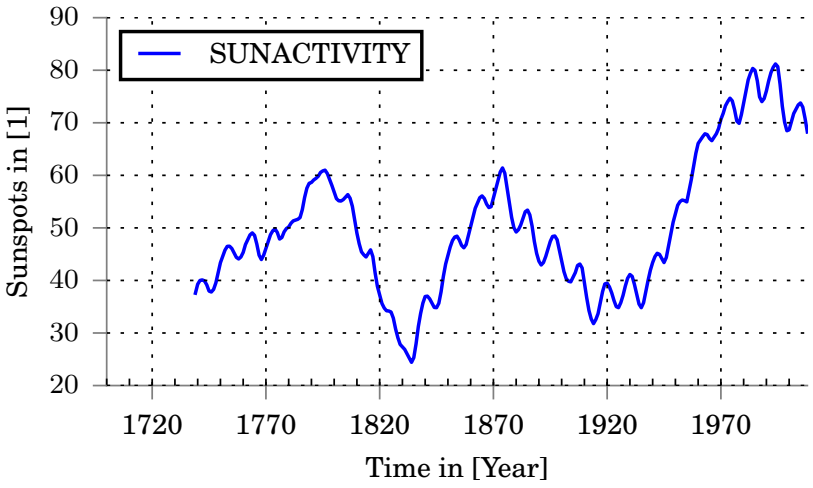

Figure 2.2.: Time plot of rolling mean over 40 samples of Figure 2.1

Autocovariance function

Given that the variance is finite, the autocovariance function of a time series x_t is defined as

$$\gamma_x(t_1, t_2) = E[(x_{t_1} - \mu_{t_1})(x_{t_2} - \mu_{t_2})]. \tag{2.2}$$

The autocovariance function measures the linear dependence between the time points t_1 and t_2. If the autocovariance function outputs 0, this means a linear dependence does not exist. However, a non-linear relationship might exist between x_{t_1} and x_{t_2}. In the case where $t_1 = t_2$, the autocovariance function equals to the variance of x_t. It is also clear that the parameters of autocovariance function are interchangable, meaning $\gamma_x(t_1, t_2) = \gamma_x(t_2, t_1)$.

Autocorrelation function (ACF)

If x_{t_2} is a linear function of x_{t_1}, that means x_{t_2} can be linearly predicted by x_{t_1}. The ACF is the measurement of the linear predictability between the points t_1 and t_2.

$$\rho(t_1, t_2) = \frac{\gamma(t_1, t_2)}{\sqrt{\gamma(t_1, t_1)\gamma(t_2, t_2)}}. \tag{2.3}$$

The difference between the autocovariance function and ACF is that the ACF is scaled and lies between the values -1 and 1. If the ACF is 0, then the two points are linearly independent. Else if the ACF is -1 or 1, then x_{t_2} is a linear function of x_{t_1}, where the sign determines the direction of change.

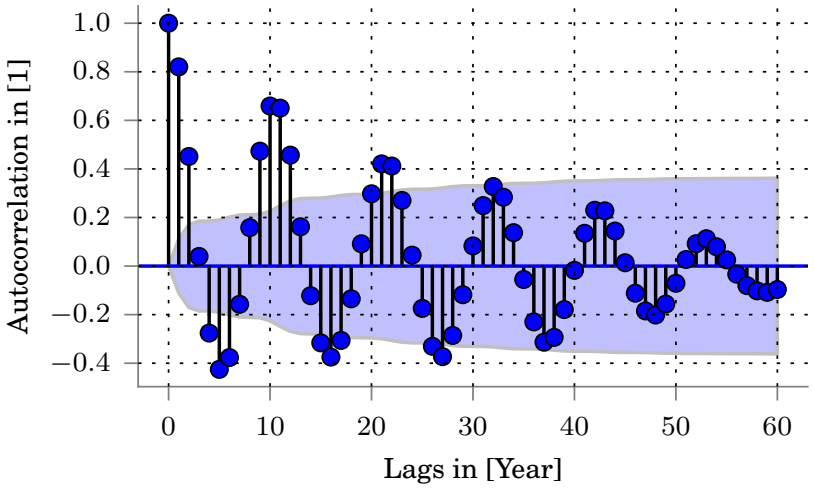

Figure 2.3.: The ACF of the sunspots time series in Figure 2.1, blue filled region depicts 95% confidence interval

2.1.2. Stationarity

The autocovariance function measures the dependence between two time points. It depends on the time points and may change along the series. When it does not depend on the location of the points, but the time difference between the points, then the series is stationary on condition that its mean function is constant. The stationarity of a time series is crucial to analyze a time series. This situation is of interest, because it is aimed to make a model which predicts the next few hours of PV power on the basis of last few hours. In other words, the next few hours of PV power is modeled as a function of the last few hours throughout the series.

A time series x_t is stationary if it has these three features: finite variance, constant mean function and its autocovariance function $\gamma(t_1, t_2)$ depends on t_1 and t_2 only through the difference $h = |t_1 - t_2|$. The term lag is often used to express h. Mathematically these conditions are defined as

1. $E[{x_t}^2] < \infty$,
2. $E[x_t] = \mu$,
3. $\gamma_x(t_1, t_2) = \gamma_x(t_1 + h, t_2 + h)$.

Following the third condition, the autocovariance function of a stationary time series is written as $\gamma_x(h) = \gamma_x(t_1, t_1 + h)$. Since the autocovariance function of a stationary series does not depend on a particular time point, it can be written as $\gamma(h)$.

The ACF of a stationary series can thus be defined as

$$\rho(h) = \frac{\gamma(h)}{\sqrt{\gamma(t_1 + h, t_1 + h)\gamma(t_1, t_1)}} = \frac{\gamma(h)}{\gamma(0)}, \tag{2.4}$$

where $\gamma(0)$ is the finite variance of the series x_t.

2.1.3. Discrete Functions for Sample Data

In practice, data sources contain sampled data as expected from data acquisition techniques and they represent only a portion of the whole series since the sources contain limited amount of data points. For that reason, it is appropriate to redefine the functions used to describe a time series x_t, which consists of n points.

The sample mean function for a stationary sampled time series is defined as

$$\overline{x} = \frac{1}{n}\sum_{t=1}^{n} x_t. \tag{2.5}$$

The sample autocovariance function of a stationary sampled time series is defined as

$$\hat{\gamma}(h) = n^{-1}\sum_{t=1}^{n-h}(x_{t+h} - \overline{x})(x_t - \overline{x}). \tag{2.6}$$

Analogously, the sample ACF is defined as

$$\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}_x(0)}. \tag{2.7}$$

2.1.4. White Noise

Many statistical modeling methods try to implement a model by transforming the time series to a white noise. Thus, after applying the model the ACF of the residuals is expected to hold the properties of a white noise sequence.

A series of uncorrelated random variables, w_t, with mean 0 and variance ${\sigma^2}_w$ is mostly used as a model for noise in practice. This stochastic process is also called white noise. Gaussian white noise, which is useful in most applications, consists of independent normal random variables and denoted as $w_t \sim iid\ N(0, {\sigma^2}_w)$.

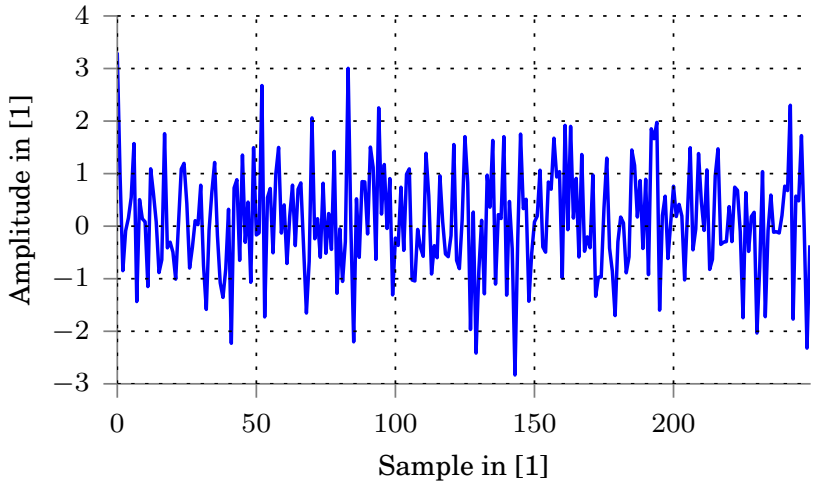

Figure 2.4.: An example time plot of Gaussian white noise

The usage of confidence intervals on the ACF helps to identify whether the sequence is accepted as white noise. The white noise sequence x_t, which consists of n samples, has mean approximately zero and its variance $\sigma^2_{\hat{\rho}_x}$ is defined as

$$\sigma^2_{\hat{\rho}_x} = \frac{1}{n}. \tag{2.8}$$

95% confidence interval is commonly used for similar tasks and its borders are calculated with $\pm 1.96\sigma_{\hat{\rho}_x}$. For a white noise sequence, 95% of the ACF $\hat{\rho}_x$ lies within these borders.

The white noise also has crucial importance since the statistical methods assume the collected time series contain white noise. Thus, the residuals of the statistical models, which are the difference between the approximation and real values, are expected to hold the characteristics of white noise in ideal case.

In general, the observed time series x_t is in the additive form

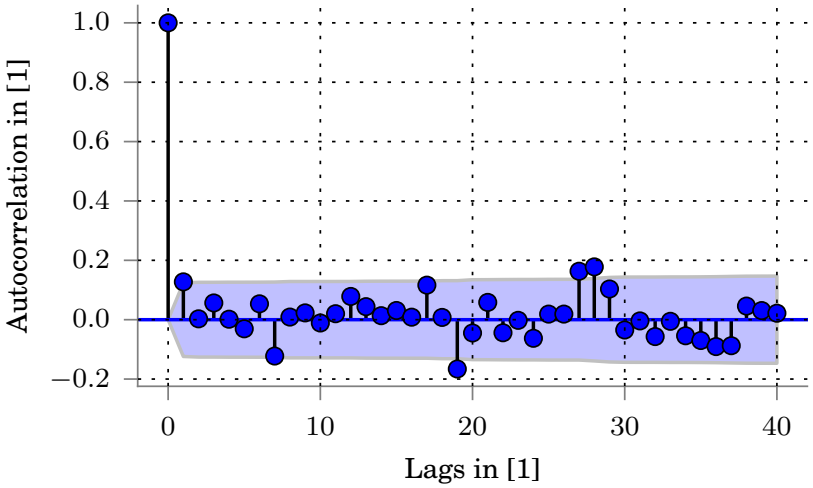

Figure 2.5.: The ACF of the Gaussian white noise in Figure 2.4

$$x_t = s_t + w_t, \tag{2.9}$$

where s_t is the time series produced by our model and w_t is the white noise. Estimating the form of s_t is the objective of the model building.

2.1.5. Bayesian Information Criterion

In order to build a successful model, it is highly important to choose the inputs of a model from significant and influential parameters. A criterion function can be used to decide between models with different parameters. The Bayesian Information Criterion (BIC) is a popular function used for similar tasks and it generally results in simpler and smaller models.

$$BIC = \ln \hat{\sigma}_k^2 + \frac{k \ln n}{n}, \tag{2.10}$$

where n is the number of samples, k is the number of parameters and $\hat{\sigma}_k^2$ is the estimator for the variance of the model and also interpreted as mean square error (MSE), which is defined as

$$\hat{\sigma}_k^2 = \frac{\sum_{t=1}^{n} (x_t - \hat{x}_t)^2}{n}, \tag{2.11}$$

where $\hat{x}_t$ is the output of the model with k parameters. The model with the smaller BIC is chosen [19].

2.1.6. Performance Criteria

The forecasting performances of different models are compared by normalized root mean square error (nRMSE) and normalized mean absolute error criteria (nMAE), which are defined as

$$nRMSE(x_t, \hat{x}_t) = \frac{\sqrt{\frac{\sum_{t=1}^{n}(x_{t,i}-\hat{x}_{t,i})^2}{n}}}{x_{max} - x_{min}}, \tag{2.12}$$

$$nMAE(x_t, \hat{x}_t) = \frac{\frac{1}{n}\sum_{t=1}^{n}|x_{t,i} - \hat{x}_{t,i}|}{x_{max} - x_{min}}, \tag{2.13}$$

where n is the number of samples, $\hat{x}_t$ is the predicted value, x_{max} and x_{min} are maximum and minimum points in the actual values x_t.

2.2. ARIMA Models

There are several ways of modeling a time series. Based on linear relations of lagged values in a time series, a systematic way to provide a model can be achieved via ARIMA models. The ARIMA models consist of autoregressive, integrated and moving average terms. These terms, which are chained together to build ARIMA, are explained in detail in Section 2.2.4. The ARIMA model is briefly presented below to give an idea about the following sections.

The ARIMA(p, d, q) model of a time series is written as

$$\theta(B)(1-B)^d x_t = \alpha + \theta(B) w_t, \tag{2.14}$$

where d is the order of differencing, w_t white noise and $\alpha = \mu(1 - \sum_{j=1}^{p} \phi_j)$ for $E[\nabla^d x_t] = \mu$. $\phi(B)$ and $\theta(B)$ are AR(p) and MA(q) operators, which are studied in Section 2.2.2 and 2.2.3.

In general, a time series with nonstationary components is firstly transformed to a stationary process via differencing. Then the mean μ of the differenced term is subtracted from the series in order to eliminate α and simplify calculations. After applying differencing and subtraction the obtained stationary process with mean zero is studied for ARMA(p, q) model [17, 19].

2.2.1. Differencing and Integrated Terms (I)

The dependence structure of a series needs to be consistent for every time point in order to build a model. It is especially important to estimate precise autocorrelations for a time series, which in general has dependence between the adjacent points. Thus, the mean and autocovariance functions are expected to satisfy stationarity conditions of a time series. The studied time series might be nonstationary as appears in most of the real cases. For that reason it is necessary to apply several methods on the nonstationary series to reduce it to a stationarity time series.

