Journal of Renewable Energy and Smart Systems Vol.1, No.1, 2024: 83-91

Renewable Energy and Smart Systems

journal homepage: https://sanad.iau.ir/journal/res

Time Series Forecasting with Artificial Neural Network Trained by Imperialist Competitive Algorithm: A Case Study of Renewable Energy Production and Consumption

Mohammad Amirkhan [1,2,*], Salman Amirkhan[2,3], Mohammad Reza Aloustani[1]

[1]*Department of Industrial Engineering, Aliabad Katoul Branch, Islamic Azad University, Aliabad Katoul, Iran*

[2]*Energy Research Center, Aliabad Katoul Branch, Islamic Azad University, Aliabad Katoul, Iran*

[3]*Department of Electrical Engineering, Aliabad Katoul Branch, Islamic Azad University, Aliabad Katoul, Iran*

Article info

Keywords:
Forecasting
Time Series
Artificial Neural Network
Imperialist Competitive Algorithm
Renewable Energy Production and Consumption.

Article history:
Received: 1 August 2024
Accepted: 15 September 2024

Abstract

The use of renewable energy sources is very valuable as an important aspect of the sustainability and progress of societies. In recent years, the importance of developing the use of renewable energy and replacing fossil energy with this type of energy is not hidden from any country, and therefore, the need for powerful models to forecast the amount of production and consumption of this energy is very necessary. The application of evolutionary neural network as a powerful forecasting technique has been gaining more and more attention in recent researches. In this paper, an efficient method based on the artificial neural network (ANN) has been presented to forecast the renewable energy production and consumption. To enhance the efficiency of the network, an evolutionary algorithm called imperialist competitive algorithm (ICA) has been applied to optimize the network weights. The performance of the hybrid ANN-ICA method is compared with ANN model on real data of the aforementioned case study and the results demonstrate the effectiveness of the ANN-ICA method.

1. Introduction

The requirement to pay more attention to the aspects of sustainability as well as progress has led societies to make the use of renewable energy sources one of their

* Corresponding author.
E-mail address: m.amirkhan.ie@gmail.com; mo.amirkhan@iau.ac.ir.

strategic planning priorities. The use of renewable energy leads to a lower amount of greenhouse gas production and air pollution, which helps to improve the quality of breathing air and reduce harmful climate changes. Substitution of renewable energy instead of fossil energy is an effective strategy to reduce carbon dioxide emissions ([A1], [A2]). The amount of renewable energy production and consumption from renewable sources can depend on various factors such as the level of progress of a country in energy industries, the geographical location of a country, the amount of consumption of citizens, the amount of fossil fuel reserves, etc. Considering the increasing importance of renewable energies in human societies, forecasting their production and consumption has become one of the basic challenges of many countries.

Time series forecasting involves gathering and analyzing historical data of a particular variable to construct a model that captures its underlying patterns, allowing for predictions of future values in the series. It predicts the behavior of a given phenomenon based solely on the past values of the same event. Forecasting problems arise in various disciplines and fields such as agriculture, energy, finance, economic, management, production, transportation or sales.

Over the past decades, numerous efforts have been dedicated to developing time series forecasting models. Two well-known and widely used techniques to predict are statistical and sot computing. The most popular statistical techniques are exponential smoothing, generalized autoregressive conditional heteroskedasticity (GARCH) volatility, and Box-Jenkins autoregressive integrated moving average (ARIMA) model among which the latter is the most used. The artificial neural network (ANN) as a data-driven learning machine and sot computing method is the most accurate and widely used forecasting model in many different areas.

The comparative studies of ANN and Box-Jenkins ARIMA modeling in time series forecasting have been conducted for specific applications. Adebiyi et al. [A3] compared the conventional ARIMA and ANN models with published stock data obtained from New York Stock Exchange. The obtained empirical results demonstrated the superiority of the ANN model over the ARIMA model. Salih et al. [A4] utilized some approaches for forecasting short-term renewable energy consumption and generating. Ali and Mohammad [A5] used a hybrid approach based on linear and non-linear models to predict the daily price of crude oil. Their linear model utilized the Box-Jenkins method, while their nonlinear model incorporated ANN and support vector machine (SVM) techniques. Zhang and Zhou [A6] presented a combined model known as ARIMA-SVR-POT, which integrates ARIMA, support vector regression (SVR), and peak over threshold (POT) techniques. This model is used to forecast the value at risk for crude oil futures. Menéndez-García et al. [A7] examined forecasting the price of platinum and 12 other commodities using both time series and machine learning models. This study made significant contributions to econometrics by showcasing the effectiveness of advanced modeling techniques in predicting the prices of precious metals, thereby offering valuable insights to the field. The research focused on predicting the spot prices of platinum traded on the New York Commodity Exchange. It employed a range of time series machine learning models, including Multivariate Adaptive Regression Splines (MARS), SVM, and a multilayer perceptron (MLP), in addition to traditional methods like ARIMA and VARMA.

