Photovoltaic power forecasting using simple data-driven models without weather data

Jorge Ángel González Ordiano[1] · **Simon Waczowicz**[1] · **Markus Reischl**[1] · **Ralf Mikut**[1] · **Veit Hagenmeyer**[1]

Abstract The present contribution offers evidence regarding the possibility of obtaining reasonable photovoltaic power forecasts without using weather data and with simple data-driven models. The lack of weather data as input stems from the fact that the constant obtainment of forecast weather data might become too expensive or that communication with weather services might fail, but still accurate planning and scheduling decisions have to be conducted. Therefore, accurate one-day ahead forecasting models with only information of past generated power as input for offline photovoltaic systems or as backup in case of communication failures are of interest. The results contained in the present contribution, obtained using a freely available dataset, provide a baseline with which more complex forecasting models can be compared. Additionally, it will also be shown that the presented weather-free data-driven models provide better forecasts than a trivial persistence technique for different forecast horizons. The methodology used in the present work for the data preprocessing and the creation and validation of forecasting models has a generalization capacity and thus can be used for different types of time series as well as different data mining techniques.

Keywords Forecasting · Data-driven models · Photovoltaics · Weather-free · Energy Lab 2.0

✉ Jorge Ángel González Ordiano
jorge.ordiano@kit.edu

[1] Institute for Applied Computer Science, Karlsruhe Institute of Technology, Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany

1 Introduction

Photovoltaics (PV), the direct conversion of sunlight into electricity, is a technology that offers a realistic way of providing electricity without using fossil fuels nor releasing pollutants into the atmosphere. This is due to the continuous efficiency increase of the PV cell [5] and the continuous drop of investment costs for PV installations in the last years [21]. Unfortunately, PV systems have the disadvantage of having a volatile electrical power generation, due to their complete dependency on the weather. The intermittent power generation makes the balancing between demand and supply in the electrical grid challenging [20]. Thus forecasting models able to predict the future PV generated electrical power are of major importance; they simplify the balancing of the future electrical demand through the mutual adjustment of photovoltaic energy sources, energy storage systems, and demand side management [19].

A type of models capable of delivering reasonable forecasts are data-driven models [1]; models created using a database containing relevant information about the PV systems (e.g., past generated power) and data mining techniques. Those models are able to take previously defined input data (e.g., weather data) to deliver in correspondence a reasonable PV power forecast. Currently there appears to be a preference for the usage of artificial neural networks (ANN), due to their capacity to describe nonlinear relations [3]. Several ANNs have been used in literature: Some refer to classical approaches like the multilayer perceptron (MLP) used by Mellit and Pavan [8] or Rashkovska et al. [13], while others use more complex networks like the ANN ensemble by Chaouachi et al. [2] or the recursive neural network by Cococcioni et al. [3] and Tao et al. [17]. Considering that the use of ANNs requires a certain degree of trial and error, especially to determine the ANN topology, other approaches

have also been used. For example, Shi et al. [15], Silva Fonseca et al. [16], and Yang et al. [22] applied the principles of support vector regression to create their forecasting models. Likewise, time series analysis has also been used for the PV power forecast like the autoregressive moving average with exogenous input model (ARMAX) used by Li et al. [7].

A shared trait of all the previously mentioned forecasting models is their use of weather data as input, whose obtainment and usage might pose several concerns. For example, if the models utilize forecast weather data as input, historical forecast data is necessary for their creation, information which must be purchased from weather services. A possibility of circumventing such purchase would be to utilize historical measured weather data as "perfect forecasts". Nonetheless, this approach results in a questionable solution since such values do not reflect the uncertainty of using true forecasts. Also, after the models' creation a constant communication with weather services is necessary in order to obtain the required inputs. Simple weather forecasts (e.g., future average temperature in a region, rain possibility) are offered free of charge, but more specific and accurate ones (e.g., future solar irradiation at the PV system in a specific temporal resolution) have to be purchased. If the communication with weather services is unwanted in view of security reasons, or most importantly if it fails (e.g., failure of internet services), models as the ones previously described become inoperative. Therefore, data-driven models able to deliver accurate forecasts by only using information of past generated power not only offer a simple low cost offline solution, but can also be provided as backup models in case communication with weather services is lost.

A further problem regarding data-driven forecasting models is the use of different datasets for the models creation (most authors use data of PV systems to which only they have access to), which makes the comparisons of models presented in literature a challenging task [18]. The dataset used in the present contribution is a freely available dataset, its preprocessing is thoroughly described, thus allowing a future comparison of results. The use of simple data-driven models is to show that even such models are able to deliver acceptable results when only using information of past generated PV power and to provide a baseline with which more complex forecasting models can be compared.

The present work results are presented as follows: first, models with a 24 h forecast horizon are created and validated using several data mining techniques and the information of past generated power 24 to 48 h prior to the forecast horizon. The selection of the 24 h forecast horizon comes from the fact that such forecasts play a major role in the scheduling of energy systems and that the quality of data-driven models without weather data with very short forecast horizons (1 and 2 h) has already been assessed, for example, in the work of Pedro and Coimbra [11]. After the validation of the one-day ahead forecasting models, the technique whose models provide the best forecasts is utilized to create models with different forecast horizons. Their results are then compared to a trivial forecast in order to demonstrate the advantage of using data-driven models over a trivial forecast, even when those lack weather information as input.

