

Article

Devising Hourly Forecasting Solutions Regarding Electricity Consumption in the Case of Commercial Center Type Consumers

Alexandru Pirjan ^{1,*}, Simona-Vasilica Oprea ², George Căruțașu ¹, Dana-Mihaela Petroșanu ^{1,3}, Adela Bâra ² and Cristina Coculescu ¹

¹ Department of Informatics, Statistics and Mathematics, Romanian-American University, Expoziției 1B, Bucharest 012101, Romania; carutasu.george@profesor.rau.ro (G.C.); danap@mathem.pub.ro (D.-M.P.); coculescu.cristina@profesor.rau.ro (C.C.)

² Department of Economic Informatics and Cybernetics, The Bucharest Academy of Economic Studies, Romana Square 6, Bucharest 010374, Romania; simona.oprea@csie.ase.ro (S.-V.O.); bara.adela@ie.ase.ro (A.B.)

³ Department of Mathematics-Informatics, University Politehnica of Bucharest, Splaiul Independenței 313, Bucharest 060042, Romania

* Correspondence: alex@pirjan.com; Tel.: +40-762-642-866

Received: 24 September 2017; Accepted: 25 October 2017; Published: 27 October 2017

Abstract: This paper focuses on an important issue regarding the forecasting of the hourly energy consumption in the case of large electricity non-household consumers that account for a significant percentage of the whole electricity consumption, the accurate forecasting being a key-factor in achieving energy efficiency. In order to devise the forecasting solutions, we have developed a series of dynamic neural networks for solving nonlinear time series problems, based on the non-linear autoregressive (NAR) and non-linear autoregressive with exogenous inputs (NARX) models. In both cases, we have used large datasets comprising the hourly energy consumption recorded by the smart metering device from a commercial center type of consumer (a large hypermarket), while in the NARX case we have used supplementary temperature and time stamps datasets. Of particular interest was to research and obtain an optimal mix between the training algorithm (Levenberg-Marquardt, Bayesian Regularization, Scaled Conjugate Gradient), the hidden number of neurons and the delay parameter. Using performance metrics and forecasting scenarios, we have obtained results that highlight an increased accuracy of the developed forecasting solutions. The developed hourly consumption forecasting solutions can bring significant benefits to both the consumers and electricity suppliers.

Keywords: energy consumption; forecasting solutions; large non-household consumers; artificial neural networks; non-linear autoregressive (NAR) model; non-linear autoregressive with exogenous inputs (NARX) model; Levenberg-Marquardt (LM); Bayesian Regularization (BR); Scaled Conjugate Gradient (SCG)

1. Introduction

For the last 45 years, worldwide energy consumption has increased by more than 2.2-fold, with the non-residential sector accounting for the most significant percentage of the whole electricity consumption, ranging from 76% in 1971 to 78% in 2015 [1]. Therefore, a very important problem that must be addressed, in order to achieve energy efficiency, consists in the accurate forecasting of the electricity consumption in the case of large non-household consumers. In order to optimize the electricity consumption for this type of consumers and attain an efficient consumption strategy, a

detailed accurate hourly forecasting solution represents an extremely useful tool that creates the premises for obtaining an appropriate management of the environment's resources.

Nowadays, the evolution of technology has made it possible to develop new methods for recording and storing different types of data in real time, by means of specialized hardware and software components, such as the ones integrated in the advanced metering infrastructure (AMI). The AMI includes specialized components for measuring, recording, sending and receiving data. An important part of the advanced metering infrastructure systems consists of the smart metering devices, specialized electronic systems that are able to record, store and make available to the customers and suppliers more detailed information than was previously possible when using regular meters. In this context, the purpose of this paper is to develop an accurate forecasting solution that uses detailed consumption data retrieved from these devices, making it possible for large non-household electricity consumers to craft a personalized efficient consumption strategy that best suits their needs, achieving significant savings on the electricity costs.

In the scientific literature, there are works that give a comprehensive review of artificial neural networks (ANNs), highlighting their evolution during the years, their mathematical formalism and countless applications in various problems [2–9]. Artificial neural networks are recognized by scientists as efficient forecasting tools as there are a lot of scientific papers that demonstrate their superiority for the forecasting accuracy when compared to other methods like statistical ones [10–13]. There are many papers that cover a wide range of artificial neural network implementations that prove their undeniable advantages in terms of performing specific tasks efficiently, with very good results [14–19].

The forecasting of time series is extensively covered in numerous scientific works [15–18,20–24]. In the case of non-linear systems and systems whose state varies dynamically, being influenced by their actual state, methods like the nonlinear autoregressive and the nonlinear autoregressive with exogenous inputs proved to be very efficient in forecasting future values of time series [19,25]. In recent years, there has been an increased interest in the scientific literature regarding the accurate forecasting of consumed electricity, and a lot of methods and associated applications have been developed for the prediction of time series that contain energy related data.

Thus, in [26], the authors developed, using an artificial neural network approach, a model for energy analysis in the case of office buildings located in subtropical climates. The model uses nine input parameters (four related to the meteorological conditions, four related to the building's design and one for the type of day—weekday or weekend days) to predict the total energy consumption for the entire building and also the consumption for several categories of users. Analyzing the obtained results, the authors concluded that they reflected a very good forecasting accuracy.

In [27], the authors developed and analyzed a new stochastic model built for the purpose of forecasting the energy consumption of a building, using a dataset comprising hourly measurements of the consumed energy for a period of 7 weeks. The authors claimed that their method offered improved results when compared to other approaches, based on artificial neural networks or hidden Markov models.

In [28], the authors proposed a short-term load forecasting without meteorological data, based on: artificial neural networks; wavelet transform and artificial neural networks; wavelet transform and radial basis function neural networks; empirical mode decomposition and radial basis function neural networks. The authors proposed a regularization method that can be applied on the forecasted results in order to improve the accuracy of the prediction and finally they state that the errors study reveals that the forecasting can be achieved successfully without using meteorological data.

Reference [29] addresses issues regarding the forecasting of short term loads in electric power systems by means of artificial neural networks. The authors used feed-forward ANNs in order to achieve short-term predictions of the load demand in the national power system of Greece. Thus, they developed a series of ANNs using various networks architectures, and proposed and analyzed optimization techniques for the developed networks.

In [30], the authors proposed a model for predicting the day ahead load forecasting in the case of the power system in Turkey, based on an artificial neural network approach that takes into account the load variation caused by national or religious holidays. In what concerns the performance metrics,

the authors used the mean absolute percentage error and the hourly absolute percentage error. They state that their work provides as its main contribution an improved method of prioritizing the resources for the cities according to their energetic needs.

In [31], the authors study the possibility to obtain a daily forecast of the energy consumption in the case of public buildings, by means of neural networks. The authors have used the data retrieved from an energy consumption monitoring system of a series of public buildings to develop NAR and NARX neural networks based on the Levenberg-Marquardt algorithm in order to forecast the energy consumption for these buildings. The authors concluded that their models are appropriate for predicting the consumed energy in public buildings and by comparing the results, they noticed that the NARX model has offered an improved forecasting accuracy.

In [32], the authors analyzed a non-linear forecasting method for managing the energy in the case of a residential household that incorporates a photovoltaic rooftop system and a secondary storage system based on a lithium-ion battery. The authors state that their results confirm a high degree of accuracy for long-term predictions.

Reference [33] focuses on forecasting the electricity consumption for the next week taking into account the dynamics of people based on analyzing mobile network traffic data using several regression models. The authors claim that their forecasting solution is useful in saving primary energy consumption and energy distribution costs.

In [34], the authors used artificial neural networks and autoregressive techniques in order to obtain a 24-hour estimation of the solar irradiance. The authors used a meteorological dataset containing hourly records over a 5-years period for developing nonlinear autoregressive, nonlinear autoregressive with exogenous inputs, auto regressive moving average with exogenous terms, neural network fitting models in order to obtain the most accurate prediction model as to optimize the direct-current microgrid functioning.

The authors in [35] used the data provided by the Independent System Operators New England Incorporation (ISO-NE Inc.) operator and targeted obtaining an accurate short-term load forecasting method, based on non-linear autoregressive artificial neural networks with exogenous variables and on the Levenberg-Marquardt training algorithm. The authors compared their results with the ones obtained through other known methods (feedforward ANNs, autoregressive–moving-average model with exogenous inputs, state space methods) and claimed to have registered an improvement of 30% in what concerns the average error, stating that in this way one can achieve significant savings by not putting into operation unnecessary power plants.

The forecasting of electricity consumption at the household level according to different behavior patterns was studied in [36]. The authors took into account the inhabitants' daily routines along with usage patterns of household appliances in order to obtain a better accuracy when performing the forecast. The authors claimed that their results have the potential to improve the efficiency of smart metering devices and bring benefits to the residents.

In this paper, we focus on a topic of great significance in regard to forecasting the hourly energy consumption in the case of large non-household electricity consumers, a segment that represents to a considerable percentage of the total energy consumption, consequently representing a key-factor in attaining energy efficiency. We have trained, tested and validated a series of dynamic neural networks for solving non-linear time series problems with the purpose of devising forecasting solutions, based on the NAR and NARX models, using the development environment MATLAB R2017a. We have acquired and used large datasets, consisting of the hourly energy consumption for an entire year, retrieved from the smart metering device of a commercial center type of consumer (a large hypermarket) for both cases. In the case of the non-linear autoregressive with exogenous inputs model, we have used supplementary data, consisting of temperature and time stamp datasets.

The implementation of smart metering is at an incipient phase in Romania (and at the hypermarket's level). Therefore, in order to forecast the hypermarket's behavior, we have used the global hourly electricity consumption data retrieved by the single existing smart meter device. In the near future, the hypermarket intends to install several smart meters that would measure electricity on specific equipment such as cooling, heating, ventilation, lighting, etc. That approach would be

able to detect and compare the individual consumption of different types of equipment and propose the shifting of the operational time in order to minimize the electricity payments.

We have paid particular attention to investigating and achieving an optimal mix between the training algorithm that has been used when developing the artificial neural networks (Levenberg-Marquardt, Bayesian Regularization, Scaled Conjugate Gradient), the number of neurons in the hidden layer and the delay parameter. After having analyzed a series of performance metrics and forecasting scenarios, we have noticed that the results reveal an increased accuracy of the developed forecasting solutions.

In the context of our paper, the main beneficiary of the hourly consumption forecasting solutions is the electricity supplier, as the forecasts influence his market strategies and have a direct impact on the electricity tariff that is linked with the electricity usage. Using the developed solutions, the consumer can use the obtained forecast in order to adjust his load profile and he can also send to the supplier the day ahead consumption schedule (his forecast), thus obtaining an optimal schedule (according to a certain objective function) that would improve the energy efficiency. Our developed hourly consumption forecasting solutions can help improve the energy management, that is beneficial to both the consumers (in order to choose the appropriate billing plan corresponding to their real consumption behavior patterns and to devise an appropriate business strategy) and to the electricity suppliers (in order to devise efficient market strategies and to elaborate accurate forecasting production reports that are mandatory to be sent to the national authorities).

2. Materials and Methods

We have developed our solution for forecasting the hourly energy consumption covering five successive stages. During the first stage of our research methodology, we have devised four steps that are executed in order to acquire and process the data. The second stage of our methodology, comprising four steps, consists in developing the artificial neural networks forecasting solutions, based on the NAR model. In the third stage of our methodology, comprising four steps, we have developed the artificial neural networks forecasting solution based on the NARX model, using as exogenous variables, the meteorological and the time stamps datasets. During the fourth stage of our methodology, comprising 4 steps, we have developed the artificial neural networks forecasting solution based on the NARX model, using as exogenous variables, the time stamps datasets. In the fifth stage of our methodology, comprising five steps, we have obtained the best forecasting solution out of the available ones. A brief synthesis of the stages and steps of our methodology is given in Table 1.

Table 1. A synthesis of the methodology's stages and steps.

Stage	Step	Final Results of the Stage
I. Acquiring and processing the data	1. Acquiring the energy consumption and the meteorological datasets	The final preprocessed datasets
	2. Preprocessing the data (filtering, reconstructing)	
	3. Constructing the time stamp dataset	
	4. Dividing datasets into 2 subsets	
II. Developing the ANN forecasting solution based on the NAR model	1. Developing ANNs based on the LM algorithm	The best forecasting solution
	2. Developing ANNs based on the BR algorithm	
	3. Developing ANNs based on the SCG algorithm	
	4. Comparing the forecasting accuracy of the obtained ANNs	
III. Developing the ANN forecasting solution based on the NARX model, using as exogenous variables the meteorological and the time stamps datasets	1. Developing ANNs based on the LM algorithm	The best forecasting solution
	2. Developing ANNs based on the BR algorithm	
	3. Developing ANNs based on the SCG algorithm	
	4. Comparing the forecasting accuracy of the obtained ANNs	
	1. Developing ANNs based on the LM algorithm	

IV. Developing the ANN forecasting solution based on the NARX model, using as exogenous variables the time stamps datasets	2. Developing ANNs based on the BR algorithm 3. Developing ANNs based on the SCG algorithm 4. Comparing the forecasting accuracy of the obtained ANNs	The best forecasting solution
V. Obtaining the best forecasting solution	1. The 3 selected networks are put into the closed loop form 2. Forecasting using the best ANN forecasting solution based on the NAR model that has been put in the closed loop form 3. Forecasting using the he best forecasting solution based on the NARX model with meteorological and time stamps exogenous data that has been put in the closed loop form 4. Forecasting using the he best forecasting solution based on the NARX model with time stamps exogenous data that has been put in the closed loop form 5. Comparing the forecasting results from steps 1–4	The forecasting solutions' hierarchy

The research methodology is broadly described in the following subsections.

2.1. Acquiring and Processing the Data

The first stage of our methodology consists in acquiring and processing the data, comprising four steps. In the first step of this stage, we have acquired a large dataset, consisting in a number of 8,784 records, containing the hourly energy consumption for the year 2016 (measured in MWh), retrieved from the smart metering device of a commercial center type of consumer (a large hypermarket). We have also acquired a dataset containing the historical hourly meteorological data regarding the temperature (measured in Celsius degrees), recorded by the meteorological sensors of a specialized institute for the year 2016, consisting in a number of 8784 records. In Figure 1, we have represented the consumption time series and in Figure 2 the temperature time series for the year 2016.

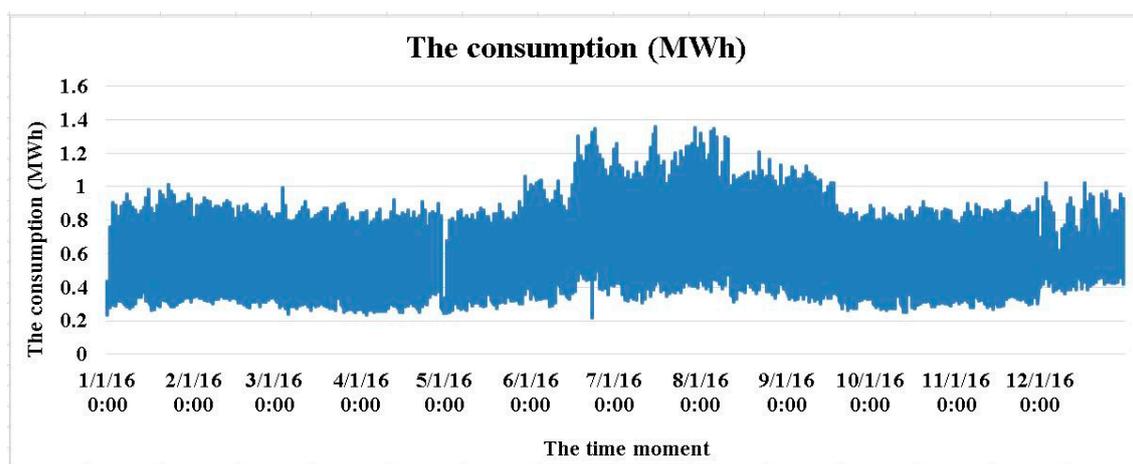


Figure 1. The consumption time series for the year 2016.