The interest in combinations of evolutionary algorithms (EAs) and ANNs has rapidly grown in recent years. Different approaches for combining EAs and ANNs have been introduced and tested by researchers over the past years. Three main types of these approaches have been reported hitherto in the literature as follows [A8]:

- *A supportive combination* involves using EAs and ANNs sequentially, with one acting as the primary problem solver and the other as the secondary.
- *A collaborative combination* involves EAs and ANNs working together simultaneously to solve the problem.
- *Amalgamated combination* in which the EA search mechanism is represented in an NN paradigm.

The purpose of evolutionary neural network is the design and training of neural networks by evolutionary algorithms. Because of importance of the neural network structure and parameters, some evolutionary algorithms such as Genetic Algorithm (GA), Simulated Annealing (SA) and Particle Swarm Optimization (PSO) have been used in various cases such as weight training, design of network topology (including the number of input and output nodes, the number of hidden layers, the number of hidden nodes and the node transfer functions) and learning rules. EAs have the potential to perform a global search and avoid local minima. Additionally, they can address learning problems with sparse feedback, enabling the training of ANNs with non-differentiable neurons [A8].

Learning in ANN has mostly been formulated as a weight training process and purpose of it is finding an optimal set of connection weights according to specified performance measures [A8]. The success of ANNs for each problem mostly depends on the quality and the adequacy of the training algorithm [A9]. ANNs have traditionally been trained with algorithms based on gradient, in particular the back-propagation (BP) algorithm and its derivatives [A10].

The gradient-based training algorithms frequently suffer from trapping in local optimum and it usually dependent on settings of initial weights [A11]. Also, the Levenberg–Marquardt (LM) algorithm based on the extension of Gauss–Newton optimization technique is powerful technique to train the network, but it requires more memory and high computational complexity. To overcome to these challenges, global search approaches like EAs which are less sensitive to local minima can be used effectively in the training process. In addition, EAs can handle large, complex, multimodal and non-differentiable functions. However, some EAs, like genetic algorithm (GA), mostly suffer from slow convergence compared with the fastest version of local algorithms, e.g., LM algorithm and have the difficulty of pre-maturity [A12]. As a result, finding a high efficiency algorithm to training ANN is one of the most important issues in the ANN application.

Due to the important and undeniable role of energies in the world's economy, renewable energy production and consumption are the key factors affecting economic decisions of countries. This paper proposes an approach to develop the ANN model for predicting the renewable energy production and consumption values. For this purpose, an evolutionary algorithm called imperialist competitive algorithm (ICA) is adopted to optimize the initial weights of ANN. The model is tested against ANNs trained by the LM algorithm using the real data extracted of renewable energy production and consumption.

The remainder of this paper is organized as follows. Section 2 describes the applications of ANNs as forecasting tools. The proposed algorithm for training ANN is presented in Section 3, so that initially, a brief review of ICA provided and then, the structure of MLP-NN to time series forecasting explained. Simulation results on the monthly data of renewable energy production and consumption are reported in Section 4 and finally, Section 5 ends the paper with conclusions and future research directions.