The present contribution is part of the research project Energy Lab 2.0 of the Karlsruhe Institute of Technology, which is funded by the Helmholtz Association, the German Federal Ministry of Education and Research (BMBF), and the Ministry of Science, Research and Art (MWK) of the State of Baden-Wuerttemberg. The Energy Lab 2.0 is planned to be a large experimental and simulation field for energy system facilities [4]. One aim of the Energy Lab 2.0 is the systematic evaluation of various big data and data mining methods. It consists of several hardware components (e.g., electrical power grid, natural gas grid, consumers, solar power storage park) and an information and communication technologies part called Smart Energy System Simulation and Control Center (SEnSSiCC) [6] in which different forecasting models are going to be implemented.

The rest of the present contribution is organized as follows: Sect. 2 describes the used data , Sect. 3 depicts the methodology for the creation and validation of different data-driven models, including the data preprocessing and the data mining process. All methods described in Sect. 3 are implemented using the open source MATLAB toolbox Gait-CAD [9]. Section 4 shows the results and offers a discussion and lastly Sect. 5 offers the conclusions and outlook to this work.

2 Data

The data used in the present work comes from the "Ausgrid Solar Home Electricity Data", which is freely offered by the state-owned Australian energy provider Ausgrid.[1] It offers electricity data from 300 different households with installed rooftop PV systems. The contained measured value which is relevant for the present work is the generated PV electrical energy from which the average generated electrical power time series P [kW] of every household can be obtained (the PV power generated under standard conditions of each system is also provided). All PV power time series extracted from the dataset, with time samples $k = 1, \ldots, K$, have a temporal resolution of 30 min and contain measurements from July 1st, 2010 to June 30th, 2013 ($K = 52608$). One of the used time series contains several missing values which are corrected through a process further described in Sect. 3. Only the information of 54 of the 300 households is used in the present contribution, those households are the ones con-

[1] http://www.ausgrid.com.au.

sidered to be part of the clean dataset defined by Ratnam et al. [14].

3 Methodology

3.1 Preprocessing

3.1.1 Overview

The following paragraphs describe the preprocessing steps applied to the raw PV power time series (P) in order to improve their quality. The preprocessing method is independent of the time series used in the present contribution. This implies that it can be used on different types of time series. Figure 1 depicts the data preprocessing (outlier detection, normalization, missing data treatment, and synchronization) using a PV power time series of the Ausgrid dataset to allow a better understanding of each step. Worth mentioning is the fact that negative values which are impossible in PV power time series are automatically set equal to zero before the application of the preprocessing method.

3.1.2 Outlier detection and elimination

The identification and elimination of outliers is carried out with the help of the Hampel filter[2] [10]. This moving window filter needs two parameters to be applied, the moving window half width parameter k_w and a threshold value t_H. The filter calculates the median $\tilde{P}$ of every subset $P[k-k_w], \ldots, P[k], \ldots, P[k+k_w]$ using the following equation:

$$\tilde{P}[k] = \text{median}(P[k-k_w], \ldots, P[k], \ldots, P[k+k_w]). \quad (1)$$

Using the obtained median, the median absolute deviation (MAD) is calculated by the equation:

$$MAD[k] = \text{median}(|P[k-k_w] - \tilde{P}[k]|, \ldots, |P[k] - \tilde{P}[k]|, \ldots, |P[k+k_w] - \tilde{P}[k]|). \quad (2)$$

Using the MAD, the MAD scale estimate S is determined:

$$S[k] = 1.4826\, MAD[k]\,. \quad (3)$$

The actual filtering occurs when the absolute difference of each value in the subset and the corresponding $\tilde{P}$ is compared to the multiplication of t_H and S; if the difference is greater than $t_H \cdot S$ the value is replaced by $\tilde{P}$:

$$P_f[k] = \begin{cases} P[k], & \text{if } |P[k] - \tilde{P}[k]| < t_H \cdot S[k] \\ \tilde{P}[k], & \text{else} \end{cases}, \quad (4)$$

P_f is the filtered time series. This filter has the advantage of only classifying as outliers values which are completely different to their neighborhood, therefore leaving trends and seasonal variations unchanged. In the present contribution $k_w = 3$ and $t_H = 3$ are used. Furthermore, values given as NaNs[3] are not considered outliers and are rather ignored by this step; those values are corrected later in the missing data treatment step.