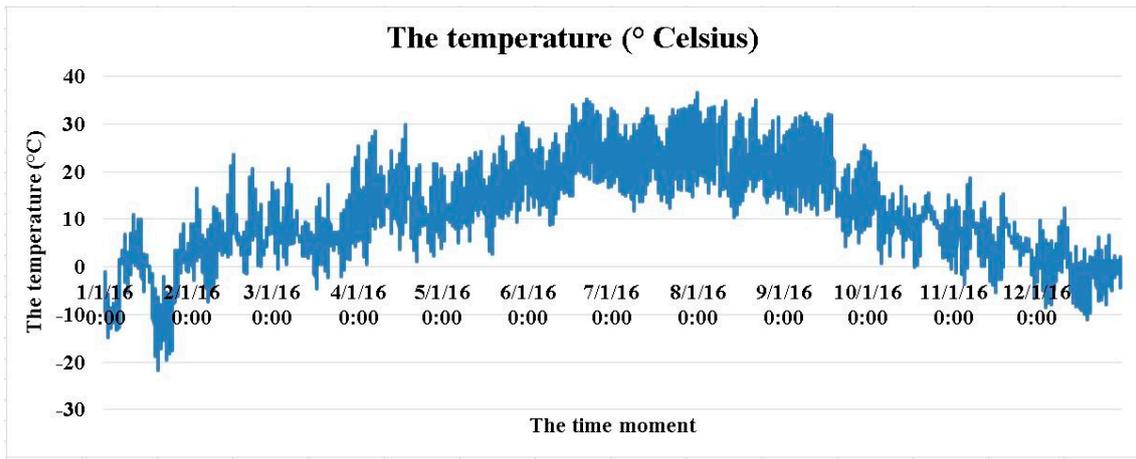


Figure 2. The temperature time series for the year 2016.

After having analyzed Figures 1 and 2, we have concluded that both time series exhibit many similar features, regarding their trend during the whole year. It is also plausible that the temperature values can influence the cooling and heating necessities of the commercial center type of consumer. Thus, the temperature series represents a good exogenous variable candidate, an external known series that significantly influences the consumption time series, useful in developing an ANN forecasting solution based on the NARX model.

We have taken into account the fact that retrieved data previously recorded by a smart metering device or by a meteorological station's sensors are sometimes prone to errors thus resulting into missing or abnormal values. Therefore, in the second step of the first stage, we have preprocessed the data, filtering the records in order to identify and correct the data recording errors, thus managing the noise and reconstructing the dataset corresponding to the hourly measurements of the whole year. In order to correct these errors, we have developed and applied a gap filling method based on the linear interpolation, taking into account the data specific features. This method is useful for approximating the values in the case of missing data or for correcting abnormal ones. In the following, we present the mathematical formalism of the developed gap filling method.

Let be the time series $x(t)$, $t \in \mathbb{N}$, $1 \leq t \leq T$, containing both valid data and missing or abnormal ones. If the terms $x(m)$ and $x(n)$, where $m, n \in \mathbb{N}$, $1 < m < n < T$ are known and the intermediate terms $x(m+1), x(m+2), \dots, x(n-1)$ must be approximated (being missing or abnormal), one first computes the step P between two neighbors of the time series, considering that all the successive steps are equidistant:

$$P = \left| \frac{x(m) - x(n)}{n - m} \right| \quad (1)$$

If one denotes by $M = \max(x(m), x(n))$, if $M = x(m)$, then we have chosen:

$$\begin{aligned} x(m+1) &= x(m) - P \\ x(m+2) &= x(m) - 2P \\ &\dots \\ x(n-1) &= x(m) - (n-m-1)P \end{aligned} \quad (2)$$

If $M = x(n)$, then we have chosen:

$$\begin{aligned} x(m+1) &= x(m) + P \\ x(m+2) &= x(m) + 2P \\ &\dots \\ x(n-1) &= x(m) + (n-m-1)P \end{aligned} \quad (3)$$

If $x(m) = x(n)$, then:

$$x(m) = x(m + 1) = x(m + 2) = \dots = x(n - 1) = x(n) \quad (4)$$

If the terms that must be approximated are the first s terms of the time series, $x(1), x(2), \dots, x(s)$ and $x(s + 1)$ is the next term and it is known, presuming that the last term of the series is also known, then the hourly consumption forecasting solutions use only the terms $x(t), t \in \mathbb{N}, s + 1 \leq t \leq T$ and all the datasets are adjusted accordingly, in order to contain the same number of elements.

If the terms that must be approximated are the last r terms of the series, $x(T), x(T - 1), \dots, x(T - r)$ and $x(T - r - 1)$ is the previous term and it is known, presuming that the first term of the series is also known, then the hourly consumption forecasting solutions use only the terms $x(t), t \in \mathbb{N}, 1 \leq t \leq T - r - 1$ and all the datasets are adjusted accordingly, in order to contain the same number of elements.

If the terms that must be approximated are the first s terms of the series ($s \in \mathbb{N}, s \geq 1$) and the last r terms of the series ($r \in \mathbb{N}, r \geq 1$), $x(1), x(2), \dots, x(s)$ and $x(T), x(T - 1), \dots, x(T - r)$, and $x(s + 1), x(T - r - 1)$ are known, then the hourly consumption forecasting solutions use only the terms $x(t), t \in \mathbb{N}, s + 1 \leq t \leq T - r - 1$ and all the datasets are adjusted accordingly, in order to contain the same number of elements.

If the abovementioned particular cases occur for both the energy consumption dataset and the meteorological dataset and it is necessary to truncate a subset of the first (or last) values of these time datasets, then one must truncate all the necessary datasets taking into account only the datasets having the indexes in the intersection of the indexes that have remained after both of the truncating processes, in order to adjust all the datasets accordingly, as to contain the same number of elements.

When we developed the gap filling method, we took into account the fact that it can become prone to errors when a certain threshold regarding the missing values is surpassed, especially consecutive ones. When running the experimental tests, we have noticed that after training, testing and validating the networks and even more importantly after having run a forecasting simulation using the developed neural networks for the month of December and compared the results with the actual ones, the errors did not influence the performance of the networks. In our case, there was only a small number of missing values. When devising the gap filling method, we have also tried to deliberately erase more consecutive values and we have noticed that when the number of consecutive values surpasses 10 for more than three consecutive days, and one uses the gap filling method to approximate the missing values, the performance of the networks does become impacted.

In our case study, the data retrieved from the smart metering device had no missing or abnormal values, but in the case of the meteorological dataset there were missing six out of 8784 values corresponding to 6 non-consecutive temperature values from different days (record no. 24 corresponding to the 1st of January at 24:00; record no. 32 corresponding to the 2nd of January at 8:00; record no. 3678 corresponding to the 2nd of June at 6:00; record no. 5552 corresponding to the 19th of August at 8:00; record no. 6415 corresponding to the 24th of September at 7:00 and record no. 4 corresponding to the 1st of December at 4:00), that were not the first either the last records of the time series. For these six values, we have used the abovementioned gap filling method. If more consecutive data values had been missing and our filling method risked becoming invalid, we would have tried to devise and implement another method or to reacquire (if possible) a valid dataset.

In our case, as the missing values of the time series that have to be computed are non-consecutive, in order to obtain each of the six missing values, the gap filling method has to compute only the arithmetic average of the previous and next values of the missing terms. As the missing data has corresponded to specific time moments that exclude the existence of any peak regarding the consumption or the temperature, the gap filling method based on the linear interpolation was a very good choice, as demonstrated by the experimental results. Therefore, the final number of samples is 8784 records, for both the energy consumption and the meteorological datasets.

Afterwards, in the third step of the first stage, we have constructed a time stamp dataset, corresponding to the time moments of the above-mentioned datasets, with the same number of records like the energy and the temperature datasets, comprising the hour of the day (ranging from 1 to 24), the day of the week (ranging from 1 to 7, where 1 corresponds to Monday), the day of the

month (ranging from 1 to the number of days of the respective month) and the month (ranging from 1 to 12, where 1 corresponds to January). In what regards the time stamp dataset, due to its mode of construction, it is not prone to errors. Detailed information about the datasets can be found in the Supplementary Materials.

In the fourth step of the first stage, we have divided each of the three datasets into two subsets. The first subset contains 8040 samples representing the January–November period and it will be used later for developing the forecasting artificial neural networks, while the second subset contains the 744 samples representing the whole month of December, that will be used for the final validation of the developed solutions, by comparing the predicted results for the month of December with the corresponding real values, contained in the second data subset.

Analyzing the real consumption recorded by the smart metering device of the hypermarket, taking into account the Romanian purchasing behavior patterns, the local traditions and customs, we have found that there are strong similarities between the activities of the hypermarket and its supply schedule in the months of November and December. Analyzing the store's sales reports, it is noteworthy that the preparation for winter holidays begins in the month of November, both in terms of supply and sales, the demand being higher than in the previous months. Even if in the period preceding Christmas, the commercial center has a prolonged schedule, which affects the consumption of lighting and the cash registers, the consumption related to the refrigeration and heating installations remains the same, as they are working around the clock.

Due to the fact that in Romania the implementation of smart metering devices is at an early phase, we only had available for our study case the global energy consumption recorded by a single smart meter, but not the detailed consumption of specific equipment. Thus, we have analyzed the types of energy consuming installations within the hypermarket (refrigeration, heating, lighting, cash register) and we have noticed that the refrigeration and heating installations are the biggest consumers. In this context, considering the fact that the data for the month of November were used in training the forecasting artificial neural networks, we have decided to select the month of December for the final validation, the methodology being confirmed by comparing the results provided by the three forecasting methods with the real data, corresponding to the month of December, measured by the smart metering device.

In the following, we present our developed artificial neural networks forecasting solutions, based on the NAR and NARX models. In order to achieve an increased efficiency when developing the ANNs, based on the NAR and NARX models, we have trained, validated and tested all the networks in an open loop, thus being able to provide the real past values to the ANN that is being developed, so that the network generates correct current results.

After having completed the training, validation and testing processes, we have modified the network, by putting it into the closed loop form in order to be able to perform a multi-step ahead prediction. If the network had remained in open loop mode, it would have been able to forecast one step ahead only, using previous values of the time series (in both the NAR and NARX models) and of the exogenous variables (in the NARX model). When the network is put into the closed loop form, it can forecast multiple steps ahead as it uses the prior available known values for the first prediction step and the predicted values of the time series starting with the second step in order to be able to obtain the forecast.

2.2. Developing the ANN Forecasting Solution Based on the NAR Model

The second stage of our methodology, comprising four steps, consists in developing the artificial neural networks forecasting solutions, based on the NAR (nonlinear autoregressive) model. In this purpose, we have used the energy consumption dataset. Of particular interest was to research and obtain an optimal mix between the training algorithm (Levenberg-Marquardt, Bayesian Regularization, Scaled Conjugate Gradient), the hidden number of neurons and the delay parameter.

The datasets involved in our research, both the inputs (the hourly energy consumption, the historical hourly meteorological data, the timestamps) and the output (the hourly forecasted energy consumption) ones, are time series, sequences of data points listed in a certain time order, in our case

being registered (or constructed) at equally spaced points in time (1 h). Regarding from the time-variable point of view, these sequences are discrete.

In many cases, the real-life situations and phenomena are described through time series that cannot be modeled by linear frameworks, as their behavior is dynamic and depends on their previous states [31]. The model developed in our research, concerning the development of hourly consumption forecasting solutions for achieving energy efficiency in the case of large electricity non-household consumers, is in the abovementioned situation as the data involved in our study is represented by discrete, non-linear time series.

One of the most useful approaches in time series predicting is the non-linear autoregressive (NAR) neural network that is modeled by the following equation:

$$y(t) = F(y(t-1), \dots, y(t-d)) + \epsilon(t) \quad (5)$$

The neural network forecasts the actual value of the time series $y(t)$, based on the previous d values of the series, where $d \in \mathbb{N}, d \geq 1$ represents the delay parameter and the terms $y(t-1), \dots, y(t-d)$ represent the feedback delays. The forecasting's purpose is to approximate as accurate as possible the unknown nonlinear function F , using neural network's specific methods.

The optimization could be attained by varying the network's weights and biases, by testing different settings regarding the number of neurons per layer and the number of hidden layers. By testing different settings, one can obtain the neural network that provides the best forecasting accuracy, but using this approach one must take into account the fact that lowering too much the number of neurons could reduce the computational power of the neural network, restricting its generalization capability, while increasing too much the number of neurons raises the system's complexity. The last term of the Equation (5), $\epsilon(t)$, represents the approximation error of the actual value of the time series $y(t)$.

In the following, the steps that we have developed for the second stage of our research methodology are presented. In the first step of the second stage, we have used the Levenberg-Marquardt (LM) training algorithm in order to develop several artificial neural networks, based on the NAR model. In the following we present a brief description of the Levenberg-Marquardt algorithm (LM), one of the most popular algorithms used in training artificial neural networks (ANNs), having extensive applications in the fields of computing and mathematics [37].

This algorithm is a curve fitting one, aiming to obtain a curve that passes through some given points that could be subjected to certain constraints. In this purpose, the method computes the errors related to a parametrized form of a function, constructed in order to approximate the given dataset, computes the sum of the squares of these errors and afterwards, targets to minimize this sum. The LM method combines two optimization methods, namely the gradient descent and the Gauss-Newton one [38].

In the following, the main mathematical formalism of the LM algorithm are briefly recalled, as they have been introduced in [37–39]. Denoting by $x = (x_1, x_2, \dots, x_n)$ an n -dimensional real array, by $r(x) = (r_1(x), r_2(x), \dots, r_m(x))$ an array whose components are the residual functions $r_k: \mathbb{R}^n \rightarrow \mathbb{R}$, $1 \leq k \leq m$, where $m \geq n$, the purpose of the method is to minimize the function f defined as:

$$f(x) = \frac{1}{2} \|r(x)\|^2 = \frac{1}{2} \sum_{k=1}^m r_k^2 \quad (6)$$

The Jacobian matrix of r is denoted by $J(x)$ and is defined by :

$$J(x) = \frac{\partial r_k}{\partial x_i}, 1 \leq k \leq m, 1 \leq i \leq n \quad (7)$$

If one considers the linear case, then all the functions r_k are linear and thus they are of the form:

$$r_k = \frac{\partial r_k}{\partial x_1} x_1 + \frac{\partial r_k}{\partial x_2} x_2 + \dots + \frac{\partial r_k}{\partial x_n} x_n + r_k(0), 1 \leq k \leq m \quad (8)$$

Equation (1) thus becomes:

$$f(x) = \frac{1}{2} \|Jx + r(0)\|^2 \quad (9)$$

Therefore, the gradient and the Hessian of the function f could be written as:

$$\begin{aligned} \nabla f(x) &= J^T(Jx + r) \\ \nabla^2 f(x) &= J^T J \end{aligned} \quad (10)$$

In order to obtain a local minimum x_{min} of the function $f(x)$, one has to impose the necessary condition $\nabla f(x) = 0$ and one obtains:

$$x_{min} = -(J^T J)^{-1} J^T r \quad (11)$$

If one considers the non-linear case, the gradient and the Hessian of the function f could be written as:

$$\nabla f(x) = \sum_{k=1}^m r_k(x) \nabla r_k(x) = J(x)^T r(x) \quad (12)$$

$$\nabla^2 f(x) = J(x)^T J(x) + \sum_{k=1}^m r_k(x) \nabla^2 r_k(x) \quad (13)$$

In the linear approximation, if r_k could be considered linear functions or the values of the residuals are small, the Hessian becomes identical to the one given by the second relation of the Equation (10). This approximation does not suit to the cases when dealing with large residual problems, as the algorithms' performance lowers.