2. Applications of ANNs as forecasting tools

ANN is essentially a powerful nonlinear modeling approach used to approximate any function. In addition, it is a beneficial tool due to its learning and forecasting capabilities. The idea of using ANNs for forecasting first adopted by Hu [A13], who used the Widrow's adaptive linear network to weather forecasting. There are several methods to apply neural network for forecasting. Most studies used the multi-layer feed-forward neural networks (MLFF-NN) and the recurrent networks. For time series forecasting problems, the model has the general form (1):

$$X_t = f(X_{t-1}, X_{t-2}, \dots, X_{t-k}) \tag{1}$$

Where $X_{t-1}, X_{t-2}, \dots$ and X_{t-k} are k independent variables, X_t is a dependent one. ANN can be trained to predict one or more future values of the dependent variable. If ANN is properly trained, forecasting performance will be satisfactory. MLF-NN is a class of neural network comprised of an input layer where external information is received, one or more hidden layers and an output layer where the problem solution is obtained. This kind of networks are applied in various fields, especially in forecasting, due to their inherent ability for arbitrary input-output mapping. An example of a simple two-layer fully-connected feed-forward neural network structure shown in Figs. 1. The first applications of the MLF-NN in forecasting purposes is performed by Lapedes and Farber [A14]. In order to use neural networks for forecasting, the input data must be clustered. This paper applied the cluster-based training procedure proposed by Khalili-Damghani and Sadi-Nezhad [A15].

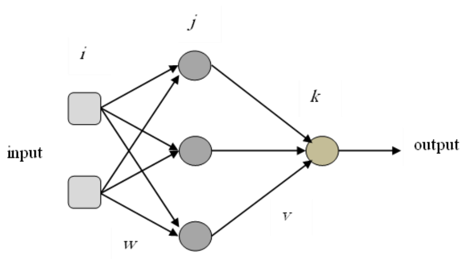

Fig. 1: A simple two-layer fully-connected feed-forward neural network [A8]

3. Updating the ANN weights using ICA

To perform any task, the network must be trained. The training process is one which determines the arc weights of ANN. The knowledge learned by the network is stored in arcs and nodes in a form of arc weights and node biases [A16]. Training of MLFF-NN is supervised and the target value of the network for each input pattern is always available.

Essentially, the network training is an unconstrained nonlinear minimization problem applied to find the weights according to a specified performance measure such as the sum of squared errors (SSE) or mean squared errors (MSE). To date, the different EAs have been used for training ANNs. The general procedure of training the ANN weights by using EAs is shown in Figs. 2. In this paper, ICA is utilized to train the weights of ANN.

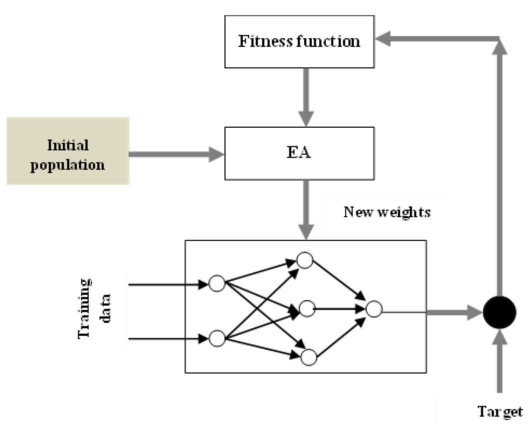

Fig. 2: Weight training of ANN by ICA [A8]

3.1 Imperialist competitive algorithm

ICA which is an evolutionary algorithm inspired from the socio-political evolution was developed by Atashpaz-Gargari and Lucas [A17]. The initial population in this algorithm called countries, that these countries themselves are divided into two categories, colonies and imperialists, which together form empires. Primarily, countries with the best fitness chosen as the imperialists and others are colonies that each one belongs to an empire. The more powerful imperialists have more colonies in this algorithm. In ICA, imperialists compete to acquire more colonies, with colonies continually moving towards the more powerful imperialists (as shown in Figs. 3). This process called assimilation. During this process, the more powerful imperialist countries are progressed and weak ones are collapsed and finally, one imperialist will just remain. For more details about ICA, readers can refer to [A17].

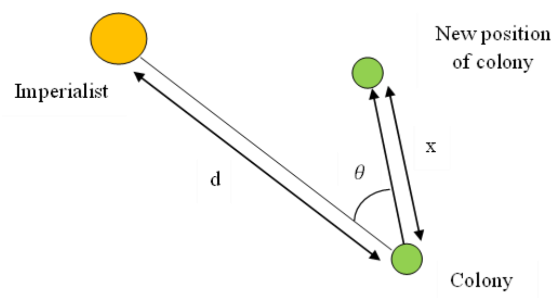

Fig. 3: Moving the colonies toward their imperialist

3.2 MLP-NN structure

It should be noted that the topology of ANN is pre-defined and fixed during the evolution, whereas optimal weights of the ANN must be formulated as a global search problem. Constructing a neural network forecaster for a specific forecasting problem involves making several crucial decisions, each of which is detailed in this section.