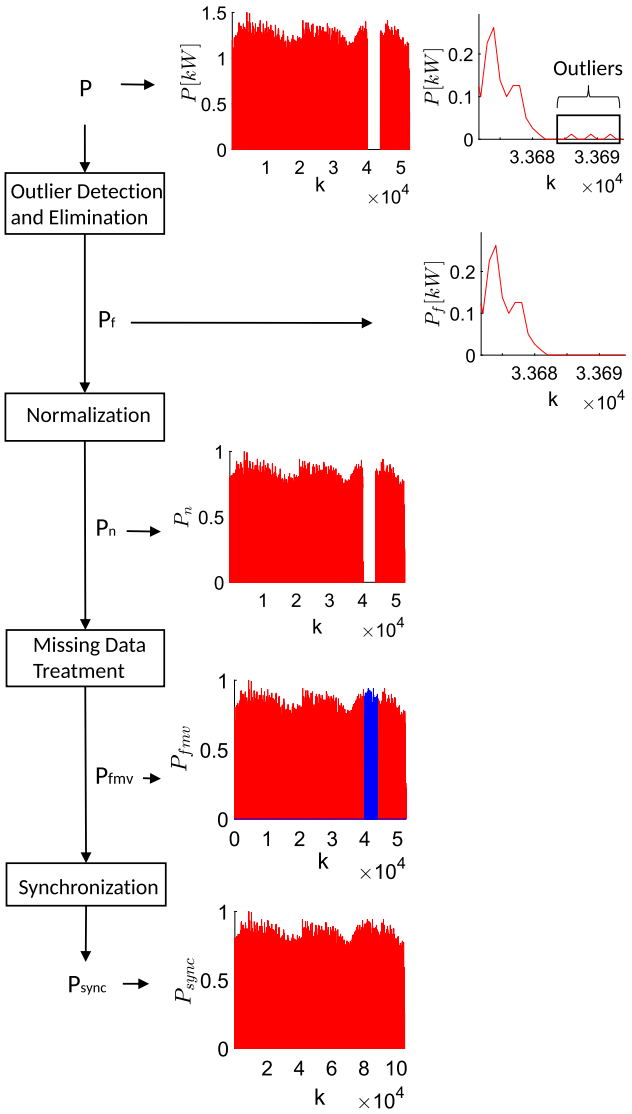

Fig. 1 Data preprocessing steps

[2] This filter can only be used during offline data analysis, due to its acausal nature.

[3] NaN: Not a Number.

3.1.3 Normalization

A normalization is applied in order to allow the comparison of time series:

$$P_n[k] = \frac{P_f[k] - \min_{k=1,\dots,K}(P_f[k])}{\max_{k=1,\dots,K}(P_f[k]) - \min_{k=1,\dots,K}(P_f[k])}. \quad (5)$$

The previous equation re-scales all time series values to values between zero and one. Due to the previous application of the outlier detection and elimination, this normalization can be applied even though it is sensitive towards outliers.

3.1.4 Missing data treatment

This step has the goal of filling in missing data in a time series with plausible values. It can be divided into a process referred to as automatic merging and a linear interpolation.

(a) Automatic merging

Samples of P have to possess similar values to other time series recorded at the same time in the same geographical area. Because the normalization makes the magnitude of time series values independent of system-specific properties (e.g., PV power under standard conditions) a merging becomes feasible. So in order to apply this method to P_n, a reference normalized time series without missing value areas and with the same type of values $P_{n,ref}$ is necessary. The reference time series are found with the help of labels which define the geographical location of the different PV systems (in this case the zip-code of the different households).[4] If a PV system has the same label as one with time series containing missing value areas, then its time series can be used as reference (if several reference time series are found only the first one to be found is utilized in this step). To determine whether adjacent missing values form a missing value area, a threshold t_{mva} has to be defined; if the number of adjacent missing values is greater than t_{mva} they are considered to be part of a missing value area. The values inside the missing value area are then replaced by the values of $P_{n,ref}$ according to the equation:

$$P_{am}[k] = P_{n,ref}[k]\,, \quad (6)$$

P_{am} is the time series after the automatic merging step. In the present contribution the number of adjacent missing values to be considered a missing value area is set equal to $t_{mva} = 3$ which represents 90 min of missing data. It is important to mention that the algorithm does not stop if it is unable to find a reference time series for one containing missing value areas. Rather, the algorithm saves a variable in its current workspace containing which time series could not be corrected, so that they can be identified later.

(b) Interpolation

The missing values which are not considered as part of a missing value area are filled in with values obtained using a linear interpolation.

The linear interpolation is undertaken as follows: if $P_{am}[k]$, with $k \in (k_1, k_2)$, is missing and $P_{am}[k_1]$ and $P_{am}[k_2]$ are the first non-missing values to the left and to the right correspondingly, then $P_{am}[k]$ is approximated using the equation:

$$P_{fmv}[k] = \frac{P_{am}[k_2] - P_{am}[k_1]}{k_2 - k_1}(k - k_1) + P_{am}[k_1], \quad k \in (k_1, k_2)\,. \quad (7)$$

P_{fmv} is referred to as the time series free of missing values.

3.1.5 Synchronization

The final part of the preprocessing step is called synchronization and aims for the standardization of the processed time series temporal resolution. The idea is to obtain new time series with a higher or lower temporal resolution through the linear interpolation or averaging of the subjected time series values. For example, if the wanted temporal resolution is 15 min but the given time series have one of 30 min (as is the case in the here used dataset) an interpolation between values is necessary. The change in resolution is accomplished using a value referred to as r_{quo}, it describes the factor with which the resolution is being increased or decreased (in the present work: $r_{quo} = 2$). The interpolation function that represents the increase in resolution of a time series is given by the following equations:[5]

$$k_r = \text{floor}\left(\frac{k_s - 1}{r_{quo}}\right) + 2\,, \quad (8)$$

$$k_l = \text{floor}\left(\frac{k_s - 1}{r_{quo}}\right) + 1\,, \quad (9)$$

$$h = \frac{P_{fmv}[k_r] - P_{fmv}[k_l]}{r_{quo}} \cdot (k_s - k_l)\,, \quad (10)$$

$$P_{sync}[k_s] = \begin{cases} P_{fmv}[k], & \exists k\colon k_s = r_{quo}(k-1) + 1 \\ h + P_{fmv}[k_l], & \text{else} \end{cases}. \quad (11)$$

[4] To search for reference time series, the PV power time series from all 300 households in the Ausgrid dataset are used.