As mentioned before, the LM method combines two optimization methods, one of them being the gradient descent method, a technique for minimizing a function by using an iterative process. In each step, one adds the gradient ∇f of the function multiplied by a negative parameter:

$$x_{i+1} = x_i - \mu \nabla f \quad (14)$$

The method's convergence can be improved by using various approaches [39]. Among these methods, the Newton's one is based on solving the equation $\nabla f(x) = 0$ by expanding the gradient in series around a certain state x_0 (considered to be the current state), using the Taylor's formula:

$$\nabla f(x) = \nabla f(x_0) + (x - x_0)^T \nabla^2 f(x_0) + \text{terms of higher order} \quad (15)$$

In the case when f is quadratic around x_0 , the terms of higher order could be neglected. If one replaces x_0 by x_i and x by x_{i+1} in the necessary condition for a local minimum $\nabla f(x) = 0$, Equation (14) becomes:

$$x_{i+1} = x_i - (\nabla^2 f(x_i))^{-1} \nabla f(x_i) \quad (16)$$

Using the Newton's method, considering the quadratic approximation of the function f , the Hessian becomes identical to the one given by Equation (10). This method offers certain advantages related to its rapid convergence, but it has some limitations with concern to the linearity around the initial point.

The Levenberg's method combines the gradient-descent method with the Gauss-Newton iteration and is more powerful than both of these methods. The update rule is chosen as a combination between the two above mentioned algorithms:

$$x_{i+1} = x_i - (H + \mu I)^{-1} \nabla f(x_i) \quad (17)$$

where the Hessian matrix computed in x_i , $\nabla^2 f(x_i)$ has been denoted by H .

Equation (17) helps to set the update rule, by adjusting the parameter μ according to the error. After each update, one evaluates the error. If it is lower, it means that the quadratic approximation fits and therefore, in the following step one reduces the parameter μ 10 times, thus reducing the gradient descent's influence. On the contrary, if the computed error increases, one increases the parameter μ 10 times, thus increasing the gradient descent's influence [39].

Using the LM algorithm, one computes the Jacobian matrix based on a back-propagation technique, while the computation of the Hessian matrix is avoided. The update rule based on the Newton's method becomes:

$$x_{i+1} = x_i - (J^T J + \mu I)^{-1} J^T(x_i) r(x_i) \quad (18)$$

where $J^T J$ represents the hessian matrix H computed in x_i and $J^T(x_i)r(x_i)$ represents the gradient $\nabla f(x_i)$.

Equation (18) offers the possibility to adjust the method by the values of the parameter μ . If one considers $\mu = 0$, one obtains the Newton method, while the gradient descent method with a small step corresponds to large values of the same parameter. The objective function is reduced at each of the algorithm's steps by adjusting the values of the parameter μ .

Considering the fact that the LM algorithm combines the performance and features of its two component parts, offering multiple undeniable advantages to the programmers, we have decided to implement it in our research, in view of devising hourly consumption forecasting solutions for achieving energy efficiency in the case of large electricity non-household consumers.

We have trained, using the LM algorithm, a number of 15 forecasting artificial neural networks based on different architectures in order to obtain an optimum mix between the number of neurons from the hidden layer and the delay parameter, as follows: one neuron for the input data corresponding to the electricity consumption of the hypermarket, n neurons in the hidden layer, where $n \in \{6, 12, 24\}$, the delay parameter $d \in \{2, 6, 12, 24, 48\}$, one neuron for the output layer and one neuron for the output data (the forecasted electricity consumption, measured in MWh).

Regarding the dataset, we have used the abovementioned first subset of the energy consumption dataset, subset that contains 8040 samples, representing the January-November period of the year 2016. In order to train, validate and test the forecasting neural networks, we have divided the dataset in the following manner: 70% of the data has been used in the training process, 15% in the validation process and the remaining 15% in the testing process. We have randomly chosen the samples corresponding to these percentages.

The data was divided using the MATLAB R2017a "net.divideFcn = 'dividerand'" instruction so that every time the network is trained, the divide function executes automatically, dividing the dataset randomly according to the division parameters "net.divideParam.trainRatio", "net.divideParam.valRatio", and "net.divideParam.testRatio", in order to obtain the training, validation and testing subsets.

We have also tried to use the MATLAB R2017a "net.divideFcn = 'divideblock'" instruction, case in which the input dataset was sequentially divided into three subsets allocated for the training, validation and testing purposes (in this specific order, validation being always in the middle), but we have noticed that the obtained results were slightly lower than when running 10 iterations with the random division of the dataset.

Taking into account that our developed artificial neural networks are dynamic, we have used in the development environment MATLAB R2017a the "net.divideMode = 'time'" instruction, that specifies to divide the target data by timestep which, according to the MATLAB official documentation [40] is the most suitable choice for the time series problems. We have set the normalization performance parameter to the standard value by using the code line "net.performParam.normalization = 'standard'"; Thus, we have instructed the development environment to compute the errors as if the output elements had values ranging from -1 to 1 . In this manner, we made sure that the data ranges do not influence the final results.

As the networks' weights and biases are initialized each time with different values and the data are allocated randomly within the percentages, we have run 10 iterations for each of the above-mentioned cases. For each iteration and each of the developed networks, we have generated plots representing the performance analysis highlighted by the mean squared error (measured in MWh^2), the error histogram, the regressions, the error autocorrelation. Comparing these plots, we have saved the network that has provided the best forecasting accuracy out of the 10 iterations, thus obtaining 15 forecasting artificial neural networks, developed based on the NAR (nonlinear autoregressive) model, trained using the LM algorithm.

In the second step of the second stage, in order to develop several artificial neural networks, based on the NAR model, we have used the Bayesian Regularization (BR) training algorithm.

In the following we present a brief description of the Bayesian Regularization (BR) algorithm that uses as objective function a linear combination of the squared errors and of the squared weights, adjusting this function in order to obtain, when the network's training process has ended, a network that offers improved generalization features [41,42]. This algorithm is based both on the LM one and on the backward propagation of errors one that is useful in computing the Jacobian of the function that has to be minimized.

The BR adjusts each variable according to the LM algorithm using also an adaptive value that is increased in order to obtain the lowest value of the objective function and is decreased after the network has been modified according to the identified changes.

The Bayesian Regularization (BR) algorithm uses two parameters denoted α and β , called Bayesian hyperparameters, whose role is to indicate the direction in which the learning process has to seek: in the one of the minimal weights or in the one of the minimal errors. The objective function is a linear combination of the sum of all the squared errors S_e and the sum of all the squared weights S_w :

$$C(i) = \alpha \cdot S_w + \beta \cdot S_e \quad (19)$$

When compared to other training algorithms, the BR algorithm has the main advantage of not requiring a validation step and this fact brings benefits, especially in the cases when reserving a data sample for the validation process cannot be achieved or results in significant costs. Another important feature offered by the Bayesian Regularization algorithm is represented by the fact that in this case the testing process of various settings related to the number of hidden neurons could be skipped, by means of a third parameter γ designed to control the network's weights and as a consequence, its complexity.

The implementations of the BR algorithm often update the hyperparameters after each step of the training process. Because of this, there are many situations when these updates might lead to weak iterations. In order to overcome this deficiency, the researchers have devised different methods for updating the parameters, relying on the computation of the Hessian matrix's inverse.

The BR algorithm computes first the Jacobian J and afterwards, the error gradient:

$$g = \nabla f(x) = J^T E \quad (20)$$

In the subsequent step, it is computed an approximation of the Hessian matrix:

$$H = J^T J \quad (21)$$

Using Equation (19), the objective function is computed and afterwards, in order to obtain the values of δ , the following equation is solved:

$$(H + \lambda I)\delta = g \quad (22)$$

These values are useful in updating the network's weights and the corresponding objective function. In the case this function does not decrease, the last computed weights are rejected and the value of λ is increased by a multiplying factor v . In the case when the function does decrease, λ is decreased by a factor v . Afterwards, the updated values of the α and β hyperparameters are computed according to one of the methods depicted in [39].

The test of time has proven the effectiveness of the BR training algorithm in developing ANNs in contrast with other available algorithms and therefore we have taken the decision to see how well this algorithm performs when developing hourly consumption forecasting solutions in view of achieving energy efficiency in the case of large electricity non-household consumers.

Using the BR algorithm, we have trained a number of 15 forecasting artificial neural networks using the same methodology as in the LM case. For the training and testing processes of the forecasting neural networks, we have divided the dataset in the same manner as in the LM case, but the rest of the data was not allocated because in the case of the BR algorithm the validation process does not take place. Therefore, we have made sure that we will use the same amount of data in the training and testing phases for all the training algorithms, hence achieving a relevant comparison. The samples were divided in percentages using the same approach as in the case of the LM training

algorithm, described above. The performance analysis and the selection of the best forecasting networks have been conducted using the same methodology as in the LM case.

In the third step of the second stage, we have used the Scaled Conjugate Gradient (SCG) training algorithm for developing the ANNs, based on the NAR model.

In the following we present a brief description of the Scaled Conjugate Gradient (SCG) algorithm, introduced in 1993, a supervised learning algorithm useful in developing ANNs. It is based on the conjugate gradient methods [43]. The algorithm is independent of user provided parameters, being completely automated. It has also a considerable advantage due to the fact that it avoids the procedures related to the determination of the step size consisting in the weight update's length.

At this point, other algorithms perform a line search during each iteration in order to compute the local minimum, a task that implies certain processing costs, being time consuming due to the fact that in order to perform the search, the network's response has to be calculated more times. In order to circumvent the line search, the Scaled Conjugate Gradient algorithm combines the Levenberg-Marquardt's model-trust region approach with the conjugate gradient one [43]. Just like other conjugate gradient based methods do, the Scaled Conjugate Gradient algorithm relies on conjugate directions. Nevertheless, the way how SCG has been implemented makes it more efficiently in what regards the processing time as it eliminates the need for a line search.

The backward propagation of errors, a frequently used method in the training process of ANNs, is implemented along with the gradient descent or other optimization methods. In order to obtain the local minimum of the objective function, the method computes this function and its gradient with respect to the network's weights. In order to decrease the objective function, the network's weights are adjusted in the steepest descent directions but this convergence is not the fastest. The algorithm minimizes the objective function along the conjugate gradient directions belonging to the previous step, thus obtaining an improved convergence when compared to the general back-propagation method. In this manner, a minimizing operation from a specific step does not have to be cancelled in the next one, as it is the case for other algorithms.

The back-propagation methods are based on the first derivatives of the objective function and thus they are first order techniques. The conjugate gradient methods make use of the second derivatives of the functions that have to be minimized and therefore they are second order techniques that imply high processing costs but offer a series of advantages when compared to the first order techniques.

The SCG can be used in the case when the components of the objective function are derivable. In the first iteration, the Scaled Conjugate Gradient searches in the direction that makes the objective function decrease most rapidly and afterwards it performs a line search that aims to obtain the distance that is used as step in advancing along the search direction [39]:

$$x_{i+1} = x_i + \alpha_i g_i \quad (23)$$

Subsequently, a new search is performed in a direction that is conjugated to the first one. At every new step, the searching direction is a combination between the searching direction of the previous step and the new direction that makes the objective function decrease most rapidly, using the parameters β_i whose computation is specific to each version of the conjugate gradient method:

$$p_i = -g_i + \beta_i g_{i-1} \quad (24)$$

Taking into account the advantages offered by the SCG algorithm, we have chosen to study how well suited it is in developing hourly consumption forecasting solutions for achieving energy efficiency in the case of large electricity non-household consumers. In order to attain an optimum mix between the number of neurons from the hidden layer and the delay parameter, we have used the same methodology as in the case of the LM algorithm.

In the fourth step of the second stage, we have compared the forecasting accuracy of the 45 artificial neural networks, trained using the LM, BR and SCG algorithms, developed based on the NAR (nonlinear autoregressive) model and we have chosen the one that has provided the best performance.

Afterwards, as we wanted to see if we can further improve the forecasting accuracy of the artificial neural networks developed based on the NAR model, we have developed an artificial neural network forecasting solution, based on the NARX (nonlinear autoregressive with exogenous inputs)

model, using as exogenous variables the meteorological and the time stamps datasets, solution that is presented in the following.

2.3. Developing the ANN Forecasting Solution Based on the NARX Model, Using as Exogenous Variables the Meteorological and the Time Stamps Datasets

During the third stage of our methodology, comprising four steps, we have developed the artificial neural networks forecasting solution based on the NARX model. In this case, we have used as exogenous variables the meteorological and the time stamps datasets. As in the case of the forecasting solution based on the NAR model, of particular interest was to research and obtain an optimal mix between the training algorithm (Levenberg-Marquardt, Bayesian Regularization, Scaled Conjugate Gradient), the hidden number of neurons and the delay parameter.

In many situations, the time series forecasting models relate the actual value of the time series not only to previous values of the series (as in the NAR model), but also to additional external time series (exogenous data). This happens when the series that has to be forecasted is correlated with other ones that are influencing it.

The nonlinear autoregressive with exogenous inputs (NARX) model [44] is described by the following equation:

$$y(t) = F(y(t-1), \dots, y(t-d), x(t), x(t-1), \dots, x(t-d)) + \epsilon(t) \quad (25)$$

Similar to the case of the NAR model, the neural network forecasts the actual value of the time series of interest $y(t)$, based on the previous d values of this series, but in addition to this it is also based on the actual and previous d values of the exogenous series, where $d \in \mathbb{N}$, $d \geq 1$ represents the delay parameter.

The forecasting's purpose is to approximate as accurate as possible the unknown nonlinear function F and this optimization could be attained by testing different settings regarding the number of neurons per layer, the number of hidden layers, by varying the network's weights and biases. Thus, one can obtain the neural network that provides the best forecasting accuracy, taking care (as in the case of the NAR model) not to lower or increase too much the number of neurons. The last term of the Equation (25), $\epsilon(t)$, represents the approximation error of the actual value of the time series $y(t)$.

During the third stage of our methodology, we have developed four steps, similar to the ones developed in the NAR case. This time, the artificial neural networks were developed based on the LM, BR and SCG training algorithms and on the NARX model, using as exogenous variables the meteorological and the time stamps datasets. The only difference between two implementation consists in the number of neurons for the input data, that is now six (five neurons for the exogenous data, one neuron for the dataset represented by the electricity consumption of the hypermarket).

The temperature dataset that we have used above as an exogenous variable limits significantly the forecasting timeframe as when using the forecasting ANN one should know beforehand the temperature for the time steps for which he needs to forecast the consumption. Therefore, as we wanted to see if we could obtain a forecasting solution based on the NARX model, for which the exogenous variables are easier to obtain and make it possible to achieve a long-term forecasting, we have developed an ANN forecasting solution, based on the NARX model, using as exogenous variable only the time stamps dataset, solution that is presented in the following.

2.4. Developing the ANN Forecasting Solution Based on the NARX Model, Using as Exogenous Variables the Time Stamps Datasets

During the fourth stage of our methodology, comprising four steps, we have developed the artificial neural networks forecasting solution based on the NARX model. The steps that we have developed in the fourth stage of our research methodology are similar with the ones presented above, in the case of the nonlinear autoregressive with exogenous inputs model, using as exogenous variables the meteorological and the time stamps datasets, the only difference being the number of neurons for the input data, as in this situation we have not used the temperature exogenous data.

Therefore, we have used five neurons for the input data (four neurons for the exogenous data, one neuron for the dataset represented by the electricity consumption of the hypermarket).

2.5. Obtaining the Best Forecasting Solution

During the fifth stage of our methodology, comprising five steps, we have obtained the best forecasting solution out of the available ones: the artificial neural networks forecasting solutions based on the NAR model, the NARX model using all exogenous data and the NARX model using the time stamps exogenous dataset. In order to obtain a final validation based on a real-world forecasting scenario, in this stage we have forecasted the electricity consumption for the month of December, using the three developed solutions and we have compared the obtained results with the corresponding real values, provided by the commercial center type of consumer.

