

Probabilistic Fuzzy Time Series Method Based on Artificial Neural Network

Erol Egrioglu^{1,*}, Eren Bas¹, Cagdas Hakan Aladag², Ufuk Yolcu³

¹Department of Statistics, Giresun University, Giresun, Turkey

²Department of Statistics, Hacettepe University, Ankara, Turkey

³Department of Statistics, Ankara University, Ankara, Turkey

Abstract Many of forecasting methods have been proposed in the literature. There are various classifications of forecasting methods. Non-probabilistic forecasting methods such as artificial neural network, fuzzy inference systems and fuzzy time series methods have been commonly used in recent years. As a consequent of this, forecasting methods can be classified into two groups as probabilistic and non-probabilistic. Fuzzy time series methods are non-probabilistic forecasting methods. In the literature, many of fuzzy time series methods have been proposed but their distributions could not be obtained for forecasts in these methods. In this study, a new probabilistic fuzzy time series method is proposed firstly. The proposed method is based on moving block bootstrap method. It is possible to obtain distributions of forecasts by using the proposed method proposed in this study. The proposed method was applied to three real world time series data and also the performance of the proposed method was examined and compared with other forecasting methods.

Keywords Fuzzy time series, Probabilistic methods, Moving block bootstrap method, Particle swarm optimization

1. Introduction

Fuzzy time series methods are known as important forecasting methods in the literature. In recent years, many of new fuzzy time series methods have been proposed. Fuzzy time series methods are based on fuzzy set theory. There are two kind of fuzzy time series approaches as time variant and time invariant methods. In the literature, the number of papers about time invariant approaches is more than the number of papers about time variant approaches. There are some other classifications of fuzzy time series methods. These are univariate and multivariate fuzzy time series methods, fuzzy time series methods based on first order and high order forecasting methods. [1] presented a good review of fuzzy time series approaches. Artificial neural networks have been employed to define fuzzy relations in the literature. [2-9] used artificial neural networks to define fuzzy relations. Multilayer perceptron artificial neural networks are generally preferred in fuzzy time series methods except a study. In the study of [9], multiplicative neuron model neural network was preferred. Because, multiplicative neuron model neural network has only one neuron, it is simpler than multilayer perceptron and it leads successful forecasting

results in fuzzy time series methods.

Because of human's incapability, the randomness is a good remedy for uncertainty. Probabilistic methods employ random variables and the values of estimators are randomly changed sample by sample. Because of this, non-probabilistic methods contain randomness like probabilistic methods. However, non-probabilistic methods do not use estimators as random variables. In fuzzy methods, fuzziness is employed as a remedy for uncertainty but these methods do not use randomness. Like randomness, fuzziness can be used for all data sets because the variables can be clustered by using fuzzy clustering methods. It is well known that fuzzy clustering can be preferred to other clustering methods for real world data. The randomness and fuzziness can be used in a forecasting method, together.

In this study, a probabilistic fuzzy time series approach was introduced. In the proposed method, fuzzification and determining of fuzzy relations were performed by using fuzzy c-means and multiplicative neuron model artificial neural networks, respectively. The moving block bootstrap method was used to determine the distributions of forecasts. In the second section, the techniques used in the proposed method are introduced. In the third section the proposed method is introduced. Application results obtained from our proposed method and some other methods proposed in the literature are given in section four. Finally, in section five discussions and conclusions are presented.

* Corresponding author:

erole1977@yahoo.com (Erol Egrioglu)

Published online at <http://journal.sapub.org/ajis>

Copyright © 2016 Scientific & Academic Publishing. All Rights Reserved

2. Techniques which use in Probabilistic Fuzzy Time Series Methods

In the presence of multicollinearity, there are several remedies recommended for avoiding from its undesirable effects on the probabilistic fuzzy time series method uses fuzzy c-means, multiplicative neuron model artificial neural network, particle swarm optimization and moving lock bootstrap techniques. In following sections, these methods are given briefly.

2.1. Fuzzy C-Means

This technique is a fuzzy clustering technique. It was proposed by [10]. In this method, the data is partitioned into fuzzy sets by minimizing sum of square error for groups.