[5] The floor operator rounds a real number to its preceding integer.

As Eq. (11) shows, the resulting time series, P_{sync}, has a greater number of discrete time steps (in the equation referred to as k_s) than P_{fmv}; those range from 1 to $r_{quo} \cdot (K-1)+1$.

To keep the nomenclature simple the discrete time steps k_s are represented further with the letter k. Likewise, the subscripts used during the present section are not going to be utilized in the following sections.

3.2 Data mining

3.2.1 Overview

After the preprocessing step the data mining process can be conducted. A second time series for every household is created and added to the used dataset. Such time series is referred to as t_{tod} and its values (which range from 0 to 1) represent the time in which every measurement of P was made. The reason behind its inclusion is to investigate if the incorporation of data which can help the models learn the periodicity of the sun improves the forecasting accuracy. An example in which the inclusion of a time of day time series as input improved the forecasting results can be found in the work of Cococcioni et al. [3]. Likewise, an advantage of using past generated power as input is that it implicitly contains specific information about the PV systems (e.g., orientation and inclination of PV modules, modules aging) and about systematic repeating effects (e.g., building shadowing effects). Figure 2 depicts an example of measurements for a single day of both P and t_{tod}.

The goal of the data mining step is to find a functional relation between input time series and the estimated future generated PV power ($\hat{P}$). The idea is to approximate the future generated PV power at the forecast horizon H using all the information contained in the input time series from time k to time $k - H_1$ (in the present work: $H_1 = 24$ h). An example of the functional relations sought after in the present contribution when the PV power as well as the combination

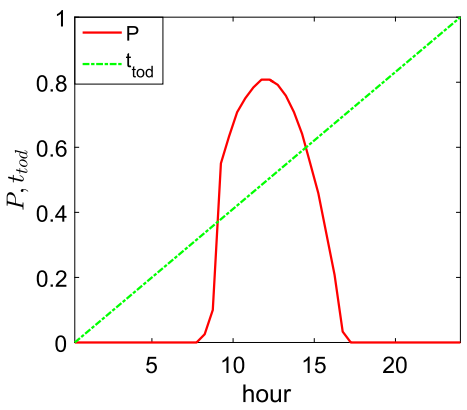

Fig. 2 Example of one day measurements for the past generated power as well as the time of day time series

of PV power and time of day time series are used as input are shown in the equations below:

$$\hat{P}[k+H] = f(P[k], \ldots, P[k-H_1]), \tag{12}$$

$$\hat{P}[k+H] = f(P[k], \ldots, P[k-H_1], t_{tod}[k], \ldots, t_{tod}[k-H_1]). \tag{13}$$

In order to simplify the notation the functional relations are going to be written further as $f(P)$ and $f(P, t_{tod})$. The following paragraphs describe first the techniques used to create the models and then the validation process applied to validate their accuracy. All data mining techniques applied in the present work can also be used with other inputs (e.g., solar irradiation, ambient temperature, PV module temperature, etc.).

3.2.2 Data mining techniques

The techniques used in the present contribution to create the data-driven models are: a so-called persistence technique, four different polynomials without bi-linear terms, and two artificial neural networks (ANN).

The first technique is referred to as persistence, it is used in order to determine whether the other techniques are able to provide more reasonable forecasts than this trivial one, which is described by the equation:

$$\hat{P}[k+H] = P[k]\,. \tag{14}$$

The four different polynomial techniques are polynomials of degree one to four (Poly1–Poly4). It is important to mention that not all input times series values are used to create the polynomial models, but rather only the four most relevant ones. For example, in order to create a Poly2 model with P as input the algorithm takes all values from $P[k]$ to $P[k - H_1]$ and raises them to the power of one and to the power of two. Afterwards, it chooses the four most relevant ones, through a stepwise process. Once the values are chosen the algorithm creates a sparse polynomial model whose parameters are determined by a least squares method.

Both ANNs are multilayer perceptrons (MLP) with one hidden layer, due to the fact that one hidden layer MLP is able to approximate a function of any complexity [2]. Furthermore, a hyperbolic tangent function is used in the hidden neurons and a linear function in the output ones. Regarding the number of hidden neurons, the first one of them has six (ANN6) and the second ten (ANN10). Both ANN's models are created by the Levenberg-Marquardt backpropagation algorithm with a maximum of twenty training epochs. Figure 3 depicts the topology of ANN6 if only P is used as an input.

No a-priori information, like PV power should be zero at night, is used during the creation of the models.

