Journal of Engg. Research Vol. 8 No. (3) September 2020 pp. 135-152 https://doi.org/10.36909/jer.v8i3.8062

خوارزمية التنبؤ بالسلاسل الزمنية المبنية على المنطق الضبابي باستخدام عناقيد الجوار الأقرب

مصباح إيروم، محمد شهزاد وإجاز محمد
قسم الهندسة الكهربائية، جامعة هيتيك، تاكسيلا، باكستان

الخلاصة

يعرض هذا البحث خوارزمية التنبؤ بالسلاسل الزمنية المبنية على المنطق الضبابي باستخدام عناقيد الجوار الأقرب. تم تطوير الخوارزمية المُقترحة باستخدام مخطط عناقيد الجوار الأقرب والنظام المُضبب الأمثل. تم اختبار الخوارزمية المُقدمة من خلال التنبؤ بالسلسلة الزمنية Mackey-Glass، والسلسلة الزمنية التي تم الحصول عليها من معادلة الخريطة اللوجيستية وسلسلة بيانات مصفاة النفط. على حد علمنا، حاول عدد قليل جداً من الدراسات التنبؤ بالخريطة اللوجيستية. تظهر نتائج المحاكاة أن الخوارزمية المُقترحة قادرة على التنبؤ بالسلاسل الزمنية الخطية وغير الخطية وحتى الفوضوية بدقة معقولة. فهي تعمل بشكل جيد، حتى في حالة عدم وجود النموذج الرياضي للسلسلة الزمنية.

Fuzzy Logic Based Time Series Prediction Algorithm Using Nearest Neighborhood Clustering

Misbah Irum*, Muhammad Shahzad* and Ejaz Muhammad**

**Department of Electrical Engineering, HITEC University, Taxila, Pakistan.*
***Corresponding Author: muhammad.shahzad@hitecuni.edu.pk*

Submitted: 19/03/2019
Revised: 11/10/2019
Accepted: 20/10/2019

ABSTRACT

This paper presents a fuzzy logic based prediction algorithm using nearest neighborhood clustering for the prediction of time series. The proposed algorithm has been developed by using nearest neighborhood clustering scheme and optimal fuzzy system. The presented algorithm is tested by the prediction of Mackey-Glass time series, time series obtained from Logistic Map equation, and an oil refinery data series. To the best of our knowledge, very few studies have attempted to predict Logistic Map. The simulation results show that the proposed algorithm is able to predict linear, nonlinear, and even chaotic time series to a reasonable accuracy. It performs well, even when the mathematical model of time series does not exist.

Keywords: chaos; Logistic Map equation; Mackey-Glass time series; nearest neighborhood clustering; optimal fuzzy system; prediction.

INTRODUCTION

In this research work, prediction refers to the process of estimating the next value in a time series. A time series can be defined as a sequence of data points taken at successive and equidistant time instants (Truong & Anh, 2012). A time series can be linear, nonlinear, and even chaotic. In case of nonlinear and especially chaotic time series, prediction is a very challenging task. In the past, several prediction models have been developed such as Autoregressive Moving Average (ARMA) and Artificial Neural Network (ANN) models. This research work uses a fuzzy logic based prediction algorithm that can predict linear, nonlinear, and even chaotic data series reasonably well. This algorithm uses nearest neighborhood clustering scheme and optimal fuzzy system. There are two phases of proposed prediction algorithm, training and prediction, respectively. In the training phase, data pairs from the time series are used to train the system and in the prediction phase, next values are predicted.

Prediction has become an active area of research these days as the concept of prediction finds applications in various fields such as meteorology, sport, finance, safety engineering, and even medicine. In recent years, numerous time series prediction techniques have been developed. A brief overview of literature search is given below.

LITERATURE REVIEW

Deng *et al.* (2015) proposed a prediction model for water quality time series prediction that uses heuristic Gaussian cloud transformation algorithm. The results were compared with those of other prediction models. Saleh *et al.* (2016) proposed a neurofuzzy model-based prediction technique for wind power generation. They used modified fuzzy C-Means (FCM) clustering to determine the optimal number of fuzzy rules. Wang *et al.* (2015) used modified fuzzy

C-means and information granules for long-term prediction of time series. Junhu *et al.* (2013) developed a fuzzy granular support vector machine to predict time series over the large scale. Goyal *et al.* (2017) used data driven fuzzy inference system to develop software fault prediction model. In their research work, Vesely *et al.* (2016) compared linear regression model to a fuzzy model in forecasting the recycling behavior. The comparison suggested that the fuzzy model performs better than linear regression model. Egrioglu *et al.* (2013) introduced a hybrid approach using fuzzy C-means clustering and artificial neural networks. They used the presented method for the prediction of enrollment data series. Abdulshahed *et al.* (2015) used adaptive neurofuzzy inference system (ANFIS) to design thermal error models. The designed models were used to compensate thermal error on CNC machine tools. The results showed that fuzzy C-means clustering model produced better results than the grid model. Goyal *et al.* (2014) designed various models of pan evaporation for subtropical climates. Results indicated that Least Squares Support Vector Regression (LS-SVR) and Fuzzy Logic based model produced better results. They also found that performance of machine learning based models was better than empirical methods. Rizwan *et al.* (2014) modeled and estimated global solar energy using a fuzzy logic approach. The proposed method used some meteorological parameters as input. Olatomiwa *et al.* (2015) developed a model using ANFIS to predict global solar radiation. The proposed ANFIS model used some meteorological parameters as input. The results indicated that the proposed model can predict global solar radiation efficiently. Hossain *et al.* (2016) devised a prediction model for estimating the kerf width in laser beam machining using the fuzzy logic. Their prediction model used triangular membership function, Mamdani implication, and centroid defuzzification method.

Das *et al.* (2016) presented a prediction model to predict the winner of Cricket World Cup. Their prediction model used various parameters that can affect the performance of a team, and the final result of the game as well. The proposed model used Analytic Hierarchy Process (AHP) to reach the conclusion. Rasoolzadeh & Tavazoei (2012) devised a methodology to search the chaotic area in the parameter space of nonsalient permanent-magnet synchronous machines. They concluded that possibility of occurrence of chaos in these machines increases with increase in input torque, size of magnet, and number of poles. Pulido *et al.* (2014) presented a hybrid method to forecast complex time series. To optimize ensemble neural network, they used Particle Swarm Optimization (PSO) technique. They tested the proposed scheme by predicting stock exchange of Mexico. Chen (2013) used PSO technique to obtain suitable parameter settings for subtractive clustering (SC). The proposed method integrated the ANFIS model to design a model for predicting business failures. Samples of companies listed on the Taiwan Stock Exchange Corporation (TSEC) were used for testing. The results indicated that the proposed model outperformed other models. Lu *et al.* (2014) proposed a methodology for time series modeling using the concept of information granules and granular computing. Their methodology first divides the time series into time window sections, then constructs fuzzy granules for each section of the time series, and finally builds the model at granular level. The prediction results showed that the proposed method is effective. Heydari et al. (2016) devised a technique to train second-order TSK fuzzy system using ANFIS and, hence, provided a new method to train and model chaotic time series. In their research work, Lu *et al.* (2016) developed an extended version of Sugeno-type fuzzy model with an additional quality of optimized allocation of information granules. They applied the presented model for the prediction of various time series.