Let u_{ij} , v_i and n represent the membership value, cluster center and the number of variables, respectively. Thus, the forms of the objective function to be minimized given by Equation 1.

$$J_{\beta}(X, V, U) = \sum_{i=1}^c \sum_{j=1}^n u_{ij}^{\beta} d^2(x_j, v_i) \quad (1)$$

In Equation 1, β is the weighting exponent ($\beta > 1$) and $d(x_j, v_i)$ is the distance measure between the observation and the cluster center. J_{β} is tried to be minimized under the constraints given by Equation 2.

$$\begin{aligned} 0 \leq u_{ij} &\leq 1, \quad \forall i, j \\ 0 < \sum_{j=1}^n u_{ij} &\leq n, \quad \forall i \\ \sum_{i=1}^c u_{ij} &= 1, \quad \forall j \end{aligned} \quad (2)$$

Minimization process in FCM is performed by using an iterative algorithm. In each iteration, the values of u_{ij} and v_i are updated by using the formulas given by Equations 3 and 4.

$$v_i = \frac{\sum_{j=1}^n u_{ij}^{\beta} x_j}{\sum_{j=1}^n u_{ij}^{\beta}} \quad (3)$$

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d(x_j, v_i)}{d(x_j, v_k)} \right)^{2/(\beta-1)}} \quad (4)$$

2.2. Multiplicative Neuron Model Artificial Neural Network

Multiplicative neuron model neural networks were proposed by [11]. This neural network does not have a hidden layer and it has only one neuron. The multiplicative neuron model artificial neural network can be preferred to the other artificial neural networks because of its simple architecture. This network uses multiplicative aggregation function instead of additive function. Multiplications of inputs provide an advantage to obtain good training results in multiplicative neuron model artificial neural network.

2.3. Particle Swarm Optimization

Particle swarm optimization (PSO) is a stochastic and artificial intelligence optimization technique. PSO technique was firstly proposed by [12]. There are many of modifications of PSO in the literature. It is well known that PSO with time varying social, cognitive coefficients and inertia weight can produce better optimization results than standard PSO.

The most important feature of PSO algorithm is to reach the optimum point from several different points simultaneously in the search space. It can avoid from local optimum traps because of its feature. Each particle in the search space has a position and velocity representing the results for the optimization problem and search direction. And also, the velocities and the positions of the particles are updated by using some equations. Those who want more information can look the study of [12].

2.4. Moving Block Bootstrap

Bootstrap techniques can be used to obtain distributions of parameters in parametric or nonparametric statistical models. In the literature, moving block bootstrap method was developed for time series data or time-dependent data. Moving block bootstrap can be applied to time series data by choosing proper block length. Determining of block length can be effective of the results. In this study, random moving block bootstrap (RMBB) technique proposed by [13] is employed. In RMBB method, block length and starting point of a block are randomly chosen by using some distributions.

3. The Proposed Method

Fuzzy time series methods are based on fuzzy set theory. In fuzzy time series methods, linguistics was employed by using fuzzification techniques. Fuzziness is an important uncertainty type and it can be handled by using fuzzy techniques. Randomness is an important uncertainty type but it is not employed in fuzzy time series methods. Fuzzy time series methods work with data samples. The results will be changed when the data sample is changed. The data sample can be obtained randomly from population so the results will be changed randomly. As a result of this fact, forecasts of

fuzzy time series methods will be randomly changed. In the statistical theory, inferences can be easily obtained for linear models. The statistical inference for distribution of parameters and forecasts in fuzzy methods are not easy because of high degree of nonlinear model structures. Bootstrap methods can be employed to obtain distribution from all statistical methods. In this study, a new probabilistic fuzzy time series method is proposed. The distributions of forecasts are obtained by using RMBB approach. The algorithm of the proposed method is given below:

Algorithm 1: Main algorithm for the proposed method

Step 1 Number of fuzzy sets (*nfs*), length of the test set and model order (*p*) are determined.

Step 2 FCM method is applied for training data and the fuzzy time series are obtained. Fuzzy time series is a series and its elements are index numbers of fuzzy sets. After applying FCM method, the highest membership value is determined for each observation than fuzzy time series are consisted by using index number of fuzzy set which has highest membership value for the observation.

Step 3 Block length and starting point are selected randomly. Then bootstrap sample is taken from the training data of fuzzy time series according to determined block length and starting point.