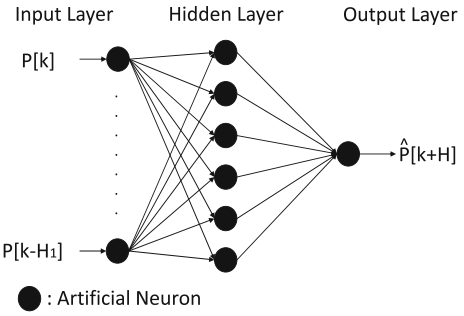

Fig. 3 Example of the ANN6 topology when only P is used as input

3.2.3 Validation

A validation is necessary to assess the performance of the models obtained with the used techniques on unknown data. The validation technique chosen in the present contribution is a fivefold cross-validation; the process consists in separating the time series to be used into five segments, using four segments (training data) to create models with the different data mining techniques, and testing those models on the remaining segment (test data). Afterwards, their performance on the test data is validated with the values commonly used for the evaluation of forecasting models, according to [12]. Those values are the mean absolute error (MAE) and the root mean square error (RMSE), likewise, the Pearson correlation coefficient between forecast and actual value ($r_{P\hat{P}}$) is also used. So if $\hat{P}$ is the forecast time series, the time series of the forecasting error (e_f) and all the previously mentioned values are calculated by the equations:

$$e_f[k] = \hat{P}[k] - P[k]\,, \tag{15}$$

$$MAE = \frac{1}{K}\sum_{k=1}^{K}|e_f[k]|\,, \tag{16}$$

$$RMSE = \sqrt{\frac{1}{K}\sum_{k=1}^{K}(e_f[k])^2}\,, \tag{17}$$

$$r_{P\hat{P}} = \frac{\sum_{k=1}^{K}(P[k]-\bar{P})\,(\hat{P}[k]-\bar{\hat{P}})}{\sqrt{\sum_{k=1}^{K}(P[k]-\bar{P})^2}\sqrt{\sum_{k=1}^{K}(\hat{P}[k]-\bar{\hat{P}})^2}}\,, \tag{18}$$

with $\bar{P}$ and $\bar{\hat{P}}$ being the averages of their corresponding time series. The average evaluation results of all households' models on their test data provide an idea of how well the different data mining techniques are at creating accurate forecasting models.

4 Results

The average validation results of the one-day ahead forecasting models ($H = H_1 = 24$ h) of every household are presented in Table 1. As it can be seen the past generated power (P) as well as the combination of power and time of day (t_{tod}) time series are used as input. The reason why t_{tod} is not used without P, is that its usage would result in a model only able to describe the periodicity of the PV power, but not its magnitude. The validation results demonstrate that the inclusion of t_{tod} has an improving effect (from approx. 0.1 to 1 %) on the overall forecasting accuracy of all the created models (with the only exceptions being the MAE values for Poly3 and Poly4). Additionally, the results show that the more complex techniques ANN6 and ANN10 possess the best results on the test data, with ANN6 using both P and t_{tod} as input being the best. Interestingly, models perform in average better than the persistence technique in regard to their RMSE and $r_{P\hat{P}}$ values, but only the ANNs obtain models whose MAE is lower than that of the trivial persistence technique. This can be attributed to the fact that most models obtained with the polynomial technique have a non-disappearing offset at night which increases their mean absolute error (such an effect can be corrected through the use of a-priori information during the models creation), but once larger errors are weighted strongly as is the case with the RMSE the persistence approach becomes the worst of them all.

As already mentioned, the ANN with six hidden neurons and both P and t_{tod} as input creates the models whose forecasting accuracy is the highest. Furthermore, the technique which does not allow any non-linearities (without considering persistence) Poly1 has the worst results regarding RMSE and $r_{P\hat{P}}$. The models' errors, when using past generated power and time of day as input, range from 6.64 to 7.25 % in the case of the mean absolute error and from 12.47 to 13.3 % for the root mean square error, while when considering the correlation coefficient the values range from 85.81 to 87.65 %. At first glance the ranges appear completely acceptable. Additionally, the fact that those ranges are obtained with the data-driven models can be attributed to the periodicity of the sun, the lack of drastic weather changes (weather stability in the dataset's region), and the fact that the models do not have to model relations between input data and system specific properties (like orientation of PV modules). The latter is already implicitly contained in the used input data. An example of forecasts obtained using both, Poly1 and ANN6 (with P and t_{tod} as input) for five different days is displayed in Fig. 4. It is clear that if the forecast day has a completely different weather as the previous two the forecast will be incorrect, because it is not possible by only using historical data of the past two days to forecast such changes without further improvements on

Table 1 Validation results obtained from models with $H = H_1 = 24$ h created using P as well as the combination of P and t_{tod} as input

$\hat{P}$	$f(P)$			$f(P, t_{tod})$		
	MAE	RMSE	$r_{P\hat{P}}$	MAE	RMSE	$r_{P\hat{P}}$
Poly1	0.0792	0.1382	0.8455	0.0724	0.1330	0.8581
Poly2	0.0737	0.1361	0.8505	0.0724	0.1325	0.8594
Poly3	0.0713	0.1352	0.8525	0.0725	0.1325	0.8595
Poly4	0.0713	0.1352	0.8525	0.0723	0.1324	0.8596
ANN6	0.0688	0.1262	0.8735	0.0664	0.1247	0.8765
ANN10	0.0685	0.1259	0.8741	0.0668	0.1248	0.8764
Persistence	0.0689	0.1543	0.8220	0.0689	0.1543	0.8220