Li *et al.* (2015) combined traditional fuzzy controller outputs with those of grey prediction fuzzy controller (GPFC) to develop a fuzzy prediction control law. This law was designed to control road tunnel ventilation. Kaya *et al.* (2016) presented a fuzzy logic based prediction system for the optimization of digital holographic setup and compared the optimization results of experimental system with those of fuzzy logic prediction system. Vafaie *et al.* (2014) presented a genetic-fuzzy system for prediction of heart diseases. The proposed system used fuzzy classifier for isolating ECG signals. Norouzi *et al.* (2016) designed a prediction model for kidney failure progression in chronic kidney disease. The proposed model is ANFIS-based and took a set of physiological parameters as input, and GFR was the predicted output. The experimental results indicated that the presented model can predict future values of GFR efficiently. Choi *et al.* (2016) proposed a model based on cascaded fuzzy neural network to predict the hydrogen gas concentration inside nuclear power plant. Hydrogen concentration above a certain level can cause huge accident in nuclear power plant.

A time series modeling and prediction methodology has been proposed by Lu *et al.* **(2014).** The proposed methodology is based on fuzzy C-means clustering and high-order fuzzy cognitive map (HFCM). The former converts the input time series into granular time series and builds the HFCM model, and the latter performs prediction by generating granular time series prediction model and performing inference in granular space. The method was applied to predict four benchmark time series including Mackey-Glass time series. Zhao & Yu (2015) presented adaptive natural gradient algorithm for multilayer perceptrons (MLPs) and radial basis function (RBF). They used this method for the prediction of Mackey-Glass time series and compared the result with that of LMA. Bodyanskiy & Vynokurova (2013) developed a hybrid adaptive wavelet-neurofuzzy system for prediction, emulation, and identification of chaotic time series and nonstationary processes. Kuremoto *et al.* (2014) presented a neural network model for the prediction of time series. They used Hinton and Salakhutdinov's deep beliefs net (DBN) composed of multilayer of restricted Boltzmann machine (RBM). They applied the short-term prediction of chaotic time series such as Lorenz chaos and logistic map time series by the presented method.

In the proposed prediction algorithm, we first train the system using nearest neighborhood clustering scheme. At the end of training phase, we get some clusters. For prediction, we first search the nearest cluster and then using the data present in this particular cluster, we predict the next value. In other words, by using nearest neighborhood clustering we divide the whole training data set into categories named as clusters and for predicting a value, we use only the data present in the nearest cluster not the whole training data set. Consequently, our algorithm uses less computational time and still obtains a good level of accuracy. To the best of our knowledge, no studies have used nearest neighborhood clustering scheme for building and selecting suitable sets of rules for prediction of time series. After each prediction, a new pair is added to the nearest cluster. Thus, the artificial intelligence of this algorithm will continue to improve with the passage of time.

Simulation results show that proposed technique can work for any data series, be it linear, nonlinear, or even chaotic. The reason is that the proposed algorithm looks at data points with respect to their location in the neighborhood of other data points. This particular aspect of the proposed algorithm forms the basis of clustering. The applications of proposed algorithm have been amply demonstrated to work in Mackey Glass series as well as Logistic Map equation. These two equations have been amply researched to represent linear, nonlinear, and chaotic data sequences and so far, they represent almost all known cases of single dimensional chaos. Our proposed algorithm has worked quite satisfactory in these two referred cases. The third application included in this paper represents a practical example of a series representation of oil production, and the related data is already in chaos (Aliev et al., 2001). The proposed algorithm has again worked quite reasonably well.

The rest of the paper is organized as follows. A brief introduction to Logistic Map and Mackey Glass equations is given in section IV. In section V, we introduce the optimal fuzzy system. Details of the proposed prediction algorithm and prediction results are presented in sections VI and VII, respectively.

LOGISTIC MAP AND MACKEY-GLASS EQUATIONS

Logistic Map Equation

Logistic Map Equation is a nonlinear equation. It was first presented by Pierre Verhulst in 1838. Logistic map equation is used by engineers and mathematicians because of its surprising behavior. Discrete-time version of logistic map equation is given as (Stavroulakis, 2005)

$$x_{n+1} = bx_n(1 - x_n) \quad (1)$$

Here, x_n is the current value of x referred to as initial condition, and x_{n+1} is the value of x at the next time instant, whereas x takes any value in the interval [0,1]. Parameter b is of great importance. For different ranges of b logistic map behaves differently. The behavior of logistic map can be divided into three broad categories (Lakshmanan & Rajaseekar, 2012).

- Logistic Map converges to a constant value for the parameter *b* between 0 and 3.
- For a value of *b* greater than 3, i.e., for $b > 3$, logistic map exhibits oscillatory behavior.
- For a value of *b* from 3.57 to 4, logistic map exhibits chaotic behavior.

We used four modes of Logistic Map for simulation. For mode-1, mode-2, mode-3, and mode-4 of logistic map, the value of parameter *b* was fixed to 2, 3.3, 3.63, and 4, respectively. The value of initial condition was fixed to 0.2.

Mackey-Glass Equation

The Mackey-Glass equation is a differential equation that can generate periodic and chaotic dynamics. The equation is given as (Glass &Mackey, 2010)

$$\frac{dx(t)}{dt} = \beta \frac{x(t-\tau)}{1+\left(x(t-\tau)\right)^n} - \gamma x(t) \tag{2}$$

where β, γ, τ and n are real numbers. The behavior of this equation depends on the values of these parameters. The equation shows chaotic behavior for $\beta = 0.2$, $\gamma = 0.1$, $n = 10$, and $\tau = 17$.

OPTIMAL FUZZY SYSTEM

Optimal fuzzy system is a fuzzy system consisting of singleton fuzzifier, product inference engine, and center average defuzzifier. Singleton fuzzifier is the most commonly used fuzzifier as it involves less mathematical computation. The singleton fuzzifier converts a real-valued input $x^* \epsilon u$ into a fuzzy singleton; that is,

$$\mu_{A'}(X) = \begin{cases} 1, & if\ x = x^* \\ 0, & otherwise \end{cases} \tag{3}$$