A time series, which increases or decreases over time has trends. The trends cause the time series to be nonstationary since the mean does not stay constant over time. In order to remove the trends from a time series, an appropriate method is differencing [17].

The first order of difference is denoted as

$$\nabla x_t = x_t - x_{t-1}, \tag{2.15}$$

which removes the linear trend from the time series x_t. Analogously, the second order of difference removes the quadratic trend and is denoted as $\nabla^2 x_t$.

Since the adjacent points in a time series have an importance for the analysis, there are some operators commonly used for time series analysis.

Shifting a time series is often required for the analysis, as seen in the differencing method. For that reason, simplifying terms such as x_{t-1} is helpful and makes it easier

to read chained operations. For instance, the backshift operator B reduces the footprint and is denoted as

$$B^k x_t = x_{t-k}, \tag{2.16}$$

where k is the order of shifting. Analogously, differencing is denoted as a function of the backshift operator B

$$\nabla^d x_t = (1 - B)^d x_t, \tag{2.17}$$

where d is the order of differencing.

The integrated ARMA, also known as ARIMA, is basically an ARMA model, which includes differencing.

A time series x_t is ARIMA(p, d, q), if

$$\nabla^d x_t = (1 - B)^d x_t \tag{2.18}$$

is ARMA(p, q).

The integrated ARMA model, also known as ARIMA(p, d, q), extends ARMA, so that a time series x_t with nonstationary components is firstly handled via differencing of order d. Then a stationary ARMA(p, q) process is studied.

2.2.2. Autoregressive Terms (AR)

The AR models assume that the current value of a time series x_t depends linearly on its p past values $x_{t_1}, x_{t_2}, \ldots, x_{t-p}$, where p is the order of the AR model and determines the number of past values required to forecast the current value .

An AR model of the order p, abbreviated as AR(p), of a stationary time series x_t with mean μ_x is formed as

$$x_t = \sum_{j=1}^{p} \phi_j x_{t-j} + w_t + \alpha, \tag{2.19}$$

where $\phi_j \neq 0$ are constants, w_t is Gaussian white noise with variance σ_w^2 and $\alpha = \mu_x(1 - \sum_{k=1}^{p} \phi_k)$. The term α helps to include the case, where the mean μ_x is not zero. To simplify calculations further, it is assumed that the mean μ_x is initially subtracted from x_t and thus $x_t = x_t - \mu_x$.

The AR(p) model of x_t with mean zero is further simplified for reading using several operators [18]. Initially AR(p) is also described as

$$(1 - \sum_{j=1}^{p} \phi_j B^j) x_t = w_t, \tag{2.20}$$

where B is the backshift operator. Introducing the autoregressive operator $\phi(B) = 1 - \sum_{j=1}^{p} \phi_j B^j$, the AR($p$) is shown as

$$\phi(B)x_t = w_t. \tag{2.21}$$

2.2.3. Moving Average Terms (MA)

The moving average (MA) models are alternative to AR models for modeling a time series, in which the time series x_t is formed from the linear combination of Gaussian white noise w_t and expressed as

$$x_t = w_t + \sum_{j=1}^{q} \theta_j w_{t-j}, \tag{2.22}$$

where $\theta_j \neq 0$ are constants. By introducing the moving average operator $\theta(B)$, x_t is equivalently expressed as

$$x_t = \theta(B)w_t, \tag{2.23}$$

where $\theta(B) = 1 + \sum_{j=1}^{q} \theta_j B^j$ [18].

2.2.4. Autoregressive Moving Average Models (ARMA)

The autoregressive moving average (ARMA) model combines the AR and MA models into one expression. The ARMA model of a stationary time series x_t with mean μ_x, which is abbreviated as ARMA(p, q), has the order (p, q) and is formed as

$$x_t = \alpha + \sum_{j=1}^{p} \phi_j x_{t-j} + \sum_{j=1}^{q} \theta_j w_{t-j}, \tag{2.24}$$

where $\theta_j \neq 0$, $\phi_j \neq 0$, $\alpha = \mu_x(1 - \sum_{j=1}^{p} \phi_j)$ analogous to AR(p) model and w_t is Gaussian white noise.

ARMA(p, q) can also be written as

$$\phi(B)x_t = \theta(B)w_t. \tag{2.25}$$

If $p = 0$, ARMA($0, q$) is equal to MA(q) and similarly if $q = 0$, then ARMA($p, 0$) is equal to AR(p).

AR, MA and ARMA models have three major problems combined, which should be avoided while building a model. These problems are studied below using polynomials since it is similar to work with operators like polynomials. The AR $\phi(B)$ and MA $\theta(B)$ operators represented as polynomials $\phi(z)$ and $\theta(z)$, where z is a complex number [17, 19].

Parameter redundancy For instance, an ARMA model might contain redundant parameters, which have to be canceled. To address this problem and obtain a truly representative model, the common factors of $\phi(z)$ and $\theta(z)$ are removed from the model [17, 19].

In addition to parameter redundancy, an ARMA model can also have the causality problem of AR models and invertibility problem of MA models, which are studied below.

Causality Causality problem can shortly be described as having an AR model which depends on the future values. In such a case, forecasting does not make sense since the model needs to know future values in order to forecast. An ARMA(p, q) model is causal, if the time series can be written as

$$x_t = \frac{\theta(B)}{\phi(B)} w_t = \sum_{j=0}^{\infty} \psi_j w_{t-j} = \psi(B) w_t, \tag{2.26}$$

where

$$\psi(B) = \frac{\theta(B)}{\phi(B)} = \sum_{j=0}^{\infty} \psi_j B^j, \tag{2.27}$$

$$\sum_{j=0}^{\infty} |\psi_j| < \infty, \quad \psi_0 = 1. \tag{2.28}$$

The causality is of ARMA(p, q) is assured for $\phi(z) \neq 0$ for $|z| \leq 1$. It can be induced that a causal ARMA(p, q), $\phi(z) = 0$, only when $|z| > 1$ [17, 19].

Invertability An ARMA (p, q) model is invertible, if the time series can be expressed as

$$\pi(B) x_t = \sum_{j=0}^{\infty} \pi_j x_{t-j} = w_t, \tag{2.29}$$

where

$$\pi(B) = \sum_{j=0}^{\infty} \pi_j B^j, \tag{2.30}$$

$$\sum_{j=0}^{\infty} |\pi_j| < \infty, \quad \pi_0 = 1. \tag{2.31}$$

The invertibility of an ARMA(p, q) model is detected by solving

$$\pi(z) = \sum_{j=0}^{\infty} \pi_j z^j = \frac{\phi(z)}{\theta(z)}, \quad |z| \leq 1, \tag{2.32}$$

which yields that $\theta(z) = 0$ only when $|z| > 1$.

To summarize, it should be paid attention that the ARMA model is causal, invertible and does not contain any redundant parameters [17].

2.2.5. Building ARMA Models

Identification of orders

The ACF and partial autocorrelation function (PACF) are used to find the order p, q of the ARMA(p, q) model. Both of them carry information about the dependence within the time series [18].

A causal ARMA(p, q) model of a mean zero time series x_t, $\phi(B)x_t = \theta(B)w_t$ can be written as

$$x_t = \sum_{j=0}^{\infty} \psi_j w_{t-j}. \tag{2.33}$$

The autocovariance function of this model is written as

$$\begin{aligned} \gamma(h) &= cov(x_{t+h}, x_t) \\ &= E[(\sum_{j=1}^{p} \phi_j x_{t+h-j} + \sum_{j=0}^{q} \theta_j w_{t+h-j})x_t] \\ &= \sum_{j=1}^{p} \phi_j \gamma(h-j) + \sigma_w^2 \sum_{j=h}^{q} \theta_j \psi_{j-h}, \quad h \geq 0, \end{aligned} \tag{2.34}$$

where the following equality is used in order to attain the second sum in the covariance function

$$E[w_{t+h-j}x_t] = E[w_{t+h-j} \sum_{k=0}^{\infty} \psi_k w_{t-k}] = \psi_{j-h}\sigma_w^2, \quad h \geq 0. \tag{2.35}$$

Once the autocovariance function $\gamma(h)$ is available, the ACF $\rho(h)$ is obtained by $\rho(h) = \gamma(h)/\gamma(0)$.

The ACF by itself is not enough to determine p since the adjacent terms in the time series are linearly dependent on each other. This causes the ACF to tail off for AR and ARMA processes. In order to attain p, the partial autocorrelation function (PACF) is used since it eliminates intermediate autocorrelations compared to the ACF [17, 19].

The **PACF** ϕ_{hh} is written as

$$\phi_{hh} = \begin{cases} corr(x_h - x_h^{h-1}, x_0 - x_0^{h-1}), & h \geq 2, \\ \phi_{11} = corr(x_1, x_0) = \rho(1), & h = 1, \end{cases} \tag{2.36}$$

	ACF	PACF
AR(p)	tail off	cut off after lag p
MA(q)	cut off after lag q	tail off
ARMA(p, q)	tail off	tail off

Table 2.1.: Behavior of the ACF and PACF for causal and invertible ARMA models [19]

where x_h^{h-1} is the minimum mean square predictor of x_h based on $\{x_{h-1}, \dots, x_1\}$ and denoted as

$$x_h^{h-1} = E[x_h | x_{h-1}, \dots, x_1] = \sum_{j=1}^{h-1} \beta_j x_{h-j}. \tag{2.37}$$

The PACF attains the correlation between two points x_t and x_{t-h} by removing the linear dependence of the points in between $\{x_{t-1}, \dots, x_{t-(h-1)}\}$ [17, 19].

Estimation of parameters

The model orders p, q are identified by evaluating the ACF and PACF. Assuming mean zero x_t is a causal and invertible ARMA(p, q) process with known p, q and w_t is white noise with variance σ_w^2, $\phi(B)x_t = \theta(B)w_t$, the goal is to estimate the model parameters $\{\phi_1, \dots, \phi_p\}$, $\{\theta_1, \dots, \theta_q\}$ and σ_w^2.

Maximum likelihood estimation is used to estimate the model parameters. The likelihood is written as

$$L(\boldsymbol{\beta}, \sigma_w^2) = \prod_{t=1}^{n} f(x_t | x_{t-1}, \dots, x_1), \tag{2.38}$$

where $\boldsymbol{\beta} = (\phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q)'$ is the vector of the unknown model parameters.

It is difficult to write the likelihood $L(\boldsymbol{\beta}, \sigma_w^2)$ explicitly, therefore it is written in terms of one-step-ahead prediction errors $x_t - x_t^{t-1}$. The conditional distribution is $x_t | x_{t-1}, \dots, x_1 \sim N(x_t^{t-1}, P_t^{t-1} = \sigma_w^2 r_t^{t-1})$, so the likelihood is also written as

$$L(\boldsymbol{\beta}, \sigma_w^2) = (2\pi\sigma_w^2)^{-n/2} [\prod_{j=1}^{n} r_j^{j-1}(\boldsymbol{\beta})]^{-1/2} \exp(-\frac{S(\boldsymbol{\beta})}{2\sigma_w^2}), \tag{2.39}$$

where

$$S(\boldsymbol{\beta}) = \sum_{t=1}^{n} \frac{(x_t - x_t^{t-1}(\boldsymbol{\beta}))^2}{r_t^{t-1}(\boldsymbol{\beta})}. \tag{2.40}$$

In order to simplify calculations for finding the maximum likelihood, the likelihood function is *log*ged and the constants in the likelihood are ignored, which results in the criterion function

$$l(\boldsymbol{\beta}) = \ln(n^{-1}S(\boldsymbol{\beta})) + n^{-1}\sum_{t=1}^{n}\ln(r_t^{t-1}(\boldsymbol{\beta})). \tag{2.41}$$

For the maximum likelihood, the criterion function $l(\boldsymbol{\beta})$ is minimized with respect to σ_w^2 and $\boldsymbol{\beta}$, so that the estimated parameters $\hat{\sigma}_w^2$ and $\hat{\boldsymbol{\beta}}$ are attained [17].

In general the minimization is provided by equating the partial derivates to zero. Nevertheless, it is difficult to obtain the derivates and therefore, numerical techniques are required to approximate them. As these numerical calculations are beyond the scope of this thesis, the matter is not discussed here and the existing solutions are used for the application.

Forecasting

The goal of this thesis is to forecast the future values of PV power. Provided that the time series consists of n values, $\boldsymbol{x} = \{x_n, x_{n-1}, \dots, x_1\}$, the goal is to predict x_{n+m}, where $m \geq 1$ is an integer forecast horizon. In other words, x_{n+m} is the m-step-ahead prediction based on the data $\boldsymbol{x}$.

It was shown in Section 2.2.5 that x_{n+m}^n is the minimum mean square predictor of x_{n+m} based on $\boldsymbol{x}$. For large n, $\tilde{x}_{n+m}^n$ is a good approximator of x_{n+m}^n based on the past values $\boldsymbol{x}$.