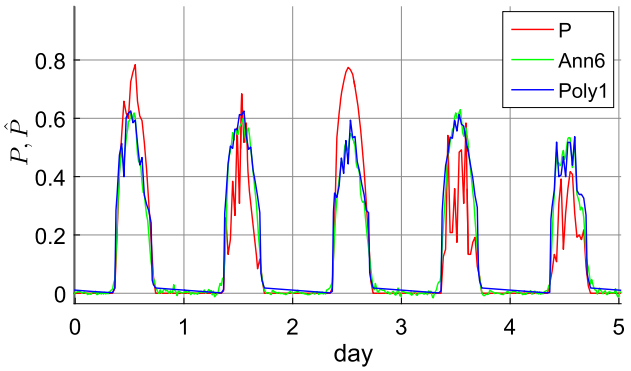

Fig. 4 Forecasts obtained from Poly1 and ANN6. *Red* Measured values, *green* ANN6 forecast, *blue* Poly1 forecast (color figure online)

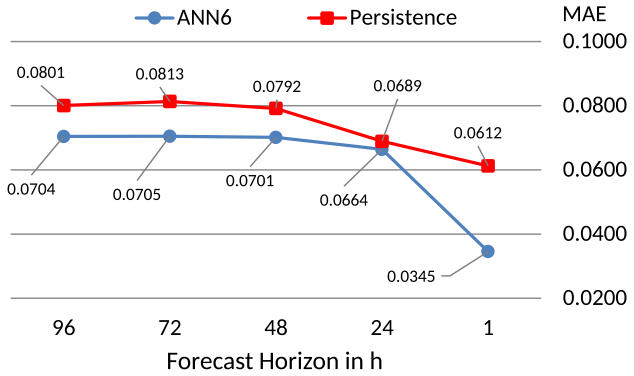

Fig. 5 Comparison of ANN6 and persistence MAE results over different forecast horizons

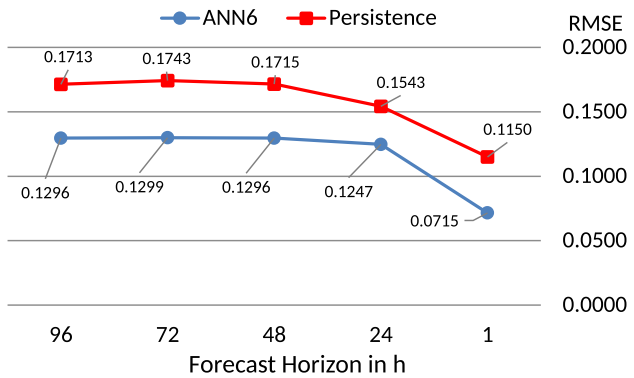

Fig. 6 Comparison of ANN6 and persistence RMSE results over different forecast horizons

the models (e.g., intra-hour corrections). So it can be concluded that weather stability is required in order to obtain accurate results with the models of the present contribution.

Since the ANN with six hidden neurons and both P as well as t_{tod} as input provides the best one-day ahead forecasting models, it is chosen to create forecasting models with different forecast horizons (here: $H = 1$, 24, 48, 72, and 96 h) using the functional relation described by Eq. (13) with $H_1 = 24$ h. Its models are then tested on unknown test data (utilizing the same fivefold cross-validation process as before). Figures 5, 6, and 7 show a comparison between the results obtained by ANN6 and the persistence technique. As it can be seen the models created using ANN6 have better results than persistence independently of the considered forecast horizon, thus demonstrating the advantages of utilizing a data-driven model instead of the trivial forecast, even when that model only uses information of the past generated power as input. Evidence regarding the weather stability of the dataset's region can be found in the results obtained for forecasts horizons greater than 24 h, which remained more or less constant for both persistence and ANN6. Such forecast horizon independence could be used as a way of qualitatively estimating the future accuracy of the weather-free data-driven forecasting models for a given dataset.

The fact that in the present contribution data-driven models seem to have better RMSE values than the trivial persistence technique, independently of the forecast horizon, might stem from their usage of two days to conduct their forecasts. For example, in a scenario in which the first day used for the forecast is sunny and the second rainy, but the forecast day is again sunny, the persistence technique is going to provide a completely inaccurate result (i.e. another rainy day), while the data-driven models are more-or-less going to deliver a weighted average of their input days as forecast. For such reason, the usage of more than one day as input allows

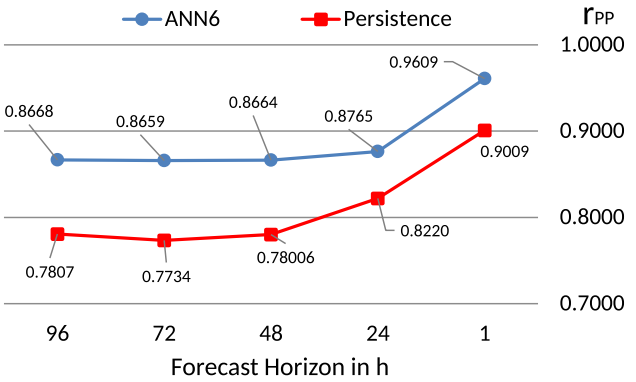

Fig. 7 Comparison of ANN6 and persistence $r_{P\hat{P}}$ results over different forecast horizons

the data-driven models to obtain an overall lower RMSE. The averaging of the input days can be discerned from the functional relations obtained via the polynomial models. For example, a common obtained Poly1 structure is described by the equation

$$\hat{P}[k+H] = \theta_0 + \theta_1 P[k] + \theta_2 P[k-H_1] + \cdots , \qquad (19)$$

with θ_i being the least squares determined parameters. As it can be seen, the first two values represent the previously mentioned averaging of input days.

