

Prediction in Photovoltaic Power by Neural Networks [†]

Antonello Rosato ¹, Rosa Altilio ¹, Rodolfo Araneo ² and Massimo Panella ^{1,*}

¹ Department of Information Engineering, Electronics and Telecommunications, University of Rome “La Sapienza”, via Eudossiana, 18, Rome 00184, Italy; antonello.rosato@uniroma1.it (A.R.); rosa.altilio@uniroma1.it (R.A.)

² Electrical Engineering Division of Department of Astronautical, Electrical and Energy Engineering, University of Rome “La Sapienza”, via Eudossiana, 18, Rome 00184, Italy; rodolfo.araneo@uniroma1.it

* Correspondence: massimo.panella@uniroma1.it; Tel.: +39-06-4458-5496

[†] This paper is an extended version of our paper published in Rosato, A.; Altilio, R.; Araneo, R.; Panella, M. Embedding of Time Series for the Prediction in Photovoltaic Power Plants. In Proceedings of the 2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC), Florence, Italy, 7–10 June 2016.

Academic Editor: Gabriele Grandi

Received: 11 April 2017; Accepted: 12 July 2017; Published: 15 July 2017

Abstract: The ability to forecast the power produced by renewable energy plants in the short and middle term is a key issue to allow a high-level penetration of the distributed generation into the grid infrastructure. Forecasting energy production is mandatory for dispatching and distribution issues, at the transmission system operator level, as well as the electrical distributor and power system operator levels. In this paper, we present three techniques based on neural and fuzzy neural networks, namely the radial basis function, the adaptive neuro-fuzzy inference system and the higher-order neuro-fuzzy inference system, which are well suited to predict data sequences stemming from real-world applications. The preliminary results concerning the prediction of the power generated by a large-scale photovoltaic plant in Italy confirm the reliability and accuracy of the proposed approaches.

Keywords: embedding technique; power forecasting; photovoltaic power plant; neural and fuzzy neural network

1. Introduction

In recent years, electrical power systems have progressively evolved from centralized bulk systems to decentralized systems in a distributed generation (DG) scenario characterized by smaller generating units connected directly to distribution networks near the consumer [1]. Several economic and environmental reasons, driven by government financial incentives [2], have considerably pushed the widespread adoption of DG in the energy marketplace, especially that from renewable resources (e.g., solar power and wind energy).

However, technical barriers often impede the entrance of DG into the current distribution infrastructure with a significant penetration level due to the intermittent nature of the renewable energy resources that often do not match the energy load demands [3]. Indeed, the transition to an energy economy primarily founded on renewable resources depends on overcoming the difficulties associated with the variability and reliability of these non-dispatchable resources [4]. These issues give rise to substantial critical issues with respect to the usual working habits of the transmission (TSO) and distribution (DSO) system operators, utility companies, as well as power producers: voltage and frequency regulation, islanding detection, harmonic distortion, distribution issues, as well as demand side management of prosumers. Certainly, energy storage systems (EES) could become a

valuable response providing specific support services for renewable energy production in order to reduce short-term output fluctuations and, consequently, improve power quality [5].

In this framework, it is of paramount importance to forecast accurately the power output of plants over the next hours or days in order to integrate in an optimal way the DG from non-programmable renewable resources into power systems on a large scale [6]. The ability to forecast the amount of power fed by renewable energy plants into substations is a necessary requirement for the TSO in order to optimize short- and middle-term decisions, such as corrections to out-of-region trade and unit commitments [7]. Besides, accurate short-term and intra-hour forecasting are required for regulatory, dispatching and load-following issues [8], while power system operators are more sensitive to intra-day forecasts [9], especially when handling multiple load zones, since they avoid possible penalties that are incurred due to deviations between forecasted and produced energy. Moreover, power system operators are mostly interested in developing methods to be used in the framework of the daily session of the electricity market where the producers must present energy bid offers for the 24 h of the following day [10].

Forecasting can be broadly divided into four major categories [11], namely long-, medium-, short- and very short-term forecast, on the basis of the time horizon's scale, which ranges from several years in the long-term forecasting to hours in the medium-term forecasting, till arriving at minutes in the very short-term forecasting. When adopted for photovoltaic (PV) power plants, forecasting techniques can be applied directly to the produced power (direct methods) or indirectly to the solar irradiation, which is the main factor related to the output power that is calculated in a second successive step (indirect methods) [12]. Both cases share similar techniques that can be broadly divided [11] into persistence methods, physical techniques, statistical techniques and hybrid models. Usually, physical methods try to obtain an accurate forecast using a white box approach, that is to say physical parametric equations. On the other hand, statistical or probabilistic methods try to predict the relevant quantities extracting dominant relations from past data that can be used to predict the future behavior. In addition, both methods can be properly mixed, originating the so-called "gray box" approaches [13].

In the prediction of time series, past observations are used to develop a method able to describe underlying relationships among data. Therefore, a mathematical model is adopted to extrapolate future values, especially for those contexts where missing and incomplete data can limit the ability to gain knowledge about the underlying, unknown process. Complexity and dynamics of real-world problems require advanced methods and tools able to use past samples of the sequence in order to make an accurate prediction [14]. Additionally, the problem of forecasting future values of a time series is often mandatory for the cost-effective management of available resources. These are well-known hard problems [6], given the non-stationarity and, often, the non-linearity of time series, which result in a complex dynamics that is hard to model adequately by using standard predictive models.

For many years, regressive statistical approaches [15] have been considered for prediction; among them, moving average (MA), auto regressive (AR), auto regressive moving average (ARMA), auto regressive integrated moving average (ARIMA) and autoregressive integrated moving average with exogenous variables (ARIMAX) are typical examples of these approaches. Unfortunately, standard structural models provide a poor representation of actual data and therefore result in poor accuracy when used for forecasting. Consequently, many worldwide research activities intend to improve the accuracy of prediction models. In this regard, computational intelligence is considered as one of the most fruitful approaches for prediction [16]. Several forecasting methods, with different mathematical backgrounds, such as fuzzy predictors, artificial neural networks (ANN), evolutionary and genetic algorithms and support vector machines, have been considered [17]. Nevertheless, dealing with noisy and missing training examples and the lack of robustness against outliers are still open problems.

In this paper, our attention is mainly concentrated on short-term forecasting with a time horizon of one day. Nevertheless, the proposed techniques can be applied to shorter or longer time horizons, yet always remaining in a short-term forecast. A huge amount of literature is available on short-term forecasting models applied to PV plants, even if most of them deal with irradiation forecasting

models: while not claiming to be exhaustive, the proposed techniques include mainly statistical models [18], stochastic predictors [19], fuzzy systems [20], ANN [21], ANN in conjunction with statistical feature parameters [22] and artificial intelligence (AI)-based techniques [23]. Some papers present also the comparison between the predictions obtained through different models based on two or more forecasting techniques [24]. Despite the large amount of literature on forecasting techniques, only a few papers describe forecasting models to be used for directly predicting the daily energy production of the PV plant. These few papers use general techniques appropriately applied to the day-ahead horizon. Starting from the input of past power measurements and meteorological forecasts of solar irradiance, the power output is estimated by means of ANN [25] or soft computing [26], while in [27], the ANN is applied only on past energy production values. Additionally, the ANN can be used in conjunction with numerical weather prediction (NWP) models [28], which can be based on satellite and land-based sky imaging [29]. NWP models have been also used to build an outperforming multi-model ensemble (MME) for day-ahead prediction of PV power generation [30]. More complex schemes are proposed in [31], where an ANN is used to improve the performance of baseline prediction models, i.e., a physical deterministic model based on cloud tracking techniques, an ARMA model and a k-nearest neighbor (kNN) model, and in [32], where a Kalman filter is used in conjunction with a state-space model (SSM).

The use of three inference systems for data regression based on neural and fuzzy neural networks is proposed here. The main distinguishing idea of our work is that we properly tailor new learning approaches to analyze data associated with renewable power sources starting from available techniques based on computational intelligence that have proven to produce very accurate predictions in several fields of application, like for example, risk management, biomedical engineering and financial forecasting [33], but also to analyze accurately the data associated with renewable power sources [34]. Actually, the stochastic and often chaotic behavior of time series in the prediction of output power of PV plants calls for ad hoc procedures to forecast power time series, which are here developed working only on the past measurements of electric quantities, as initially tested in our previous work [35].

We introduce a prediction problem as a two-fold process. First, the observed time series is embedded in order to select, among the available samples, the ones to be used to predict the next one. Then, such samples are used to feed three different function approximation models and techniques based on neural and fuzzy neural networks, which are suited to predict data sequences that often show a chaotic behavior: radial basis function (RBF) [36]; the adaptive neuro-fuzzy inference system (ANFIS) [37]; the higher-order neuro-fuzzy inference system (HONFIS) [38]. As explained in the body of the paper, a predictor can be based on such function approximation models, whose parameters are estimated by data-driven learning procedures.

The rest of the paper is organized as follows. In Section 2, we introduce the function approximation models that are used for time series prediction. The application is ascertained by several benchmark results and extensive computer simulations reported in Section 3 and discussed in Section 4 proving the validity of the proposed learning procedure and showing a favorable comparison with other well-known prediction techniques. Finally, our conclusions are drawn in Section 5.

2. Materials and Methods

There is a huge amount of literature pertaining to time series prediction techniques that rely on neural and fuzzy neural networks [39], which are actually of high importance in this field since they can be used to solve data regression problems [40]. A sequence $S(t)$ can be predicted by considering a function approximation $y = f(\underline{x})$, $f : \mathbb{R}^N \rightarrow \mathbb{R}$. For instance, by using linear models each input vector $\underline{x}_t$ is made of N subsequent samples of $S(t)$, and the output y_t is the sample to be predicted:

$$\underline{x}_t = [S(t) \ S(t-1) \dots S(t-N+1)] , \ y_t = S(t+m) , \ f_{\text{lin}}(\underline{x}_t) = \sum_{j=1}^N \lambda_j x_{tj} \quad (1)$$

Then, we obtain:

$$\tilde{S}(t+m) = \sum_{j=1}^N \lambda_j S(t-j+1) \quad (2)$$

where $\tilde{S}(t+m)$ is the prediction of the true value $S(t+m)$ at the time distance m . Considering the statistical properties of $S(t)$, as the autocorrelation function, it is possible to determine the parameters $\lambda_j, j = 1 \dots N$, of the function $f_{\text{lin}}(\cdot)$.