Product inference engine uses individual rule based inference with union combination, Mamdani's product implication, and algebraic product for all t-norm operators and max for all s-norm operators. Mathematically, it is given as (Wang, 1999)

$$\mu_{B'}(y) = max_{l=1}^{M}[supX_{\in\mu}(\mu_{A'}(X)\Pi_{i=1}^{n}\mu_{A_i^l}(x_i)\mu_{B^l}(y))] \tag{4}$$

where A' in U is the fuzzy input and B' in V is the fuzzy output of inference engine. There are many types of defuzzifiers such as center of gravity, maximum, and center average. We have used center average defuzzifier; it takes the weighted average of the centers of fuzzy sets, whereas weights are the heights of the corresponding fuzzy sets. If w_l are the heights and y^{-l} are the centers of l fuzzy sets, then the crisp output y^* using center average defuzzifier (Wang, 1999) is given by

$$y^* = \frac{\Sigma_{l=1}^{M} w_l y^{-l}}{\Sigma_{l=1}^{M} w_l} \tag{5}$$

Substituting Equation 3 into Equation 4, we have

$$\mu_{B'}(y) = max_{l=1}^{M}[\Pi_{i=1}^{n}\mu_{A_i^l}(x_i^*)\mu_{B^l}(y)] \tag{6}$$

Crisp output using center average defuzzifier is given as

$$y^* = \frac{\Sigma_{l=1}^{M} y^{-l}(\Pi_{i=1}^{n}\ \mu_{A_i^l}(x_i^*))}{\Sigma_{l=1}^{M}(\Pi_{i=1}^{n}\ \mu_{A_i^l}(x_i^*))} \tag{7}$$

The optimal fuzzy system for $N\ (x,y)$ pairs using Gaussian membership function is given as (Wang, 1999)

$$f(x) = \frac{\Sigma_{l=1}^{N} y_l exp\left(-\frac{|x - x_l|^2}{\sigma^2}\right)}{\Sigma_{l=1}^{N} exp\left(-\frac{|x - x_l|^2}{\sigma^2}\right)} \tag{8}$$

where y_l are the centers of output fuzzy sets and σ is the spread of membership functions. Optimal fuzzy system $f(x)$ can match all $l = 1, 2, 3, \ldots .N$ numbers of (x_l, y_l) pairs. The optimal fuzzy system $f(x)$ (with value of $\sigma = 0.1$) for six input-output pairs (-3,1), (-2,1), (-1,0), (0,2), (1,2), and (2,1) is plotted in Figure 1.

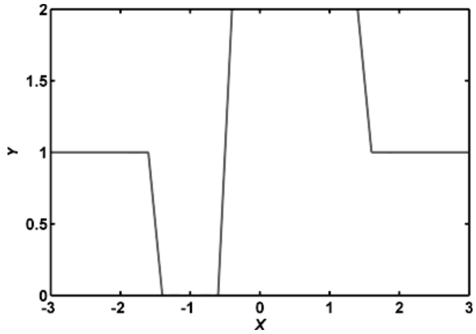

Figure 1. The Optimal Fuzzy System with σ=0.1.

Input-output matching ability of optimal fuzzy system is highly dependent on σ. The effect of choosing a specific value of σ is shown in Figure 2. Larger values of σ produce more smooth results but with less accuracy, whereas smaller values result in more accurate output; hence, the value of σ used for the computation of prediction results is 0.1.

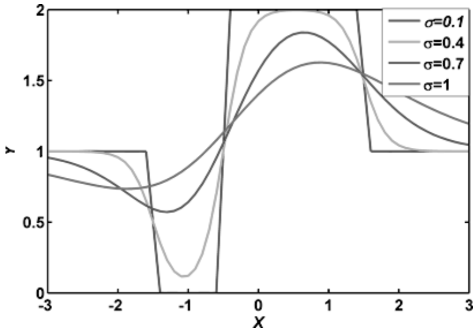

Figure 2. The optimal fuzzy system with σ from 0.1 through 1.

THE PREDICTION ALGORITHM

The proposed prediction algorithm consists of two phases; the first phase is training, and the second is prediction. During training phase, training pairs are placed in appropriate clusters using nearest neighborhood clustering scheme. In prediction phase, we predict the next value in the data series.

Training phase

First, we take the input data series and convert this series into pairs. Suppose we have a series $x_1, x_2, x_3, \ldots$.. The first pair that we can get from this series is (x_1, x_2), second pair (x_2, x_3), and so on. In the first pair, x_2 will be treated as output of x_1 so it can be renamed as y_1 and in the second pair, x_3 will be treated as output of x_2, so we can call it as y_2 and so on.

Now after obtaining pairs, we can use knowledge of x values to divide given data into clusters. As mentioned earlier, division is based on the principle of nearest neighborhood for which a parameter r called cluster radius is important. Cluster radius is the distance on x-axis from cluster center x_c to allowable farthest point x_f. It is given as

$$\boldsymbol{r} = |x_c - x_f| \tag{9}$$

Stepwise details of near neighborhood clustering algorithm are given as follows (Wang, 1999).

1) Select a cluster radius r.

2) The input x_1 of the first input-output pair is set as the center of first cluster.

3) Suppose that we consider the nth pair. Calculate the distances of the input x_n to all cluster centers and let "d" be the smallest distance, that is, the distance from the nearest cluster center.

(a) If the smallest distance d is greater than r, set x_n as a new cluster center.

(b) If the smallest distance d is less than or equal to r, make (x_n, y_n) pair member of the nearest cluster.

Using the above-mentioned procedure, all the pairs are placed in appropriate clusters. In this research work, process of placing pairs in appropriate clusters is called training. After training, we have the knowledge of all cluster centers and member pairs present in each cluster. This knowledge is used in prediction phase. Flow chart of training phase is shown in Figure 3.

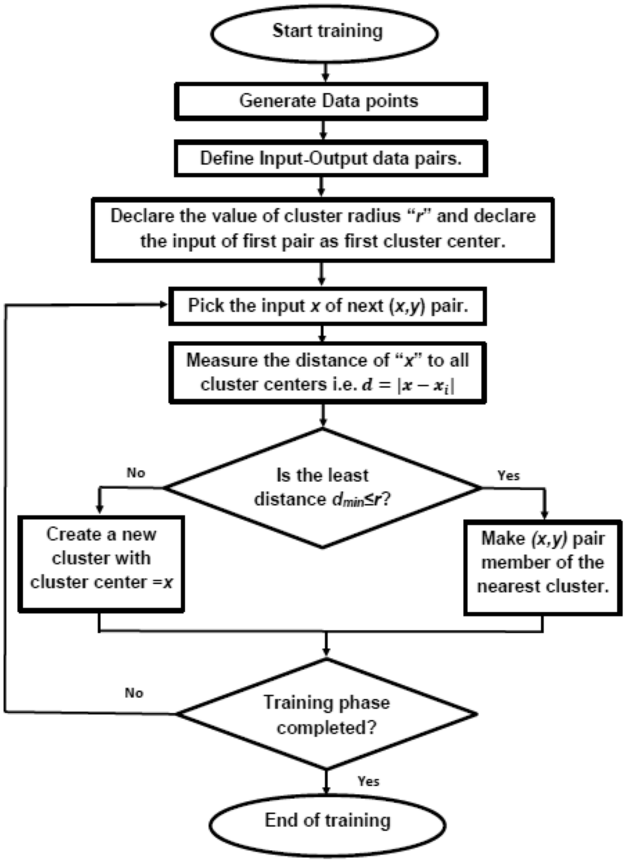

Figure 3. Flow chart of the training phase.