For an ARMA(p, q) with mean zero, $\phi(B)x_t = \theta(B)w_t$, where w_t is white noise, the causal and invertible forms of x_{n+m} are expressed as

$$x_{n+m} = \sum_{j=0}^{\infty}\psi_j w_{n+m-j} \quad \text{and} \quad \psi_0 = 1, \tag{2.42}$$

$$w_{n+m} = \sum_{j=0}^{\infty}\pi_j x_{n+m-j} \quad \text{and} \quad \pi_0 = 1. \tag{2.43}$$

Considering the fact that the time series is limited and consists of $\boldsymbol{x}$, the predictor for the available data $1 \leq t \leq n$ is defined as

$$\tilde{x}_t^n = \begin{cases} x_t & 1 \leq t \leq n, \\ 0 & t \leq 0, \\ \tilde{x}_{n+m}^n & t > n \end{cases} \tag{2.44}$$

similarly the predictor of the white noise w_t is given by

$$\tilde{w}_t^n = \begin{cases} 0 & t \leq 0, t > n, \\ \phi(B)\tilde{x}_t^n - \sum_{j=1}^{q}\theta_j\tilde{w}_{t-j}^n & 1 \leq t \leq n. \end{cases} \tag{2.45}$$

Based on the prediction of the invertible form 2.43, it is deducted that

$$\tilde{w}_{n+m}^{n} = \tilde{x}_{n+m}^{n} + \sum_{j=1}^{n+m-1} \pi_j \tilde{x}_{n+m-j} = 0, t > n, \tag{2.46}$$

therefore the predictor $\tilde{x}_{n+m}^{n}$ is also written as

$$\tilde{x}_{n+m}^{n} = -\sum_{j=1}^{m-1} \pi_j \tilde{x}_{n+m-j} - \sum_{j=m}^{n+m-1} \pi_j x_{n+m-j}, \quad t > n. \tag{2.47}$$

Then the prediction $\tilde{x}_{n+m}^{n}$ from Equation 2.47 is evaluated recursively as one-step-ahead predictor until desired prediction horizon, thereby starting with $m = 1$ and incrementing $m = 2, 3, ...$

The mean square prediction error P_{n+m}^{n} is given by

$$P_{n+m}^{n} = E[x_{n+m} - \tilde{x}_{n+m}^{n}]^2 = \sigma_w^2 \sum_{j=0}^{m-1} \psi_j^2, \tag{2.48}$$

where $x_{n+m} - \tilde{x}_{n+m}^{n}$ is found by using Equation 2.47 and written as

$$x_{n+m} - \tilde{x}_{n+m}^{n} = \sum_{j=0}^{m-1} \psi_j w_{n+m-j}. \tag{2.49}$$

The predictions are correlated and this leads to the correlation of the prediction errors, which is given by

$$E[(x_{n+m} - \tilde{x}_{n+m}^{n})(\tilde{x}_{n+m+k} - \tilde{x}_{n+m+k}^{n})] = \sigma_w^2 \sum_{j=0}^{m-1} \psi_j \psi_{j+k}. \tag{2.50}$$

Deriving from the results above, while the forecast horizon grows, $m \to \infty$,the ψ_m dampens to zero. Therefore, as $m \to \infty$, the forecasts of an ARMA model approaches the mean of x_t, $\mu_x = 0$ in the case. Furthermore, the mean square prediction error P_{n+m}^{n} converges to the constant $\sigma_w^2 \sum_{j=0}^{\infty} \psi_j^2$ [17]. The prediction intervals are helpful to indicate the precision of the forecast. In the ACF, 95% confidence intervals are provided by $\pm 1.96\sigma_w$ [17, 19]. Analogously, the prediction intervals are provided by

$$\tilde{x}_{n+m}^{n} \pm 1.96\sqrt{P_{n+m}^{n}}. \tag{2.51}$$

2.2.6. SARIMAX

By making some modifications to the ARIMA model, it is possible to include seasonal behaviour of the time series. The seasonality component of a time series can be explained by the dependence of the current data on the multiples of a seasonal lag *s*. For example,

an hourly captured data depends on the same hour of the previous day's. In this case, $s = 24$ would express the daily dependence.

The seasonal lags are captured with seasonal autoregressive $\Phi_P(B^s)$ and seasonal moving average $\Theta_Q(B^s)$ polynomials. An ARIMA(p, d, q) model, $\phi(B)\nabla^d x_t = \alpha + \theta(B)w_t$, is improved to capture seasonal dependencies, which results in the SARIMA model (p, d, q)x$(P, D, Q)_s$ and is given by

$$\Phi_P(B^s)\phi(B)\nabla_s^D\nabla^d x_t = \alpha + \Theta_Q(B^s)\theta(B)w_t, \tag{2.52}$$

where w_t is the white noise, $\phi(B)$ is the ordinary autoregressive polynomial of order p, $\theta(B)$ is the ordinary moving average polynomial of order q, $\Phi_P(B^s)$ is the seasonal autoregressive polynomial of order P, $\Theta_Q(B^s)$ is the seasonal moving average polynomial of order Q, $\nabla^d = (1 - B)^d$ is the ordinary difference component and $\nabla_s^D = (1 - B^s)^D$ is the seasonal difference component [17, 19].

In order to make forecasts with higher precision, it is advantageous to include exogenous inputs in the SARIMA model, which the time series depends on. For example, meteorological conditions, e.g. temperature, cloudiness, humidity etc, might provide information about PV power. Thus, it is necessary to improve our SARIMA model to include exogenous inputs [17].

The ARMAX model is based on the ARMA, which also uses exogenous inputs in the model in addition to the ARMA. The SARIMAX model is an extended version of the ARMAX, which takes into account seasonality and integrated terms, in order to better specify the model. The ARMAX model describes a time series, whose parameters are its lagged values, the lagged values of white noise, as well as the current and lagged values of exogenous inputs [17]. In other words, in the ARMAX model the forecasts of exogenous inputs are available in order to predict the PV power in our case.

The SARIMAX model (p, d, q)x$(P, D, Q)_s$ with a vector of exogenous inputs u_t is denoted as

$$\Phi_P(B^s)\phi(B)\nabla_s^D\nabla^d(x_t - \beta' u_t) = \Theta_Q(B^s)\theta(B)w_t, \tag{2.53}$$

where β is the vector of regression coefficients. In order to determine β, intermediate calculations are necessary. Considering the regression model

$$x_t = \beta' u_t + r_t, \tag{2.54}$$

an ordinary regression of x_t on u_t produces the residuals $\hat{r}_t = y_t - \hat{\beta}' u_t$. $\hat{r}_t$ is applied ARMA, $\hat{\phi}(B)\hat{r}_t = \hat{\theta}(B)w_t$, then polynomials $\hat{\phi}(B)$ and $\hat{\theta}(B)$ are applied to Equation 2.54, which transforms r_t to white noise w_t

$$c_t = \hat{\beta} d_t + w_t, \tag{2.55}$$

where

$$c_t = \frac{\hat{\phi}(B)}{\hat{\theta}(B)} x_t, \tag{2.56}$$

$$d_t = \frac{\hat{\phi}(B)}{\hat{\theta}(B)} u_t. \tag{2.57}$$

Ordinary least squares regression is applied to estimate $\hat{\beta}$ and this procedure is repeated until convergence [17].

2.3. ANN

Artificial neural networks are inspired by biological nervous systems to build nonlinear functional systems. The relationships between the inputs and outputs may not easily identifiable or the formulation of a mathematical model may be too complex. In such systems, the ANN is powerful for modeling based on the experimental data. The ANN is very flexible and powerful as compared to other approaches since it can learn the system from experimental data and does not require detailed information about the system. Due to these powerful features of the ANN, it is widely used in regression and classification problems [20].

2.3.1. Feedforward Neural Network

Feedforward neural network is a simple type of ANN, which maps inputs $\boldsymbol{x}$ to onto outputs $\boldsymbol{y}$. A feedforward neural network contains many interconnected nodes, which act as processing units. The nodes of a feedforward neural network are divided by the input layer, which collects the inputs $\boldsymbol{x}$ at input nodes, the hidden layer and the output layer, which produces the outputs $\boldsymbol{y}$ at the output nodes. In a feedforward neural network, the connections between the nodes are unidirectional and the information flows from the input layer to the output layer through hidden layers.

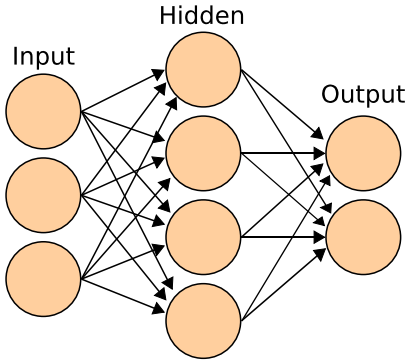

Figure 2.6.: Example feedforward neural network with 3 layers, 2 input nodes and 2 output nodes[1]

As seen in Figure 2.6, the nodes in each layer is fully connected to the nodes of the previous layer with a unidirectional connection. Each node in the layers receive weighted inputs and produces an output by a nonlinear activation function based on the sum of weighted inputs. The activation $\boldsymbol{a}^{(i)}$ of the nodes in layer i is calculated as

[1] Figure is taken from `https://upload.wikimedia.org/wikipedia/commons/e/e4/Artificial_neural_network.svg`

$$\boldsymbol{a}^{(i)} = g(\boldsymbol{z}^{(i)}), \tag{2.58}$$

$$\boldsymbol{z}^{(i)} = \boldsymbol{\Theta}^{(i-1)}\boldsymbol{a}^{(i-1)}, \tag{2.59}$$

where g is the activation function of the nodes, $\boldsymbol{\Theta}$ is the matrix of connection weights and $\boldsymbol{z}$ is the sum of products from activations and weights [21].

2.3.2. Training algorithm

The weights of an ANN are adjusted for a specific problem by a training algorithm. Backpropagation is a popular algorithm for training feedforward neural networks. A training data set, which consists of inputs $\boldsymbol{x}$ and desired outputs $\boldsymbol{y}$, is used to train a feedforward network iteratively. The weight coefficients $\boldsymbol{\Theta}$ are modified for every input-output pair. An iteration on one input-output pair is summarized with the following procedure:

1. Perform forward propagation for the inputs x_i.
2. Find the error at the output by $y_i - \hat{y}_i$.
3. Perform backward propagation and find the errors at layer j by
$$\delta^{(j)} = (\boldsymbol{\Theta}^{(j)})^T \delta^{(j+1)} .* g'(\boldsymbol{z}^{(j)}).$$
4. Modify the weights in the opposite direction of the backward propagation errors δ.

An iteration over all input-output pairs of the training set is called an epoch and the network is then trained many epochs until the validation error is minimized, which is obtained by processing the validation set on the current network [21].

As seen in the iteration procedure above, the backpropagation training algorithm requires a differentiable activation function. tanh is a continuous and differentiable function and used for all of the nodes in the application. The tanh function produces an output between -1 and 1 and is defined as

$$\tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{e^{2x} + 1}{e^{2x} - 1} \tag{2.60}$$

3. Analysis

3.1. Experimental Setup

The goal is to make a model of the PV power for forecasting. The available data for analysis is collected by different stations in Germany, which are geographically distanced from each other. The PV power data is collected by a household from 2013-08-01 to 2014-12-31, and the meteorological data is collected by Deutscher Wetterdienst (DWD) at these stations [1]. The frequency of data acquisition at the PV power station is 10 minutes, whereas at the DWD stations the data is sampled hourly.

Station data	Distance	Unit	Min	Max
Production	-	[W]	0	4382
Air temperature, humidity	20 km	[°C], [%]	-5.7, 18	35.5, 100
Cloudiness	10 km	[1]	0	8
Solar irradiance, sun angle	150 km	$[J/cm^2]$, [°]	0, 30.15	340, 149.8

Table 3.1.: Collected data information

3.2. Exploratory Analysis

Initially, time plots are observed in Figure 3.1 to start the analysis. As seen in the time plots, the PV power can be very volatile during the day due to the environmental conditions. Furthermore, this makes the calculations harder to work with data sets acquired with different frequencies. The PV power data is therefore smoothed by averaging hourly. With this technique, the average PV power data for hour $h-1$ is stored in hour h. It is important not to store any data for hour $h+1$ at hour h, otherwise there will be unintended correlations in the data, which cause to faulty models.

Secondly, lag scatter plots are shown in Figure A.1, which help to explore the data further. The lag scatter plots show the current PV power at time t versus its lagged values at time $t-h$. These scatter plots give a rough idea about the linear and nonlinear relationship between the PV power and its lagged values.

It is also worthwhile to examine the PV power by plotting lagged values of exogenous inputs versus the PV power. The resulting lag scatter plots will be useful to notice, which lags are significant for linear and nonlinear relationships.

[1] DWD data is taken from `ftp://ftp-cdc.dwd.de/pub/CDC/observations_germany/climate/hourly`

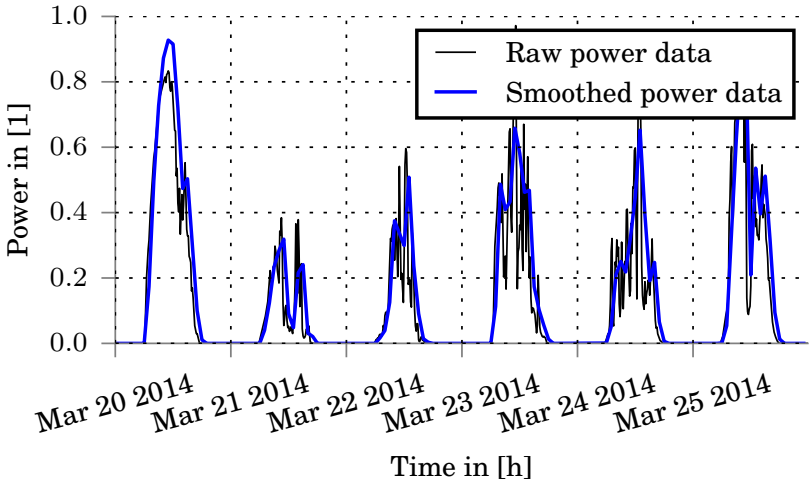

Figure 3.1.: Raw and smoothed PV power output, scaled from 0 to 1

In Figure A.2, the lag scatter plots show the relationship between the PV power at time t and the lagged values of air temperature at time $t - h$. It can be observed that there is a positive relationship for $h = 0$. In addition, for $h = 0$ the points are close to each other, whereas the points spread as the lag h grows.

Figure A.3 illustrates the relationship between the PV power and the lagged values of humidity, analogously. It can be observed that the relationship is negative for $h = 0$ and $h = 1$.

The scatter plot of the PV power and cloudiness in Figure A.4 is different from the other scatter plots since the cloudiness measurements are integer values between $0 - 8$. Therefore, the measurement for cloudiness is not precise enough for further usage. According to the plots, there is no obvious linear relationship between cloudiness and PV power.