Generally speaking, input vectors $\underline{x}_t$ are obtained through the so-called “embedding technique”, which makes use of previous samples of $S(t)$ to build the vectors themselves. There are two parameters to be set in this regard, the embedding dimension D and the time lag T , resulting in the following embedding structure:

$$\underline{x}_t = [S(t) \ S(t-T) \ S(t-2T) \dots S(t-(D-1)T)] \quad (3)$$

These two parameters are estimated in two different ways, by using the false nearest neighbors (FNN) algorithm for the embedding dimension and the average mutual information (AMI) criterion for the time lag [41]. In the following part of the paper, D and T will be chosen by such algorithms via the VRA software [42].

The performances of a predictor that relies on a linear approximation model, as the one in Equation (1), are very poor when it is applied to data sequences in real environments. They often have noisy and chaotic components that force one to wisely choose the embedding parameters, as well as the function approximation model by using appropriate procedures [43]. In this context, the underlying, unknown system is observed through $S(t)$, and its state-space evolution is obtained by the trajectories of embedded vectors.

Overall, the estimated sample $\tilde{S}(t+m)$ predicted at a distance m will be:

$$\tilde{S}(t+m) = f(\underline{x}_t) \quad (4)$$

where $f(\cdot)$ is the regression model to be determined. Thus, the function $f(\cdot)$ will approximate the link from the reconstructed state $\underline{x}_t$ to the corresponding output y_t [41]. It is important to note that $f(\cdot)$ must be non-linear since the considered system has a complex behavior. In the following, we will consider the usual case of a “one-step-ahead” prediction, that is $m = 1$.

By driving the function approximation model with embedding data, neural networks and fuzzy logic are very suited to solving a prediction problem dealing with real-world sequences. In this paper, three models of this kind are proposed for the study of the considered photovoltaic time series. They are introduced in the following, considering as a baseline for benchmarking in the successive tests: the linear predictor reported in Equation (1), whose parameters are estimated through a common least-squares estimator (LSE) [44]; the well-known ARIMA model, which handles non-stationarity and seasonality and whose parameters are determined by a maximum likelihood approach [45]. Both of these models will be estimated by using lagged samples at the input, as determined by the embedding procedure, similarly to the adopted neural networks.

2.1. RBF

An RBF neural network is used to build up a function approximation model having the following structure:

$$f(\underline{x}) = \sum_{i=1}^M \lambda_i \phi(\|\underline{x} - \underline{c}_i\|) \quad (5)$$

where $\underline{x} \in \mathbb{R}^N$ is the input vector, $\phi(\cdot)$ is a radial basis function centered in $\underline{c}_i$ and weighted by an appropriate coefficient λ_i . The choice of $\phi(\cdot)$ and $\underline{c}_i$ must be considered for the approximation capability of the network. Commonly-used types of radial basis functions include:

- Gaussian:

$$\phi(r) = e^{-(\epsilon r)^2} \quad (6)$$

- Multiquadric:

$$\phi(r) = \sqrt{1 + (\epsilon r)^2} \quad (7)$$

- Inverse quadratic:

$$\phi(r) = \frac{1}{1 + (\epsilon r)^2} \quad (8)$$

Several methods can be used to minimize the error between desired output and model output and, hence, to identify the parameters $\underline{c}_i$ and λ_i [46].

2.2. ANFIS

An ANFIS neural network implements a fuzzy inference system to approximate the function $y = f(\underline{x})$, $f: \mathbb{R}^N \rightarrow \mathbb{R}$. It is composed of M rules of the Sugeno first-order type, where the k -th rule, $k = 1 \dots M$, is:

$$\text{If } x_1 \text{ is } B_1^{(k)}, \dots, \text{ and } x_N \text{ is } B_N^{(k)} \text{ then } y^{(k)} = \sum_{j=1}^N a_j^{(k)} x_j + a_0^{(k)} \quad (9)$$

where $\underline{x} = [x_1 \ x_2 \ \dots \ x_N]$ is the input to the network and $y^{(k)}$ is the output of the rule. The antecedent part of the rule depends on the Membership Functions (MFs) $\mu_{B_j^{(k)}}(x_j)$ of the fuzzy input variables $B_j^{(k)}$, $j = 1 \dots N$; the consequent part is determined by the coefficients $a_j^{(k)}$, $j = 0 \dots N$, of the crisp output $y^{(k)}$. By using standard options for composing the input MFs and combining the rule outputs [40], the output of the ANFIS network is represented by:

$$\tilde{y} = \frac{\sum_{k=1}^M \mu_{B^{(k)}}(\underline{x}) y^{(k)}}{\sum_{k=1}^M \mu_{B^{(k)}}(\underline{x})} \quad (10)$$

where $\tilde{y}$ is the estimation of y and $\mu_{B^{(k)}}(\underline{x})$ is the composed MF of the k -th rule.

The ANFIS learning of network coefficients is based on a classical back-propagation technique associated with a least-squares estimation [47].

2.3. HONFIS

The HONFIS model adopted for data regression is a Takagi–Sugeno (TS) fuzzy inference system that is a generalization of ANFIS where the consequent part in Equation (9) is a polynomial of order greater than one combining the input values x_j , $j = 1 \dots N$. It is made of M different fuzzy rules, and the coefficients are obtained through the use of a clustering procedure in the joint input-output space [48].

Let $\Gamma = \{\Gamma_1, \Gamma_2, \dots, \Gamma_M\}$ be a set of M clusters (each associated with a rule output), and let every pattern of the training set be assigned randomly to one of these clusters. Then, the clustering procedure is fundamentally an alternating optimization technique that aims at identifying the M cluster prototypes [44]. At the end of the iterations, a label q , $1 \leq q \leq M$, is associated with each pattern, representing the rule it has been assigned to during the last iteration. In this way, a classification model able to assign a fuzzy label $L(\underline{x})$ to any pattern $\underline{x}$ of the input space is obtained:

$$L(\underline{x}) = [\mu_{B^{(1)}}(\underline{x}) \ \mu_{B^{(2)}}(\underline{x}) \ \dots \ \mu_{B^{(M)}}(\underline{x})] \quad (11)$$

where the k -th element of $L(\underline{x})$ represents the fuzzy membership of the pattern to the k -th class, that is the firing strength $\mu_{\underline{B}^{(k)}}(\underline{x})$ in Equation (10).

In the following, we will adopt a K -nearest neighbor (K -NN) strategy for classification of the fuzzy label during the prediction tests. This is done by using the input value $\underline{x}$ only, and hence, the fuzzy label of $\underline{x}$ will be:

$$L(\underline{x}) = \frac{1}{K} \sum_{q=1}^K L(\underline{x}_{t_q}) \quad (12)$$

where $\underline{x}_{t_1}, \underline{x}_{t_2}, \dots, \underline{x}_{t_K}$ are the K patterns of the training set that score the smallest Euclidean distance from $\underline{x}$, with their fuzzy labels determined during the training phase.

The output $\tilde{y}$ is calculated through Equation (10) by means of the firing strengths contained in the fuzzy label $L(\underline{x})$ and the output consequents whose parameters have been previously determined by clustering in the joint input-output space. For instance, in the case of a 3rd order polynomial as adopted in the following, the k -th rule's consequent, $k = 1 \dots M$, is:

$$\begin{aligned} y^{(k)} = & a_{31}^{(k)} x_1^3 + \dots + a_{3N}^{(k)} x_N^3 + \\ & a_{21}^{(k)} x_1^2 + \dots + a_{2N}^{(k)} x_N^2 + \\ & a_{11}^{(k)} x_1 + \dots + a_{1N}^{(k)} x_N + a_0^{(k)} \end{aligned} \quad (13)$$

Regarding the generalization capability, the optimal number of rules is automatically determined on the basis of learning theory [49]. The best model is chosen by evaluating a cost function based on network complexity and approximation error [50]. As a measure of the network performance on the training set, the mean squared error (MSE) is adopted:

$$E = \frac{1}{N_T} \sum_{t=1}^{N_T} (y_t - \tilde{y}_t)^2 \quad (14)$$

where N_T is the number of samples in the training set to be predicted and $\tilde{y}_t$ is the output generated by HONFIS in correspondence with the t -th input pattern $\underline{x}_t$. The optimal network is selected by minimizing the following cost function that depends on the maximum allowed number of the HONFIS rules:

$$F(M) = (1 - \lambda) \frac{E(M) - E_{\min}}{E_{\max} - E_{\min}} + \lambda \frac{M}{N_T} \quad (15)$$

where $E_{\min}$ and $E_{\max}$ are the extreme values of the performance E that are encountered during the analysis of the different models, while λ is a non-critical weight, $0 \leq \lambda \leq 1$, set to 0.5 by default.

3. Results

The prediction performances of the considered models were investigated by means of several simulation tests carried out using data from the De Nittis Power Plant in the Apulia Region, Italy (latitude $\phi = 41^\circ 26' 16''$, longitude $\lambda = 15^\circ 45' 47''$). Data were relative to a single photovoltaic plant, organized with eight cabins with two inverters each. The output current was used as the variable to be predicted. The value was sampled at the source with a 5-min sample interval, and it was collected from 6:00 a.m.–10:55 p.m., resulting in 204 samples per day. The used data stream comes from a single inverter of a single cabin, and it is relative to the year 2012. In order to provide a statistical characterization of the handled time series, whose samples are measured in Amperes, we have computed the first four statistical moments of the whole 2012 dataset, whose histogram is also reported in Figure 1: mean 85.1291; variance 1.0244×10^4 ; skewness 0.8151; kurtosis 2.1082. Successively, the whole time series has been normalized linearly between zero and one in order to cope with the numerical requirements of learning algorithms, especially for neural network models.

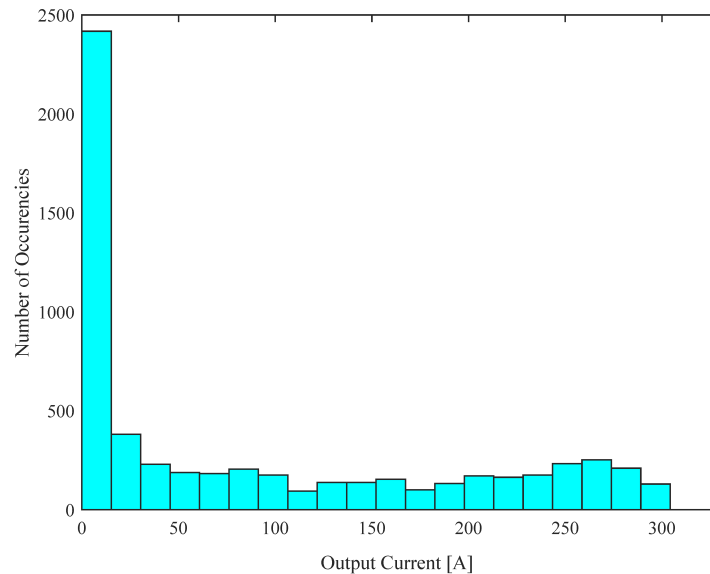


Figure 1. Histogram of the output current.