Prediction phase

After training, the system through "p" number of pairs, prediction, is started as shown in Figure 4. At the end of the training phase, suppose we have M number of clusters. Now, the algorithm picks up the input x_{p+1} of the $(p+1)^{th}$ pair (whose output we want to predict). Then it measures the distances of x_{p+1} to all cluster centers. Suppose the smallest distance is $d = |x_{p+1} - x_c^k|$, i.e., the distance from k^{th} cluster center; if this distance is less than or equal to cluster radius r, the nearest cluster to x_{p+1} is the k^{th} cluster. If the distance d is greater than r, the input x_{p+1} is skipped. We shall see that, choosing suitable cluster radius, only negligible number of inputs are skipped. After searching the nearest cluster, the output y_{p+1} is computed using optimal fuzzy system. The predicted output according to the proposed algorithm is given by

$$f(x_{p+1}) = \frac{\Sigma_{l=1}^{N} y_l exp\left(-\frac{|x_{p+1} - x_l|^2}{\sigma^2}\right)}{\Sigma_{l=1}^{N} exp\left(-\frac{|x_{p+1} - x_l|^2}{\sigma^2}\right)} \tag{10}$$

where N is the number of (x, y) pairs present in the nearest cluster (including the center). Another point that should be noted is that N rules are being used in the above computation (one rule per pair). Hence, the output is being statistically calculated, and this statistical calculation is called prediction. In fact, the process of searching the nearest cluster is the process of finding appropriate set of rules for making a prediction. After the calculation of output, the pair (x_{p+1}, y_{p+1}) is placed in the already searched nearest cluster, whereas y_{p+1} is the real value of the output. After this, the algorithm moves towards the next prediction.

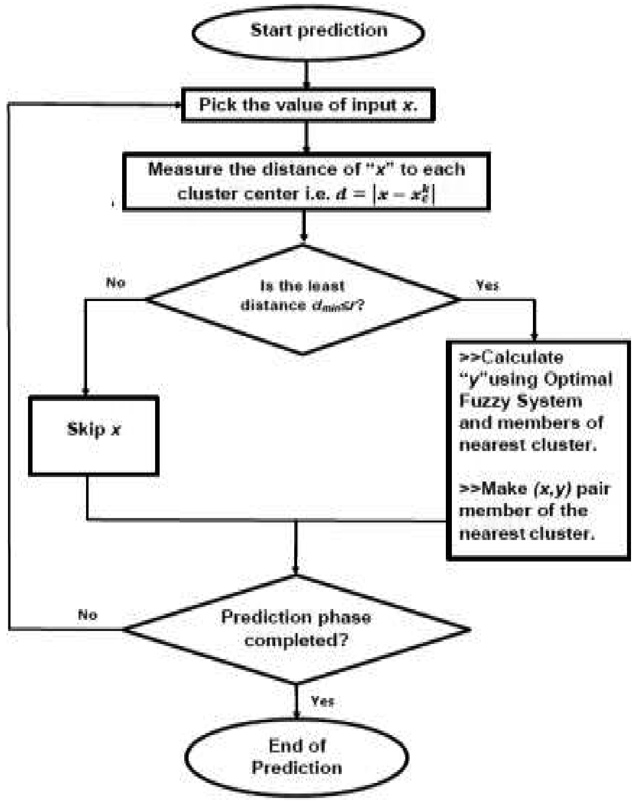

Figure 4. Flow chart of prediction phase.

PREDICTION RESULTS

Using the proposed prediction algorithm, we tried to predict various benchmark time series. These input time series are Logistic map time series, Mackey-Glass time series, and petroleum production data series. The process involves generating data points of input time series, converting these data points into (x, y) pairs, and then using some of these pairs in the learning phase of our algorithm and subsequently predicting the remaining values. It is also pointed out that, in this research work, learning or training phase amounts to making clusters as explained earlier. The algorithm was implemented in MATLAB.

Prediction Results of Different Modes of Logistic Map Time Series

For mode-1, referring to Equation 1, the value $b = \mathbf{2}$ was used to generate 350 data points, for which it converges to a constant value of 0.5. The first 300 out of 350 data points were used as training data. Next 50 data points were predicted, i.e., 301 to 350, while keeping the value of cluster radius equal to 0.01. For this cluster radius, the algorithm created four clusters. No input was skipped by the algorithm during the prediction phase. Figure 5 illustrates the predicted and actual values. It can be observed that the predicted and actual values are nearly the same. Mean and variance of the error for fifty predicted outputs are 4.3×10^{-7} and 3.90×10^{-16}, respectively.

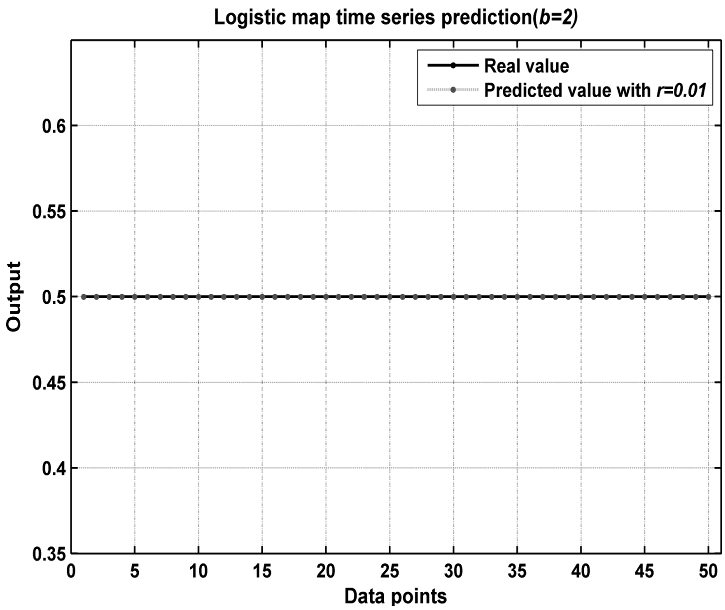

Figure 5. Prediction of Mode-1 with $r = 0.01$.

When the radius was increased to 0.3, the algorithm created only one cluster. Figure 6 illustrates the prediction results of the next fifty points. Mean and variance of prediction error are 6.7×10^{-5} and 9.30×10^{-12}, respectively. No input was skipped during the prediction phase. It may be noted that Figure 5 and Figure 6 are apparently similar. However, each has a different error, but the magnitude of error is so small that it cannot be seen on the scale of the figures.

For mode-2, the parameter b=3.3 was substituted into Equation 1 to generate 350 points. The first 300 out of 350 data points were used for training. Next 50 data points were predicted, while keeping the value of cluster radius equal to 0.1. Mean of prediction error is 8.74×10^{-6} and variance is 1.44×10^{-11}. The algorithm created three clusters. Figure 7 shows the predicted and actual values.