The scatter plot for sun angle as depicted in Figure A.5 shows a negative pattern. As compared to the other exogenous inputs, the scatter plot for sun angle has the points close to each other, and the graph seems like a phase shift as the lag h grows. This indicates that the sun angle changes in a consistent manner, as expected.

As seen in Figure A.6, there is a strong positive relationship between solar irradiance and PV power for $h = 0$ and $h = 1$. The current and one hour previously measured solar irradiance could be used as a factor to forecast the PV power with a horizon of 1 hour.

The relationship between the PV power and exogenous inputs can be analyzed further by finding the correlation. In this case, however, instead of finding a scalar for the whole time series, the correlations are calculated for 48 hour window of data set. By having such a rolling correlation plot, it can be seen whether the correlation is volatile throughout the time series.

According to Figure 3.2, the correlation between the current and previous sample of PV power is very strong but varying throughout the year. The variability of this correlation is a sign of increasing complexity of fitting the data to the whole data set.

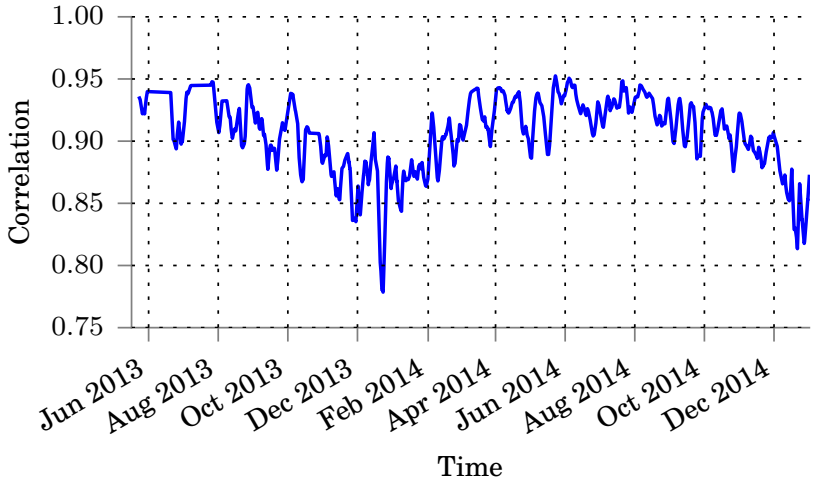

Figure 3.2.: Rolling correlation over 48 hours of PV power and its 1 hour lagged values

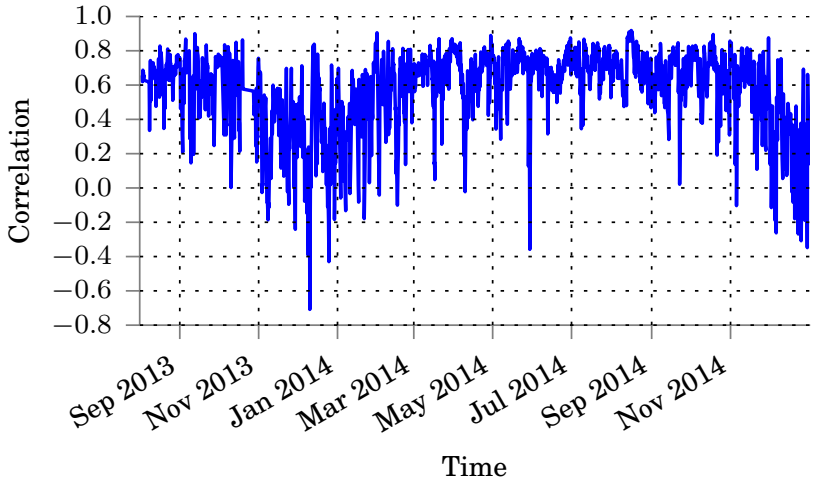

Figure 3.3.: Rolling correlation over 48 hours of PV power and air temperature

In Figure 3.3, the correlation between PV power and air temperature is plotted. The plot

shows that the correlation is on average above 0.6. Nevertheless the correlation decreases and is close to zero during winter.

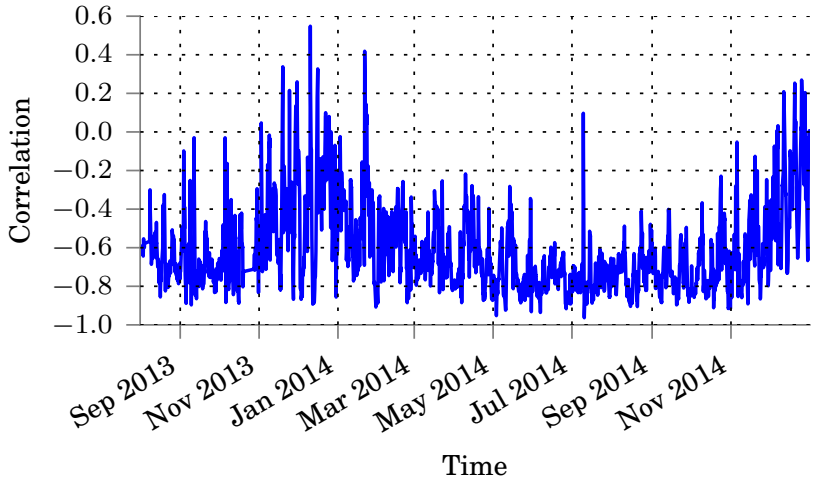

Figure 3.4.: Rolling correlation over 48 hours of PV power and humidity

Figure 3.4 shows that the correlation between PV power and humidity is negative most of the time and on average 0.6. Again, the correlation approaches zero during winter.

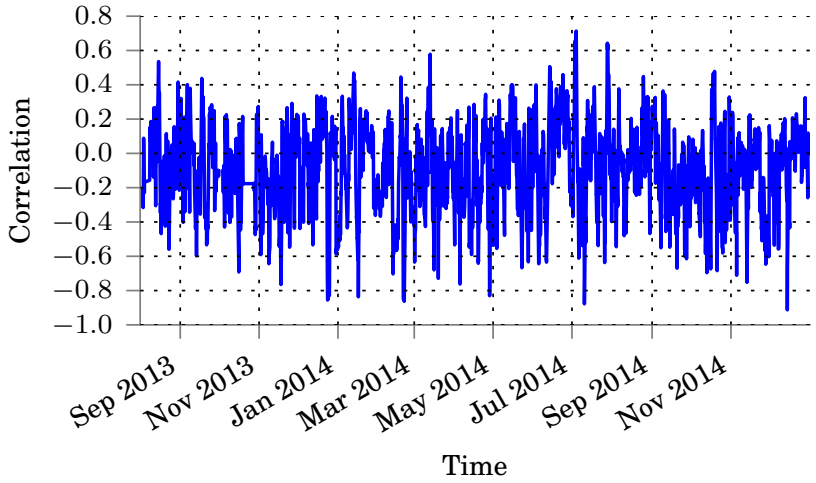

Figure 3.5.: Rolling correlation over 48 hours of PV power and cloudiness

As seen in Figure 3.5, the correlation between PV power and cloudiness is almost zero

throughout the set. From this plot, it can be concluded that cloudiness does not carry significant information for a linear model.

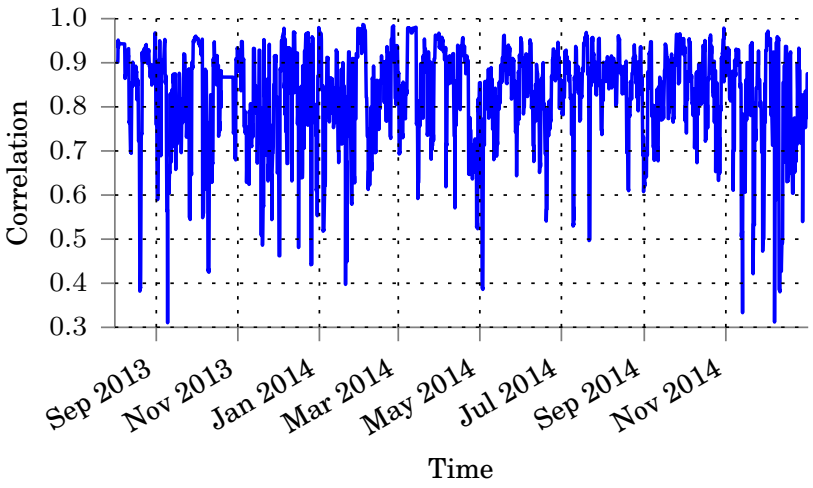

Figure 3.6.: Rolling correlation over 48 hours of PV power and solar irradiance

Figure 3.6 shows that the correlation between PV power and solar irradiance is very high and around 0.8 on average. This correlation plot does not show any indication of approaching zero during winter. Nevertheless, it can be seen that the variability of the correlation is very high. Thus, it is questionable whether solar irradiance can be used as a reliable source for forecasting.

The correlation between PV power and sun angle is negative, as expected, which is shown in Figure 3.7. The plot shows that the correlation drops during winter.

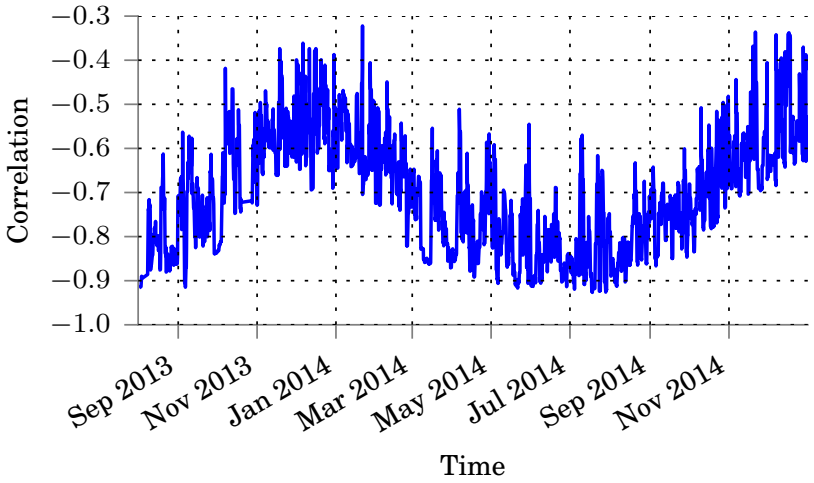

Figure 3.7.: Rolling correlation over 48 hours of PV power and sun angle

4. Results

In this chapter, modeling and forecasting the PV power are explained. Three different methods are preferred for modeling PV power. These are SARIMA, SARIMAX and ANN models.

The models are compared by their forecasting performances up to 24 steps ahead. The performance criteria used in this work are nRMSE and nMAE. Since nRMSE and nMAE are representations of the ratio of prediction errors, the model with lower ones achieves higher performance.

The forecasts are based on dynamic predictions. In a dynamic prediction, each forecasting provides 1-step-ahead prediction, and the obtained forecast is used for further step-ahead predictions.

4.1. Modeling

In order to handle non-constant variability and simplify interpretations, the PV power and exogenous inputs are scaled between 0 and 1 by using

$$\frac{x_t - x_{min}}{x_{max} - x_{min}}, \tag{4.1}$$

where x_{min} and x_{max} are the minimum and maximum points of the time series x_t, respectively.

4.1.1. SARIMA

For modeling, the training data is chosen from a small section of the whole data set. It is important to note that fitting a model for the whole time series is complex, and it is suggested to fit a model based on training data of shorter time series. Besides, SARIMA failed to fit the whole data set in an experiment.

The following results are based on the data from 2014-05-15 to 2014-06-15.

According to Figure 4.1, the ACF of the PV power has a sinusoid shape, which attenuates very slowly. Slow attenuation is an indication of nonstationarity, therefore differencing is required to remove the trends. It can be seen that the slow attenuation occurs in the seasonal and nonseasonal manners. Considering the seasonal attenuation, the sinusoid has a period of 24 samples, which is an indication of the seasonal term. Thus, differencing of 24 backshifted samples will help to remove seasonal trends in this context. In addition, differencing is also required to remove nonseasonal trends.

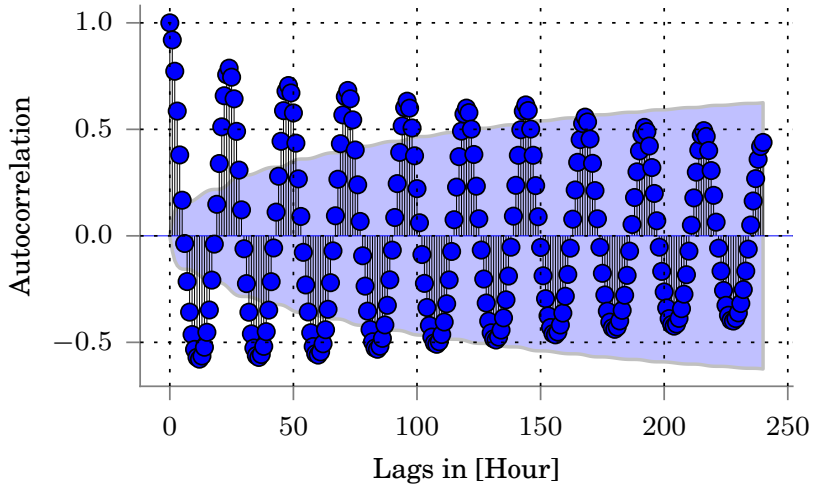

Figure 4.1.: The ACF of the smoothed PV power, blue filled region depicts 95% confidence interval

The raw PV power values are differenced once with nonseasonal integration order $d = 1$ and seasonal integration order $D = 1$ with seasonal term $s = 24$. The applied differencing is mathematically expressed as

$$(1 - B)(1 - B^{24})x_t, \tag{4.2}$$

where x_t is the PV power. After differencing, the ACF of the differenced values is inspected to ensure stationarity.