All of the computational models were trained using time series subsampled at an interval of 1 h, thus resulting in 17 samples per day. Four different kinds of training conditions were considered: 1-, 3-, 7- and 30-day, associated with training sets composed of 17, 51, 119 and 510 samples, respectively. Every computational model estimated by using one of these training sets was applied to test the day following the latest one in the training set, that is on a test set of 17 samples. The successive experiments would consider for testing one day for each month of 2012, and we have chosen the 15th of each month for the sake of uniformity (the last 15 samples of 2011 are also used for the 30-day training of 15 January). For the same reason, after the preliminary analysis we carried out for determining the embedding parameters, as the latter did not show a considerable sensitivity to seasonality and to the length of the training set, which was relatively small with respect to the usual length of time series processed by the AMI and FNN methods, we had adopted as the optimal choice the values $T = 5$ and $D = 3$ for every training set and the related experiment. We remark that all of the test sets had different irradiation and meteorological conditions, and thus, they could be considered as a good ensemble for representing the typical behaviors that might be encountered.

All of the experiments had been performed using MATLAB® 2016b, running on a 3.1-GHz Intel Core i7 platform equipped with 16 GB of memory. As previously mentioned, in addition to the three proposed neural and fuzzy neural models, the linear (LSE) and ARIMA predictors were adopted for benchmarking. LSE did not have parameters to be set in advance, while for ARIMA, RBF and ANFIS, we had adopted the default options provided by the software platform for training and model regularization (ARIMA, RBF and ANFIS models were trained by using the supported functions in the econometrics toolbox, neural network toolbox and fuzzy logic toolbox of MATLAB, respectively). Regarding HONFIS, a preliminary analysis had been performed in order to evaluate the best model order for the rule consequents. In the following, the results of the third-order model are reported, as it was able to obtain the best performance in almost all of the considered cases. The input space classification was obtained by a three-NN classifier (i.e., $K = 3$). The optimal number of rules was determined by Equation (15), using $\lambda = 0.5$ and varying M from 1–50% of N_T .

The prediction performances were measured by two metrics commonly adopted for energy time series, which were independent of the said procedure for data normalization:

- Normalized Mean Square Error (NMSE):

$$\text{NMSE} = \frac{\sum_{t=1}^{N_T} (y_t - \tilde{y}_t)^2}{\sum_{t=1}^{N_T} (y_t - \bar{y})^2} \quad (16)$$

where $\bar{y}$ is the average value of samples y_t in the test set;

- Mean Absolute Range Error (MARE):

$$\text{MARE} = \frac{1}{N_T} \sum_{t=1}^{N_T} \frac{|y_t - \tilde{y}_t|}{y_{\max} - y_{\min}} \quad (17)$$

where $y_{\max}$ and $y_{\min}$ are the maximum and minimum value of samples y_t in the test set, respectively.

The numerical results for each tested day are reported in Tables 1–12, where the performance of the best model is marked in bold font. As per the following discussion, being HONFIS the model that assures the best performance in most of the situations, the graphical illustration of the actual time series (blue line) and the one predicted by HONFIS (red line), over the four training conditions, is reported in Figures 2–13, respectively. In the x -axis is reported the cumulative index of samples of the considered day, starting from Index 1 for the first sample of 1 January 2012 (which is a leap year) and considering 17 samples per day. The output current reported in the y -axis is a dimensionless value between zero and one, as the whole dataset has been normalized before the model processing. Some negative values may occur because of possible numerical issues of a trained model when its performance is inadequate.

Table 1. Prediction results for 15 January .

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.4690	0.4429	0.7591	0.3523	0.0921	0.1312	0.0457	0.0884
LSE test	0.8230	0.4394	0.5463	0.4347	0.1843	0.1547	0.1542	0.1544
ARIMA training	1.9929	0.8843	0.6605	0.9852	0.3968	0.2910	0.2232	0.7345
ARIMA test	1.8852	0.5194	0.4838	0.2691	0.5822	0.2010	0.2222	0.1597
RBF training	0.1137	0.1393	0.1847	0.2901	0.0512	0.0687	0.0675	0.0801
RBF test	1.3472	0.2739	0.2959	0.4399	0.9586	0.1146	0.1134	0.1461
ANFIS training	0.0745	0.0005	0.2698	0.3383	0.0522	0.0040	0.0888	0.0910
ANFIS test	2.2393	0.0004	0.4415	0.5227	0.3895	0.0046	0.1597	0.1723
HONFIS (3rd ord.) training	0.0137	0.0005	0.0032	0.0650	0.0181	0.0027	0.0040	0.0174
HONFIS (3rd ord.) test	1.4820	0.0008	0.0009	0.0511	0.2373	0.0041	0.0044	0.0212

Table 2. Prediction results for 15 February.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.0978	0.2167	0.3480	0.2631	0.0906	0.1123	0.1355	0.1213
LSE test	0.4072	0.3690	0.3653	0.3786	0.1681	0.1411	0.1426	0.1527
ARIMA training	0.3354	0.7523	1.0269	1.2555	0.5194	0.2254	0.1786	0.1139
ARIMA test	0.4480	0.6653	1.4407	0.5348	0.2117	0.2257	0.3612	0.2266
RBF training	0.0060	0.0147	0.1239	0.1479	0.0253	0.0285	0.0784	0.0768
RBF test	0.3212	0.0056	0.0860	0.2660	0.1508	0.0219	0.0797	0.1129
ANFIS training	0.0001	0.0067	0.0894	0.0156	0.0003	0.0097	0.0581	0.0786
ANFIS test	0.5220	0.0071	0.0295	0.2602	0.1480	0.0113	0.0472	0.1147
HONFIS (3rd ord.) training	0.0081	0.0081	0.0067	0.0269	0.0002	0.0134	0.0065	0.0157
HONFIS (3rd ord.) test	0.7961	0.0052	0.0068	0.1235	0.1733	0.0094	0.0106	0.0644

Table 3. Prediction results for 15 March.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.0399	0.0428	0.1248	0.2093	0.0663	0.0682	0.0946	0.1061
LSE test	0.0393	0.0388	0.0412	0.0463	0.0657	0.0651	0.0673	0.0720
ARIMA training	0.6048	0.3498	2.3712	0.4085	0.2417	0.1752	0.0692	0.0688
ARIMA test	0.0091	0.0190	0.6537	0.5065	0.0292	0.0437	0.0426	0.0687
RBF training	0.0041	0.0002	0.0323	0.1225	0.0168	0.0037	0.0476	0.0715
RBF test	0.0063	0.0002	0.0153	0.0128	0.1508	0.0040	0.0355	0.0310
ANFIS training	0.0001	0.0005	0.0134	0.1268	0.0002	0.0011	0.0269	0.0720
ANFIS test	0.0021	0.0003	0.0086	0.0216	0.0120	0.0010	0.0208	0.0399
HONFIS (3rd ord.) training	0.0001	0.0001	0.0039	0.0012	0.0002	0.0012	0.0090	0.0035
HONFIS (3rd ord.) test	0.9437	0.0002	0.0062	0.0003	0.1985	0.0013	0.0129	0.0035

Table 4. Prediction results for 15 April.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.4307	0.4117	0.4059	0.2502	0.1193	0.1482	0.1322	0.1126
LSE test	0.4356	0.4205	0.4115	0.4252	0.1882	0.1851	0.1916	0.1981
ARIMA training	1.2368	3.8105	0.8170	0.6961	0.4845	0.3825	0.1961	0.2525
ARIMA test	1.2658	0.9881	0.2392	0.1691	0.3113	0.2767	0.1543	0.1216
RBF training	0.1533	0.1309	0.1145	0.1225	0.0900	0.0828	0.0647	0.0763
RBF test	1.2685	0.0754	0.1500	0.2697	0.2595	0.0720	0.1018	0.1506
ANFIS training	0.0002	0.0066	0.1469	0.1444	0.0184	0.0011	0.0803	0.0801
ANFIS test	2.3168	0.0036	0.2457	0.1991	0.0120	0.0171	0.1437	0.1252
HONFIS (3rd ord.) training	0.0025	0.0006	0.0039	0.0003	0.0096	0.0039	0.0068	0.0030
HONFIS (3rd ord.) test	1.6248	0.0004	0.0203	0.0003	0.3250	0.0037	0.0172	0.0016

Table 5. Prediction results for 15 May.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.7847	0.5205	0.3469	0.2998	0.1499	0.1671	0.1442	0.1290
LSE test	0.7678	0.5437	0.5586	0.5714	0.2479	0.1896	0.2006	0.2058
ARIMA training	6.9047	1.2811	0.4089	0.9808	0.5694	0.2333	0.1765	0.2757
ARIMA test	1.9263	1.5920	1.2429	0.3656	0.3716	0.3131	0.2694	0.2795
RBF training	0.2155	0.1940	0.1347	0.1404	0.0728	0.1066	0.0946	0.0854
RBF test	7.9004	0.1939	0.2250	0.2900	2.0610	0.1077	0.1327	0.1386
ANFIS training	0.0001	0.0080	0.0762	0.1718	0.0004	0.0098	0.0564	0.0972
ANFIS test	1.1388	0.0225	0.0957	0.3253	0.3458	0.0181	0.0605	0.1483
HONFIS (3rd ord.) training	0.0119	0.0218	0.0017	0.0005	0.0123	0.0041	0.0030	0.0027
HONFIS (3rd ord.) test	1.8555	0.0002	0.0003	0.0002	0.3427	0.0009	0.0016	0.0029

Table 6. Prediction results for 15 June.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.0108	0.0121	0.0096	0.0957	0.0363	0.0383	0.0318	0.0579
LSE test	0.0119	0.0118	0.0118	0.0137	0.0376	0.0375	0.0372	0.0361
ARIMA training	0.7354	1.3923	0.3083	0.3240	0.2934	0.3922	0.1314	0.1667
ARIMA test	0.8885	0.0115	0.0670	0.0205	0.3397	0.0327	0.0831	0.0473
RBF training	0.0003	0.0002	0.0002	0.0367	0.0054	0.0049	0.0046	0.0326
RBF test	0.0005	0.0002	0.0002	0.0075	0.0073	0.0049	0.0048	0.0266
ANFIS training	0.0001	0.0005	0.0003	0.0423	0.0005	0.0017	0.0053	0.0353
ANFIS test	0.0010	0.0006	0.0004	0.0118	0.0099	0.0019	0.0064	0.0303
HONFIS (3rd ord.) training	0.0001	0.0006	0.0001	0.0053	0.0001	0.0010	0.0022	0.0068
HONFIS (3rd ord.) test	0.1763	0.0004	0.0006	0.0004	0.0482	0.0008	0.0012	0.0045