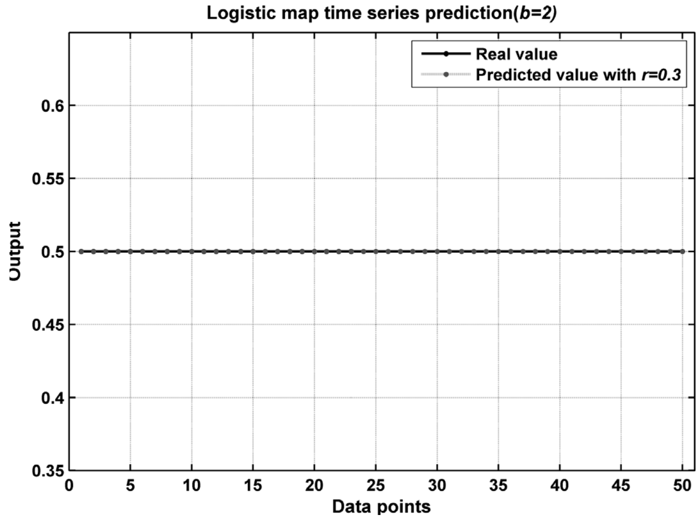

Figure 6. Prediction of Mode-1 with $r = 0.3$.

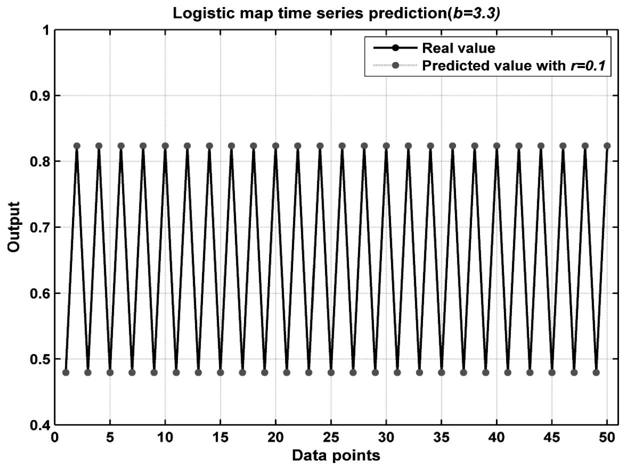

Figure 7. Prediction of mode-2 with $r = 0.1$.

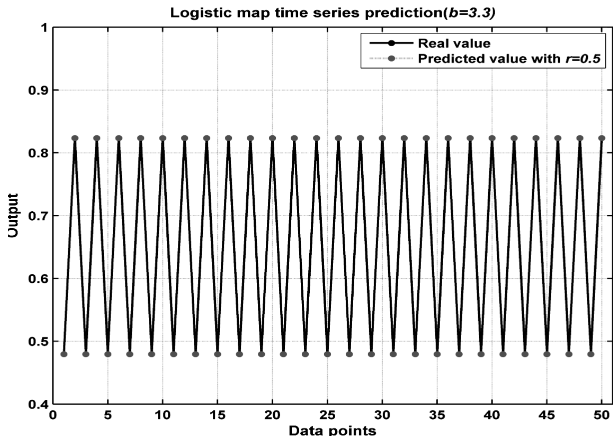

Figure 8. Prediction of mode-2 with $r = 0.5$

Then we increased the cluster radius to 0.5 for which the algorithm created two clusters. Figure 8 shows the fifty predicted outputs. Mean of error is 2.71×10^{-5} and variance is 2.22×10^{-10}. So, as we increased the cluster radius from 0.1 to 0.5, both the mean and variance of error increased.

After mode-1 and mode-2, results for mode-3 of logistic map were also evaluated. For mode-3 of logistic map, the value of parameter *b* was fixed to 3.63. Training was completed using 300 points, and the outputs of the next fifty pairs were predicted. For cluster radius equal to 0.1, the algorithm created five clusters and did not skip any input during the prediction phase. Figure 9 shows the prediction results. Mean and variance of prediction error are 0.0358 and 0.0013, respectively. When the radius was increased to 0.5, the number of clusters reduced to two. Training was done using three hundred pairs, and the next fifty outputs were predicted. Figure 10 shows the prediction results. Mean of error is 0.0566, and variance of error is 0.0023. As radius increased from 0.1 to 0.5, both the mean and variance of error increased.

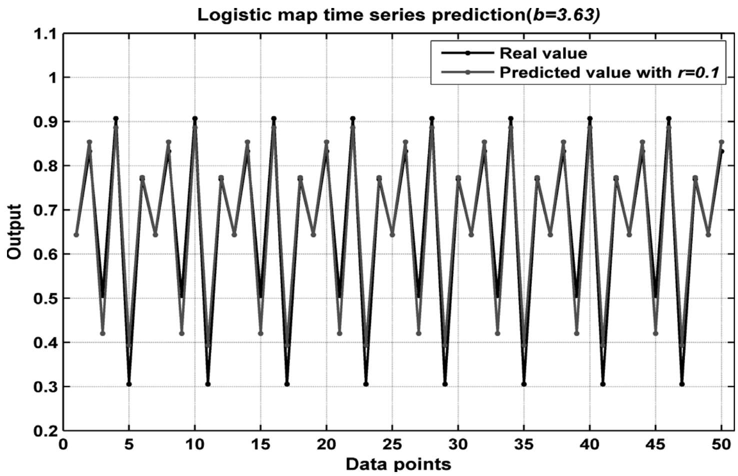

Figure 9. Prediction of mode-3 with $r = 0.1$.

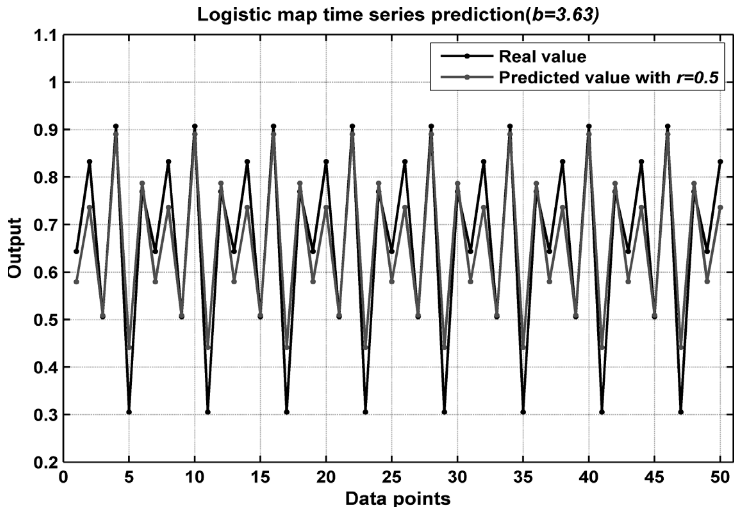

Figure 10. Prediction of mode-3 with $r = 0.5$.

For Mode-4 of logistic map, we used value of parameter b=4. This mode shows highly chaotic behavior and, hence, is difficult to predict. We first selected the value of cluster radius r=0.1 for which eight clusters were created. After training the system through 300 pairs, we predicted the next fifty outputs. Figure 11 shows the prediction results. Mean and variance of prediction error are 0.0623 and 0.0024, respectively. This error is tolerable because this mode of time series is highly chaotic, and predicting such a time series is a very challenging task.