According to Figure 4.2, the nonstationary terms are removed via differencing since slow attenuation does not exist in the ACF anymore. As the seasonal and nonseasonal integration orders are determined, it is now required to further analyze the differenced values to find the autoregressive p and moving average q orders of a SARIMA process $(p, d = 1, q)x(P, D = 1, Q, s = 24)$. The PACF is shown, which is also required to identify the orders.

In Figure 4.2, the ACF has a peak at the lag $l = 24$. When the PACF in Figure 4.3 is observed, it is easy to recognize that the peaks occur at the multiples of the the seasonal term $s = 24$, which tail off. According to Table 2.1, a moving average process with order q cuts off at q of the PACF and tails off in the ACF. Therefore, it is attained that the differenced time series is a moving average process with order $Q = 1$. It can be noted that there is no indication of seasonal autoregressive terms for this process, so $P = 0$.

To determine the nonseasonal orders, the first lags in the ACF and PACF are analyzed, which are plotted in Figure 4.4 and 4.5. The first lags of the ACF and PACF seem to tail off, which is an indication of an ARMA process according to Table 2.1. Nevertheless, the

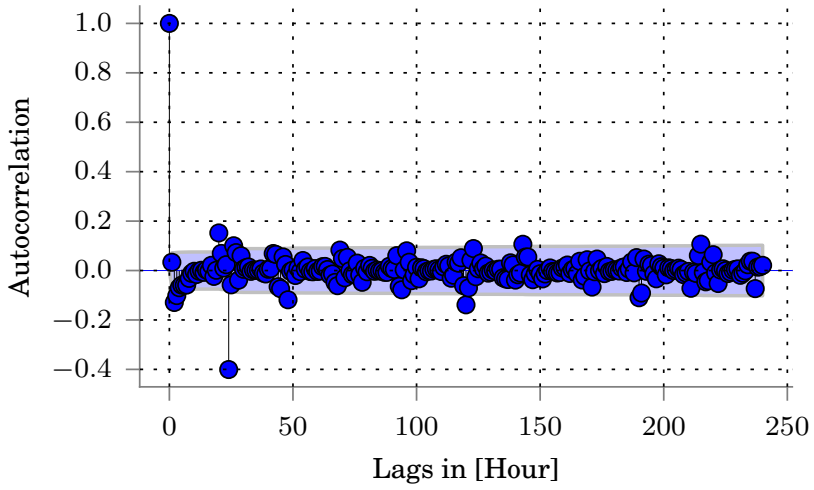

Figure 4.2.: The ACF of the differenced smoothed PV power at lags $l = \{1, 24\}$

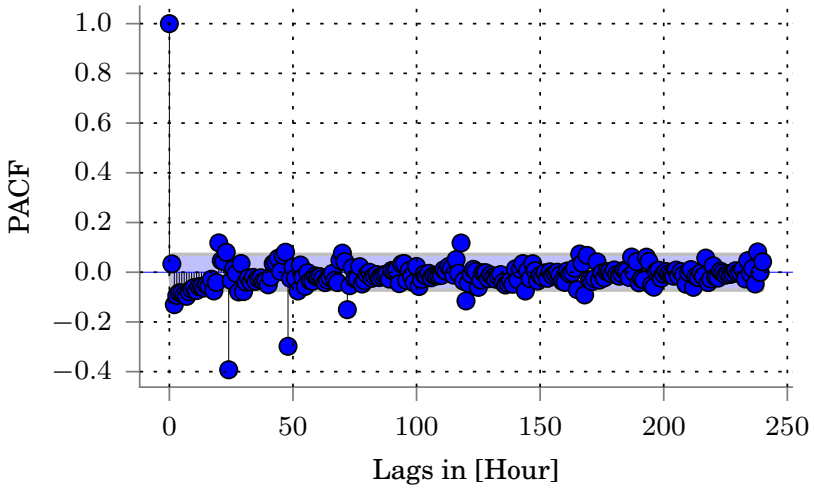

Figure 4.3.: The PACF of the differenced smoothed PV power at lags $l = \{1, 24\}$

behaviour of the first lags is not obvious and open to judgment. Therefore, several models are fit with different orders and their criterions are compared to identify the satisfying order.

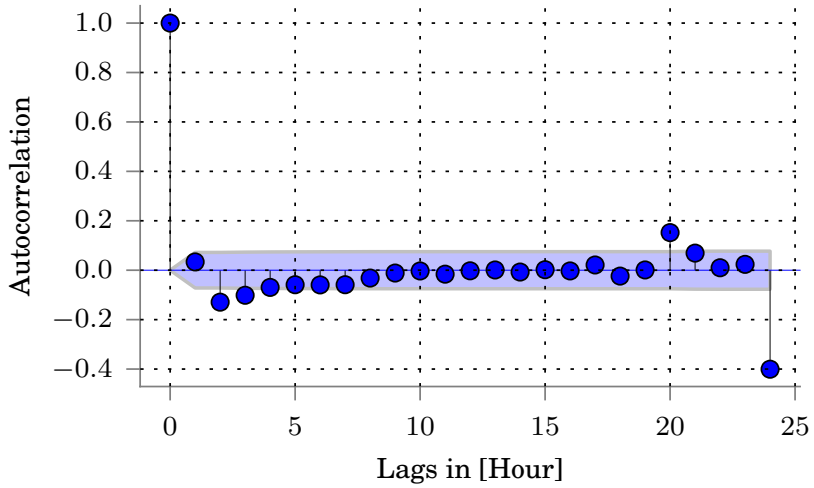

Figure 4.4.: The ACF of the differenced smoothed PV power at lags $l = \{1, 24\}$

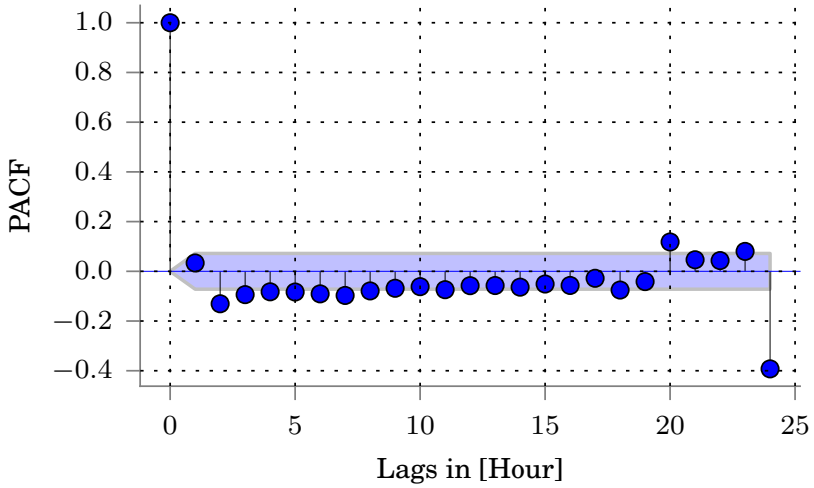

Figure 4.5.: The PACF of the differenced smoothed PV power in Figure 3.1 at lags $l = \{1, 24\}$

The model orders p, q in inspection are the combinations of $p = \{0, 1, 2\}$ and $q = \{0, 1, 2\}$.

p	q	BIC	σ^2
0	0	-1415	0.007
0	1	-1410	0.007
0	2	-1408	0.007
1	0	-1410	0.007
1	1	-1460	0.007
1	2	-1466	0.006
2	0	-1408	0.007
2	1	-1467	0.006

Table 4.1.: List of criterion results of SARIMA model fitting with p, q combinations

The information about the model with $p = 2, q = 2$ is not available since the model parameters results in a non-invertible model.

The model order selection can be done via comparing the variance of the residuals σ^2, which can be interpreted as errors. Smaller variance of errors is preferred for a more consistent model. The other criterion used is BIC as mentioned in Equation 2.10. Smaller BIC indicates a compact and better performing model.

A crucial criterion of model selection is to obey parsimony, so that overfitting is avoided. It is possible that the variance of residuals and BIC of different models are very close to each other. In that case, selecting the model with smaller orders help to avoid overfitting.

According to the results of the model fittings available in Table 4.1, the models with ($p = 1, q = 1$), ($p = 1, q = 2$) and ($p = 2, q = 1$) have relatively smaller BIC and σ^2 in comparison to the other models and have similar values. To obey parsimony, the chosen model is $(1, 1, 1)x(0, 1, 1)_{24}$.

After a model is fit to the data, its adequacy is examined through residual analysis. The residuals are obtained by the subtraction of the model values from the input data. The adequacy of the model is ensured, if the time series is transformed to white noise. Therefore, the residuals are expected to behave like white noise.

As seen in Figure 4.6 and 4.7, the correlations lie mostly between 95% confidence intervals. There are a few outliers, which can be omitted.

ϕ_1	0.8298
θ_1	-0.9999
Θ_1	-1.0000
σ^2	0.007
nRMSE	0.088
nMAE	0.052

Table 4.2.: Model parameters for SARIMA $(1, 1, 1)x(0, 1, 1)_{24}$

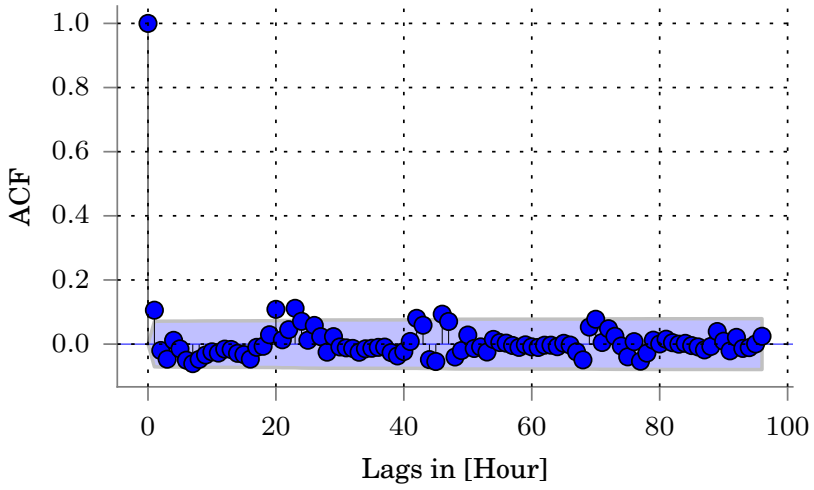

Figure 4.6.: The ACF of the residuals of the model $(1,1,1)x(0,1,1)_{24}$

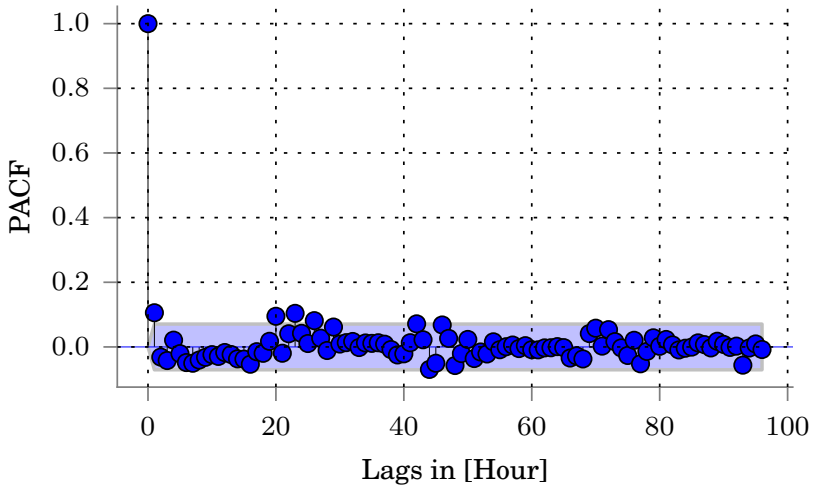

Figure 4.7.: The PACF of the residuals of the model $(1,1,1)x(0,1,1)_{24}$

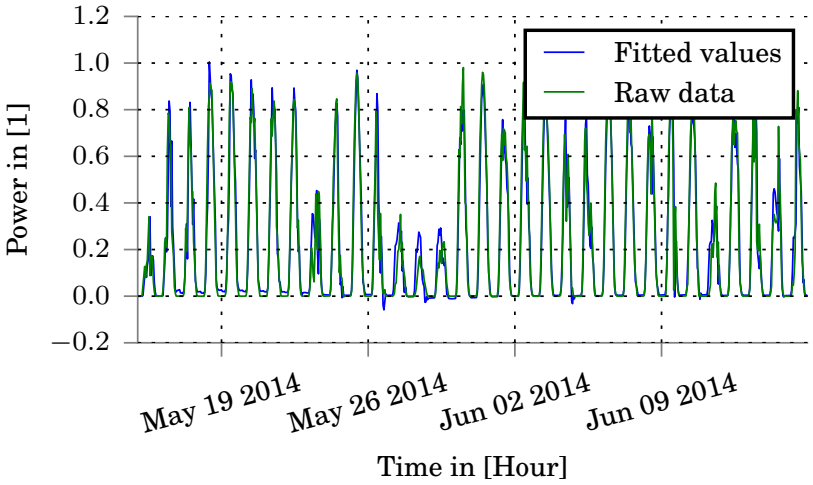

Figure 4.8.: Time plot of the fitted model $(1,1,1)x(0,1,1)_{24}$ and raw production data

Forecasting Result

The model is trained with the PV power data from 2014-05-15 to 2014-06-15. The test dates are from 2014-06-16 to 2014-07-01.