Table 7. Prediction results for 15 July.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.0112	0.4319	0.1981	0.1226	0.0340	0.1576	0.0949	0.0700
LSE test	0.2471	0.3071	0.2432	0.2399	0.0905	0.1597	0.1117	0.0960
ARIMA training	1.0052	2.1191	0.2111	0.8671	0.5624	0.4436	0.1372	0.2310
ARIMA test	0.9954	0.1042	0.1541	0.1041	0.5820	0.0786	0.1170	0.0788
RBF training	0.0007	0.1733	0.0471	0.0538	0.0080	0.0848	0.0328	0.0438
RBF test	0.1519	0.1249	0.0698	0.0840	0.0909	0.0860	0.0452	0.0690
ANFIS training	0.0001	0.0004	0.0214	0.0707	0.0003	0.0315	0.0037	0.0501
ANFIS test	0.2438	0.0005	0.0397	0.0891	0.0002	0.0399	0.0054	0.0621
HONFIS (3rd ord.) training	0.0001	0.0008	0.0005	0.0023	0.0009	0.0053	0.0037	0.0044
HONFIS (3rd ord.) test	2.3358	0.0005	0.0011	0.0004	0.4013	0.0059	0.0054	0.0035

Table 8. Prediction results for 15 August.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.0295	0.0861	0.0406	0.1226	0.0517	0.0614	0.0442	0.0853
LSE test	0.1973	0.1941	0.1963	0.1927	0.0942	0.0868	0.0846	0.0984
ARIMA training	0.8843	0.2923	0.2207	0.2990	0.6698	0.1163	0.1179	0.1566
ARIMA test	0.9124	0.2968	0.1316	0.1081	0.7546	0.1786	0.1155	0.0747
RBF training	0.0020	0.0117	0.0071	0.0758	0.0143	0.0244	0.0122	0.0529
RBF test	0.1174	0.1249	0.0157	0.0841	0.0613	0.0297	0.0452	0.0614
ANFIS training	0.0001	0.0004	0.0214	0.0707	0.0003	0.0315	0.0037	0.0501
ANFIS test	0.2438	0.0005	0.0307	0.0891	0.0002	0.0399	0.0213	0.0621
HONFIS (3rd ord.) training	0.0001	0.0085	0.0075	0.0929	0.0147	0.0053	0.0102	0.0525
HONFIS (3rd ord.) test	0.0107	0.0005	0.0350	0.0701	0.0177	0.0059	0.0236	0.0448

Table 9. Prediction results for 15 September.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.7511	0.3141	0.2900	0.2971	0.1845	0.1280	0.1281	0.1192
LSE test	0.7526	0.7263	0.7265	0.7262	0.2181	0.2088	0.2099	0.2114
ARIMA training	1.3369	1.3875	0.9725	0.3571	0.4424	0.2483	0.2423	0.1741
ARIMA test	0.4237	0.6639	0.5273	0.3427	0.3354	0.2334	0.1888	0.1604
RBF training	0.2025	0.1243	0.1076	0.1840	0.1016	0.0855	0.0708	0.0883
RBF test	1.2412	0.2118	0.3201	0.3445	0.3014	0.1155	0.1379	0.1362
ANFIS training	0.0001	0.0001	0.0498	0.1953	0.0001	0.0001	0.0467	0.0903
ANFIS test	0.4766	0.0001	0.0633	0.4668	0.1738	0.0008	0.0529	0.1582
HONFIS (3rd ord.) training	0.0001	0.0001	0.0122	0.0032	0.0001	0.0010	0.0138	0.0050
HONFIS (3rd ord.) test	1.7032	0.0002	0.0068	0.0092	0.3168	0.0011	0.0136	0.0125

Table 10. Prediction results for 15 October.

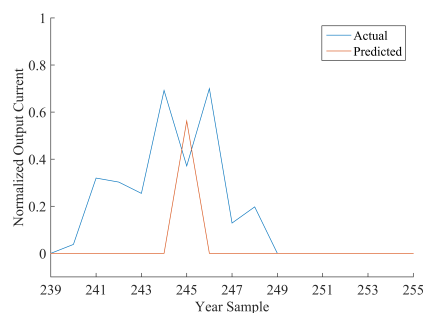
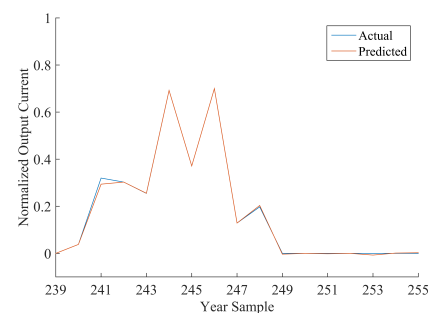
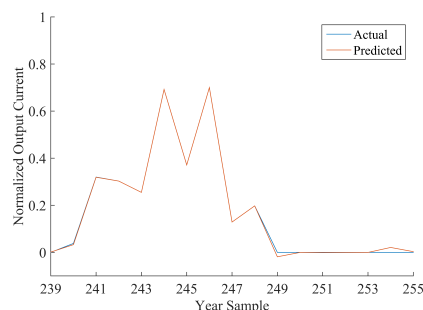
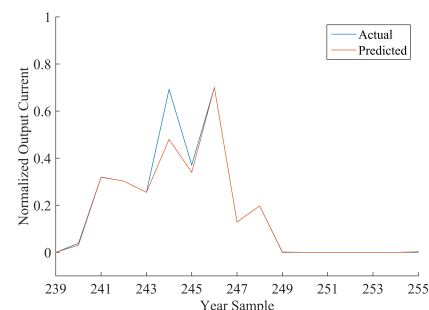
Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.1889	0.1198	0.0924	0.2720	0.0947	0.0983	0.0942	0.1047
LSE test	0.2071	0.1901	0.1899	0.1986	0.1108	0.1180	0.1194	0.1121
ARIMA training	1.5475	1.0569	0.5870	0.6301	0.3616	0.2311	0.1772	0.1965
ARIMA test	0.8957	0.6734	1.3392	0.3758	0.2439	0.2097	0.3119	0.1545
RBF training	0.0182	0.0231	0.0057	0.1583	0.0291	0.0426	0.0184	0.0677
RBF test	0.5444	0.0333	0.0126	0.1006	0.1772	0.0491	0.0273	0.0862
ANFIS training	0.0002	0.0001	0.0048	0.1692	0.0002	0.0003	0.0160	0.0702
ANFIS test	1.0372	0.0003	0.0086	0.1574	0.2217	0.0004	0.0188	0.1019
HONFIS (3rd ord.) training	0.0029	0.0036	0.0010	0.0145	0.0077	0.0078	0.0048	0.0112
HONFIS (3rd ord.) test	1.6475	0.0065	0.0004	0.0008	0.3124	0.0132	0.0030	0.0036

Table 11. Prediction results for 15 November.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.8224	0.4429	0.7591	0.3523	0.1139	0.1312	0.0457	0.0884
LSE test	0.5896	0.4394	0.5463	0.4347	0.1575	0.1547	0.1542	0.1544
ARIMA training	1.0569	0.7870	0.6644	1.1759	0.2311	0.2698	0.1937	0.2855
ARIMA test	0.6734	0.9403	4.4649	0.9554	0.2097	0.2776	1.2247	0.2504
RBF training	0.0725	0.1662	0.4894	0.2160	0.0335	0.0868	0.0379	0.0638
RBF test	8.5147	0.1844	0.9832	3.1527	0.6348	0.0913	0.2168	0.3780
ANFIS training	0.0001	0.0247	0.0480	0.2363	0.0001	0.0323	0.0650	0.0687
ANFIS test	5.5850	0.0197	0.0599	2.6863	0.4980	0.0294	0.0688	0.3969
HONFIS (3rd ord.) training	0.0009	0.0566	0.1659	0.0911	0.0007	0.0219	0.0048	0.0279
HONFIS (3rd ord.) test	2.3314	0.0433	0.6480	0.3782	0.5552	0.0349	0.1045	0.1206

Table 12. Prediction results for 15 December.

Prediction Model	NMSE				MARE			
	1-Day	3-Day	7-Day	30-Day	1-Day	3-Day	7-Day	30-Day
LSE training	0.6335	0.5107	0.5513	0.3814	0.1109	0.1110	0.0970	0.0984
LSE test	0.4326	0.3922	0.4051	0.4512	0.1436	0.1552	0.1526	0.1518
ARIMA training	1.0407	1.1536	1.9333	1.1766	0.3000	0.2067	0.2194	0.2016
ARIMA test	0.9998	0.8886	0.5563	0.4208	0.2272	0.2090	0.1557	0.1769
RBF training	0.0754	0.0379	0.0540	0.2106	0.0412	0.0481	0.0383	0.0665
RBF test	0.9442	0.0793	0.1061	0.1270	0.2207	0.0768	0.0627	0.0684
ANFIS training	0.0609	0.0043	0.0556	0.2272	0.0401	0.0064	0.0410	0.0719
ANFIS test	1.2890	0.0057	0.1005	0.1443	0.2599	0.0074	0.0639	0.0643
HONFIS (3rd ord.) training	0.0060	0.0039	0.0198	0.1012	0.0083	0.0069	0.0094	0.0316
HONFIS (3rd ord.) test	1.1232	0.0079	0.0544	0.1017	0.1951	0.0083	0.0298	0.0416

**(a)** Prediction behavior for one-day training**(b)** Prediction behavior for three-day training**(c)** Prediction behavior for seven-day training**(d)** Prediction behavior for 30-day training**Figure 2.** Prediction using the best model (HONFIS) for 15 January.