In the first step of the fifth stage, as we wanted to use the ANNs in order to forecast more future time series values, we have first put the three selected networks into the closed loop form, due to the aspects that have been analyzed in Section 2.1.

Afterwards, in the next three steps of the fifth stage, we have forecasted the hourly energy consumption for the month of December 2016, consisting in 744 values, using the closed loop form of the ANN forecasting solutions based on the following models: NAR, NARX using as exogenous variables the meteorological and the time stamps datasets and NARX using as exogenous variables the time stamps datasets.

In the fifth step of the fifth stage, we have compared the values obtained after having executed the previous three steps with the real ones, contained in the second data subset that has been previously selected and set apart, consisting in 744 samples that represent the hourly energy consumption for the whole month of December. Thus, we have obtained a final validation of the ANN forecasting solutions based on the NAR and NARX models along with an eloquent metric, useful in comparing the forecasting accuracy of the three developed forecasting solutions. In this purpose, we have first plotted on the same chart the real consumption and the three forecasted ones. Thus, we are able to compare the values provided by the three forecasting solutions and also to compare them with the real consumption values.

Afterwards, in order to obtain a more eloquent comparison, we have plotted on a second chart the absolute values of the differences between the real consumption and each of the three forecasted ones. As we are interested in obtaining forecasting results as close as possible to the real values, based on this second chart we are able to identify the network that provides the best forecast, for which the absolute values of the differences are closest to the zero value.

The block diagram of our devised forecasting methodology, described in the above sections, is synthesized in Figure 3. In the following, we present the results obtained after having run the experimental tests.

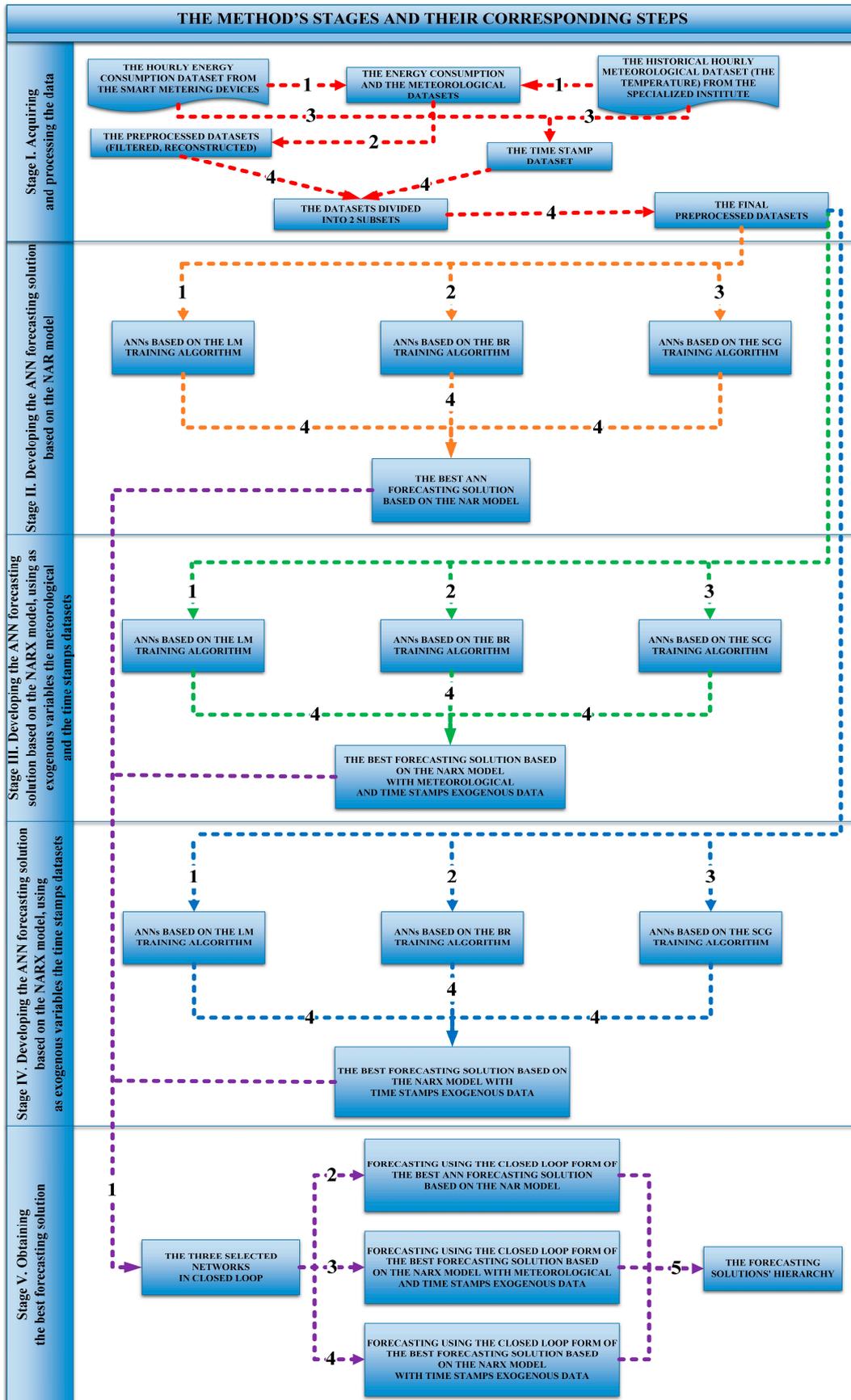


Figure 3. The block diagram of the devised forecasting methodology.

3. Results

In developing, testing and using the ANN forecasting solutions based on the NAR and NARX models we have used as a hardware configuration the central processing unit Intel i7-5960x operating at 3.0 GHz with 32 GB (4×8 GB) of 2144 MHz, DDR4 quad channel and the GeForce GTX 1080 TI NVIDIA graphics card with 11 GB GDDR5X 352-bit from the Pascal architecture. The software configuration that we have used consists in the Windows 10 Educational Version 1703 operating system and the MATLAB R2017a development environment.

3.1. Results Regarding the Developed ANN Forecasting Solution Based on the NAR Model

According to the abovedescribed methodology, when we have developed the artificial neural networks forecasting solution based on the NAR model, in order to select the optimum mix between the number n of neurons from the hidden layer and the delay parameter d and to select the best training algorithm, we have run different settings for the pair (n, d) , for each of the LM, BR and SCG training algorithms. After having run 10 iterations for each case, we have saved the network that has provided the best hourly forecasting accuracy for each case and we have synthesized the obtained results, highlighting the values of the mean squared error MSE and the correlation coefficient R between the network targets and the network outputs, for the whole dataset (Table 2).

Table 2. The synthesis of the experimental results when developing the artificial neural networks forecasting solution based on the NAR model.

The Levenberg-Marquardt Training Algorithm						
n	d	2	6	12	24	48
6	MSE	0.0029307	0.0030022	0.0019403	0.0018349	0.00091808
	R	0.97626	0.98049	0.9867	0.99199	0.99272
12	MSE	0.0025306	0.0022852	0.0018363	0.001085	0.0015098
	R	0.97696	0.98416	0.98811	0.99324	0.99184
24	MSE	0.0037219	0.0026784	0.0013428	0.0019062	0.0017245
	R	0.97486	0.98197	0.98986	0.99255	0.99254
The Bayesian Regularization Training Algorithm						
n	d	2	6	12	24	48
6	MSE	0.0029627	0.002228	0.0014656	0.00094866	0.00072544
	R	0.97709	0.983	0.98909	0.99244	0.99422
12	MSE	0.0028051	0.0019579	0.001046	0.00068836	0.00056501
	R	0.97871	0.98468	0.992	0.99402	0.99448
24	MSE	0.0026605	0.0017508	0.00085038	0.00057053	0.00048272
	R	0.9799	0.98549	0.99399	0.99355	0.99526
The Scaled Conjugate Gradient Training Algorithm						
n	d	2	6	12	24	48
6	MSE	0.0035269	0.0032701	0.0021393	0.0015375	0.0010847
	R	0.97191	0.97206	0.98145	0.98853	0.99123
12	MSE	0.0052954	0.0046987	0.0028487	0.00093968	0.00083771
	R	0.97055	0.9691	0.98061	0.9916	0.99241
24	MSE	0.0038286	0.0027353	0.0021494	0.0016137	0.0013316
	R	0.97033	0.9762	0.9806	0.98971	0.98819

Having analyzed the results, we have observed that in the case of the LM training algorithm, the network developed according to the architecture that comprises $n = 6$ neurons in the hidden layer and a delay parameter of $d = 48$ provides the best hourly forecasting accuracy, having the lowest value of the mean squared error (0.00091808) and the value of the correlation coefficient computed for the whole dataset very close to 1 (0.99272). Contrariwise, the network developed using $n = 24$ neurons in the hidden layer and the delay parameter of $d = 2$ provides the worst hourly forecasting accuracy, having the highest value of the mean squared error (0.0037219) and the value of the correlation coefficient computed for the whole dataset furthest from 1 (0.97486). Despite being the

network that provides the lowest hourly forecasting accuracy out of the 15 networks trained based on the LM algorithm, it still provides a good hourly forecasting accuracy.

After analyzing the obtained results, we have noticed that in the case of the BR training algorithm, the network developed using $n = 24$ neurons in the hidden layer and a delay parameter of $d = 48$ offers the best forecasting in terms of accuracy, having the lowest value of the mean squared error (0.00048272) and the value of the correlation coefficient computed for the whole dataset closest to 1 (0.99526). On the opposite side, the network developed according to the architecture that comprises $n = 6$ neurons in the hidden layer and a delay parameter of $d = 2$ provides the worst hourly forecasting accuracy, having the highest value of the mean squared error (0.0029627) and the value of the correlation coefficient computed for the whole dataset furthest from 1 (0.97709).

We have analyzed the results obtained in the case of the SCG training algorithm and we have observed that the best hourly forecasting accuracy is provided by the network developed using $n = 12$ neurons in the hidden layer and a delay parameter of $d = 48$ that registered the lowest value of the mean squared error (0.00083771) and the value of the correlation coefficient computed for the whole dataset closest to 1 (0.99241). In what concerns the worst prediction accuracy, it has been provided by the network developed according to the architecture that comprises $n = 2$ neurons in the hidden layer and a delay parameter of $d = 12$, having the highest value of the mean squared error (0.0052954) and the value of the correlation coefficient computed for the whole dataset among the furthest from 1 (0.97055). Although this network has the lowest hourly forecasting accuracy out of the 15 networks trained based on the SCG algorithm, it still provides a good hourly forecasting accuracy.

In all of the analyzed situations, the hourly forecasting accuracy improves along with the increase of the delay parameter value, the best hourly forecasting accuracy being obtained for $d = 48$ for all the training algorithms. One can observe that this value of d is a multiple of 24, which means that in order to obtain an improved hourly forecasting accuracy one needs 2 consecutive days values.

Comparing the forecasting accuracy obtained by the three selected networks (corresponding to the three training algorithms) that have provided the best results, based on the NAR model, we have observed that the best hourly forecasting accuracy is provided by the ANN developed using the BR algorithm, with $n = 24$ neurons in the hidden layer and a delay parameter of $d = 48$, entitled ANN_NAR_BR (Figure 4).

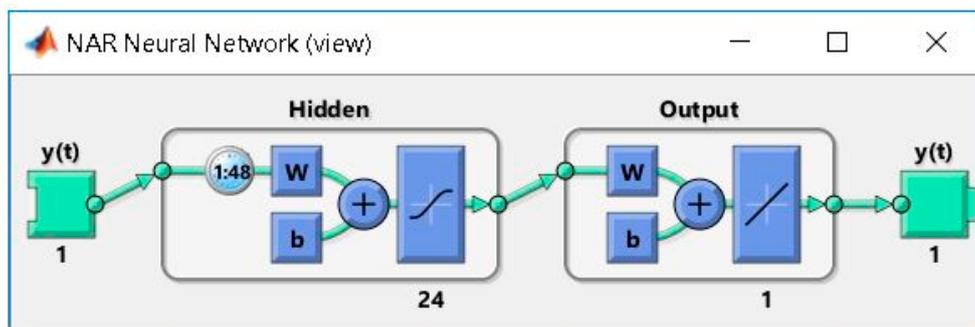


Figure 4. The ANN_NAR_BR's architecture.

In order to analyze the training performance when forecasting the electricity consumption for the month of December 2016, using the ANN developed based on the BR algorithm and on the NAR model, we have first plotted the training and testing curves. We have obtained the best training performance at the 493th epoch, when the mean squared error has the value of 0.00048272. Analyzing this plot, we have noticed that the devised forecasting solution has proven to be stable (as the curves do not increase after they converge). When analyzing the results, we have also taken into consideration the testing curve evaluation and its comparison to the training one. Thus, we have noticed that the testing curve does not increase significantly before the training curve and therefore we can conclude that in this case the overfitting process does not occur. This aspect reflects the fact that the data sets have been divided appropriately and the network training has been conducted

efficiently. This chart confirms the high level of accuracy and performance of the forecasting ANN, developed based on the BR algorithm and on the NAR model (Figure 5).

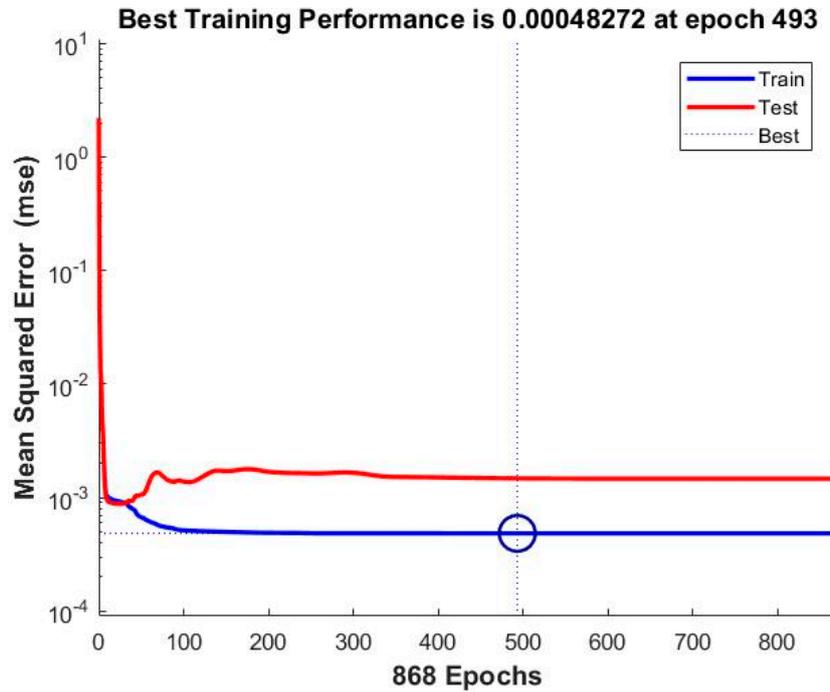


Figure 5. The best training performance when forecasting the electricity consumption for the month of December, using the ANN_NAR_BR network.

Afterwards, we have represented the error histogram, when forecasting the electricity consumption for the month of December using the above-mentioned forecasting ANN (Figure 6).