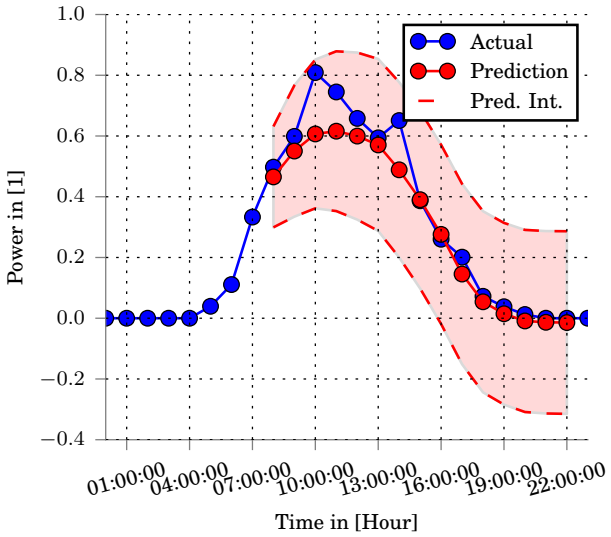

Figure 4.9.: 12-step-ahead dynamic forecasts with the selected model $(1,1,1)x(0,1,1)_{24}$ on 2014-06-27

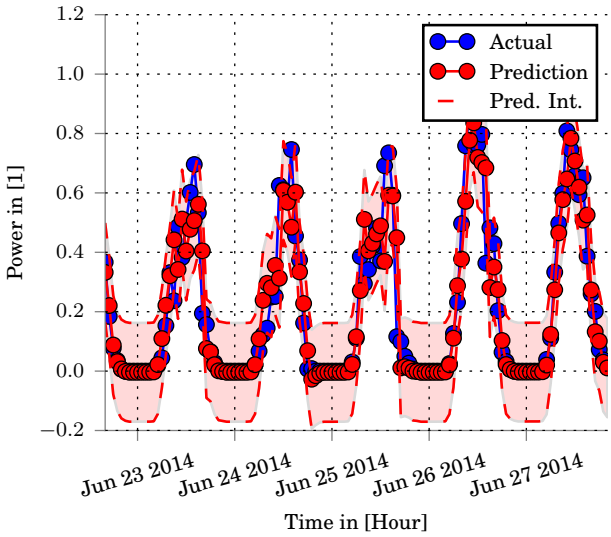

Figure 4.10.: One-step-ahead forecasts with the selected SARIMA model $(1,1,1)x(0,1,1)_{24}$

In Figure 4.9, forecasts are calculated dynamically for 14 hours. Figure 4.10 demonstrates one-step-ahead forecasts for the dates from 2014-06-23 to 2014-06-27. The performance criteria nRMSE and nMAE for multiple-step-ahead forecasts are listed in Table 4.4. nRMSE is between 9.8% − 13.5% and nMAE is between 5.6% − 8.4%. Both of the results are satisfying for a model, which does not have information about the forecasts of exogenous inputs.

4.1.2. SARIMAX

The model fitting can be improved by introducing exogenous inputs, such as meteorological data. The SARIMAX model handles exogenous inputs by building a linear regression based on the exogenous inputs. Then the regression errors are fit by a SARIMA process. As seen in the scatter plots in Chapter 3, the linear relationship between the PV power and exogenous inputs is not very strong. Therefore, for model fitting, SARIMAX may have similar results to SARIMA without exogenous inputs. Nevertheless, SARIMAX can have less errors of prediction owing to available forecasted exogenous inputs. It should be noted, however, that solar irradiance cannot be used as exogenous input for SARIMAX since forecasted values of solar irradiance are not available.

It is important to note that due to the different set of inputs, the BIC of this model cannot be compared with the model from Section 4.1.1. The models are compared with different set of exogenous inputs and model orders and the model with minimal BIC is selected. In Chapter 3, it is seen that the humidity and cloudiness do not have significant linear relationship with the PV power and thus resulted in higher BIC and insignificant coeffi-

cients. Accordingly, the selected SARIMAX model contains air temperature and sun angle as exogenous inputs. The selected model is $(1, 0, 0)x(1, 0, 1)_{24}$.

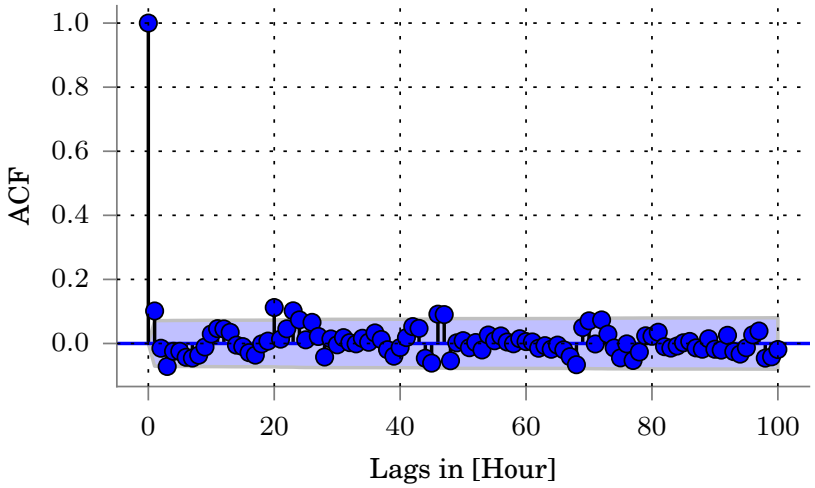

Figure 4.11.: The ACF of the residuals of the model $(1, 0, 0)x(1, 0, 1)_{24}$ with air temperature and sun angle as exogenous inputs

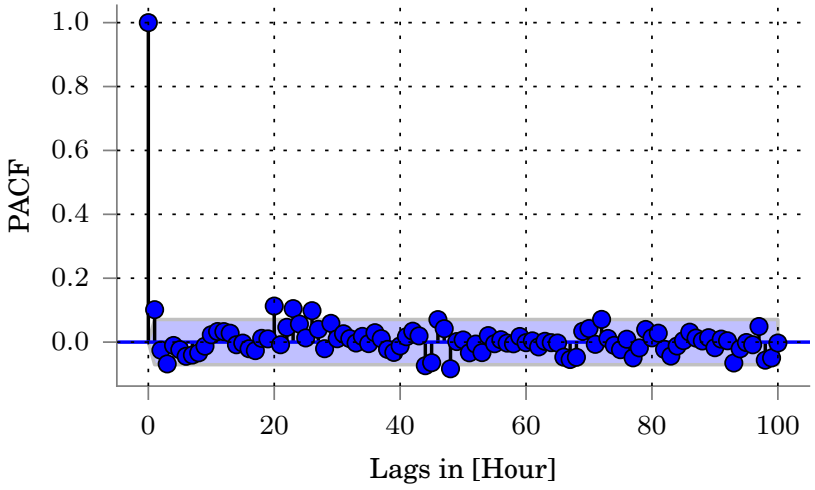

Figure 4.12.: The PACF of the residuals of the model $(1, 0, 0)x(1, 0, 1)_{24}$ with air temperature and sun angle as exogenous inputs

As seen in Figure 4.11 and Figure 4.12, the ACF and PACF of residuals do not contain significant peaks outside confidence intervals.

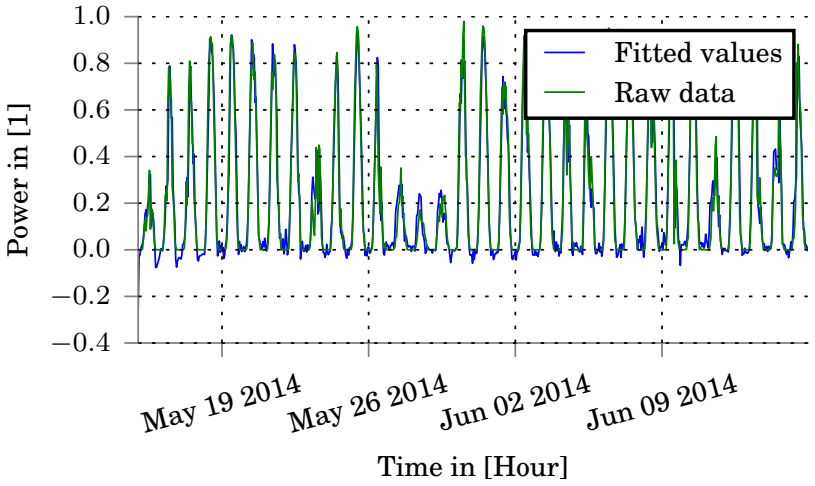

Figure 4.13.: Time plot of the fitted SARIMAX $(1,0,0)x(1,0,1)_{24}$ with exogenous inputs and raw production data

air temperature	0.8396
sun angle	-0.6845
ϕ_1	0.8328
Φ_1	0.9990
Θ_1	-0.9731
σ^2	0.0065
nRMSE	0.083
nMAE	0.053

Table 4.3.: Model parameters for SARIMAX $(1,0,0)x(1,0,1)_{24}$ with exogenous inputs

Forecasting Result

In Figure 4.14, forecasts are calculated dynamically for 14 hours. In Figure 4.15, one-step-ahead forecasts are available for the dates from 2014-06-23 to 2014-06-27. The performance criteria nRMSE and nMAE for multiple-step-ahead forecasts are available in Table 4.4.

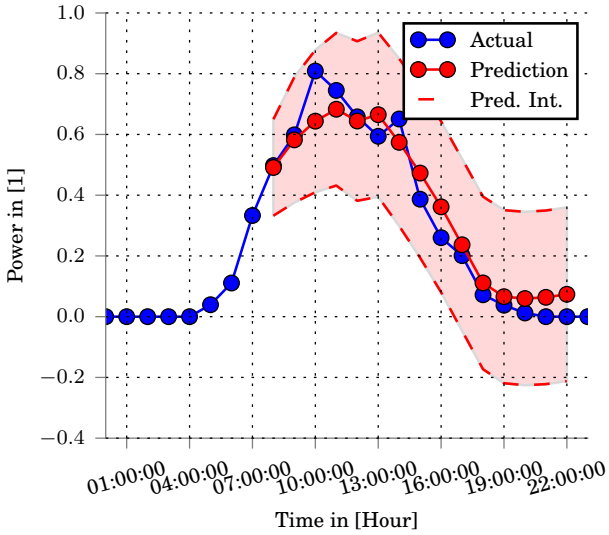

Figure 4.14.: 14-step-ahead dynamic forecasts with the selected model $(1, 0, 0)x(1, 0, 1)_{24}$ on 2014-06-27

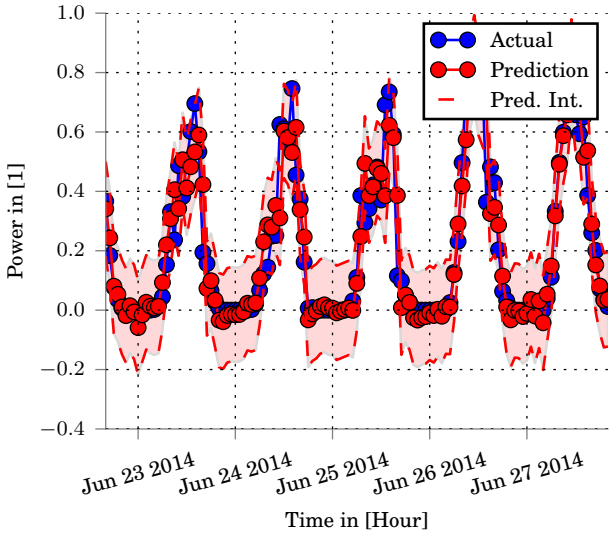

Figure 4.15.: One-step-ahead forecasts with the selected SARIMAX model $(1, 0, 0)x(1, 0, 1)_{24}$

4.1.3. ANN

In order to build a neural network model for PV power, the parameters to be determined are the number of input nodes, the number of nodes in the hidden layer and the transfer function of the hidden nodes. These parameters are determined by trial and error. The number of input nodes depends on which exogenous inputs are included and how many recent samples of exogenous inputs and PV power is required to predict the next PV power. After trying for each combination of number of input and hidden nodes, a neural network model is selected by comparing the nRMSE of the fitted model. The model based on 4 hours of past data, and air temperature and sun angle as exogenous inputs with 4 hidden nodes results in the lowest error.

Forecasting Result

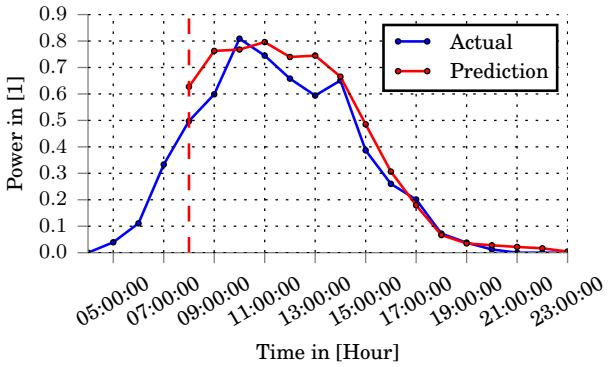

Figure 4.16.: 16-step-ahead dynamic forecasts with the selected neural network model on 2014-06-27

Figure 4.16 demonstrates the dynamical forecasts for 16 hours. In Figure 4.17, one-step-ahead forecasts are available for the dates from 2014-06-23 to 2014-06-27. The performance criteria nRMSE and nMAE for multiple-step-ahead forecasts are listed in Table 4.4.

According to Table 4.4, the SARIMA model has overall the highest nRMSE and nMAE, therefore, the lowest accuracy among all models. As compared to the errors of the SARIMA and SARIMAX models, the ANN model has higher forecasting accuracy due to lower errors.