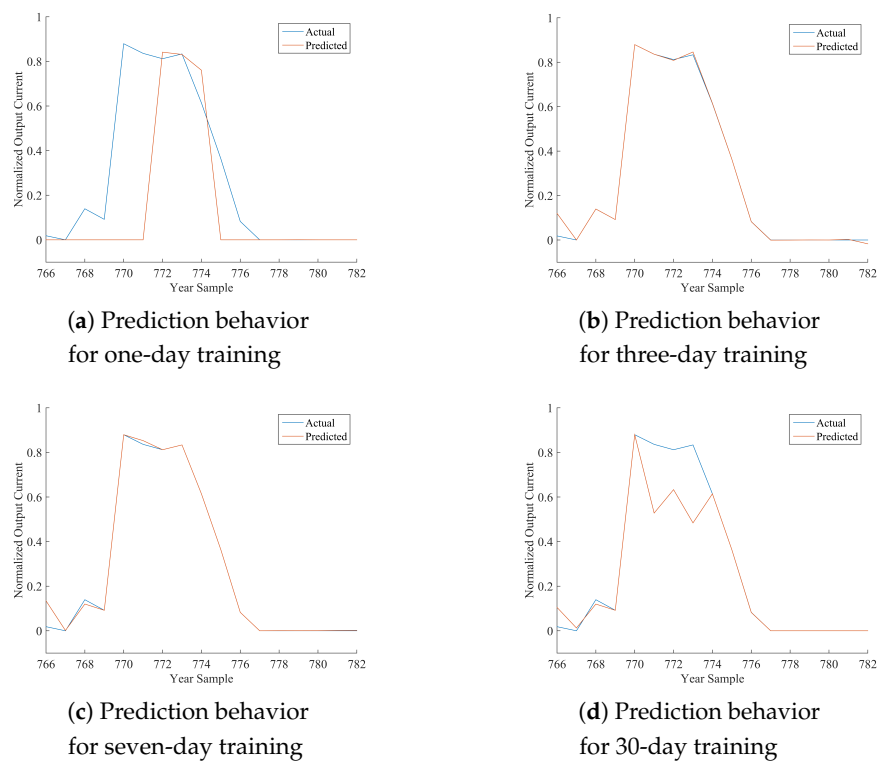


Figure 3. Prediction using the best model (HONFIS) for 15 February.

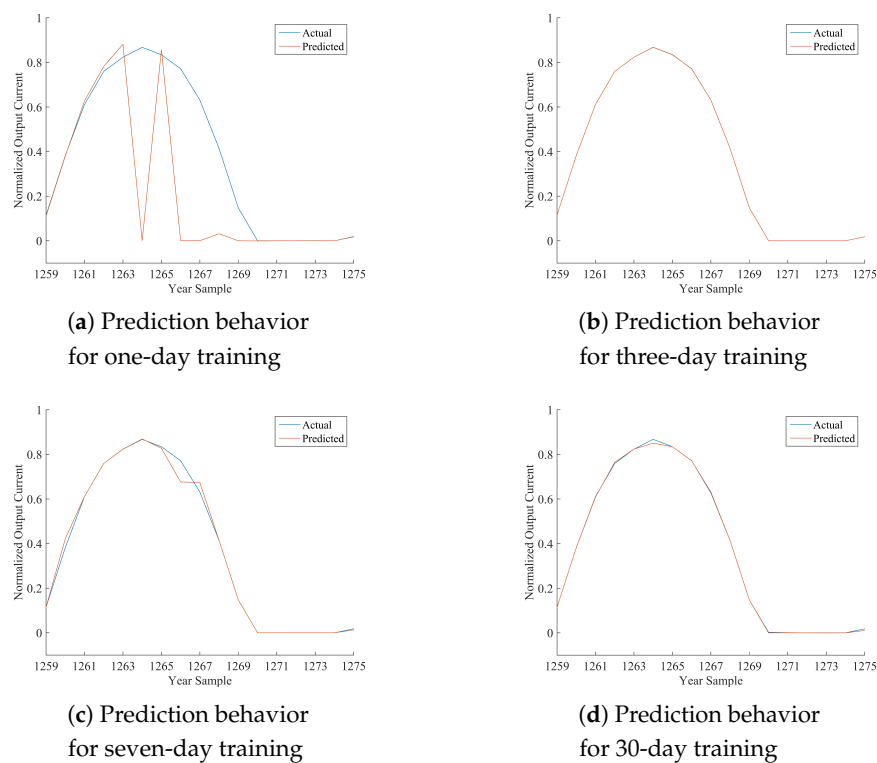
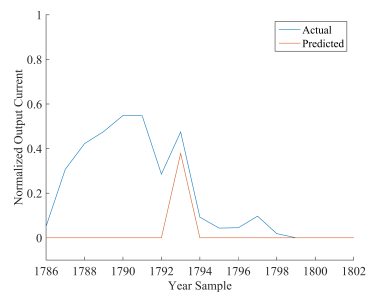
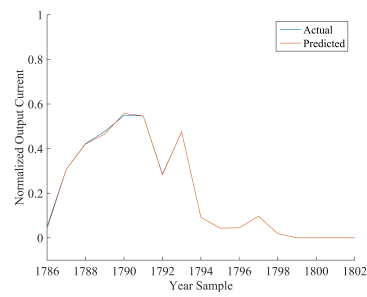


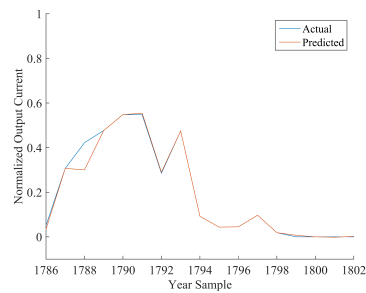
Figure 4. Prediction using the best model (HONFIS) for 15 March.



(a) Prediction behavior for one-day training



(b) Prediction behavior for three-day training

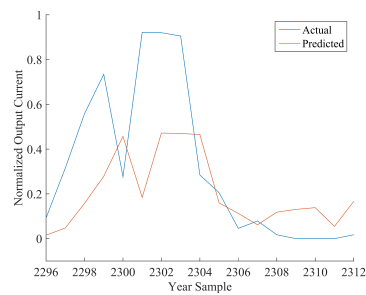


(c) Prediction behavior for seven-day training

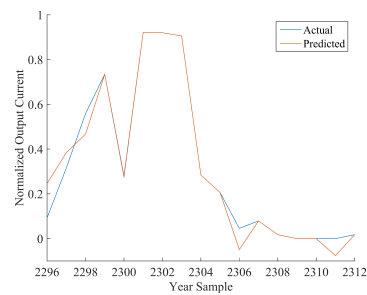


(d) Prediction behavior for 30-day training

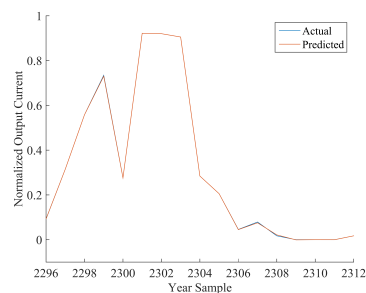
Figure 5. Prediction using the best model (HONFIS) for 15 April.



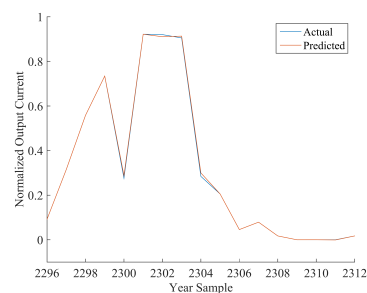
(a) Prediction behavior for one-day training



(b) Prediction behavior for three-day training



(c) Prediction behavior for seven-day training



(d) Prediction behavior for 30-day training

Figure 6. Prediction using the best model (HONFIS) for 15 May.

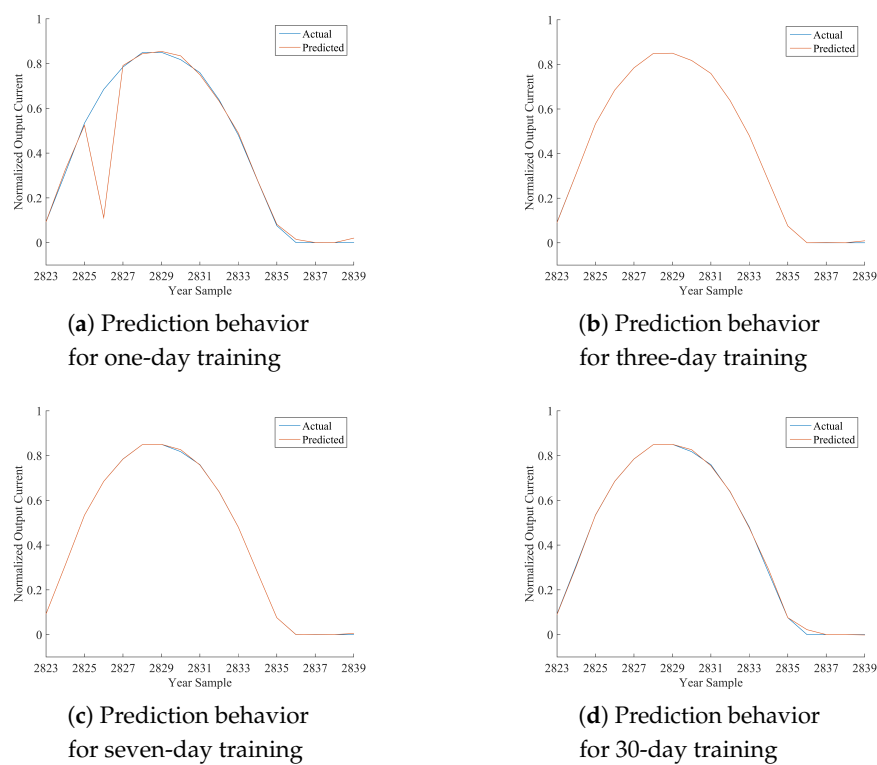


Figure 7. Prediction using the best model (HONFIS) for 15 June.