a) The number of hidden layers and nodes

Kumar [A18] concluded that general problems typically require only one hidden layer, while more complex problems benefit from adopting two hidden layers. Most researchers use only one hidden layer for forecasting task (for example, see [A19] and [A20]); They believed that ANNs with a single hidden layer can approximate any complex nonlinear function to any level of desired precision. Also, some researchers suggested using two hidden layers for certain specific problems. The most common approaches for determining the number of hidden nodes in the literature involve conducting experiments or using a trial-and-error method. Tang and Fishwick [A20] concluded that while the number of hidden nodes does influence forecast performance, the effect is relatively minor. Generally, the try-error approach is used to find the optimal number of nodes in the hidden layers. Che [A21] suggested that it is better to adopt the optimal number between half the total of the number of nodes in the input and out-put layers, and twice the number of nodes in the input layers. Zhang et al. [A16] believed that if the number of hidden nodes be equals to the number of input nodes, the results of forecasting is better. Hecht-Nielsen [A22] proposed that the number of hidden nodes to be considered ‘’2n+1”, which n is the number of input nodes.

b) *The number of input nodes*

Like the case of hidden nodes, there is no systematic and efficient ways to determine the number of input nodes. However, identifying the appropriate number of input nodes is generally easier than determining the number of hidden nodes. In the time series forecasting task, the number of input nodes usually is equals to the number of lagged observations used to discover the underlying pattern in a time series [A16].

c) *The number of output nodes*

The number of output nodes usually is considered equal to the forecasting horizon ([A16], [A23]). There are two types of forecasting consist of one-step-ahead and multi-step-ahead forecasting and also, two methods of making multi-step forecasts include the iterative and direct ways are reported in the literature [A16].

d) *The interconnection of the nodes*

Basically, the connections between nodes of various layers in a neural network specify the behavior of the network. For most time series forecasting problems, the network is considered as fully connected (for example, see [A17], [A19]).

e) *Activation function*

The activation or transfer function specifies the relationship between inputs and outputs of a node and a network [A16]. There are many activation functions, but in practice, a few of them have high efficiency. The most popular activation functions are sigmoid (logistic), hyperbolic tangent (tanh), sine or cosine and linear.

f) *Training sample and test sample*

To construct ANN, training and test sets are needed. Hence, the data sets must be divided into these two classes. Practically, training and test sets is used for ANN model development and evaluating the forecasting ability of the model, respectively. To avoid the overfitting problem and validate robustness of network simulation, sometimes a third one called the validation set is adopted. In most cases, both validation and testing sets are considered as one test set. It is notable that the division rate of original data sets into two (or three) classes is one of the parameters of ANN [A16].

g) *Training algorithm*

The neural network training is an unconstrained nonlinear minimization problem in which arc weight values of a network are iteratively modified to improve Performance measures of network [A16]. The most widely used training algorithm in ANN is back-propagation. This algorithm is a gradient steepest descent method which is very sensitive to the choice of the learning rate, slow to train and possibly stuck at a local minimum [A24].

h) *Data normalization*

Some activation functions need data to be scaled into a special interval. To do so, efficient preprocessing techniques must be adopted to scale data to the needed interval. The most popularly approaches for input normalization are as follows:

1- Along channel normalization: for this case, the linear transformation to [0,1] is used as follows [A25]:

$$x_n = \frac{x_o - x_{min}}{x_{max} - x_{min}} \quad (2)$$

2- Across channel normalization: for this case, the linear transformation to [a,b] is used as equation (3) [A26]:

$$x_n = (b-a)\frac{x_o - x_{min}}{x_{max} - x_{min}} + a \quad (3)$$

3- Mixed channel normalization: for this case, the statistical normalization is used as equation (4) [A27]:

$$x_n = \frac{x_o - \bar{x}}{s} \quad (4)$$

4- External normalization: for this case, the simple normalization is used as equation (5) [A28]:

$$x_n = \frac{x_o}{x_{max}} \quad (5)$$

where x_n and x_o represent the normalized and original data, respectively. Also, $\bar{x}$, x_{min}, x_{max} and s are the average, minimum, maximum, and standard deviation, respectively.