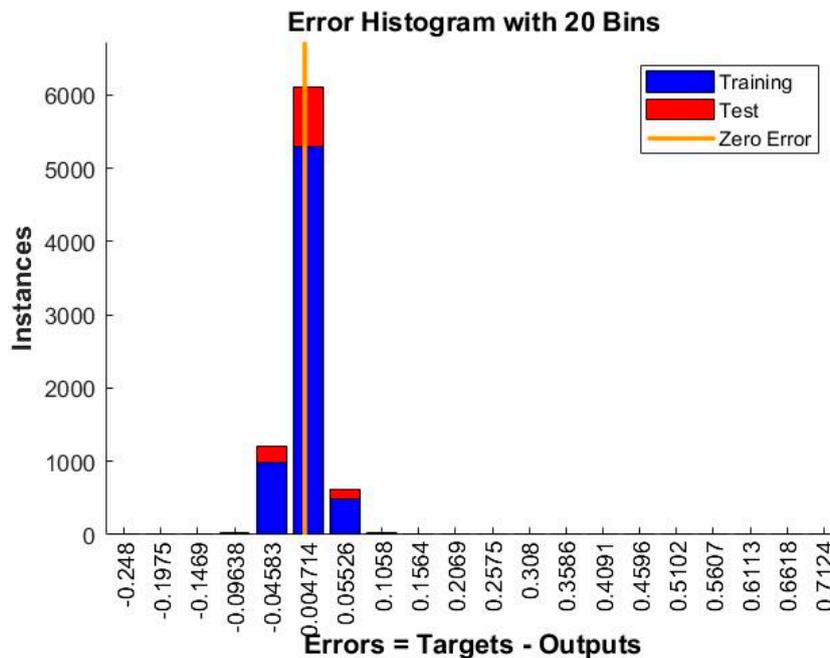


Figure 6. The error histogram when forecasting the electricity consumption for the month of December, using the ANN_NAR_BR network.

In this plot, the blue bars represent the training data, while the red bars represent the testing data. Analyzing the plot, we have noticed that in this case, most of the errors fall between -0.04583

and 0.05526, a very narrow interval. There are only very few training points for which the errors fall outside this range. Thus, the error histogram highlights very good results in the case of forecasting the electricity consumption for the month of December, using the ANN developed based on the BR algorithm and on the NAR model.

Another important plot that we have computed and represented in order to analyze the forecasting accuracy is represented by the regressions between the network targets and outputs (Figure 7).

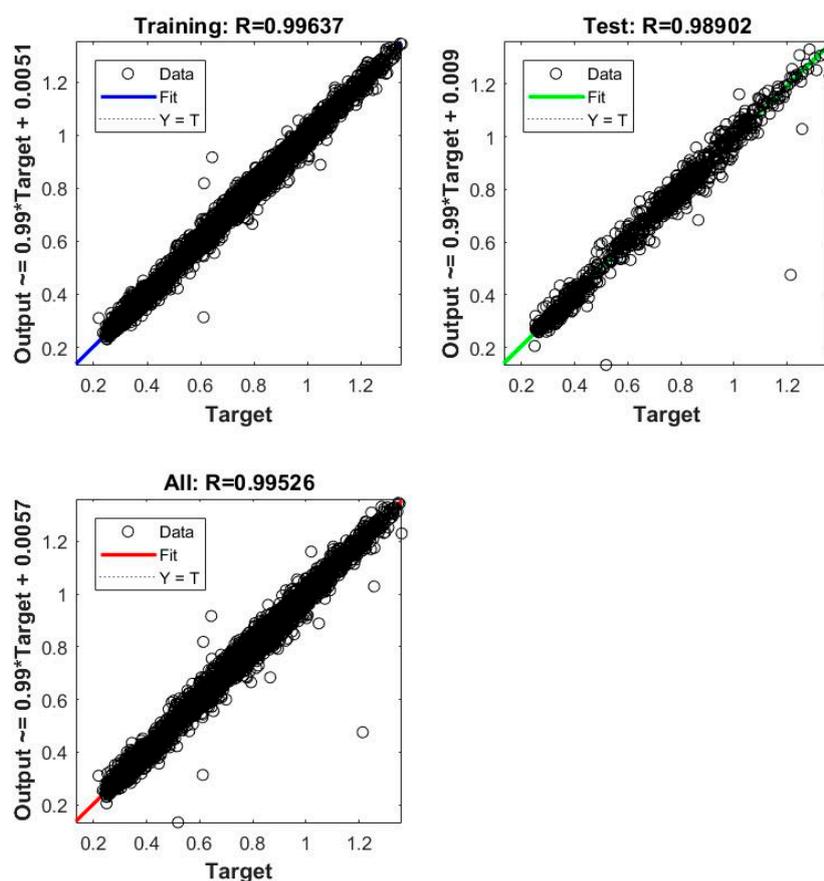


Figure 7. The regressions between the network targets and network outputs when forecasting the electricity consumption for the month of December, using the ANN_NAR_BR network.

We have obtained three regression plots corresponding to the training, testing and to the whole data set, in which the solid line represents the best-fitting linear regression line between the targets and outputs in each case, while the dashed line represents the ideal case, when the outputs and targets are identical. Analyzing these plots, we have noticed that we have obtained a very good fit as the values of the correlation coefficient R are very close to 1, all of them being greater than 0.98902.

In order to validate the network's performance, we have also represented the error autocorrelation function that depicts the way in which the forecasting errors are interlinked in time. As we have used the non-normalized autocorrelations, the autocorrelation function has as units of measurement the input data's units squared (MWh^2). In the ideal case, the plot should contain only a single nonzero value of this function that corresponds to the zero lag and represents the MSE. In this situation, all the errors would have been completely uncorrelated between them. In our case, besides the zero-lag correlation, most of them fall within the 95% confidence limits around zero and thus, it is confirmed the validity of the forecasting method (Figure 8).

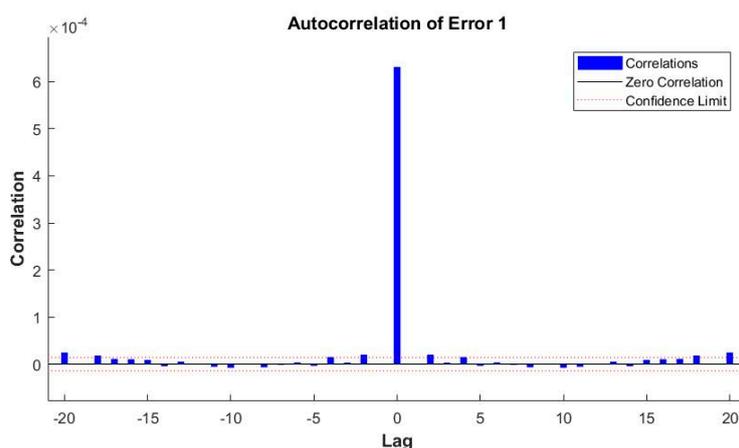


Figure 8. The error autocorrelation function when forecasting the electricity consumption for the month of December, using the ANN_NAR_BR network.

In the following, we present an analysis of the results that we have obtained using the artificial neural network forecasting solution, based on the NARX model, using the meteorological and the time stamps datasets as exogenous variables.

3.2. Results Regarding the Developed ANN Forecasting Solution Based on the NARX Model, Using as Exogenous Variables the Meteorological and the Time Stamps Datasets

Pursuing to the above presented methodology, we have developed the artificial neural networks forecasting solution based on the NARX model, using as exogenous variables the meteorological and the time stamps datasets and we have synthesized the obtained results, highlighting the values of the mean squared error (MSE) and the correlation coefficient (R) between the network targets and the network outputs, for the whole dataset (Table 3).

Table 3. The synthesis of the experimental results when developing the artificial neural networks forecasting solution based on the NARX model, using as exogenous variables the meteorological and the time stamps datasets.

The Levenberg-Marquardt Training Algorithm						
<i>n</i>	<i>d</i>	2	6	12	24	48
6	MSE	0.0012052	0.001088	0.00081051	0.00080346	0.00077934
	R	0.99036	0.99218	0.99291	0.99346	0.99394
12	MSE	0.00080556	0.00086619	0.00087007	0.00070308	0.00084926
	R	0.99227	0.99254	0.99386	0.99402	0.99375
24	MSE	0.00085223	0.00087959	0.00089669	0.00096489	0.00090853
	R	0.99364	0.9926	0.99338	0.99425	0.99401
The Bayesian Regularization Training Algorithm						
<i>n</i>	<i>d</i>	2	6	12	24	48
6	MSE	0.0013377	0.0009888	0.00071482	0.00054033	0.00043039
	R	0.98865	0.99258	0.99387	0.99529	0.99561
12	MSE	0.00085773	0.00073291	0.00054438	0.00044708	0.00035744
	R	0.9918	0.99445	0.99555	0.9952	0.99616
24	MSE	0.00068362	0.00066045	0.0004577	0.00033796	0.0002294
	R	0.99444	0.99491	0.99578	0.9967	0.99701
The Scaled Conjugate Gradient Training Algorithm						
<i>n</i>	<i>d</i>	2	6	12	24	48
6	MSE	0.0022016	0.0020858	0.0012062	0.0010987	0.00078457
	R	0.98175	0.98928	0.99079	0.99142	0.99293
12	MSE	0.0035236	0.002518	0.0016623	0.00077795	0.00088082
	R	0.97808	0.98292	0.98814	0.99254	0.99299
24	MSE	0.0024664	0.0014422	0.0011421	0.00091561	0.0011415
	R	0.97996	0.98956	0.98974	0.99227	0.99231

Analyzing the results, we have noticed that in the case of the LM training algorithm, the network developed according to the architecture that comprises $n = 12$ neurons in the hidden layer and a delay parameter of $d = 24$ provides the best hourly forecasting accuracy, as it has the lowest value of the mean squared error (0.00070308) and the value of the correlation coefficient computed for the whole dataset very close to 1 (0.99402). In contrast to this, the network developed using $n = 6$ neurons in the hidden layer and the delay parameter of $d = 2$ provides the worst hourly forecasting accuracy, as it provides the highest value of the mean squared error (0.0012052) and the value of the correlation coefficient computed for the whole dataset furthest from 1 (0.99036). Even if this network provides the lowest hourly forecasting accuracy out of the 15 networks trained based on the LM algorithm, it still provides a good hourly forecasting accuracy.

We have observed that in the case of the BR training algorithm, the network developed using $n = 24$ neurons in the hidden layer and a delay parameter of $d = 48$ offers the best forecasting in terms of accuracy, because it has the lowest value of the mean squared error (0.0002294) and the value of the correlation coefficient computed for the whole dataset closest to 1 (0.99701). On the contrary, the network developed according to the architecture that comprises a delay parameter of $d = 2$ and $n = 6$ neurons in the hidden layer provides the worst hourly forecasting accuracy, as it has the highest value of the mean squared error (0.0013377) and the value of the correlation coefficient computed for the whole dataset furthest from 1 (0.98865).

After analyzing the results obtained in the case of the SCG training algorithm, we have perceived that the network developed using $n = 12$ neurons in the hidden layer and a delay parameter of $d = 24$ provides the best hourly forecasting accuracy, as it has registered the lowest value of the mean squared error (0.00077795) and the value of the correlation coefficient computed for the whole dataset closest to 1 (0.99254). The network developed according to the architecture that comprises $n = 12$ neurons in the hidden layer and a delay parameter of $d = 2$ has provided the worst prediction accuracy, because it has registered the highest value of the mean squared error (0.0035236) and the value of the correlation coefficient computed for the whole dataset furthest from 1 (0.97808). Despite having the lowest hourly forecasting accuracy out of the 15 networks trained based on the SCG algorithm, this network still provides a good hourly forecasting accuracy.

The obtained results show that the best hourly forecasting accuracy is obtained when the delay parameter is 24 for the LM and SCG algorithms, while for the BR algorithm it has the value 48. In all the cases, the value of d is a multiple of 24, which means that in order to obtain an improved hourly forecasting accuracy one needs the precedent values for a full day or for 2 consecutive days.

As we were trying to further increase the delay, we have faced two drawbacks: the first one consists in a very long period of time necessary for training the networks, especially in the case of the BR algorithm (in the case of $n = 24$ neurons and $d = 48$, there were necessary almost 2 hours in order to train one iteration, on the hardware and software configuration mentioned above) and the second one consists in the fact that the performance gains that appear in the case of some iterations, when increasing the delay over 48 h, were negligible and could not be accounted on the delay value but rather on the random division of the dataset.

Once we have identified for each training algorithm the network that has provided the best results, we have compared the parameters of these 3 networks and we have observed that the best hourly forecasting accuracy is provided by the ANN developed using the BR algorithm, based on the NARX model, using as exogenous variables the meteorological and the time stamps datasets, with $n = 24$ neurons in the hidden layer and a delay parameter of $d = 48$ entitled ANN_NARX_BR_ALL (Figure 9).

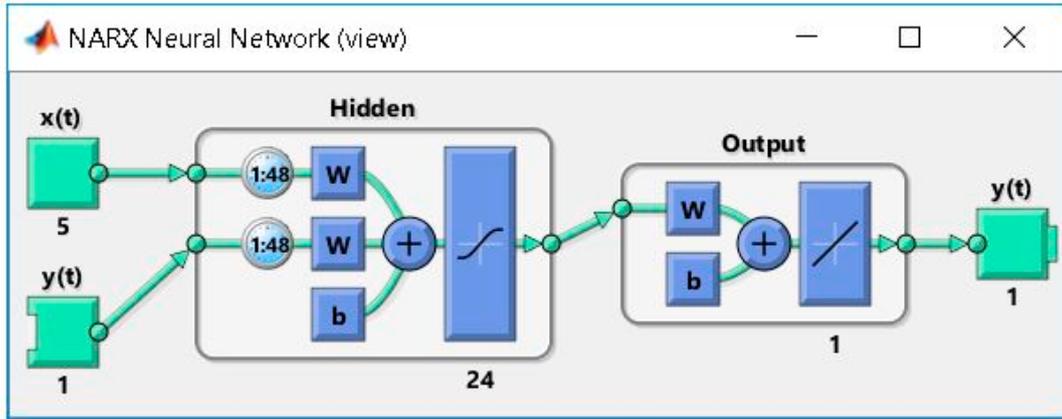


Figure 9. The ANN_NARX_BR_ALL’s architecture.

In order to analyze the training performance when forecasting the electricity consumption for the month of December 2016, using the ANN developed based on BR algorithm and on the NARX model, using as exogenous variables the meteorological and the time stamps datasets, we have first plotted the training and testing curves. In this case, we have registered the best training performance at the 1000th epoch, when the mean squared error has the value of 0.0002294. As in the case of the ANN_NAR_BR neural network, analyzing this plot we can conclude that the devised forecasting solution is stable, the overfitting process does not occur and the ANN_NARX_BR_ALL neural network offers a high level of accuracy and performance (Figure 10).

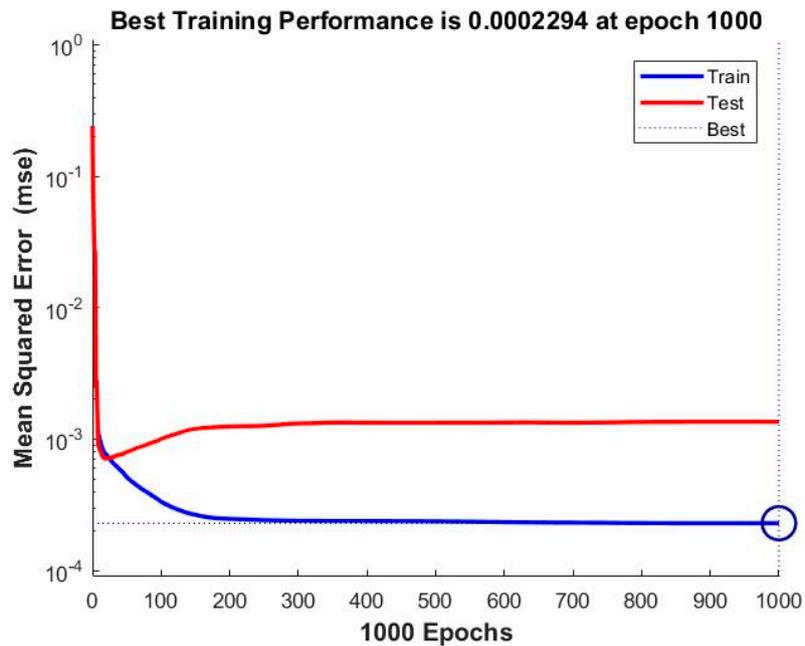


Figure 10. The best training performance when forecasting the electricity consumption for the month of December, using the ANN_NARX_BR_ALL network.