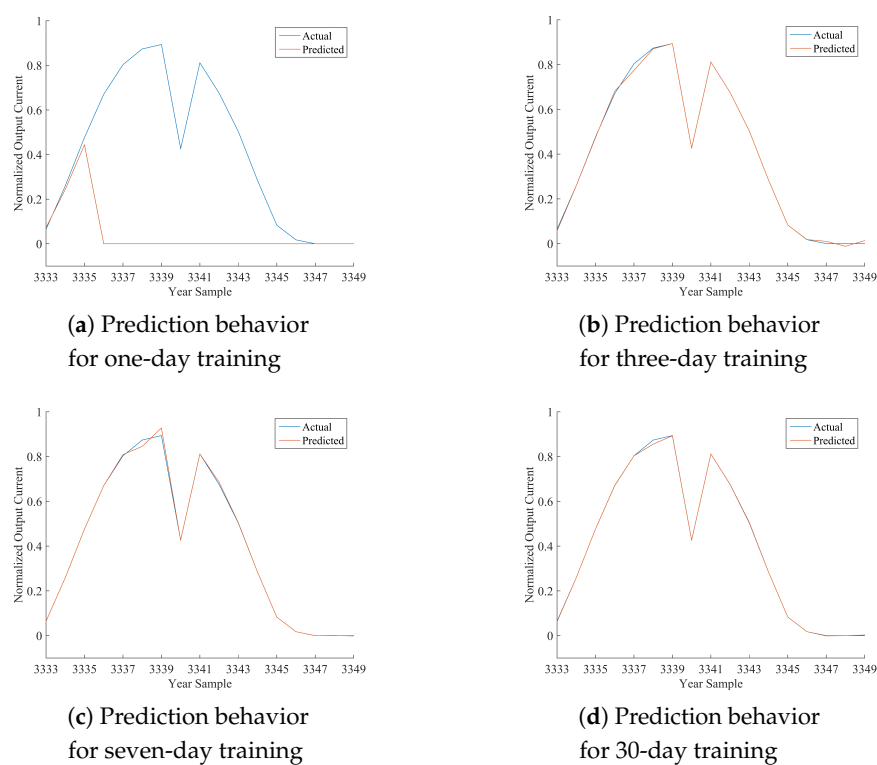
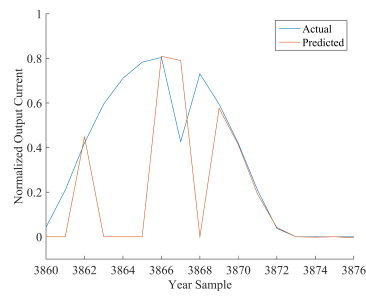
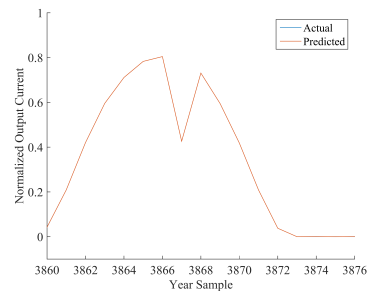


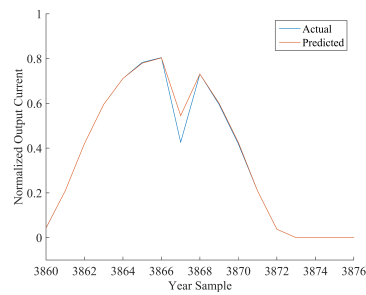
Figure 8. Prediction using the best model (HONFIS) for 15 July.



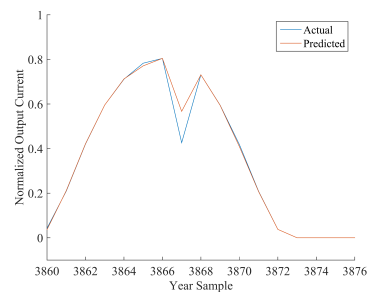
(a) Prediction behavior for one-day training



(b) Prediction behavior for three-day training

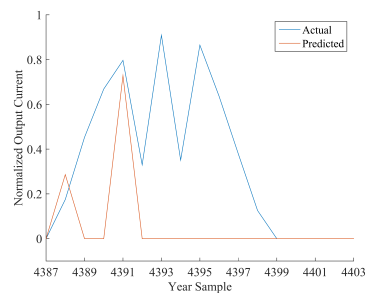


(c) Prediction behavior for seven-day training

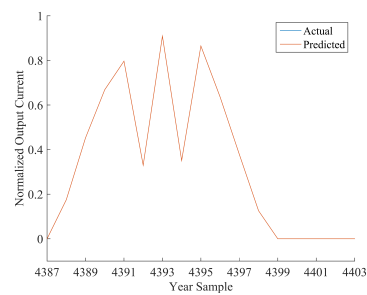


(d) Prediction behavior for 30-day training

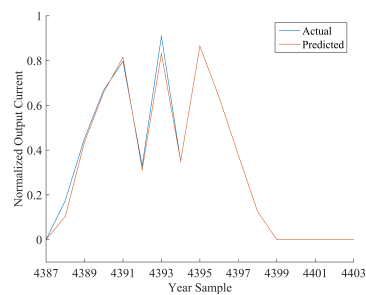
Figure 9. Prediction using the best model (HONFIS) for 15 August.



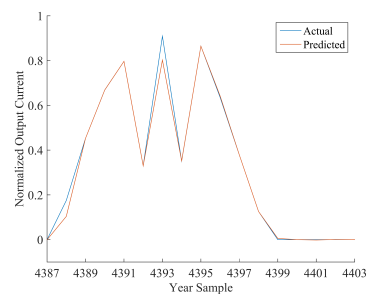
(a) Prediction behavior for one-day training



(b) Prediction behavior for three-day training

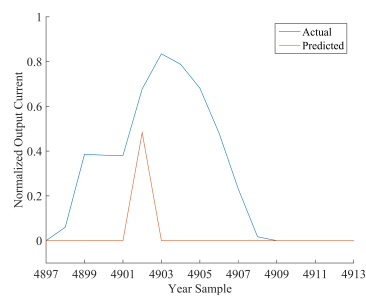


(c) Prediction behavior for seven-day training

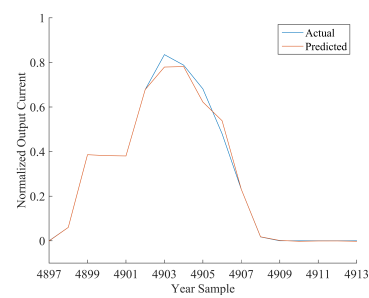


(d) Prediction behavior for 30-day training

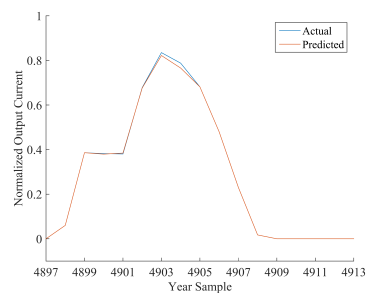
Figure 10. Prediction using the best model (HONFIS) for 15 September.



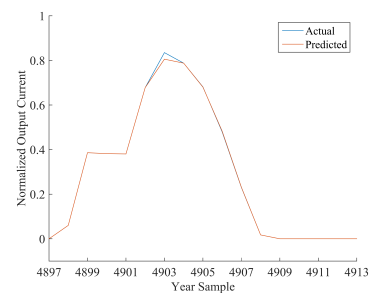
(a) Prediction behavior for one-day training



(b) Prediction behavior for three-day training

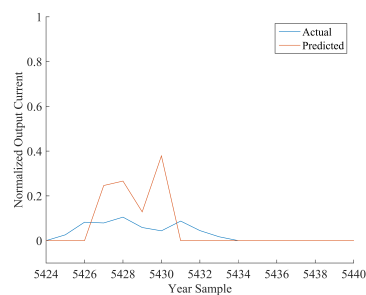


(c) Prediction behavior for seven-day training

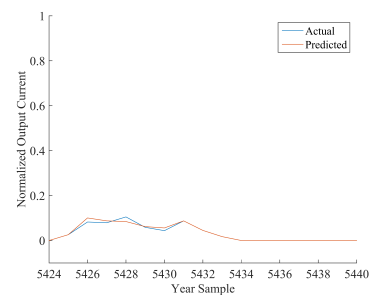


(d) Prediction behavior for 30-day training

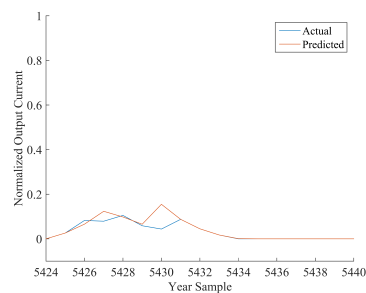
Figure 11. Prediction using the best model (HONFIS) for 15 October.



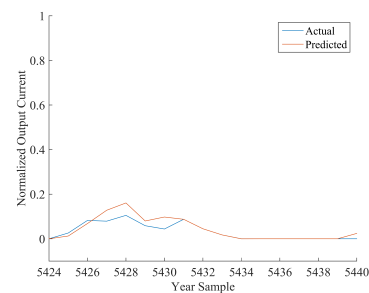
(a) Prediction behavior for one-day training



(b) Prediction behavior for three-day training



(c) Prediction behavior for seven-day training



(d) Prediction behavior for 30-day training

Figure 12. Prediction using the best model (HONFIS) for 15 November.

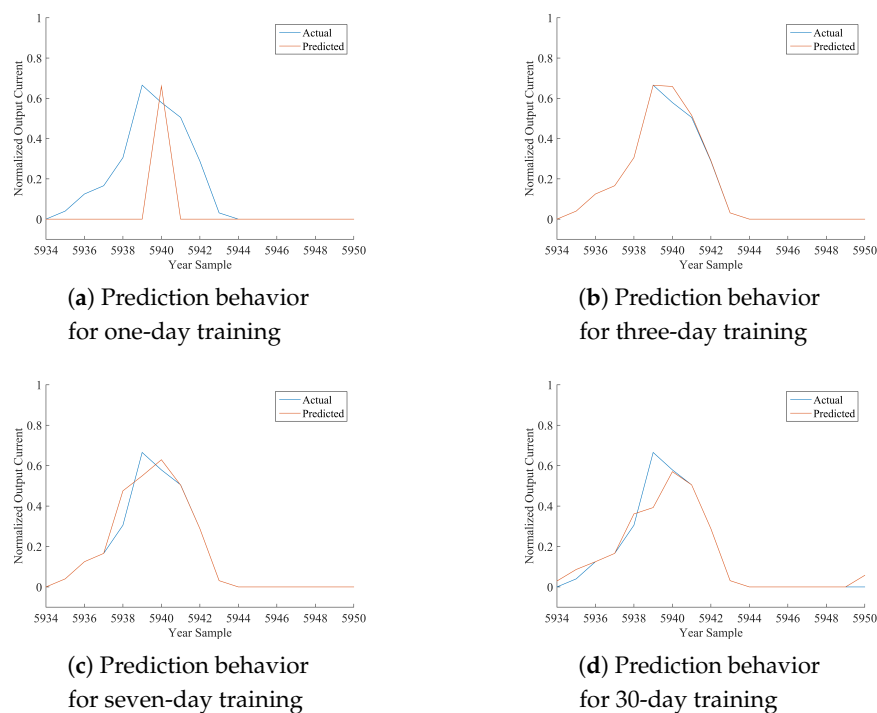


Figure 13. Prediction using the best model (HONFIS) for 15 December.

It is important to remark that the number of samples that should be predicted in a test set is strictly related to the computational cost of the learning procedure, as well as to the desired horizon of predictability, for which it is necessary to follow a given strategy for the energy management and to broadcast the relevant information to the involved operators. For instance, as the learning times of the previous models were in the order of some minutes and the considered time series are subsampled at 1 h, any model could be re-trained on a hourly basis for a new sample to be predicted (i.e., using a test set with one sample only), thus incorporating the new information given by the the latest available sample that was measured.