5 Conclusions and outlook

The present contribution utilizes a freely available dataset as well as a systematic preprocessing and data-mining procedure to create a series of simple data-driven forecasting models. The thorough description of the applied methodology and the availability of the used dataset allow the reproducibility as well as the comparison of the presented results. Hence, it fulfills the goal of providing a comparative baseline for future research and more complex forecasting approaches.

Furthermore, the results offer evidence that acceptable one-day ahead forecasting models can be obtained by using past generated power and the time in which all those measurements were taken. The apparent low errors (MAE $=$ $6.64 \ldots 7.25$ % and RMSE $=$ $12.47 \ldots 13.3$ %) and high correlation coefficients ($r_{P\hat{P}} = 85.81 \ldots 87.65$ %) obtained from the data-driven models (with past generated power and time of day as input) show that, at least for the used dataset, they are able to provide satisfactory results. At the same time, the impossibility of the presented models to provide accurate one-day ahead forecasts for days with completely different weather conditions as their predecessors demonstrates that their usage should be reserved to regions with mostly stable weather (as possible low cost offline solutions) or as emergency backup for cases in which traditional weather dependent forecasting models become inoperative. Additionally, the results acquired by comparing the trivial persistence technique and models obtained from an artificial neural network across several forecast horizons offer evidence regarding the advantages of using forecasting data-driven models over a trivial forecast—even when those data-driven models lack weather data as input.

More complex weather-free forecasting models, like artificial neural networks with different structures, have to be further investigated in order to estimate how accurate such type of models can get. Likewise, a comparison between models obtained with the present contribution's data-mining techniques in regions with more unstable weather is necessary. Such comparison will provide further evidence regarding the possibility of using weather-free data-driven models as backup, in case of communication failure with weather services.

Another scenario of interest for future studies, that allow the usage of weather-free models in regions with more unstable weather, consists of a regional energy grid comprised of a central agent and several decentralized ones. The central agent possesses forecast weather data, as well as an additional constraint regarding the impossibility to communicate such data with the other agents (e.g. due to confidentiality clauses with weather services). The decentralized agents lack completely forecast weather data. In such a scenario, decentralized agents could utilize weather-free forecasting models as a simple low cost solution, while the centralized one (who has information regarding the decentralized agents' forecasting models) could estimate forecast deviations utilizing its knowledge about future weather and act accordingly to assure the balancing of the electrical load. Specifics about such possible procedures have to be clarified and studied in future related works.

Acknowledgements The present contribution is supported by the Helmholtz Association under the Joint Initiative "Energy System 2050–A Contribution of the Research Field Energy".

References

1. Almonacid F, Rus C, Pérez-Higueras P, Hontoria L (2011) Calculation of the energy provided by a PV generator. comparative study: conventional methods vs. artificial neural networks. Energy 36(1):375–384
2. Chaouachi A, Kamel RM, Nagasaka K (2010) Neural network ensemble-based solar power generation short-term forecasting. JACIII 14(1):69–75
3. Cococcioni M, D'Andrea E, Lazzerini B (2012) One day-ahead forecasting of energy production in solar photovoltaic installations: an empirical study. Intell Decis Technol 6(3):197–210
4. Düpmeier C, Stucky KU, Mikut R, Hagenmeyer V (2015) A concept for the control, monitoring and visualization center in Energy Lab 2.0. In: Energy informatics. Springer, pp 83–94