Afterwards, we have represented the error histogram, when forecasting the electricity consumption for the month of December, using the above-mentioned forecasting ANN (Figure 11).

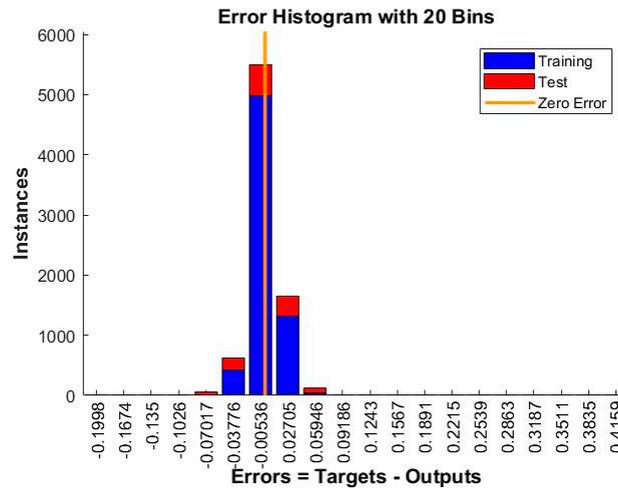


Figure 11. The error histogram when forecasting the electricity consumption for the month of December, using the ANN_NARX_BR_ALL network.

Analyzing the plot, we have noticed that most of the errors fall between -0.03776 and 0.02705 , a very narrow interval. It's only for a very few training points that the errors fall outside this range. Thus, the error histogram highlights very good results in this case.

Afterwards, in order to analyze the forecasting accuracy, we have computed and represented another important plot, the regressions between the network targets and outputs. The values of the correlation coefficient R are very close to 1, all of them being greater than 0.99701. Therefore, we can conclude that we have obtained a very good fit (Figure 12).

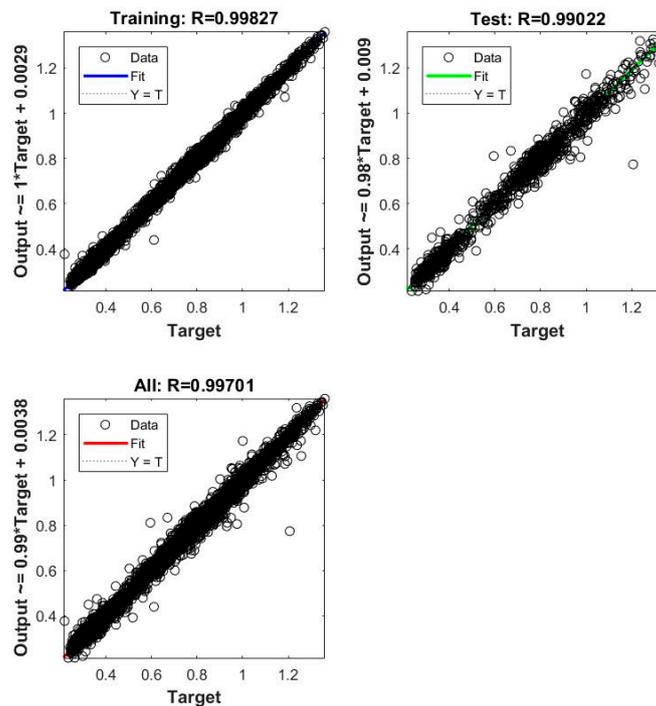


Figure 12. The regressions between the network targets and network outputs when forecasting the electricity consumption for the month of December, using the ANN_NARX_BR_ALL network.

In order to validate the network's performance, we have also analyzed the way in which the forecasting errors are interlinked in time, through the error autocorrelation function. In our case, besides the zero-lag correlation, the rest of them fall within the 95% confidence limits around zero and thus, the validity of the forecasting method is confirmed (Figure 13).

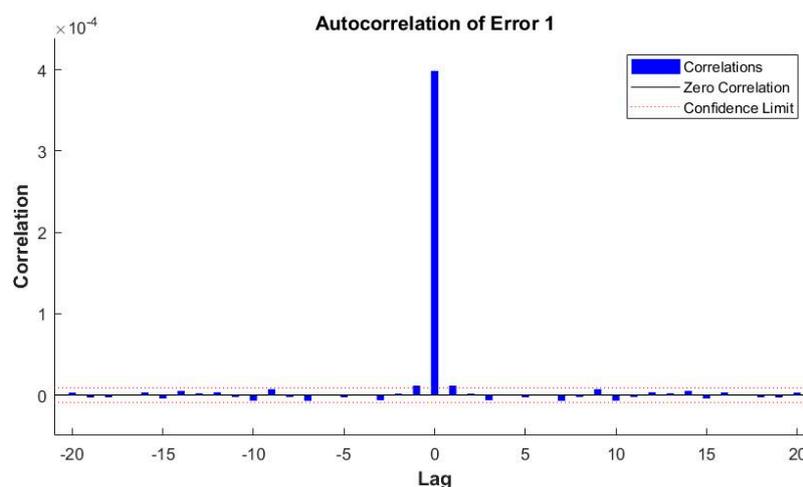


Figure 13. The error autocorrelation function when forecasting the electricity consumption for the month of December, using the ANN_NARX_BR_ALL network.

In the following, we present an analysis of the results that we have obtained using the artificial neural network forecasting solution, based on the NARX model, using only the time stamps datasets as exogenous variables.

3.3. Results Regarding the Developed ANN Forecasting Solution Based on the NARX Model, Using as Exogenous Variables the Time Stamps Datasets

Carrying out the above presented methodology, after developing the artificial neural networks forecasting solution based on the NARX model, using as exogenous variables the time stamps datasets, we have synthesized the obtained results, highlighting the values of the mean squared error MSE and the correlation coefficient R between the network targets and the network outputs, for the whole dataset (Table 4).

Table 4. The synthesis of the experimental results when developing the artificial neural networks forecasting solution based on the NARX model, using as exogenous variables the time stamps datasets.

The Levenberg-Marquardt Training Algorithm						
<i>n</i>	<i>d</i>	2	6	12	24	48
6	MSE	0.001455	0.0019983	0.00084518	0.00087728	0.00084281
	R	0.9878	0.99162	0.99325	0.99266	0.9932
12	MSE	0.00094045	0.0012128	0.00096237	0.00076798	0.0009589
	R	0.99159	0.99163	0.99347	0.99388	0.99382
24	MSE	0.00098655	0.0011564	0.0018315	0.0018474	0.0010069
	R	0.99288	0.99228	0.99377	0.99306	0.99352
The Bayesian Regularization Training Algorithm						
<i>n</i>	<i>d</i>	2	6	12	24	48
6	MSE	0.001611	0.0010559	0.00072309	0.00061356	0.00050737
	R	0.98814	0.99231	0.99393	0.99459	0.99543
12	MSE	0.00086453	0.00075894	0.00058258	0.00047763	0.00038671
	R	0.99256	0.99317	0.99475	0.995	0.99597
24	MSE	0.00070019	0.00066089	0.00049556	0.00039866	0.00032274
	R	0.99456	0.99381	0.9948	0.99604	0.99623
The Scaled Conjugate Gradient Training Algorithm						
<i>n</i>	<i>d</i>	2	6	12	24	48
6	MSE	0.0026367	0.0019962	0.001349	0.001295	0.0009588
	R	0.97828	0.98305	0.98974	0.99148	0.99208
12	MSE	0.0039318	0.0028443	0.002392	0.00085282	0.000881
	R	0.97681	0.98497	0.98638	0.99254	0.99231
24	MSE	0.0036967	0.0016997	0.0013259	0.0010325	0.0010516
	R	0.97367	0.98609	0.98908	0.99164	0.99249

Analyzing the results, we have noticed that in the case of the LM training algorithm, the best hourly forecasting accuracy is registered by the network developed according to the architecture that comprises $n = 12$ neurons in the hidden layer and a delay parameter of $d = 24$, as it has the lowest value of the mean squared error (0.00076798) and the value of the correlation coefficient computed for the whole dataset closest to 1 (0.99388). Contrariwise, the network developed using $n = 6$ neurons in the hidden layer and the delay parameter of $d = 6$ provides the worst hourly forecasting accuracy, having the highest value of the mean squared error (0.0019983) and the value of the correlation coefficient computed for the whole dataset among the furthest from 1 (0.99162). Even though this network provides the lowest hourly forecasting accuracy out of the 15 ones trained based on the LM algorithm, it still provides a good hourly forecasting accuracy.

In the case of the BR training algorithm, we have observed that the network developed using $n = 24$ neurons in the hidden layer and a delay parameter of $d = 48$ offers the best forecasting accuracy, as it has the lowest value of the mean squared error (0.00032274) and the value of the correlation coefficient computed for the whole dataset closest to 1 (0.99623). On the opposite side, the network developed according to the architecture that comprises a delay parameter of $d = 2$ and $n = 6$ neurons in the hidden layer provides the worst hourly forecasting accuracy, as it has the highest value of the mean squared error (0.001611) and the value of the correlation coefficient computed for the whole dataset furthest from 1 (0.98814).

After analyzing the results obtained in the case of the SCG training algorithm, we have perceived that the network developed using $n = 12$ neurons in the hidden layer and a delay parameter of $d = 2$ provides the best hourly forecasting accuracy, as it has registered the lowest value of the mean squared error (0.00085282) and the value of the correlation coefficient computed for the whole dataset closest to 1 (0.99254). The network developed according to the architecture that comprises $n = 12$ neurons in the hidden layer and a delay parameter of $d = 2$ has provided the worst prediction accuracy, having registered the highest value of the mean squared error (0.0039318) and the value of the correlation coefficient computed for the whole dataset among the furthest from 1 (0.97681). Despite having the lowest hourly forecasting accuracy out of the 15 networks trained based on the SCG algorithm, this network still provides a good hourly forecasting accuracy.

The best hourly forecasting accuracy is obtained when the delay parameter is 24 for the LM and SCG algorithms, while for the BR algorithm it has the value 48. In these cases, the value of d is a multiple of 24, which means that in order to obtain an improved hourly forecasting accuracy, one needs the previous values for a full day or for 2 consecutive days.

When we were trying to further increase the delay, we have faced the same drawbacks as in the case of the developed ANN forecasting solution based on the NARX model using as exogenous variables the meteorological and the time stamps datasets.

After we have identified the network that has provided the best results for each of the training algorithms, we have compared the parameters of these three networks and we have observed that the best hourly forecasting accuracy is provided by the ANN developed using the BR algorithm, based on the NARX model, using as exogenous variables the time stamps datasets, with $n = 24$ neurons in the hidden layer and a delay parameter of $d = 48$, entitled ANN_NARX_BR_TS (Figure 14).

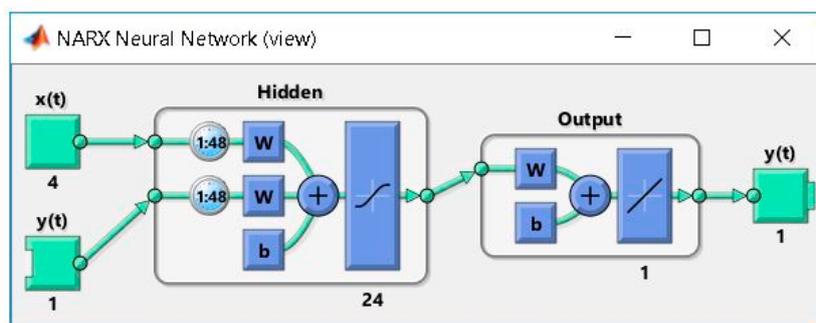


Figure 14. The ANN_NARX_BR_TS's architecture.

In order to analyze the training performance when forecasting the electricity consumption for the month of December 2016, using the ANN developed based on BR algorithm and on the NARX model, using as exogenous variables the time stamps datasets, we have first plotted the training and testing curves. In this case, we have registered the best training performance at the 898th epoch, when the mean squared error has the value of 0.00032274. As in the previous case of the ANN_NAR_BR and ANN_NARX_BR_ALL networks, analyzing the performance plot, one can notice that the forecasting solution is stable, the overfitting process does not occur and thus the chart confirms the high level of accuracy and performance of the ANN_NARX_BR_TS neural network (Figure 15).

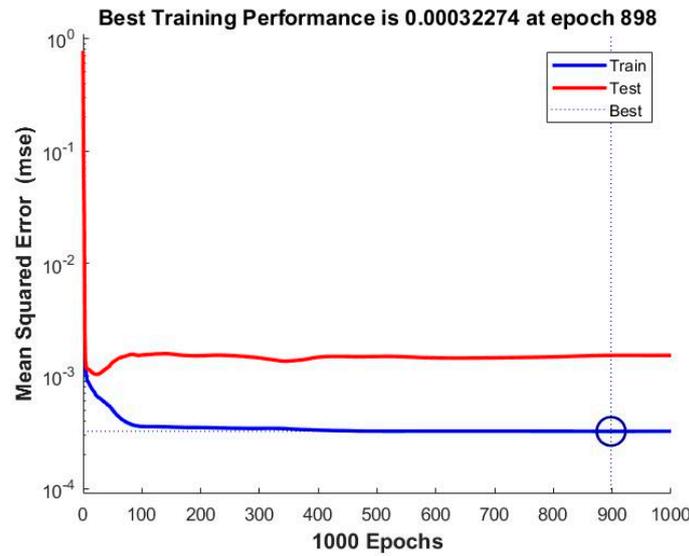


Figure 15. The best training performance when forecasting the electricity consumption for the month of December, using the ANN_NARX_BR_TS network.

Subsequently, we have represented the error histogram, when forecasting the electricity consumption for the month of December, using the above-mentioned forecasting ANN (Figure 16). Analyzing the plot, we have noticed that most of the errors fall between -0.002324 and 0.02492 , a very narrow interval. The errors fall outside this range only for a very few training points and therefore the error histogram highlights very good results.

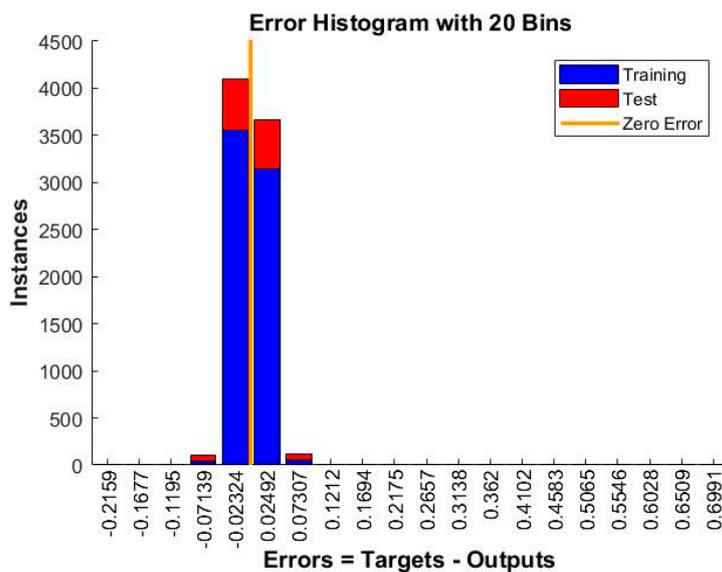


Figure 16. The error histogram when forecasting the electricity consumption for the month of December, using the ANN_NARX_BR_TS network.

Afterwards, in order to analyze the forecasting accuracy, we have computed and represented the regressions between the network targets and outputs. The values of the correlation coefficient R are very close to 1, all of them being greater than 0.99623. Therefore, we can conclude that we have obtained a very good fit (Figure 17).