However, in order to ensure a wide range of action, it is worth forecasting more than one sample at a time, as previously done considering an entire day. Although longer test sets might not be so useful and accurate, we consider in the following also weekly test sets, so as to report the prediction results for more days in a year and to evaluate the sensitivity of the proposed approach not only to different training, but also to different test conditions. The numerical results for each tested week are reported in Tables 13–24. In such a case, we have considered seven-day or 30-day training sets only, that is longer than the test set or of the same length. As the starting day in the test set is still the 15th of each month, the training results are the same as the ones reported in Tables 1–12, respectively.

Table 13. Prediction results from 15–21 January.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.6304	0.6464	0.0720	0.0739
ARIMA test	1.2130	1.2252	0.1192	0.0957
RBF test	5.9196	0.5863	0.1500	0.0734
ANFIS test	1.0452	0.5863	0.1239	0.0800
HONFIS (3rd ord.) test	3.4869	0.2296	0.1651	0.0185

Table 14. Prediction results from 15–21 February.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.2490	0.2390	0.1242	0.1239
ARIMA test	1.4324	1.0998	0.3048	0.2697
RBF test	1.4771	0.1224	0.2359	0.0758
ANFIS test	0.6664	0.1281	0.1895	0.0755
HONFIS (3rd ord.) test	1.0124	0.0197	0.2236	0.0147

Table 15. Prediction results from 15–21 March.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.1402	0.1414	0.0998	0.1007
ARIMA test	0.4567	0.2935	0.1934	0.1669
RBF test	0.2806	0.0801	0.1298	0.0657
ANFIS test	0.2732	0.0945	0.1091	0.0667
HONFIS (3rd ord.) test	3.4978	0.0008	0.3570	0.0039

Table 16. Prediction results from 15–21 April.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.4062	0.4418	0.1493	0.1627
ARIMA test	0.7751	0.6039	0.2667	0.2392
RBF test	1.1634	0.1780	0.1938	0.1004
ANFIS test	0.2311	0.1044	0.3102	0.1223
HONFIS (3rd ord.) test	1.8455	0.0004	0.3343	0.0019

Table 17. Prediction results from 15–21 May.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.4062	0.3882	0.1493	0.1299
ARIMA test	0.6885	0.4344	0.2492	0.1996
RBF test	4.4347	0.1571	0.1298	0.0773
ANFIS test	0.2962	0.1044	0.1109	0.0778
HONFIS (3rd ord.) test	1.8219	0.0003	0.3592	0.0033

Table 18. Prediction results from 15–21 June.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.2334	0.2295	0.1031	0.0981
ARIMA test	0.3503	0.3825	0.1124	0.1502
RBF test	0.9345	0.1238	0.2062	0.0688
ANFIS test	0.5972	0.1655	0.1650	0.0818
HONFIS (3rd ord.) test	0.9560	0.01213	0.1935	0.0108

Table 19. Prediction results from 15–21 July.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.1932	0.1886	0.0712	0.0725
ARIMA test	0.4167	0.4025	0.1918	0.1839
RBF test	9.8115	0.0695	0.4956	0.0400
ANFIS test	0.4592	0.0854	0.1229	0.045
HONFIS (3rd ord.) test	1.0701	0.0057	0.1905	0.0053

Table 20. Prediction results from 15–21 August.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.1445	0.1408	0.0750	0.0815
ARIMA test	0.1283	0.1487	0.0620	0.0763
RBF test	6.2338	0.0521	1.2980	0.0473
ANFIS test	0.1523	0.0849	0.0691	0.0478
HONFIS (3rd ord.) test	0.9875	0.0045	0.1884	0.0065

Table 21. Prediction results from 15–21 September.

Prediction model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.1803	0.1806	0.1032	0.1023
ARIMA test	0.7250	0.3781	0.2216	0.1668
RBF test	0.4751	0.0898	0.1671	0.0670
ANFIS test	1.7361	0.1031	0.2823	0.0678
HONFIS (3rd ord.) test	1.8185	0.0017	0.3105	0.0048

Table 22. Prediction results from 15–21 October.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.2266	0.2138	0.1169	0.1098
ARIMA test	0.8376	0.3808	0.2021	0.1626
RBF test	1.6818	0.0898	0.2237	0.0670
ANFIS test	1.0212	0.1225	0.1782	0.0650
HONFIS (3rd ord.) test	0.8892	0.0179	0.1884	0.0169

Table 23. Prediction results from 15–21 November.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.4126	0.3083	0.1274	0.1166
ARIMA test	1.2414	1.3639	0.2193	0.2285
RBF test	2.6564	0.1509	0.2237	0.0686
ANFIS test	1.3567	0.1445	0.1766	0.1630
HONFIS (3rd ord.) test	1.6437	0.1159	0.2320	0.0347

Table 24. Prediction results from 15–21 December.

Prediction Model	NMSE		MARE	
	7-Day	30-Day	7-Day	30-Day
LSE test	0.3800	0.3790	0.1347	0.1348
ARIMA test	1.3209	1.1432	0.2322	0.2059
RBF test	3.2432	0.2143	0.2649	0.0823
ANFIS test	0.4764	0.2168	0.1253	0.0886
HONFIS (3rd ord.) test	3.6065	0.0684	0.2519	0.0364

4. Discussion

Considering the experimental results reported in this paper, it can be noted that all of the proposed approaches are suitable for the prediction of the considered time series, depending on the chosen training set. The numerical results in terms of either NMSE or MARE basically agree, and they are coherent with the graphical behaviors, reported in the figures, between actual and predicted time series. Note that a perfect match is obtained when the NMSE is smaller than about 5×10^{-4} .

In the days with the most stable meteorological conditions (i.e., 15 March or 15 June), the prediction results for daily test sets are better than the others. Furthermore, there are differences among the results associated with diverse training procedures. The tests with one-day training sets have a high variability for all of the algorithms because the prediction is highly affected by the difference in irradiation between the day of the training and the day of the test. In fact, when the meteorological condition varies greatly from the training day to the sequent test day, the algorithms are trained on a set that is almost not correlated with the test set. It can also be noted that when the irradiation is very similar (no clouds, good weather), the performance is very good also for the one-day training set.

A similar discussion can be made, with smaller variability, for the three-day training sets. It has to be noted that we have inserted the one-day and three-day training sets mainly to show the sensitivity of the models to different training conditions. The results for seven-day training sets are much more stable for all of the algorithms and the days. The results for 30-day training sets are stable as well, but the sequences are much longer; thus, if the days are variable in terms of meteorological conditions, the model has a worse performance because the training process is more difficult. Training sets longer than 30 days were also considered in some preliminary tests, and the results confirm such a trend for which the performance decreases progressively as much as the length of the training set increases.

Overall, a one-day training is based on a very small number of samples in the training set, which is interesting for the computational cost of the learning procedure but it is an issue when many model parameters must be estimated, which is the well-known curse of dimensionality for neural networks. In fact, the third-order polynomial adopted for HONFIS makes it the most complex model among the ones considered in this paper, and its performances improve with respect to the other models as much as the number of samples increases in the training set, although a larger training set may not be the optimal choice.

Although of relative usefulness, the numerical results for weekly test sets are similar to the previous ones. In the case of seven-day training sets, all of the neural models suffer from the curse of dimensionality, as the information given by a training set of the same length of the test set is too small to ensure a robust estimation of the model parameters. In such cases, the LSE predictor yields better performances than the others models albeit too shallow in absolute terms. In the case of 30-day training sets, HONFIS is able to obtain the best results for weekly test sets, with a numerical score in terms of both NMSE and MARE that is stable enough with respect to shorter test sets of one day only.

The performance of the prediction is highly affected by the intrinsic seasonality of the time series considered. Anyway, HONFIS achieves the best results for almost all of the training sets with respect to the other proposed neural models, and all of them outperform both the LSE benchmark

and the ARIMA approach, which is not a feasible solution in this case mainly for the intrinsic chaotic properties of the sequence. This reinforces the fact that the prediction of photovoltaic production is a promising field for the application of neural and fuzzy neural approaches along with the use of a suitable embedding procedure.

5. Conclusions

In this paper, we presented new embedding approaches based on neural and fuzzy neural networks that have been properly tailored to be efficiently applied to PV time series prediction. The models were applied to the time series of sampled output current of a test PV plant, predicting a single day of operation. The models were trained using different training sets, namely 1-day, 3-day, 7-day or 30-day.

The obtained results demonstrate that in normal operation conditions, all of the solutions are suitable for the proposed goal, except for the shorter training sets, which result in less reliable performances. In particular, the approaches were shown to be very accurate when considering a seven-day training set to forecast the successive day, mostly because in a single-week window, the solar irradiation is more stable.

The results are very promising and suggest several opportunities for future work. We could make use of detrending techniques, such as mean-reverting approaches, in order to remove seasonal differences and spike outliers, as well as to improve the training accuracy. Additionally, it could be useful to test distributed learning approaches [51], by which the results could be improved sharing the data from different cabins of the same plant or from different nearby plants.

Author Contributions: Massimo Panella and Rodolfo Araneo conceived of the theoretical approach and designed the experiments. Antonello Rosato and Rosa Altilio performed the experiments and analyzed the data, after having organized the materials and set up the analysis tools. All of the authors wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

The following main nomenclature is adopted in the paper:

$S(t)$:	actual sample of the energy time series (output current) at time t .
$\tilde{S}(t)$:	predicted sample of the energy time series at time t .
$f(\underline{x})$:	parametric model used to solve the prediction problem by data regression.
$\underline{x}_t$:	embedded vector feeding the input of $f(\underline{x})$ that estimates the predicted sample $\tilde{S}(t)$.
y_t :	output at $f(\underline{x})$ representing the predicted sample $\tilde{S}(t)$.
D :	embedding dimension, that is the number of samples of the time series in the embedded vector $\underline{x}_t$.
T :	time lag between the samples of the time series that are embedded in $\underline{x}_t$.
LSE:	linear regression model for $f(\underline{x})$.
ARIMA:	Auto regressive integrated moving average regression model for $f(\underline{x})$.
RBF:	Radial basis function regression model for $f(\underline{x})$.
ANFIS:	Adaptive neuro-fuzzy inference system regression model for $f(\underline{x})$.
HONFIS:	Higher-order neuro-fuzzy inference system regression model for $f(\underline{x})$.
NMSE:	Normalized mean square error measure.
MARE:	Mean absolute range error measure.