5. Green MA (2005) Silicon photovoltaic modules: a brief history of the first 50 years. Prog Photovolt Res Appl 13(5):447–455
6. Hagenmeyer V, Cakmak HK, Düpmeier C, Faulwasser T, Isele J, Keller HB, Kohlhepp P, Kühnapfel U, Stucky U, Waczowicz S, Mikut R (2016) Information and communication technology in Energy Lab 2.0: Smart energies system simulation and control center with an open-street-map-based power flow simulation example. Energy Technol 4:145–162
7. Li Y, Su Y, Shu L (2014) An ARMAX model for forecasting the power output of a grid connected photovoltaic system. Renew Energy 66:78–89
8. Mellit A, Pavan AM (2010) A 24-h forecast of solar irradiance using artificial neural network: application for performance prediction of a grid-connected PV plant at Trieste, Italy. Sol Energy 84(5):807–821
9. Mikut R, Burmeister O, Braun S, Reischl M (2008) The open source Matlab toolbox Gait-CAD and its application to bioelectric signal processing. In: Proc., DGBMT-Workshop Biosignalverarbeitung, Potsdam, pp 109–111
10. Pearson RK (2002) Outliers in process modeling and identification. IEEE Trans Control Syst Technol 10(1):55–63
11. Pedro HT, Coimbra CF (2012) Assessment of forecasting techniques for solar power production with no-exogenous inputs. Sol Energy 86(7):2017–2028
12. Pelland S, Remund J, Kleissl J, Oozeki T, De Brabandere K (2013) Photovoltaic and solar forecasting: state of the art. IEA PVPS Task, vol 14
13. Rashkovska A, Novljan J, Smolnikar M, Mohorčič M, Fortuna C (2015) Online short-term forecasting of photovoltaic energy production. In: Proc., the Sixth IEEE conference on innovative smart grid technologies (ISGT2015)
14. Ratnam EL, Weller SR, Kellett CM, Murray AT (2015) Residential load and rooftop PV generation: an Australian distribution network dataset. Int J Sustain Energy, pp 1–20
15. Shi J, Lee WJ, Liu Y, Yang Y, Wang P (2012) Forecasting power output of photovoltaic systems based on weather classification and support vector machines. IEEE Trans Ind Appl 48(3):1064–1069
16. Silva Fonseca JG, Oozeki T, Takashima T, Koshimizu G, Uchida Y, Ogimoto K (2012) Use of support vector regression and numerically predicted cloudiness to forecast power output of a photovoltaic power plant in Kitakyushu, Japan. Prog Photovolt Res Appl 20(7):874–882
17. Tao C, Shanxu D, Changsong C (2010) Forecasting power output for grid-connected photovoltaic power system without using solar radiation measurement. In: Proc., 2nd IEEE International symposium on power electronics for distributed generation systems (PEDG). IEEE, pp 773–777
18. Ulbricht R, Fischer U, Kegel L, Habich D, Donker H, Lehner W (2014) ECAST: a benchmark framework for renewable energy forecasting systems. In: EDBT/ICDT Workshops. Citeseer, pp 148–155
19. Waczowicz S, Reischl M, Hagenmeyer V, Mikut R, Klaiber S, Bretschneider P, Konotop I, Westermann D (2015) Demand response clustering-how do dynamic prices affect household electricity consumption? In: Proc., IEEE PowerTech, Eindhoven. IEEE, pp 1–6
20. Waczowicz S, Reischl M, Klaiber S, Bretschneider P, Konotop I, Westermann D, Hagenmeyer V, Mikut R (2016) Virtual storages as theoretically motivated demand response models for enhanced smart grid operations. Energy Technol 4:163–176
21. Wirth H, Schneider K (2013) Recent facts about photovoltaics in Germany. Report from Fraunhofer Institute for Solar Energy Systems, Germany
22. Yang HT, Huang CM, Huang YC, Pai YS (2014) A weather-based hybrid method for 1Day ahead hourly forecasting of PV power output. IEEE Trans Sustain Energy 5(3):917–926

Jorge Ángel González Ordiano received his Master of Science degree in mechanical engineering from the Karlsruhe Institute of Technology (KIT), Germany, in 2016. Since 2016 he works as a Ph.D. student at KIT's Institute for Applied Computer Science (IAI). He is an expert in the fields of data mining, time series analyses, and time series forecasting.

Simon Waczowicz received his diploma degree in mechanical engineering in 2013 at the Karlsruhe Institute of Technology (KIT), Germany. Since 2013 he is a research associate at the Institute for Applied Computer Science (IAI) at the KIT. Simon Waczowicz is an expert in the fields of data mining, time series analyses, demand side management and smart grids.

Markus Reischl received the diploma degree and his Ph.D. degree in mechanical engineering from the Karlsruhe Institute of Technology (KIT), Germany, in 2001 and 2006, respectively. Since 2001, he is with the Institute for Applied Computer Science. He heads the project TELMYOS with a focus on man-machine interfaces, bioinformatics and data-mining.

Ralf Mikut received the diploma degree in Automatic Control from the University of Technology, Dresden, Germany, in 1994, and the Ph.D. and Habilitation degree in mechanical engineering from the University of Karlsruhe, Karlsruhe, Germany, in 1999 resp. 2007. Since 2011, he is Associate Professor at the Faculty of Mechanical Engineering and Head of the Research Group "Automated Image and Data Analysis" at the Institute for Applied Computer Science of the Karlsruhe Institute of Technology (KIT), Germany. His current research interests include data mining, image processing, big data, computational intelligence, life science applications, and smart grids.

Veit Hagenmeyer studied engineering cybernetics in Stuttgart, Germany, and Berkeley, CA, USA. He wrote his dissertation in Paris, France, in the area of differential flatness and electrical drives. After Postdoctoral stays in Paris and Stuttgart, he worked in several positions at BASF SE, Ludwigshafen, Germany, mainly in the fields of Automation Technology, Advanced Process Control and Verbund Simulation. In his final position at BASF SE, he was responsible for the three power plants and energy grids at Ludwigshafen site. Now he is a Professor for energy informatics and a Research Director at the Karlsruhe Institute of Technology (KIT), Germany. In this position, together with his team, he is responsible for the Smart Energy System Simulation and Control Center, the ICT-part of the Energy Lab 2.0, and of the research field "Tools" of the Energy 2050 Initiative of the Helmholtz Association.