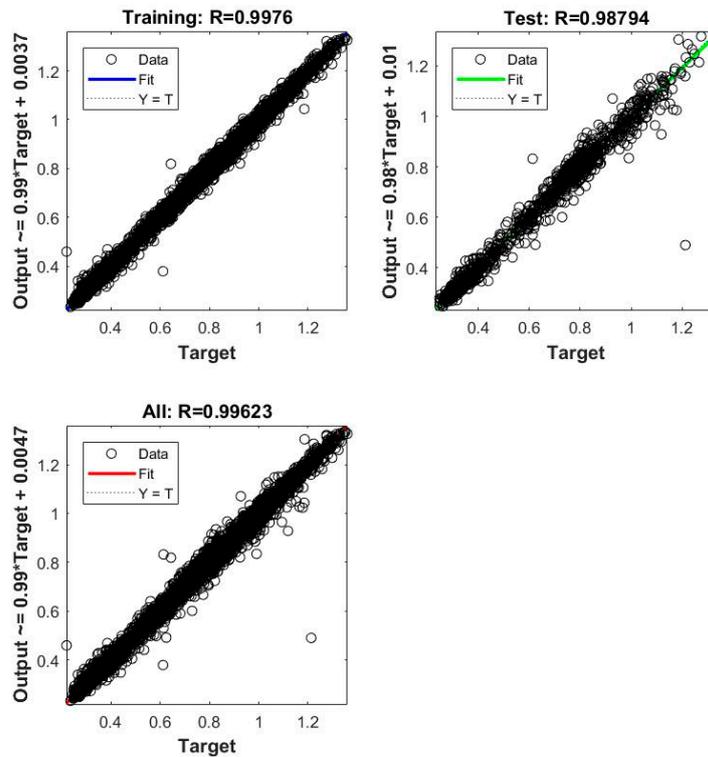


Figure 17. The regressions between the network targets and network outputs when forecasting the electricity consumption for the month of December, using the ANN_NARX_BR_TS network.

In order to validate the network’s performance, we have also represented the error autocorrelation function that depicts the way in which the forecasting errors are interlinked in time. Besides the zero-lag correlation, the rest of the errors fall within the 95% confidence limits around zero and thus, the validity of the forecasting method is confirmed (Figure 18).

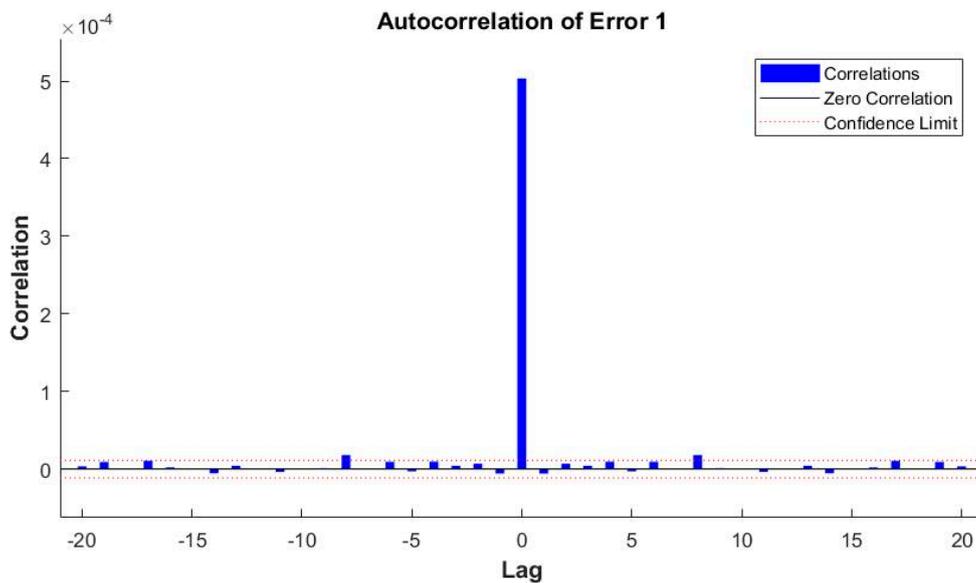


Figure 18. The error autocorrelation function when forecasting the electricity consumption for the month of December, using the ANN_NARX_BR_TS network.

In the following, we present the way in which, according to the above presented methodology, we have obtained the best forecasting solution out of the three developed ones, depicted above and denoted by ANN_NAR_BR, ANN_NARX_BR_ALL and ANN_NARX_BR_TS.

3.4. Results Concerning the Best Forecasting Solution

According to our research methodology, in order to obtain a final validation based on a real-world forecasting scenario, in the next step we have forecasted the electricity consumption for the month of December, using the forecasting ANNs selected above (ANN_NAR_BR, ANN_NARX_BR_ALL and ANN_NARX_BR_TS) and we have compared the obtained results with the corresponding real values, provided by the commercial center type of consumer.

In this purpose, we have first plotted on the same chart the real consumption and the forecasted ones. Analyzing the plot, we have noticed that the four curves are very close to each other, fact that reflects the high level of accuracy of the three developed forecasting solutions (Figure 19).

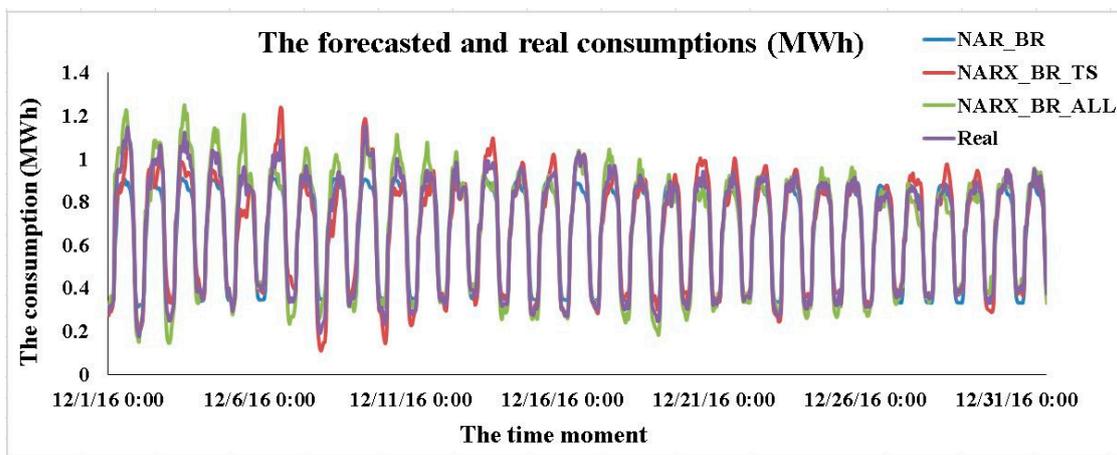


Figure 19. The real and forecasted consumptions.

Afterwards, in order to obtain a more eloquent comparison, we have computed and plotted on a second chart the absolute values of the differences between the real consumption and each of the three forecasted ones (measured in MWh) (Figure 20).

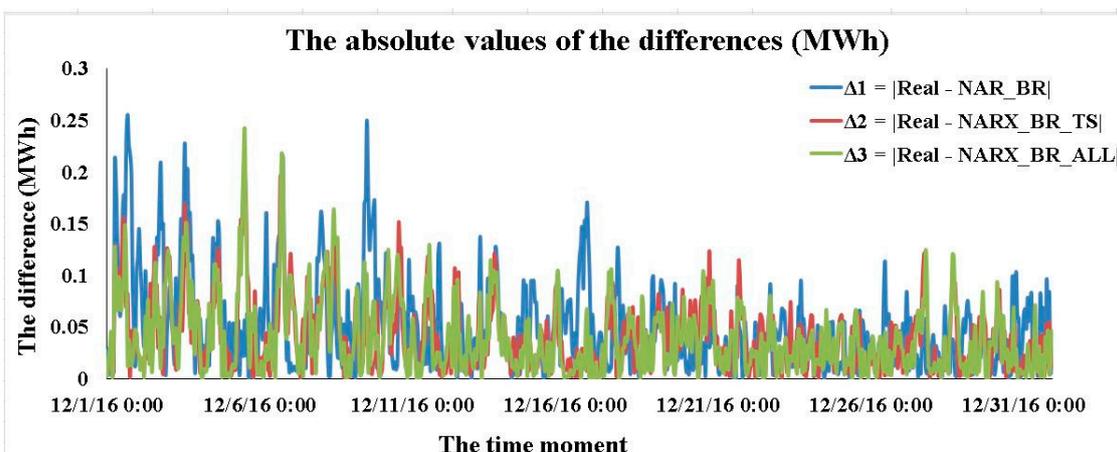


Figure 20. The absolute values of the differences between the real consumption and each of the three forecasted ones.

Based on this second chart, we are able to identify that the network that provides the best forecasting results, as close as possible to the real values, is ANN_NARX_BR_ALL, as in this case the absolute values of the differences are closest to the zero value. On the second place is situated the

ANN_NARX_BR_TS neural network. Detailed information about the predicted datasets can be found in the Supplementary Materials.

We also wanted to verify if the hierarchy of networks is confirmed by comparing their relevant parameters (the values of the mean squared error MSE and the correlation coefficient R between the network targets and the network outputs, for the whole dataset). In this purpose, we have selected the values from the above presented Tables 2–4.

Analyzing the results synthetized in this table, we have noticed that the lowest value of the mean squared error and the value of the correlation coefficient closest to 1 are those registered for the ANN_NARX_BR_ALL forecasting solution, followed by the ANN_NARX_BR_TS one. This comparison confirms the results obtained when analyzing the networks' performance in a real-world forecasting scenario for the month of December 2016, highlighted in Figures 19 and 20. The hierarchy of the three developed forecasting solutions, obtained by analyzing Table 5 is matching the one obtained by analyzing Figure 19 along with Figure 20, this fact confirming the relevance and correctness of our devised methodology.

In all the analyzed situations (the three ANNs forecasting solutions, developed based on the NAR and NARX models), we have also identified the networks that have provided the lowest performance and we have noticed that these networks still offer a good hourly forecasting accuracy.

Table 5. The relevant parameters of the ANN_NAR_BR, ANN_NARX_BR_ALL and ANN_NARX_BR_TS forecasting solutions.

No.	The Forecasting Solution	MSE	R
1	ANN_NAR_BR	0.00048272	0.99526
2	ANN_NARX_BR_ALL	0.0002294	0.99701
3	ANN_NARX_BR_TS	0.00032274	0.99623

4. Discussion

The research methodology has focused on devising three hourly consumption forecasting solutions for achieving energy efficiency in the case of large electricity non-household consumers. Thus, we have developed a series of artificial neural networks based on the NAR and NARX approaches, using LM, BR, SCG training algorithms and different settings regarding the hidden number of neurons n and the delay parameter d . According to our devised methodology, we have afterwards compared the forecasting accuracy of these networks and finally, we have obtained the network that provides the best forecasting results, as close as possible to the real values. Detailed information about the predicted datasets can be found in the Supplementary Materials.

We have conducted a detailed examination of the obtained results in order to obtain a more refined analysis. In this purpose, we have studied the results synthetized in the Tables 2–4 noticing that in most of the analyzed situations, the hourly forecasting accuracy improves along with the increase of the delay parameter value, the best results being obtained when d was 24 or 48 for all the training algorithms, in both the cases of NAR and NARX models. Therefore, in order to obtain an improved hourly forecasting accuracy, one needs 1 or 2 consecutive days' values. The optimum mix (n, d) and the corresponding performance metrics are synthetized in Table 6.

Table 6. The optimum mix (n, d) and the corresponding performance metrics.

The Training Algorithm	The Model		
	NAR	NARX with Meteorological and Timestamps Exogenous Data	NARX with Timestamps Exogenous Data
LM	$n = 6, d = 48$	$n = 12, d = 24$	$n = 12, d = 24$
	$MSE = 0.00091808$	$MSE = 0.00070308$	$MSE = 0.00076798$
	$R = 0.99272$	$R = 0.99402$	$R = 0.99388$
BR	$n = 24, d = 48$	$n = 24, d = 48$	$n = 24, d = 48$
	$MSE = 0.00048272$	$MSE = 0.0002294$	$MSE = 0.00032274$
	$R = 0.99526$	$R = 0.99701$	$R = 0.99623$
SCG	$n = 12, d = 48$	$n = 12, d = 24$	$n = 12, d = 24$
	$MSE = 0.00083771$	$MSE = 0.00077795$	$MSE = 0.00085282$
	$R = 0.99241$	$R = 0.99254$	$R = 0.99254$

In addition, in all these cases the BR has proved to be the best training algorithm, as the network developed using $n = 24$ neurons in the hidden layer and a delay parameter of $d = 48$ has offered the best forecasting accuracy, having the lowest value of the mean squared error and the value of the correlation coefficient computed for the whole dataset closest to 1. The neural networks that have provided the lowest prediction accuracy out of the 135-developed forecasting neural networks are synthetized in Table 7.

Table 7. The worst mix (n, d) and the corresponding performance metrics.

The Training Algorithm	The Model		
	NAR	NARX with Meteorological and Timestamps Exogenous Data	NARX with Timestamps Exogenous Data
LM	$n = 24, d = 2$	$n = 6, d = 2$	$n = 6, d = 6$
	$MSE = 0.0037219$	$MSE = 0.0012052$	$MSE = 0.0019983$
	$R = 0.97486$	$R = 0.99036$	$R = 0.99162$
BR	$n = 6, d = 2$	$n = 6, d = 2$	$n = 6, d = 2$
	$MSE = 0.0029627$	$MSE = 0.0013377$	$MSE = 0.001611$
	$R = 0.97709$	$R = 0.98865$	$R = 0.98814$
SCG	$n = 12, d = 2$	$n = 12, d = 2$	$n = 12, d = 2$
	$MSE = 0.0052954$	$MSE = 0.0035236$	$MSE = 0.0039318$
	$R = 0.97055$	$R = 0.97808$	$R = 0.97681$

Analyzing these results, one can conclude that in most of the cases the worst results were obtained when the value of the delay parameter was $d = 2$, while the number of neurons in the hidden layer varies. Even if these networks provide the lowest hourly forecasting accuracy within a certain model and training algorithm, they still offer a good hourly forecasting accuracy. Analyzing and comparing the results presented in Tables 6 and 7, we have noticed that the improvement of the mean squared error of the best mix when compared to the worst one, ranges between 41.66% and 84.18%, while in the case of the correlation coefficient R between the network targets and the network outputs, for the whole dataset, the improvement ranges between 0.23% and 2.25% (Table 8).

Table 8. The performance metrics improvement when comparing the best with the worst mix (n, d).

The Training Algorithm	The Model		
	NAR	NARX with Meteorological and Timestamps Exogenous Data	NARX with Timestamps Exogenous Data
LM	75.33% for the MSE	41.66% for the MSE	61.57% for the MSE
	1.83% for R	0.37% for R	0.23% for R
BR	83.71% for the MSE	82.85% for the MSE	79.97% for the MSE
	1.86% for R	0.85% for R	0.82% for R
SCG	84.18% for the MSE	77.92% for the MSE	78.31% for the MSE
	2.25% for R	1.48% for R	1.61% for R

Even if the first developed forecasting solution, based on the NAR model, has proved to provide a very good prediction accuracy, as we wanted to further improve the accuracy of the obtained results, we have first developed a forecasting solution based on the NARX model, using as exogenous datasets the temperature and the time stamps datasets. By applying this method, the forecasting accuracy has been improved but the temperature dataset used as an exogenous variable limits significantly the forecasting timeframe as when using the forecasting ANN one should know beforehand the temperature for the time steps for which he needs to forecast the consumption.

Therefore, as we wanted to see if we could obtain a forecasting solution based on the NARX model, for which the exogenous variables are easier to see if we can in the same time achieve an accurate long-term forecasting, we have developed an ANN forecasting solution, based on the NARX model, using as exogenous variable only the time stamps dataset. This solution provides a forecasting accuracy that is slightly lower but very close to the one offered by the solution that uses the two datasets of exogenous variables, while having the undeniable advantage of being easier to develop and being possible to achieve a longer forecasting timeframe, as it does not require the acquisition of a supplementary dataset (the temperature).