References

1. Lopes, J.P.; Hatziargyriou, N.; Mutale, J.; Djapic, P.; Jenkins, N. Integrating distributed generation into electric power systems: A review of drivers, challenges and opportunities. *Electr. Power Syst. Res.* **2007**, *77*, 1189–1203.
2. Dio, V.D.; Favuzza, S.; Cascia, D.L.; Massaro, F.; Zizzo, G. Critical assessment of support for the evolution of photovoltaics and feed-in tariff(s) in Italy. *Sustain. Energy Technol. Assess.* **2015**, *9*, 95–104.
3. Du, Y.; Lu, D.D.C.; James, G.; Cornforth, D.J. Modeling and analysis of current harmonic distortion from grid connected PV inverters under different operating conditions. *Sol. Energy* **2013**, *94*, 182–194.
4. Samadi, A.; Eriksson, R.; der, L.S.O.; Rawn, B.G.; Boemer, J.C. Coordinated active power-dependent voltage regulation in distribution grids with PV systems. *IEEE Trans. Power Del.* **2014**, *29*, 1454–1464.
5. Han, X.; Liao, S.; Ai, X.; Yao, W.; Wen, J. Determining the minimal power capacity of energy storage to accommodate renewable generation. *Energies* **2017**, *10*, 468.
6. Inman, R.H.; Pedro, H.T.; Coimbra, C.F. Solar forecasting methods for renewable energy integration. *Prog. Energy Combust. Sci.* **2013**, *39*, 535–576.
7. Garrigle, E.M.; Leahy, P. Quantifying the value of improved wind energy forecasts in a pool-based electricity market. *Renew. Energy* **2015**, *80*, 517–524.
8. Liu, N.; Tang, Q.; Zhang, J.; Fan, W.; Liu, J. A hybrid forecasting model with parameter optimization for short-term load forecasting of micro-grids. *Appl. Energy* **2014**, *129*, 336–345.
9. Kardakos, E.G.; Alexiadis, M.C.; Vagropoulos, S.I.; Simoglou, C.K.; Biskas, P.N.; Bakirtzis, A.G. Application of time series and artificial neural network models in short-term forecasting of PV power generation. In Proceedings of the 2013 48th International Universities' Power Engineering Conference (UPEC), Dublin, Ireland, 2–5 September 2013; pp. 1–6.
10. Clements, A.; Hurn, A.; Li, Z. Strategic bidding and rebidding in electricity markets. *Energy Econ.* **2016**, *59*, 24–36.
11. Raza, M.Q.; Nadarajah, M.; Ekanayake, C. On recent advances in PV output power forecast. *Sol. Energy* **2016**, *136*, 125–144.
12. Antonanzas, J.; Osorio, N.; Escobar, R.; Urraca, R.; de Pison, F.M.; Antonanzas-Torres, F. Review of photovoltaic power forecasting. *Sol. Energy* **2016**, *136*, 78–111.
13. Al-Messabi, N.; Goh, C.; Li, Y. Heuristic grey-box modelling for photovoltaic power systems. *Syst. Sci. Control Eng.* **2016**, *4*, 235–246.
14. Coyle, D.; Prasad, G.; McGinnity, T.M. A time-series prediction approach for feature extraction in a brain-computer interface. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2005**, *13*, 461–467.
15. Bennett, C.; Stewart, R.A.; Lu, J. Autoregressive with exogenous variables and Neural Network short-term load forecast models for residential low voltage distribution networks. *Energies* **2014**, *7*, 2938–2960.
16. Tzafestas, S.; Tzafestas, E. Computational intelligence techniques for short-term electric load forecasting. *J. Intell. Robot. Syst.* **2001**, *31*, 7–68.
17. Proietti, A.; Panella, M.; Leccese, F.; Svezia, E. Dust detection and analysis in museum environment based on pattern recognition. *Measurement* **2015**, *66*, 62–72.
18. Safi, S.; Zeroual, A.; Hassani, M. Prediction of global daily solar radiation using higher order statistics. *Renew. Energy* **2002**, *27*, 647–666.
19. Kaplanis, S.; Kaplani, E. Stochastic prediction of hourly global solar radiation for Patra, Greece. *Appl. Energy* **2010**, *87*, 3748–3758.
20. Iqdour, R.; Zeroual, A. Prediction of daily global solar radiation using fuzzy systems. *Int. J. Sustain. Energy* **2007**, *26*, 19–29.
21. Voyant, C.; Muselli, M.; Paoli, C.; Nivet, M.L. Optimization of an artificial neural network dedicated to the multivariate forecasting of daily global radiation. *Energy* **2011**, *36*, 348–359.
22. Wang, F.; Mi, Z.; Su, S.; Zhao, H. Short-term solar irradiance forecasting model based on artificial Neural Network using statistical feature parameters. *Energies* **2012**, *5*, 1355–1370.
23. Mellit, A.; Kalogirou, S.A. Artificial intelligence techniques for photovoltaic applications: A review. *Prog. Energy Combust. Sci.* **2008**, *34*, 574–632.
24. Hocaoglu, F.O.; Gerek, O.N.; Kurban, M. Hourly solar radiation forecasting using optimal coefficient 2-D linear filters and feed-forward neural networks. *Sol. Energy* **2008**, *82*, 714–726.

25. Chen, C.; Duan, S.; Cai, T.; Liu, B. Online 24-h solar power forecasting based on weather type classification using artificial neural network. *Sol. Energy* **2011**, *85*, 2856–2870.
26. Monteiro, C.; Fernandez-Jimenez, L.A.; Ramirez-Rosado, I.J.; Muñoz-Jimenez, A.; Lara-Santillan, P.M. Short-term forecasting models for photovoltaic plants: analytical versus soft-computing techniques. *Math. Probl. Eng.* **2013**, *2013*, 767284.
27. Izgi, E.; Öztöpal, A.; Yerli, B.; Kaymak, M.K.; Şahin, A.D. Short-mid-term solar power prediction by using artificial neural networks. *Sol. Energy* **2012**, *86*, 725–733.
28. Fernandez-Jimenez, L.A.; Munoz-Jimenez, A.; Falces, A.; Mendoza-Villena, M.; Garcia-Garrido, E.; Lara-Santillan, P.M.; Zorzano-Alba, E.; Zorzano-Santamaria, P.J. Short-term power forecasting system for photovoltaic plants. *Renew. Energy* **2012**, *44*, 311–317.
29. Barbieri, F.; Rajakaruna, S.; Ghosh, A. Very short-term photovoltaic power forecasting with cloud modeling: A review. *Renew. Sustain. Energy Rev.* **2017**, *75*, 242–263.
30. Pierro, M.; Bucci, F.; Felice, M.D.; Maggioni, E.; Moser, D.; Perotto, A.; Spada, F.; Cornaro, C. Multi-Model Ensemble for day ahead prediction of photovoltaic power generation. *Sol. Energy* **2016**, *134*, 132–146.
31. Chu, Y.; Urquhart, B.; Gohari, S.M.; Pedro, H.T.; Kleissl, J.; Coimbra, C.F. Short-term reforecasting of power output from a 48 MWe solar PV plant. *Sol. Energy* **2015**, *112*, 68–77.
32. Takeda, H. Short-term ensemble forecast for purchased photovoltaic generation. *Sol. Energy* **2017**, *149*, 176–187.
33. Panella, M. Advances in biological time series prediction by neural networks. *Biomed. Signal Process. Control* **2011**, *6*, 112–120.
34. Mirowski, P.; Chen, S.; Ho, T.K.; Yu, C.N. Demand forecasting in smart grids. *Bell Labs Tech. J.* **2014**, *18*, 135–158.
35. Rosato, A.; Altilio, R.; Araneo, R.; Panella, M. Embedding of time series for the prediction in photovoltaic power plants. In Proceedings of the IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC), Florence, Italy, 7–10 June 2016; pp. 1–4.
36. Ko, C.N.; Lee, C.M. Short-term load forecasting using SVR (support vector regression)-based radial basis function neural network with dual extended Kalman filter. *Energy* **2013**, *49*, 413–422.
37. Osório, G.; Matias, J.; Ao, J.C. Short-term wind power forecasting using adaptive neuro-fuzzy inference system combined with evolutionary particle swarm optimization, wavelet transform and mutual information. *Renew. Energy* **2015**, *75*, 301–307.
38. Castro, J.R.; Castillo, O.; Sanchez, M.A.; Mendoza, O.; Rodríguez-Díaz, A.; Melin, P. Method for higher order polynomial Sugeno Fuzzy inference systems. *Inf. Sci.* **2016**, *351*, 76–89.
39. Masulli, F.; Studer, L. Time Series Forecasting and Neural Networks. In Proceedings of the International Joint Conference on Neural Networks, Washington, DC, USA, 10–16 July 1999.
40. Jang, J.; Sun, C.; Mizutani, E. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*; Prentice-Hall: Upper Saddle River, NJ, USA, 1997.
41. Abarbanel, H. *Analysis of Observed Chaotic Data*; Springer: New York, NY, USA, 1996.
42. Kononov, E. Visual Recurrence Analysis (VRA 4.9). Available online: <http://visual-recurrence-analysis.software.informer.com/4.9/> (accessed on 15 July 2017).
43. Panella, M.; Rizzi, A.; Mascioli, F.M.F.; Martinelli, G. ANFIS synthesis by hyperplane clustering. In Proceedings of the Joint 9th IFSA World Congress and 20th NAFIPS International Conference, Vancouver, BC, Canada, 25–28 July 2001; Volume 1, pp. 340–345.
44. Panella, M.; Liparulo, L.; Proietti, A. A Higher-Order Fuzzy Neural Network for Modeling Financial Time Series. In Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN), Beijing, China, 6–11 July 2014; pp. 3066–3073.
45. Box, G.; Jenkins, G.; Reinsel, G. *Time Series Analysis: Forecasting and Control*, 3rd ed.; Englewood Cliffs: Prentice Hall, NJ, USA, 1994.
46. Broomhead, D.S.; Lowe, D. Multivariable Functional Interpolation and Adaptive Networks. *Complex Syst.* **1988**, *2*, 321–355.
47. Jang, J.S. ANFIS: Adaptive-Neural-Based Fuzzy Inference System. *IEEE Trans. Syst. Man Cybern.* **1993**, *23*, 665–685.
48. Panella, M. A Hierarchical Procedure for the Synthesis of ANFIS Networks. *Adv. Fuzzy Syst.* **2012**, *2012*, 1–12.

49. Sharma, A.; Podolsky, R.; Zhao, J.; McIndoe, R.A. A modified hyperplane clustering algorithm allows for efficient and accurate clustering of extremely large datasets. *Bioinformatics* **2009**, *25*, 1152–1157.
50. Haykin, S. *Neural Networks, a Comprehensive Foundation*, 2nd ed.; Prentice-Hall: Englewood Cliffs, NJ, USA, 1999.
51. Scardapane, S.; Fierimonte, R.; Wang, D.; Panella, M.; Uncini, A. Distributed music classification using Random Vector Functional-Link nets. In Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN), Killarney, Ireland, 12–17 July 2015; pp. 1–8.



© 2017 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).