i) Performance measures

An appropriate measure of accuracy is often based on the forecasting error obtained from the difference between the actual and the predicted value. The most important Performance measures in ANN [A16] are listed in Table 1:

Table 1: Performance measures in ANN

Performance measures	**Abbreviation**	**Mathematical relation**
Sum of Squared Error	SSE	$\sum (e_t)^2$
Mean Squared Error	MSE	$\frac{\sum (e_t)^2}{N}$
Root Mean Squared Error	RMSE	$\sqrt{MSE}$
Mean Absolute Deviation	MAD	$\frac{\sum \lvert e_t \rvert}{N}$
Mean Absolute Percentage Error	MAPE	$\frac{1}{N}\sum \left\lvert \frac{e_t}{y_t} \right\rvert (100)$

where e_t, y_t and N are the individual forecast error, the actual value and the number of error terms, respectively. However, to enhance the accuracy and due to the limitations associated with each individual measure, most researchers use multiple performance measures in a particular forecasting problem.

4. Simulation results

To evaluate the performance of the proposed approach, we used the real data concerning renewable energy production (including wood energy, biofuels, and total biomass energy) and consumption (including hydroelectric power, geothermal energy, solar energy, wind energy, wood energy, waste energy, biofuels, and total biomass energy) of countries from January 2000 to April 2024. The data are monthly as well as available at the web address " *https://www.eia.gov/renewable/* ". Figs. 4 and Figs. 5 show the trends of these data.

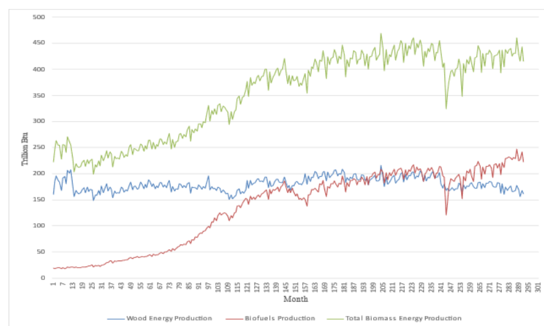

Fig. 4: Renewable energy production from January 2000 to April 2024

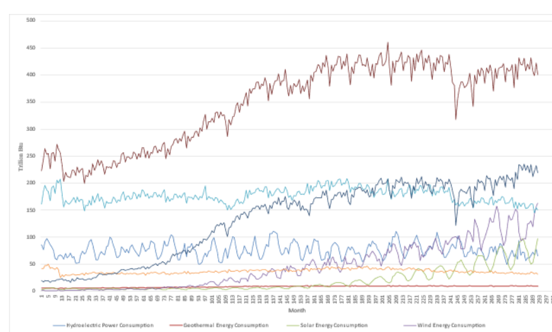

Fig. 5: Renewable energy consumption from January 2000 to April 2024

Steps of the research to implement the proposed approach are shown in Figs. 6.

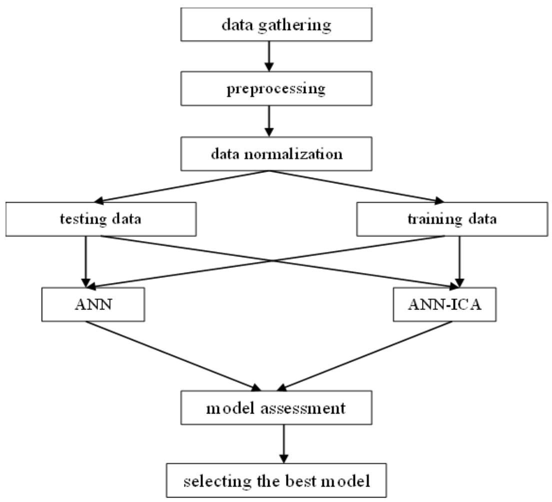

Fig. 6: Conceptual framework of the current research

To construct an ANN predictor, a two-layer fully-connected feed-forward neural network is considered. Various numbers of input nodes (or lag) have been tried to determine the best structure of the mentioned network for the time series forecasting. It should be noted that increasing the number of hidden nodes may benefit the system dynamic, but in return the computational complexity of the problem increases [A30]. The parameters of our MLF-NN have adjusted as Table 2.