The results obtained using the NARX model reveal a strong relation between the exogenous variables and the electricity consumption that makes our proposed forecasting method a viable alternative to other methods from the literature. For example, comparing with [28], in which the authors do not consider the meteorological data and with [30], in which the authors consider only national and religious holidays, we have introduced some specific exogenous variables (meteorological and timestamps datasets) that influence the hypermarket's consumption. Comparing with [29], in which the authors have used the feed-forward ANNs in order to develop a short-term forecasting of the Greek Power System load demand, our method is based on ANNs developed based on the NAR and NARX models, that are more suitable in forecasting the time series' future terms.

In contrast with [31], where the authors have used only the Levenberg-Marquardt training algorithm for developing the forecasting neural networks, we have used three training algorithms (LM, BR and SCG) and finally we have chosen the one that has provided the best results, reflected by the best forecasting accuracy. Moreover, in order to choose the networks' parameters (the delay parameter and the number of neurons in the hidden layer), the authors have developed a testing methodology based on a "trial-and-error procedure", in which they adjusted first the delay parameter and afterwards, using this parameter, they searched the number of neurons using the same approach. In our paper, we have developed an approach based on testing different pairs of (n, d) parameters, in the same time. Therefore, our search for obtaining the best mix between the training algorithm, the delay parameter and the number of neurons in the hidden layer was conducted in a tridimensional approach, while the approach used in [31] was twice unidimensional: the training algorithm is always LM, the searching of the best delay parameter and afterwards the one regarding the number of neurons in the hidden layer are conducted separately, in two unidimensional searches. Both the approaches (the one depicted in [31] and our method) offer a very good forecasting accuracy.

Nevertheless, it is hard to obtain a 100% relevant comparison between different existing forecasting methods in the literature, without using the same dataset and moreover the same case study. Some methods can out best others in certain scenarios, while others can be the best choice, depending on the particularities of the case study. An important criterion in assessing the effectiveness of a forecasting method consists in its ability to be generalized as to be able to apply it to a wide range of case studies. Our devised method can be adapted and tested on other datasets for non-household consumers: offices with restaurant, hotels, entertainment centers, due to the ease of building the exogenous timestamps dataset. Our analyzed case study considers the electricity consumption of a hypermarket with exclusive commercial activities. The proposed forecasting methods can also be adapted for industrial activities. In this case, more input variables could be considered whether the electricity measurement is done on separate industrial equipment. In the same time, the industrial activity specificity must be taken into account (night shifts, the availability of production line, scheduled maintenance). Moreover, additional algorithms for input data could be employed in accordance with

the requirements. Therefore, our proposed method can be adapted for other types of consumers using supplementary exogenous variables in accordance with their specific activities.

5. Conclusions

The research presented in this paper aimed to devise forecasting solutions for the hourly energy consumption in the case of large non-household electricity consumers, a prerequisite in attaining energy efficiency. The forecasting solutions have been developed using artificial neural networks based on the NAR and NARX models and on the LM, BR, SCG training algorithms. For all of these ANNs, the involved dataset consists in the hourly energy consumption recorded by smart metering device from a large hypermarket chain, while in the NARX case exogenous data have also been used, represented by meteorological and time stamp datasets.

By testing different settings regarding the training algorithm, the number of neurons in the hidden layer, the delay parameter, the networks' weights and biases and by comparing the obtained results, we have obtained in each case the neural network that provides the best prediction accuracy. Afterwards, by analyzing the forecasting performance of the selected networks (highlighted by the mean squared error, the error histogram, the regressions, the error autocorrelation plots), we have noticed that the best hourly forecasting accuracy is provided by the ANN developed using the BR algorithm, based on the NARX model, using as exogenous variables the meteorological and the time stamps datasets, with $n = 24$ neurons in the hidden layer and a delay parameter of $d = 48$ (ANN_NARX_BR_ALL).

In order to obtain a final validation based on a real-world forecasting scenario, we have forecasted the electricity consumption using the selected ANNs and we have compared the obtained results with the corresponding real values, provided by the large electricity non-household consumer thus confirming the forecasting accuracy of our developed solutions.

By applying our method, based on the flexibility of the consumer, the optimum operation can be scheduled. As a further step, the consumer can send to the supplier the day ahead consumption schedule (his forecast) and obtain based on his flexibility the optimal schedule (considering a certain objective function) that would improve the energy efficiency. Our method can be further improved by adding the consumption of individual appliances or consumption devices. This topic is complex and could be approached in a future research paper.

The hourly consumption forecasting solutions developed within this paper offer a high level of accuracy, having the potential of being useful tools for both the large electricity non-household consumers and electricity producers, offering them the means to attain an appropriate management of resources and energy efficiency.

Supplementary Materials: The following are available online at www.mdpi.com/1996-1073/10/11/1727/s1, the input datasets used for developing the solutions: electricity consumption for the whole 2016 year, the environmental temperature for the whole 2016 year, the time stamps dataset; the predicted datasets for the month of December 2016; the developed ANNs.

Acknowledgments: This work was funded by a grant of the National Research Council (CNCS), the Advisory Council for Research, Development and Innovation (CCCDI), The Executive Agency for Higher Education, Research, Development and Innovation Funding (UEFISCDI), project number PN-III-P2-2.1-BG-2016-0286 "Informatics solutions for electricity consumption analysis and optimization in smart grids" and contract no. 77BG/2016, within the National Plan for Research, Development and Innovation for the period 2015-2020 (PNCDI III).

Author Contributions: All authors contributed equally to this work.

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

References

- World Energy Balances: Overview (2017 Edition). Available online: <http://www.iea.org/publications/freepublications/publication/WorldEnergyBalances2017Overview.pdf> (accessed on 9 January 2017).
- Rigatos, G. *Advanced Models of Neural Networks: Nonlinear Dynamics and Stochasticity in Biological Neurons*; Springer: Berlin, Germany, 2015; ISBN 9783662437636.
- Krawczak, M. *Multilayer Neural Networks: A Generalized Net Perspective*; Springer Publishing Company: Heidelberg, Germany, 2013; ISBN 3319002473, 9783319002477.
- Chakraverty, S.; Mall, S. *Artificial Neural Networks for Engineers and Scientists: Solving Ordinary Differential Equations*; CRC Press: Boca Raton, FL, USA, 2017; ISBN 9781351651318.
- Da Silva, I.N.; Spatti, D.H.; Flauzino, R.A.; Liboni, L.H.B.; dos Reis Alves, S.F. *Artificial Neural Networks: A Practical Course*; Springer International Publishing: Basel, Switzerland, 2016; ISBN 9783319431628.
- Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning (Adaptive Computation and Machine Learning Series)*; MIT Press: Cambridge, UK, 2016; ISBN 9780262035613.
- Almonacid, F.; Fernandez, E.F.; Mellit, A.; Kalogirou, S. Review of techniques based on artificial neural networks for the electrical characterization of concentrator photovoltaic technology. *Renew. Sustain. Energy Rev.* **2017**, *75*, 938–953, doi:10.1016/j.rser.2016.11.075.
- Qazi, A.; Fayaz, H.; Wadi, A.; Raj, R.G.; Rahim, N.A.; Khan, W.A. The artificial neural network for solar radiation prediction and designing solar systems: A systematic literature review. *J. Clean. Prod.* **2015**, *104*, 1–12, doi:10.1016/j.jclepro.2015.04.041.
- Yadav, A.K.; Chandel, S.S. Solar radiation prediction using Artificial Neural Network techniques: A review. *Renew. Sustain. Energy Rev.* **2014**, *33*, 772–781, doi:10.1016/j.rser.2013.08.055.
- Du, K.L.; Swamy, M.N.S. *Neural Networks and Statistical Learning*; SpringerLink: London, UK, 2013; ISBN 9781447155713.
- Kumar, R.; Aggarwal, R.K.; Sharma, J.D. Comparison of regression and artificial neural network models for estimation of global solar radiations. *Renew. Sustain. Energy Rev.* **2015**, *52*, 1294–1299, doi:10.1016/j.rser.2015.08.021.
- Yadav, A.K.; Chandel, S.S. Identification of relevant input variables for prediction of 1-minute time-step photovoltaic module power using Artificial Neural Network and Multiple Linear Regression Models. *Renew. Sustain. Energy Rev.* **2017**, *77*, 955–969, doi:10.1016/j.rser.2016.12.029.
- Paliwal, M.; Kumar, U.A. Neural networks and statistical techniques: A review of applications. *Expert Syst. Appl.* **2009**, *36*, 2–17, doi:10.1016/j.eswa.2007.10.005.
- Bou-rabee, M.; Sulaiman, S.A.; Saleh, M.S.; Mara, S. Using artificial neural networks to estimate solar radiation in Kuwait. *Renew. Sustain. Energy Rev.* **2017**, *72*, 434–438, doi:10.1016/j.rser.2017.01.013.
- Xiao, Q.; Xing, L.; Song, G. Time series prediction using optimal theorem and dynamic Bayesian network. *Opt. Int. J. Light Electron Opt.* **2016**, *127*, 11063–11069, doi:10.1016/j.ijleo.2016.09.002.
- Proskuryakov, A. Intelligent System for Time Series Forecasting. *Procedia Comput. Sci.* **2017**, *103*, 363–369.
- Xiao, Q. Time series prediction using bayesian filtering model and fuzzy neural networks. *Opt. Int. J. Light Electron Opt.* **2017**, *140*, 104–113, doi:10.1016/j.ijleo.2017.03.096.
- Tealab, A.; Hefny, H.; Badr, A. Forecasting of nonlinear time series using ANN. *Future Comput. Inform. J.* **2017**, *2*, 39–47, doi:10.1016/j.FCIJ.2017.05.001.
- Ibrahim, M.; Jemei, S.; Wimmer, G.; Hissel, D. Nonlinear autoregressive neural network in an energy management strategy for battery/ultra-capacitor hybrid electrical vehicles. *Electr. Power Syst. Res.* **2016**, *136*, 262–269, doi:10.1016/j.epsr.2016.03.005.
- Yang, D.; Dong, Z.; Lim, L.H.I.; Liu, L. Analyzing big time series data in solar engineering using features and PCA. *Sol. Energy* **2017**, *153*, 317–328, doi:10.1016/j.solener.2017.05.072.
- Hirata, Y.; Aihara, K. Improving time series prediction of solar irradiance after sunrise: Comparison among three methods for time series prediction. *Sol. Energy* **2017**, *149*, 294–301, doi:10.1016/j.solener.2017.04.020.
- Zheng, T.; Chen, R. Dirichlet ARMA models for compositional time series. *J. Multivar. Anal.* **2017**, *158*, 31–46, doi:10.1016/j.jmva.2017.03.006.
- Deb, C.; Zhang, F.; Yang, J.; Lee, S.E.; Shah, K.W. A review on time series forecasting techniques for building energy consumption. *Renew. Sustain. Energy Rev.* **2017**, doi:10.1016/j.rser.2017.02.085.
- Balestrassi, P.P.; Popova, E.; Paiva, A.P.; Marangon Lima, J.W. Design of experiments on neural network's training for nonlinear time series forecasting. *Neurocomputing* **2009**, *72*, 1160–1178, doi:10.1016/j.neucom.2008.02.002.

25. Benmouiza, K.; Cheknane, A. Forecasting hourly global solar radiation using hybrid k-means and nonlinear autoregressive neural network models. *Energy Convers. Manag.* **2013**, *75*, 561–569, doi:10.1016/j.enconman.2013.07.003.
26. Wong, S.L.; Wan, K.K. W.; Lam, T.N.T. Artificial neural networks for energy analysis of office buildings with daylighting. *Appl. Energy* **2010**, *87*, 551–557, doi:10.1016/j.apenergy.2009.06.028.
27. Mocanu, E.; Nguyen, P.H.; Gibescu, M.; Kling, W.L. Comparison of machine learning methods for estimating energy consumption in buildings. In Proceedings of the 13th International Conference on Probabilistic Methods Applied to Power Systems, Durham, UK, 7–10 July 2014.
28. Esener, I.I.; Yüksel, T.; Kurban, M. Short-term load forecasting without meteorological data using AI-based structures. *Turk. J. Electr. Eng. Comput. Sci.* **2015**, *23*, 370–380, doi:10.3906/elk-1209-28.
29. Tsekouras, G.J.; Kanellos, F.D.; Mastorakis, N. Short Term Load Forecasting in Electric Power Systems with Artificial Neural Networks. In *Computational Problems in Science and Engineering*; Mastorakis, N., Bulucea, A., Tsekouras, G., Eds.; Springer International Publishing: Cham, Germany, 2015; pp. 19–58, ISBN 978-3-319-15765-8.
30. Tanıdır, Ö.; Tor, O.B. Accuracy of ANN based day-ahead load forecasting in Turkish power system: Degrading and improving factors. *Neural Netw. World* **2015**, *25*, 443–456, doi:10.14311/NNW.2015.25.023.
31. Ruiz, L.; Cuéllar, M.; Calvo-Flores, M.; Pegalajar Jiménez, M.; Del, C. An Application of Non-Linear Autoregressive Neural Networks to Predict Energy Consumption in Public Buildings. *Energies* **2016**, *9*, 684, doi:10.3390/en9090684.
32. Sun, C.; Sun, F.; Moura, S.J. Nonlinear predictive energy management of residential buildings with photovoltaics & batteries. *J. Power Sources* **2016**, *325*, 723–731, doi:10.1016/j.jpowsour.2016.06.076.
33. Bogomolov, A.; Lepri, B.; Larcher, R.; Antonelli, F.; Pianesi, F.; Pentland, A. Energy consumption prediction using people dynamics derived from cellular network data. *EPJ Data Sci.* **2016**, *5*, doi:10.1140/epjds/s13688-016-0075-3.
34. Mauledoux, M.; Aviles, O.; Mejia-Ruda, E.; Caldas, O.I. Analysis of autoregressive predictive models and artificial neural networks for irradiance estimation. *Indian J. Sci. Technol.* **2016**, *9*, doi:10.17485/ijst/2016/v9i38/86214.
35. Buitrago, J.; Asfour, S. Short-term forecasting of electric loads using nonlinear autoregressive artificial neural networks with exogenous vector inputs. *Energies* **2017**, *10*, doi:10.3390/en10010040.
36. Gajowniczek, K.; Zabkowski, T. Electricity forecasting on the individual household level enhanced based on activity patterns. *PLoS ONE* **2017**, *12*, doi:10.1371/journal.pone.0174098.
37. Levenberg, K. A method for the solution of certain non-linear problems in least squares. *Q. J. Appl. Math.* **1944**, *2*, 164–168, doi:10.1017/CBO9781107415324.004.
38. Marquardt, D.W. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *J. Soc. Ind. Appl. Math.* **1963**, *11*, 431–441, doi:10.1137/0111030.
39. Kişi, Ö.; Uncuoğlu, E. Comparison of three back-propagation training algorithms for two case studies. *Indian J. Eng. Mater. Sci.* **2005**, *12*, 434–442.
40. Neural Network Object Properties—MATLAB & Simulink. Available online: <https://www.mathworks.com/help/nnet/ug/neural-network-object-properties.html#bss4hk6-48> (accessed on 12 October 2017).
41. MacKay, D.J.C. Bayesian Interpolation. *Neural Comput.* **1992**, *4*, 415–447, doi:10.1162/neco.1992.4.3.415.
42. Foresee, F.D.; Hagan, M.T. Gauss-Newton approximation to bayesian learning. In Proceedings of the International Conference on Neural Networks, Houston, TX, USA, 9–12 June 1997; pp. 1930–1935.
43. Möller, M.F. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Netw.* **1993**, *6*, 525–533, doi:10.1016/S0893-6080(05)80056-5.
44. Lin, T.; Horne, B.G.; Tino, P.; Giles, C.L. Learning long-term dependencies in NARX recurrent neural networks. *IEEE Trans. Neural Netw.* **1996**, *7*, 1329–1338, doi:10.1109/72.548162.