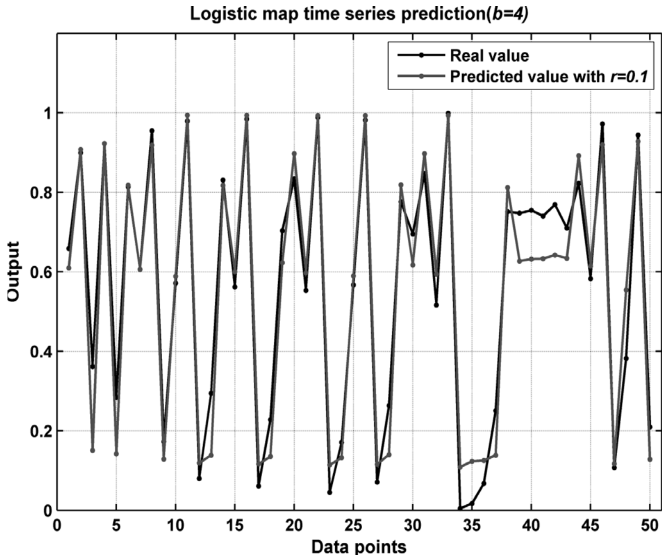

Figure 11. Prediction of mode-4 with r = 0.1.

For radius equal to 0.5, the algorithm produced two clusters and did not skip any input during the prediction phase. Figure 12 shows the prediction results. Mean of error is 0.0801 and variance of error is 0.0014. So, for this mode mean error increased but variance decreased with the increase in radius. As explained earlier, the Logistic Map Equation generates nonlinear, oscillatory, and chaotic time series for various values of b parameter. In Figure 9 and Figure 10, b = 3.63 and in Figure 11 and Figure 12, b = 4. Both cases show chaotic data series. Hence, the error in prediction increases as compared to nonlinear and oscillatory data series.

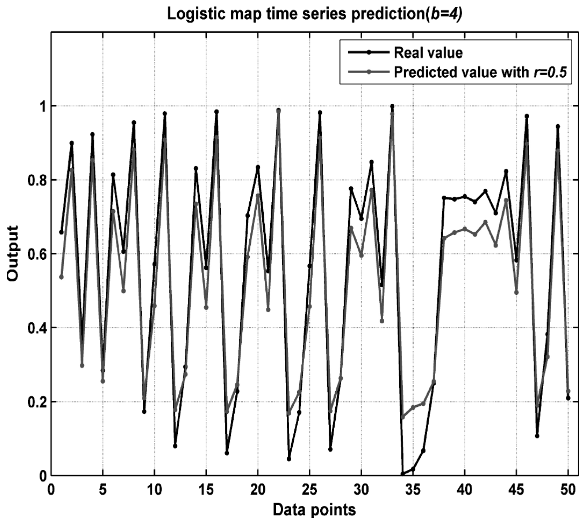

Figure 12. Prediction of mode-4 with r = 0.5.

Simulation Results of Mackey Glass Time Series

Another benchmark time series is Mackey-Glass time series that was given as input to the proposed algorithm. The Mackey-Glass chaotic time series was generated using Equation 2 with parameters $\beta = 0.2$, $\gamma = 0.1$, $n = 10$, and $\tau = 17$. Training was done through 300 pairs, and the next 300 points were predicted. Cluster radius was set equal to 0.1 for which the algorithm created nine clusters and skipped six inputs. Figure 13 illustrates the prediction results. Mean value of prediction error is 0.0303 and variance is 3.805×10^{-4}.

When cluster radius was increased to 0.5, the number of clusters decreased to two. Figure 14 shows the 300 predicted outputs. Mean of error is 0.0410 and variance is 6.86×10^{-4}. So, the increase in cluster radius caused increase in prediction error. With cluster radius of 0.5 mean error for Mackey-Glass time series is 0.0410 and for logistic map mode-4, it is 0.0801. This is because of the fact that Mackey-Glass is apparently periodic, whereas logistic map mode-4 lacks any kind of periodicity. That is why prediction error is comparatively higher in case of logistic map mode-4.

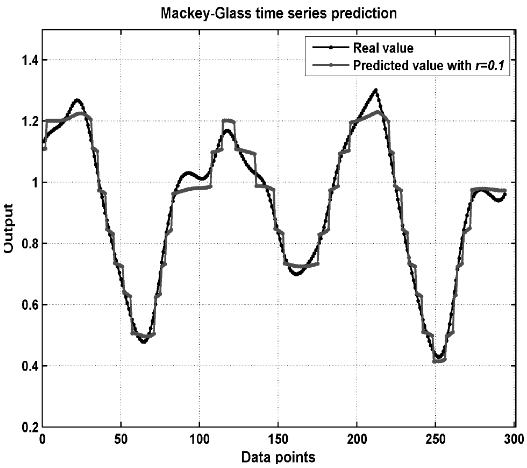

Figure 13. Prediction of Mackey-glass time series with r=0.1.

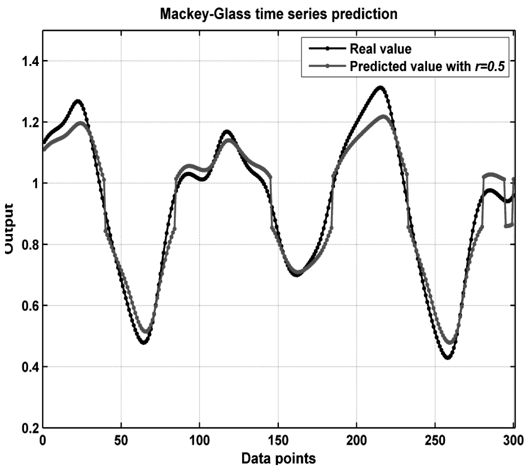

Figure 14. Prediction of Mackey-glass time series with r=0.5.

Comparison of Prediction of Mackey Glass Time Series

Fuzzy system for data points grouped into clusters using nearest neighborhood clustering can also be represented as (Wang, 1999)

$$f(x) = \frac{\Sigma_{l=1}^{M} A^l(k) exp\left(-\frac{|x - x_c^l|^2}{\sigma}\right)}{\Sigma_{l=1}^{M} B^l(k) exp\left(-\frac{\left|x - x_c^l\right|^2}{\sigma}\right)} \tag{11}$$

A^l is the sum of output values of data pairs present in the l^{th} cluster having the center x^l. B^l is the number of data pairs present in the cluster. M is the total number of clusters. It may be noted that, whichever is the nearest cluster, this function uses the data of all the clusters when used for prediction, whereas the proposed algorithm uses parameters of only the nearest cluster. Figure 15 shows the prediction result when we used the function given by Equation 11 instead of the optimal fuzzy system. A comparison of Figure 14 and Figure 15 shows that the proposed algorithm gives better results.