Step 4 The multiplicative neuron model neural network is trained by using training data and PSO method. The inputs of the network are lagged variables of fuzzy time series ($F(t-1), F(t-2), \dots, F(t-p)$), output of the network is fuzzy time series ($F(t)$).

Step 5 The forecasts are obtained for test data and they are kept.

Step 6 Steps 3-5 are repeated until it is reached to possible bootstrap sample number.

Step 7 The distributions of the forecasts and arithmetic means of bootstrap samples of forecasts are obtained. The results of forecasts of the method are arithmetic means of bootstrap samples of forecasts.

Algorithm 2: Training algorithm for the multiplicative neuron model artificial neural network

Step 1 Initial velocity and position vectors of the particles are generated randomly. All initial positions of the particles are generated randomly from (0, 1) interval. On the other hand, velocities are generated randomly from $(-v_{maps}, v_{maps})$ interval according to the pre-determined *vmaps* limit value.

Step 2 Fitness function value for each particle is calculated. By using outputs calculated from the network for learning samples as fitness function, root-mean squared error (RMSE) value is obtained by using Equation 5.

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (F_t - \hat{F}_t)^2} \quad (5)$$

Where \hat{F}_t is the output of the network, F_t is the target value and n is the number of training examples. For the calculation of RMSE value for each particle, outputs of the

network $\hat{F}_t, t = 1, 2, \dots, n$ are calculated.

Step 3 *Pbest* and *Gbest* are updated. If the fitness value of *Gbest* is under a certain value of ϵ , or it is reached to the maximum number of iterations, the process is stopped. Otherwise, moves to Step 4.

Step 4 Velocity values of the positions and positions are updated and return to Step 2. Formulas given in Equations (6) and (7) are used.

$$v_{ij}^{k+1} = w \times v_{ij}^k + c_1 r_1 (pbest_{ij} - x_{ij}) + c_2 r_2 (gbest_j - x_{ij}) \quad (6)$$

$$x_{ij}^{k+1} = x_{ij}^k + v_{ij}^{k+1} \quad (7)$$

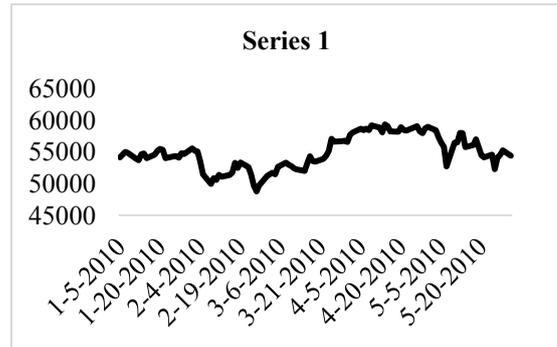


Figure 1. Time series graph of Series 1

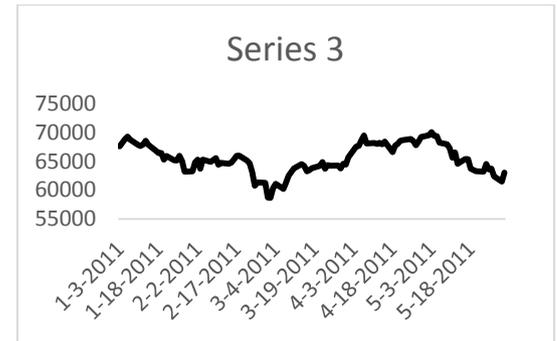


Figure 2. Time series graph of Series 2

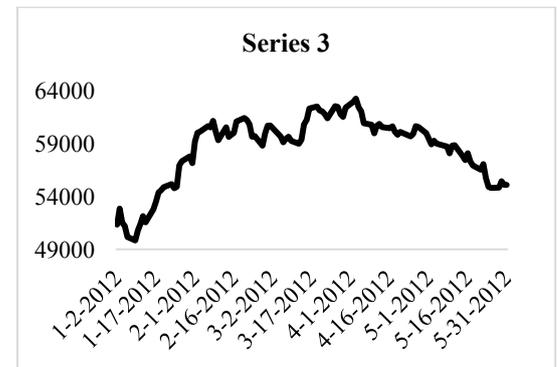


Figure 3. Time series graph of Series 3

4. Implementation

For the evaluation of the forecasting performance of the proposed method, three real world time series data were analyzed. These time series include daily data 2010, 2011

and 2012 Istanbul Stock Exchange Market BIST100 index.