Tables 2: Network architecture of current paper

Parameters	Settings values
Data	monthly
Delays (lags)	[1,2,12,24,36]
Input nodes	5
Hidden layer: nodes	1:11
Output nodes	1
Transfer function hidden: output	Tansig: purelin
Training algorithm	LM, ICA
Training:	80:20:00
Data normalization	Across channel normalization with interval [-1,1]
Performance measure	$\frac{RMSE(ts)}{2} * 100$

To reduce random variation of the propose algorithm itself, each experiment is run 50 times and the mean is presented. The results of ANN and ANN-ICA models are compared in this section. To compare the results, the following performance measure (6) is employed:

$$\text{Percent of the forecasting error} = \frac{RMSE(ts)}{2} * 100 \qquad (6)$$

The simulation results are given in Table 3 and Figs. 7. The results show that the ANN-ICA model is superior to the ANN model for all sub-sections, exception two sub-sections (wood energy and total biomass energy). Also, the mean value of measures of the ANN-ICA model is better than the ANN one.

Tables 3: The simulation results for two presented models

Section	Sub-section	Performance measure	
		ANN model	**ANN-ICA model**
Production	wood energy	18.4878	11.2182
	biofuels	12.0689	7.8361
	total biomass energy	18.6459	13.8117
Consumption	hydroelectric power	17.0629	11.3169
	geothermal energy	9.1844	5.527
	solar energy	16.78	9.2196
	wind energy	9.5359	6.1043
	wood energy	8.9429	11.1017
	waste energy	16.9279	13.6133
	biofuels	19.1364	8.0015
	total biomass energy	10.7122	11.8913
Mean		14.3169	9.9674

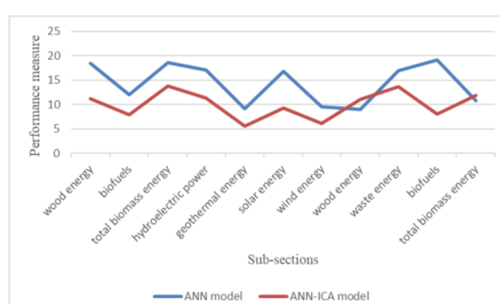

Fig. 7: Simulation results of ANN model and ANN-ICA model

5. Conclusion and future research directions

The main purpose of this paper is to utilize artificial intelligence approaches for time series forecasting. Hence, a hybrid MLP-NN model in which the ICA is used to train the connected weights is presented. The model is implemented on the monthly real data of renewable energy production (including wood energy, biofuels, and total biomass energy) and consumption (including hydroelectric power, geothermal energy, solar energy, wind energy, wood energy, waste energy, biofuels, and total biomass energy) of countries from January 2000 to April 2024. Then, a performance measure to evaluate the proposed hybrid model is defined. Simulation results show good performance of the proposed model in terms of quality of forecasting and convergence rate. From the comparison results of the proposed approaches based on the aforementioned measure, it can be concluded that the hybrid ANN-ICA model is superior to the ANN model in forecasting the renewable energy production and consumption.

For future research, the proposed model can be expanded to use evolutionary algorithms to simultaneously improve the topology of the artificial neural network and its weights so that time series prediction can be performed with higher quality.