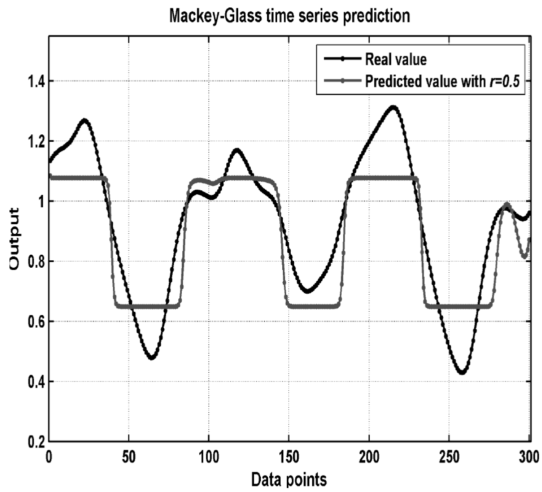

Figure 15. Prediction of Mackey-glass time series using Li-Xin Wang's function.

Prediction Results of Oil Refinery Data Series

Petroleum production data series was also given as input to the algorithm. This data series is a rescaled representation of petroleum production of Azerbaijan's oil refinery (Aliev *et al.*, 2001). The data series is shown in Figure 16. The mathematical model of this time series does not exist, so predicting it is a good test to validate the performance of the prediction algorithm. We first fixed the value of cluster radius equal to 0.01, used sixty-one pairs for training, and predicted the next one hundred outputs from which no point was skipped by the algorithm. At the end of the training phase, eleven clusters appeared. Prediction results are given in Figure 17. Mean of error is 0.0125 and variance of error is 1.27×10^{-4}.

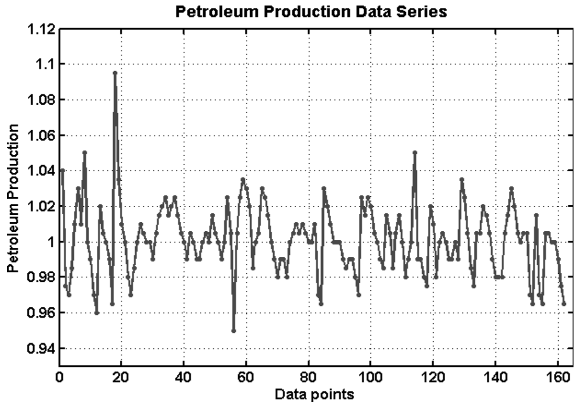

Figure 16. Petroleum production data series (Rescaled data representation).

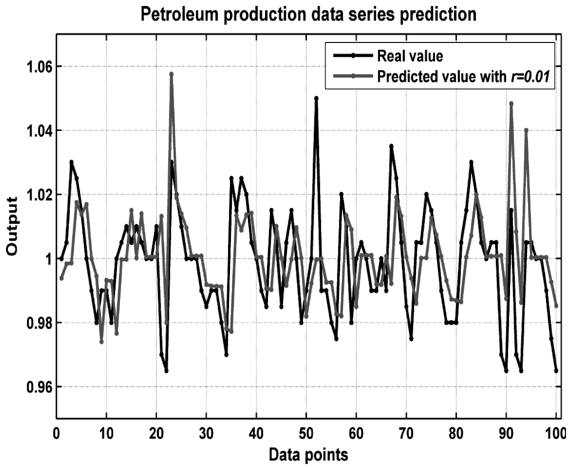

Figure 17. Prediction of oil refinery data series with r=0.01.

When cluster radius was set equal to 0.05, three clusters came into being. Figure 18 shows the prediction of one hundred outputs. No input was skipped by the algorithm during the prediction phase. Mean of error is 0.0133 and variance of error is 1.11×10^{-4}. A comparison of Figure 17 and Figure 18 suggests that an increase in cluster radius decreased the prediction accuracy.

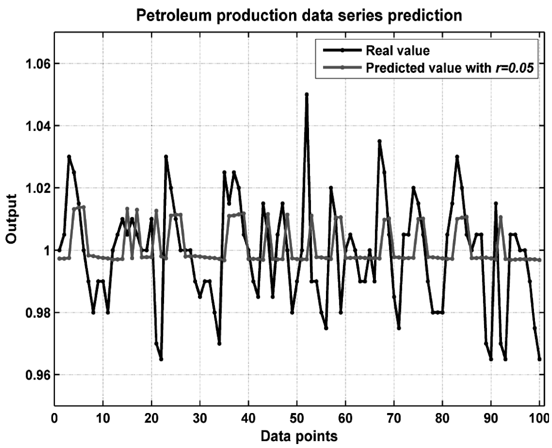

Figure 18. Prediction of oil refinery data series with r=0.05.

So, for all input time series prediction error increased with the increase in cluster radius expect for few cases where error variance decreased with the increase in cluster radius. Mean and variance of prediction error for all input time series are given in Table 1. This table also compares the accuracy of Logistic Map and Mackey Glass time series as predicted by the proposed algorithm.

Table 1. Mean and variance of prediction error.

Time series	Cluster radius 'r'	Number of clusters	Mean error	Error Variance
LME Mode-1	0.01	4	4.3×10^{-7}	3.9×10^{-16}
	0.3	1	6.7×10^{-5}	9.3×10^{-12}
LME Mode-2	0.1	3	8.74×10^{-6}	1.44×10^{-11}
	0.5	2	2.71×10^{-5}	2.22×10^{-10}
LME Mode-3	0.1	5	0.0358	0.0013
	0.5	2	0.0566	0.0023
LME Mode-4	0.1	8	0.0623	0.0024
	0.5	2	0.0801	0.0014
Mackey-Glass	0.1	9	0.0303	3.805×10^{-4}
	0.5	2	0.0410	6.866×10^{-4}
Petroleum production	0.01	11	0.0125	1.27×10^{-4}
	0.05	3	0.0133	1.11×10^{-4}

CONCLUSION

In this paper, we introduced a new fuzzy logic based prediction algorithm using the nearest neighborhood clustering scheme. The simulation results indicate that the proposed prediction algorithm is able to predict nonlinear and oscillatory data series with a very high accuracy, depending on the number of data clusters. The algorithm can also predict chaotic series with reasonable accuracy. However, the prediction error is comparatively higher for chaotic time series. The presented algorithm uses the nearest neighborhood clustering scheme for which cluster radius is an important parameter. For each input time series, we evaluated results for two different values of cluster radius. Cluster radius controls the number and size of clusters. Increasing the cluster radius increases the cluster size and lessens the number of clusters, and decreasing the cluster radius decreases the cluster size and increases the number of clusters. Simulation results show that the prediction error is small for smaller radius and large for larger radius. So, we conclude that prediction accuracy is high for smaller cluster radius because for smaller radius, the set of rules used for making a prediction becomes more specific. To the best of our knowledge, no studies have used the nearest neighborhood clustering scheme for building and selecting suitable sets of rules for prediction of time series. With each prediction, a new pair is added to the nearest cluster. This cluster update keeps on increasing the subsequent prediction accuracy.