Series 1: BIST100 index values with 104 observations observed between 01.04.2010 and 05.31.2010 (Figure 1)

Series 2: BIST100 index values with 106 observations observed between 01.03.2011 and 05.31.2011 (Figure 2)

Series 3: BIST100 index values with 106 observations observed between 01.02.2012 and 05.31.2012 (Figure 3)

In the analysis of the time series, the last 7 and 15 values of each series were taken as test set. Alternative forecasting methods used in the analysis and implementation details of the methods are listed below.

ARIMA: Autoregressive Integrated Moving Average Model, The best model was determined Box-Jenkins Procedure.

ES: Exponential Smoothing, Simple, Holt and winters exponential smoothing methods were applied and the best model was selected.

MLP-ANN: Multi-Layer Perceptron Artificial Neural Network, The number of inputs and hidden layer neurons were changed from 1 to 5 and the best architecture was selected by trial&error method. Levenberg Marquardt training algorithm was used as learning algorithm.

SC: Song and Chissom time invariant fuzzy time series method ([14]) the numbers of fuzzy sets were changed from 5 to 15 and the best number of fuzzy sets were selected.

FF: Fuzzy function approach ([15]). The model order and the number of fuzz sets were changed from 1 to 5 and from 5 to 15, respectively.

Besides these methods, the proposed method was also compared with Fuzzy Time Series Network (FTS-N) proposed by [16].

For the comparison of the methods, values of RMSE and Mean Absolute Percentage Error (MAPE) criteria calculated for the test set were used. MAPE criteria is calculated by using Equation (4)

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{X_t - \hat{X}_t}{X_t} \right| \quad (4)$$

In the implementation of the proposed method, the model order (p) was varied between 1 and 5; the number of fuzzy sets in the proposed method was varied between 5 and 15 and bootstrap iteration number was taken as 100. Best-case results were obtained in all possible situations mentioned above. In the implementation of PSO, $w = 0.9$, $c_1 = c_2 = 2$, $pn = 30$, $vm = 0.1$ and the maximum number of iterations was taken as 200.

Tables 1 and 2 summarize the results obtained from test set for Series 1 in terms of RMSE and MAPE criteria when the length of test set (n_{test}) is 7 and 15, respectively.

In Table 1, the best result of the proposed method was obtained from third-order model when the number of fuzzy sets was 13. In Table 2, the best forecasting performance was obtained when the model order is 3 and the number of fuzzy set is 14. Analysis of Tables 1 and 2 reveals that the proposed method exhibits more successful and superior forecasting performance when compared with other methods in terms of MAPE and RMSE performance measures. The graphs of

actual values of the test set and forecasts of the proposed method were given in Figures 4 and 5, respectively when the length of the test set is 7 and 15 for Series 1.

Table 1. Comparison of Series 1 forecasts for test set in terms of RMSE and MAPE criteria when $n_{test}=7$

Methods	RMSE	MAPE
ARIMA	1221	0.0183
ES	1208	0.0185
MLP-ANN	1077	0.0143
SC	1128	0.0182
FF	1180	0.0179
FTS-N	1050	0.0159
Proposed Method	1028	0.0165

Table 2. Comparison of Series 1 forecasts for test set in terms of RMSE and MAPE criteria when $n_{test}=15$

Methods	RMSE	MAPE
ARIMA	1612	0.0220
ES	1612	0.0220
MLP-ANN	1603	0.0220
SC	1742	0.0265
FF	1852	0.0264
FTS-N	1357	0.0202
Proposed Method	1313	0.0209

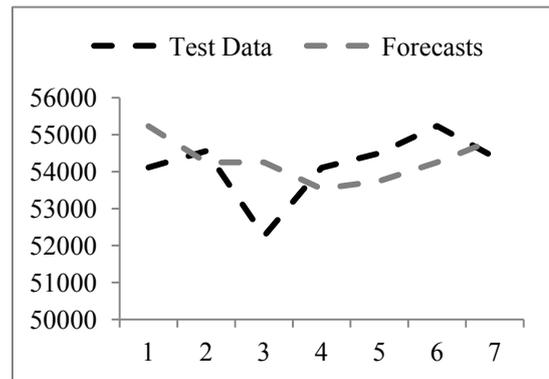


Figure 4. Time series graph of test data ($n_{test}=7$) and forecasts obtained from proposed method for Series 1