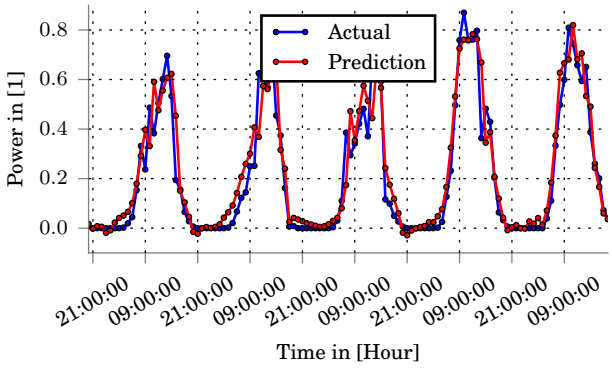

Figure 4.17.: One-step-ahead forecasts with the selected neural network model

Model	SARIMA		SARIMAX		ANN	
Steps-Ahead	nRMSE	nMAE	nRMSE	nMAE	nRMSE	nMAE
1	0.098607	0.055318	0.093163	0.056303	0.085051	0.052914
2	0.114780	0.071805	0.105359	0.070516	0.093544	0.062900
3	0.120626	0.080542	0.108471	0.076679	0.097168	0.065595
4	0.130312	0.087975	0.115296	0.082006	0.100131	0.067555
5	0.134650	0.091933	0.116946	0.083905	0.099200	0.067706
6	0.135951	0.093425	0.116250	0.083683	0.098502	0.068366
7	0.136039	0.094376	0.115017	0.082810	0.099145	0.070292
8	0.135754	0.094431	0.113729	0.082342	0.099185	0.071232
9	0.134858	0.093317	0.112717	0.081298	0.098394	0.071368
10	0.134306	0.092741	0.112216	0.079863	0.098127	0.071308
11	0.133852	0.091985	0.111935	0.079133	0.097636	0.070793
12	0.133266	0.090719	0.111504	0.078544	0.097381	0.070373
13	0.131475	0.088798	0.110569	0.077345	0.097194	0.069995
14	0.130065	0.086781	0.110140	0.076928	0.097284	0.069897
15	0.128056	0.084864	0.109227	0.076074	0.097396	0.069834
16	0.126759	0.083481	0.108814	0.075855	0.097530	0.069824
17	0.125536	0.082224	0.108429	0.075623	0.097674	0.069880
18	0.125003	0.081423	0.108475	0.075725	0.097814	0.069996
19	0.124702	0.080925	0.108439	0.075898	0.097955	0.070112
20	0.124792	0.080808	0.108522	0.076078	0.098087	0.070266
21	0.124875	0.080807	0.108647	0.076270	0.098213	0.070441
22	0.125274	0.081029	0.108919	0.076553	0.098330	0.070640
23	0.125697	0.081508	0.109142	0.076753	0.098425	0.070819
24	0.126348	0.082067	0.109482	0.077014	0.098524	0.071006

Table 4.4.: nRMSE and nMAE performances of the SARIMA, SARIMAX and ANN models according to multiple-step-ahead dynamical forecasts

5. Discussion

As seen in Figure 4.9, SARIMA model predicts a time series by grabbing its general features and produces smooth outputs as dynamical predictions. This is an expected behaviour since the SARIMA process is based purely on historical data and does not contain any information about the future of the environmental factors, which have impact on PV power. Due to that reason, it can be claimed that the SARIMA model is not the best candidate, which reacts to environmental changes.

On the other hand, SARIMAX model additionally makes use of meteorological forecasts. In Figure 4.14, it is visible that the SARIMAX process can produce volatile outputs in accordance with the forecasts. By comparing the nRMSE and nMAE values of SARIMA and SARIMAX in Table 4.4, it is seen that SARIMAX performs overall better than SARIMA, as expected. Nevertheless, it can be said that the errors are still considerably high despite the exogenous inputs. One possible reason is that the SARIMAX cannot make use of exogenous inputs at their highest potential since its intrinsic linear regression is inadequate to represent the PV power in terms of exogenous inputs. The other possibility is that the exogenous inputs are not adequate at its best to represent the PV power. In that case, either more inputs are required for the model or the quality of the exogenous inputs should be increased by moving the meteorological measurements closer to the PV power station.

SARIMA and SARIMAX cannot fit a single model for a long time series, the whole time series data in this case. The reason is that model fitting requires a lot of manual interrogation and analysis of the ACF and PACF. At its best, if the model parameters are used repetitively for different parts of the time series, the model parameters should be re-estimated. Due to that reason, it is claimed that SARIMA and SARIMAX are good model candidates, where the model can be fit adequately based on the recent observations regardless of the variability and cyclic behaviours in the long term. Therefore, it is suggested to implement time-varying SARIMA and SARIMAX models, which re-estimates the model parameters for the long term usage.

Neural network model produced overall better results than SARIMA and SARIMAX, as seen by comparing the nRMSE and nMAE for different models in Table 4.4. The reason of better performance can be explained by the fact that neural networks are good at capturing nonlinear relationships among PV power and exogenous inputs. The other advantage of the neural networks is that the training process requires less interrogations of the user for the long term model fitting in comparison with the SARIMA and SARIMAX models. Due to its nature, having more training data improves the model and it learns to represent the time series better by itself. For that reason, re-estimation of the model parameters is not required for the long term usage.

Another improvement suggestion is to include exogenous inputs, which are measured at the same frequency as the PV power. In this work, the available PV power is collected every 10 minutes, whereas the sampling rate of the exogenous inputs is 1 hour. Due to that reason, the PV power is averaged hourly and upsampled to 1 hourly spaced data. Nevertheless, the available exogenous inputs consist of hourly measured data instead of hourly averaged data.

6. Conclusion

Statistical and neural network models are developed and implemented in this work to predict PV power up to 24 hours ahead. The goal is to achieve high precision in forecasts with the available data by implementing and comparing different time series models. For this purpose, PV power of a station is initially analyzed to recognize its characteristics and relationships with meteorological data. Then according to the analysis, several data-driven models are based on the past PV power data and meteorological data.

According to the analysis, it can seen that the available PV power data is volatile during daylight, therefore it is averaged hourly to obtain a smoother time series. Then the correlation of the PV power is analyzed by ACF and PACF and it is concluded that the PV power is correlated with its lagged values and also has has seasonal daily behaviour.

SARIMA is a good candidate of modeling PV power, due to its nature of representing seasonal and on-the-past dependant data. A good selected SARIMA captures the main characteristics of the PV power and produces forecasts regardless of the environmental changes. The disadvantage of SARIMA model is that it is unaware of the environmental factors, which have impact on the PV power.

SARIMAX is an improved version of SARIMA and includes exogenous inputs in the model, which contains predicted information. SARIMAX performs better than SARIMA, since PV power depends highly on the environmental factors. As compared to SARIMA, SARIMAX can produce more volatile results in accordance with the exogenous inputs. Nevertheless, it makes use of exogenous inputs based only on linear regression and the relationship between the PV power and exogenous inputs can be nonlinear.

SARIMA and SARIMAX are inadequate to fit the training data of PV power, which includes long term cyclic behaviours. Due to that reason, SARIMA and SARIMAX are fit for recent observations.

Neural network model can be fed with long term training data, which improves its results, too. As compared to SARIMA and SARIMAX, neural networks perform better at predicting PV power due its ability to represent nonlinear relationships as well.

A. Appendix

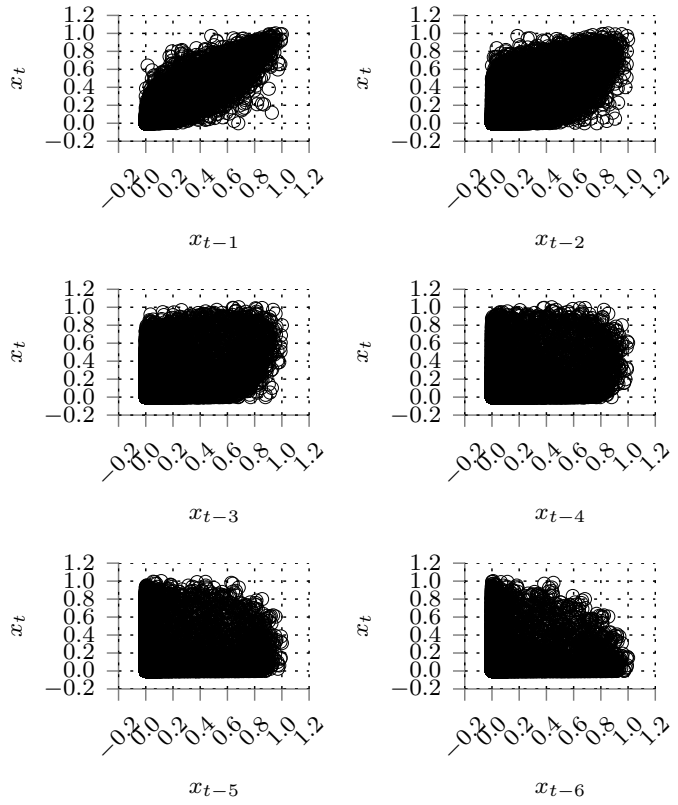

Figure A.1.: Scatter plot of current photovoltaic production x_t to past production values x_{t-h}

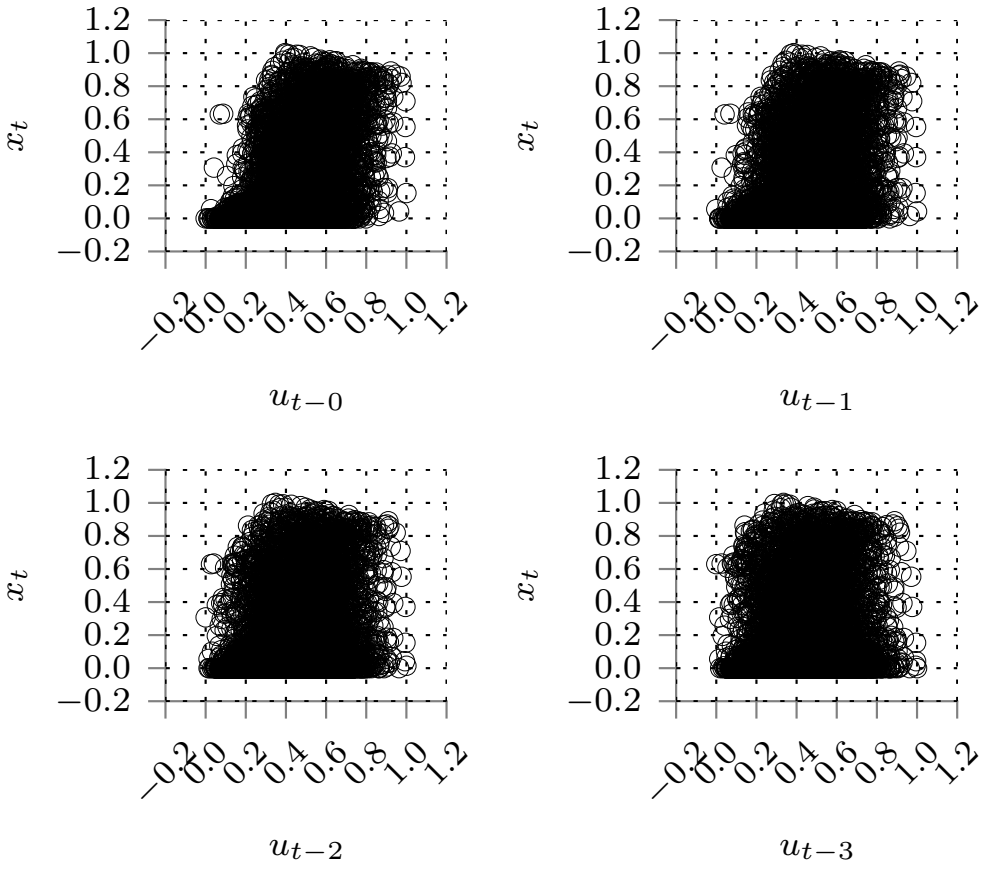

Figure A.2.: Scatter plot of current photovoltaic production x_t and air temperature u_{t-h}

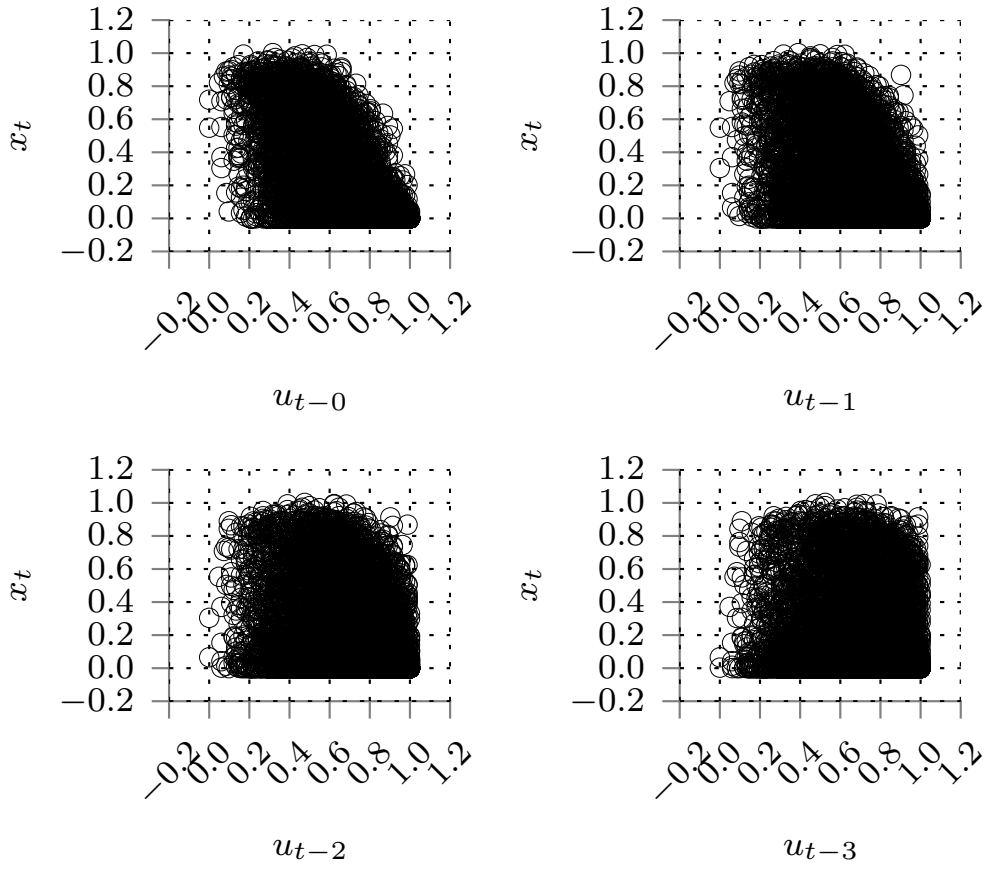

Figure A.3.: Scatter plot of current photovoltaic production x_t and humidity u_{t-h}