References

[A1] Bhattacharya, M., S. R., Paramati, I., Ozturk, Bhattacharya, S., The effect of renewable energy consumption on economic growth: Evidence from top 38 countries, Applied energy, 162 (2016) 733-741.
[A2] Wang, Q., F. Zhang, and R. Li, Revisiting the environmental kuznets curve hypothesis in 208 counties: The roles of trade openness, human capital, renewable energy and natural resource rent, Environmental Research, 216(3) (2023) 114637.
[A3] Adebiyi, A.A., A.O. Adewumi, and C.K. Ayo, Comparison of ARIMA and Artificial Neural Networks Models for Stock Price Prediction, Journal of Applied Mathematics, 2014 (2014) 1-7.
[A4] Abu-Salih, B., et al., Short-term renewable energy consumption and generation forecasting: A case study of Western Australia, Heliyon, 8(3) (2022) e09152.
[A5] Ali, N.S.M. and F.A. Mohammed, The use of ARIMA, ANN and SVR models in time series hybridization with practical application, International Journal of Nonlinear Analysis and Applications, 14(3) (2023) 87-102.
[A6] Zhang, C. and X. Zhou, Forecasting value-at-risk of crude oil futures using a hybrid ARIMA-SVR-POT model, Heliyon, 10(1) (2024) e23358.
[A7] Menéndez-García, L.A., et al., Time series analysis for COMEX platinum spot price forecasting using SVM, MARS, MLP, VARMA and ARIMA models: A case study, Resources Policy, 95 (2024) 105148.
[A8] Siddique, N. and H. Adeli, Computational intelligence: synergies of fuzzy logic, neural networks and evolutionary computing, 2013: John Wiley & Sons.
[A9] Peyghami, M.R. and R. Khanduzi, Predictability and forecasting automotive price based on a hybrid train algorithm of MLP neural network, Neural Computing and Applications, 21(1) (2012) 125-132.
[A10] Haykin, S., Neural networks: a comprehensive foundation, 1994. Mc Millan, New Jersey, 2010.
[A11] Wang, R.L., Z. Tang, and Q.P. Cao, An efficient approximation algorithm for finding a maximum clique using Hopfield network learning, Neural computation, 15(7) (2003) 1605-1619.
[A12] Tayefeh Mahmoudi, M., Forouzideh, N., Lucas, C., Taghiyareh, F., Artificial neural network weights optimization based on imperialist competitive algorithm. in 7th International conference on computer science and information technologies (CSIT'09), Yerevan. 2009.
[A13] Hu, M.J.-C., Application of the adaline system to weather forecasting, 1964, Department of Electrical Engineering, Stanford University.
[A14] Lapedes, A. and R. Farber, Nonlinear signal processing using neural networks. 1987.
[A15] Khalili-Damghani, K. and S. Sadi-Nezhad, Application of multi-layer recurrent neural network in chaotic time series prediction: a real case study of crude oil distillation capacity, International Journal of Artificial Intelligence and Soft Computing, 2(4) (2011) 367-380.
[A16] Zhang, G., B. Eddy Patuwo, and M. Y Hu, Forecasting with artificial neural networks: The state of the art, International journal of forecasting, 14(1) (1998) 35-62.
[A17] Atashpaz-Gargari, E. and C. Lucas. Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. in Evolutionary Computation, 2007. CEC 2007. IEEE Congress on. 2007. IEEE.
[A18] Kumar, S., Neural networks: a classroom approach. 2004: Tata McGraw-Hill Education.
[A19] Peyghami, M.R. and R. Khanduzi, A hybrid model based on neural network and hybrid genetic algorithm for automotive price forecasting, International Journal of Applied Mathematics and Computation, 3(3) (2011) 158-168.

[A20] Tang, Z. and P.A. Fishwick, Feedforward neural nets as models for time series forecasting, ORSA Journal on Computing, 5(4) (1993) 374-385.

[A21] Che, Z., PSO-based back-propagation artificial neural network for product and mold cost estimation of plastic injection molding, Computers & Industrial Engineering, 58(4) (2010) 625-637.

[A22] Hecht-Nielsen, R. Kolmogorov's mapping neural network existence theorem. in Proceedings of the international conference on Neural Networks. 1987. New York: IEEE Press.

[A23] Zhang, X., Time series analysis and prediction by neural networks, Optimization Methods and Software, 4(2) (1994) 151-170.

[A24] Lee, C.-M. and C.-N. Ko, Time series prediction using RBF neural networks with a nonlinear time-varying evolution PSO algorithm, Neurocomputing, 73(1) (2009) 449-460.

[A25] Azoff, E.M., Neural network time series forecasting of financial markets. 1994: John Wiley & Sons, Inc.

[A26] Lapedes, A.S. and R.M. Farber. How neural nets work. Conference of Neural Information Processing Systems, Denver, Colorado, USA, 1987.

[A27] Srinivasan, D., A. Liew, and C. Chang, A neural network short-term load forecaster, Electric Power Systems Research, 28(3) (1994) 227-234.

[A28] Weigend, A.S., B.A. Huberman, and D.E. Rumelhart. Predicting sunspots and exchange rates with connectionist networks. in santa fe institute studies in the sciences of complexity-proceedings volume. 1992. addison-wesley publishing co.

[A29] Lachtermacher, G. and J.D. Fuller, Back propagation in time- series forecasting, Journal of Forecasting, 14(4) (1995) 381-393.

[A30] Cai, X., et al., Time series prediction with recurrent neural networks trained by a hybrid PSO–EA algorithm, Neurocomputing, 70(13) (2007) 2342-2353.