REFERENCES

Abdulshahed, A.M., Longstaff, A.P. & Fletcher, S., 2015. The application of ANFIS prediction models for ther- mal error compensation on CNC machine tools. Applied Soft Computing, **27**: 158-168.

Aliev, R.A. & Aliev, R.R. 2001. Soft computing and its applications. World Scientific Publishing Company.

Bodyanskiy, Y. & Vynokurova, O. 2013. Hybrid adaptive wavelet-neuro-fuzzy system for chaotic time series identification. Information Sciences, **220**: 170-179.

Chen, M.Y. 2013. A hybrid ANFIS model for business failure prediction utilizing particle swarm optimization and subtractive clustering. Information Sciences, **220**: 180-195.

Choi, G.P., Kim, D.Y., Yoo, K.H. & Na, M.G. 2016. Prediction of hydrogen concentration in nuclear power plant containment under severe accidents using cascaded fuzzy neural networks. Nuclear Engineering and Design, **300**: 393-402.

Deng, W., Wang, G. & Zhang, X. 2015. A novel hybrid water quality time series prediction method based on cloud model and fuzzy forecasting. Chemometrics and Intelligent Laboratory Systems, **149**: 39-49.

Das, A., Parida, A.R. & Srivastava, P.R. 2016. ICC cricket world cup prediction model. In Information Systems Design and Intelligent Applications Springer, New Delhi: 529-539

Egrioglu, E., Aladag, C.H. & Yolcu, U. 2013. Fuzzy time series forecasting with a novel hybrid approach combining fuzzy c-means and neural networks. Expert Systems with Applications, **40**(3): 854-857.

Goyal, R., Chandra, P. & Singh, Y. 2017. Fuzzy inferencing to identify degree of interaction in the development of fault prediction models. Journal of King Saud University-Computer and Information Sciences, **29**(1): 93-102.

Goyal, M.K., Bharti, B., Quilty, J., Adamowski, J. & Pandey, A., 2014. Modeling of daily pan evaporation in sub tropical climates using ANN, LS-SVR, Fuzzy Logic, and ANFIS. Expert systems with applications, **41**(11): 5267- 5276.

Glass, L. and Mackey, M. 2010. Mackey-glass equation. Scholarpedia. **5**(3): 6908.

Hossain, A., Hossain, A., Nukman, Y., Hassan, M.A., Harizam, M.Z., Sifullah, A.M. & Parandoush, P., 2016. A fuzzy logic-based prediction model for kerf width in laser beam machining. Materials and Manufacturing Processes, **31**(5): 679-684.

Heydari, G., Vali, M. & Gharaveisi, A.A. 2016. Chaotic time series prediction via artificial neural square fuzzy inference system. Expert Systems with Applications, **55**: 461-468.

Kaya, G.U., Erkaymaz, O. & Sarac, Z. 2016. Optimization of digital holographic setup by a fuzzy logic predic- tion system. Expert Systems With Applications, **56**: 177-185.

Kuremoto, T., Kimura, S., Kobayashi, K. & Obayashi, M., 2014. Time series forecasting using a deep belief network with restricted Boltzmann machines. Neurocomputing, **137**: 47-56.

Lu, W., Pedrycz, W., Liu, X., Yang, J. & Li, P. 2014. The modeling of time series based on fuzzy information granules. Expert Systems with Applications, **41**(8): 3799-3808.

Lu, W., Zhang, L., Pedrycz, W., Yang, J. & Liu, X. 2016. The granular extension of Sugeno-type fuzzy models based on optimal allocation of information granularity and its application to forecasting of time series. Applied Soft Computing, **42**: 38-52.

Li, Y., Ling, L. & Chen, J. 2015. Combined grey prediction fuzzy control law with application to road tunnel ventilation system. Journal of applied research and technology, **13**(2): 313-320.

Lu, W., Yang, J., Liu, X. & Pedrycz, W. 2014. The modeling and prediction of time series based on synergy of high-order fuzzy cognitive map and fuzzy c-means clustering. Knowledge-Based Systems, **70**: 242-255.

Lakshmanan, M. & Rajaseekar, S. 2012. Nonlinear dynamics: integrability, chaos and patterns. Springer Sci- ence Business Media.

M., Melin, P. & Castillo, O. 2014. Particle swarm optimization of ensemble neural networks with fuzzy aggrega- tion for time series prediction of the Mexican Stock Exchange. Information Sciences, **280**: 188-204.

Norouzi, J., Yadollahpour, A., Mirbagheri, S.A., Mazdeh, M.M. & Hosseini, S.A. 2016. Predicting renal failure progression in chronic kidney disease using integrated intelligent fuzzy expert system. Computational and mathe- matical methods in medicine.

Olatomiwa, L., Mekhilef, S., Shamshirband, S. & Petkovic´, D. 2015. Adaptive neuro-fuzzy approach for solar radiation prediction in Nigeria. Renewable and Sustainable Energy Reviews, **51**: 1784-1791.

Ruan, J., Wang, X. & Shi, Y. 2013. Developing fast predictors for large-scale time series using fuzzy granular support vector machines. Applied soft computing, **13**(9): 3981-4000.

Rizwan, M., Jamil, M., Kirmani, S. & Kothari, D.P. 2014. Fuzzy logic based modeling and estimation of global solar energy using meteorological parameters. Energy **70**: 685-691.

Rasoolzadeh, A. & Tavazoei, M.S. 2012. Prediction of chaos in non-salient permanent-magnet synchronous ma- chines. Physics Letters A. **377**(1-2): 73-79.

Saleh, A.E., Moustafa, M.S., Abo-Al-Ez, K.M. & Abdullah, A.A. 2016. A hybrid neuro-fuzzy power prediction system for wind energy generation. International Journal of Electrical Power Energy Systems, **74**: 384-395.

Stavroulakis, P. ed., 2005. Chaos applications in telecommunications. CRC press.

Truong, C.D. & Anh, D.T., 2012, December. Time series prediction using motif information. In International Workshop on Multi-disciplinary Trends in Artificial Intelligence. Springer, Berlin, Heidelberg:110-121.

Vesely, S., Klockner, C.A. & Dohnal, M. 2016. Predicting recycling behaviour: Comparison of a linear regression model and a fuzzy logic model. Waste management, **49**: 530-536.

Vafaie, M.H., Ataei, M. & Koofigar, H.R. 2014. Heart diseases prediction based on ECG signals' classification using a genetic-fuzzy system and dynamical model of ECG signals. Biomedical Signal Processing and Control, **14**: 291-296.

Wang, L.X., 1999. A course in fuzzy systems.

Wang, W., Pedrycz, W. & Liu, X. 2015. Time series long-term forecasting model based on information granules and fuzzy clustering. Engineering Applications of Artificial Intelligence, **41**: 17-24.

Zhao, J. & Yu, X. 2015. Adaptive natural gradient learning algorithms for Mackey Glass chaotic time prediction. Neurocomputing **157**:41-45