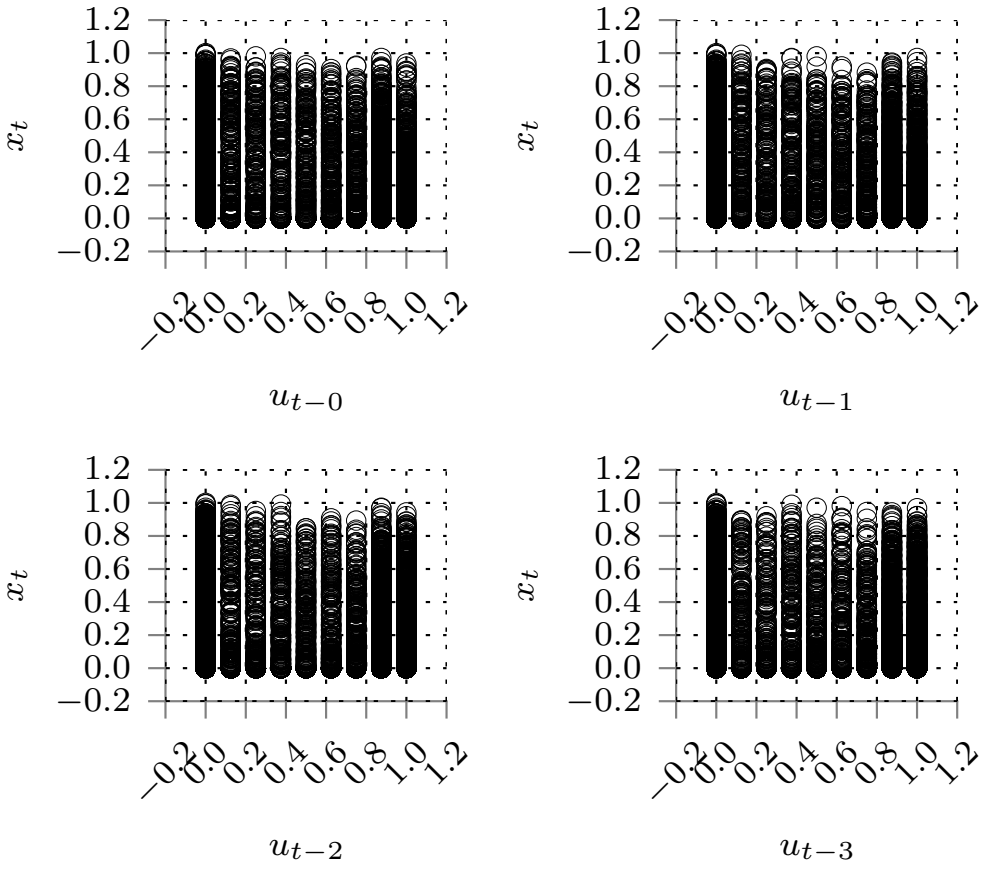

Figure A.4.: Scatter plot of current photovoltaic production x_t and cloudiness u_{t-h}

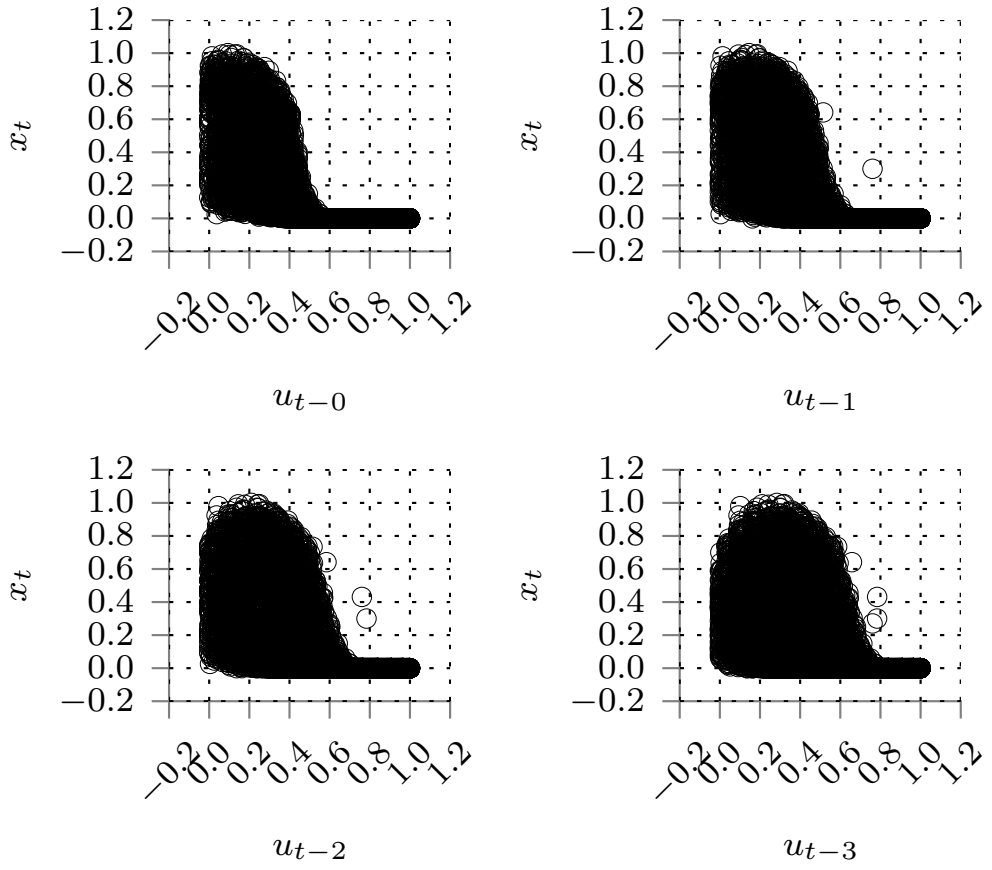

Figure A.5.: Scatter plot of current photovoltaic production x_t and sun angle u_{t-h}

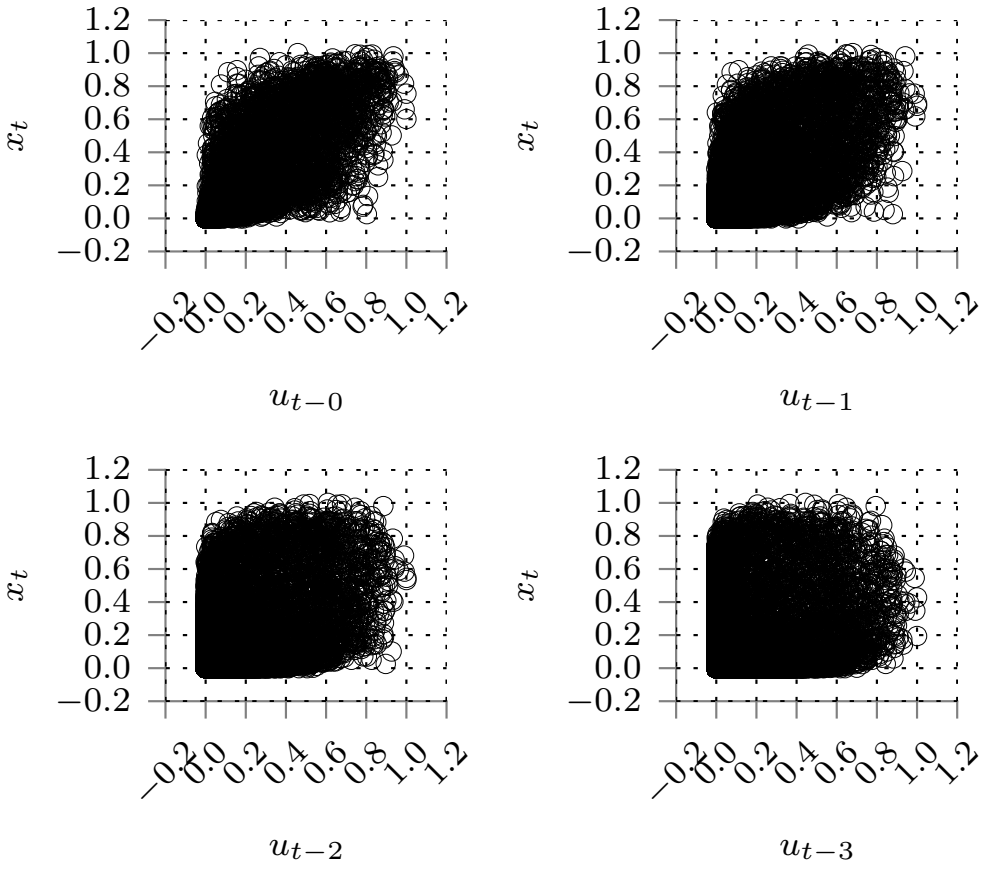

Figure A.6.: Scatter plot of current photovoltaic production x_t and solar irradiance u_{t-h}

List of Figures

Bibliography

1. C. Monteiro, T. Santos, L. A. Fernandez-Jimenez, I. J. Ramirez-Rosado, and M. S. Terreros-Olarte, “Short-term power forecasting model for photovoltaic plants based on historical similarity,” *Energies*, vol. 6, no. 5, pp. 2624–2643, 2013.

2. Y. Zhang, M. Beaudin, H. Zareipour, and D. Wood, “Forecasting solar photovoltaic power production at the aggregated system level,” in *North American Power Symposium*, 2014, pp. 1–6.

3. C. Rodriguez and G. Amaratunga, “Dynamic stability of grid-connected photovoltaic systems,” in *Power Engineering Society General Meeting*, 2004, pp. 2193–2199.

4. V. Lo Brano, G. Ciulla, and M. Di Falco, “Artificial neural networks to predict the power output of a pv panel,” *International Journal of Photoenergy*, 2014.

5. Y. Su, L.-C. Chan, L. Shu, and K.-L. Tsui, “Real-time prediction models for output power and efficiency of grid-connected solar photovoltaic systems,” *Applied energy*, vol. 93, pp. 319–326, 2012.

6. L. Prokop, S. Mišák, T. Novosád, P. Krömer, J. Platoš, and V. Snášel, “Artificially evolved soft computing models for photovoltaic power plant output estimation,” in *IEEE International Conference on Systems, Man, and Cybernetics*, 2012, pp. 1011–1016.

7. B. Parsons, M. Milligan, B. Zavadil, D. Brooks, B. Kirby, K. Dragoon, and J. Caldwell, “Grid impacts of wind power: a summary of recent studies in the United States,” *Wind Energy*, vol. 7, no. 2, pp. 87–108, 2004.

8. M. Ortega-Vazquez, D. S. Kirschen *et al.*, “Assessing the impact of wind power generation on operating costs,” *IEEE Transactions on Smart Grid*, vol. 1, no. 3, pp. 295–301, 2010.

9. J. L. Angarita-Márquez, C. A. Hernandez-Aramburo, and J. Usaola-Garcia, “Analysis of a wind farm’s revenue in the british and spanish markets,” *Energy Policy*, vol. 35, no. 10, pp. 5051–5059, 2007.

10. M. Cucumo, A. De Rosa, V. Ferraro, D. Kaliakatsos, and V. Marinelli, “Performance analysis of a 3kw grid-connected photovoltaic plant,” *Renewable energy*, vol. 31, no. 8, pp. 1129–1138, 2006.

11. T. Hove, “A method for predicting long-term average performance of photovoltaic systems,” *Renewable Energy*, vol. 21, no. 2, pp. 207–229, 2000.

12. W. Durisch, B. Bitnar, J.-C. Mayor, H. Kiess, K.-h. Lam, and J. Close, “Efficiency model for photovoltaic modules and demonstration of its application to energy yield estimation,” *Solar energy materials and solar cells*, vol. 91, no. 1, pp. 79–84, 2007.

13. M. Bouzerdoum, A. Mellit, and A. M. Pavan, “A hybrid model (sarima–svm) for short-term power forecasting of a small-scale grid-connected photovoltaic plant,” *Solar Energy*, vol. 98, pp. 226–235, 2013.

14. Y. Li, Y. Su, and L. Shu, “An armax model for forecasting the power output of a grid connected photovoltaic system,” *Renewable Energy*, vol. 66, pp. 78–89, 2014.

15. A. Mellit, S. Sağlam, and S. Kalogirou, “Artificial neural network-based model for estimating the produced power of a photovoltaic module,” *Renewable Energy*, vol. 60, pp. 71–78, 2013.

16. A. Yona, T. Senjyu, T. Funabashi, and C.-H. Kim, “Determination method of insolation prediction with fuzzy and applying neural network for long-term ahead pv power output correction,” *IEEE Transactions on Sustainable Energy*, vol. 4, no. 2, pp. 527–533, 2013.

17. R. H. Shumway and D. S. Stoffer, *Time series analysis and its applications*. New York, USA: Springer Science & Business Media, 2013.

18. G. E. Box, G. M. Jenkins, and G. C. Reinsel, *Time series analysis: forecasting and control*. New Jersey, USA: John Wiley & Sons, 2011.

19. D. C. Montgomery, C. L. Jennings, and M. Kulahci, *Introduction to time series analysis and forecasting*. New Jersey, USA: John Wiley & Sons, 2015.

20. M. T. Hagan, H. B. Demuth, M. H. Beale, and O. D. Jesus, *Neural network design*, Accessed date: 2015-08-03. [Online]. Available: http://hagan.okstate.edu/NNDesign.pdf

21. R. Rojas, *Neural networks: a systematic introduction*. Berlin, Germany: Springer-Verlag Berlin Heidelberg, 2013.