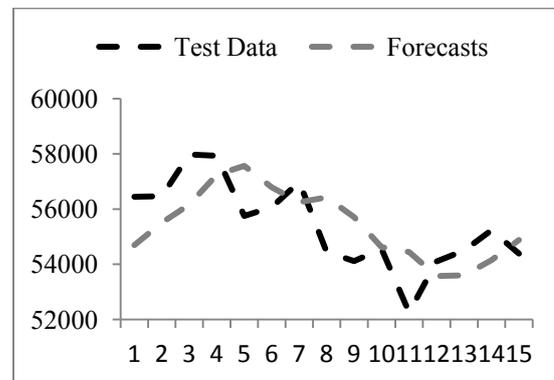


Figure 5. Time series graph of test data ($n_{test}=15$) and forecasts obtained from proposed method for Series 1

Similarly, obtained results for Series 2 were summarized in Tables 3 and 4, respectively.

Table 3. Comparison of Series 2 forecasts for test set in terms of RMSE and MAPE criteria when $n_{test}=7$

Methods	RMSE	MAPE
ARIMA	651	0.0084
ES	651	0.0084
MLP-ANN	775	0.0111
SC	1292	0.0183
FF	1034	0.0162
FTS-N	590	0.0084
Proposed Method	546	0.0060

Table 4. Comparison of Series 2 forecasts for test set in terms of RMSE and MAPE criteria when $n_{test}=15$

Methods	RMSE	MAPE
ARIMA	621	0.0088
ES	621	0.0088
MLP-ANN	783	0.0117
SC	1047	0.0153
FF	1038	0.0161
FTS-N	582	0.0087
Proposed Method	575	0.0077

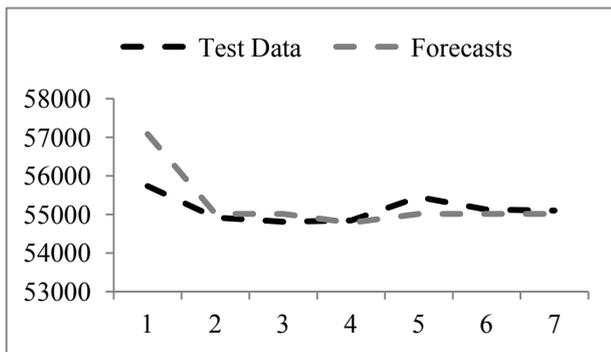


Figure 6. Time series graph of test data ($n_{test}=7$) and forecasts obtained from proposed method for Series 2

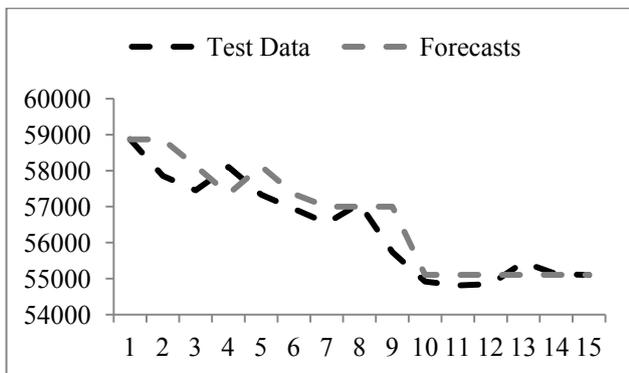


Figure 7. Time series graph of test data ($n_{test}=15$) and forecasts obtained from proposed method for Series 2

In Table 3, the best result of the proposed method was obtained from fourth-order model when the number of fuzzy

sets is nine. In Table 4, the best forecasting performance was obtained when the model order is 1 and the number of fuzzy set is 15. Analysis of Tables 3 4 reveals that the proposed method exhibits more successful and superior forecasting performance compared to other methods in terms of MAPE and RMSE performance measures. The graphs of actual values of the test set and forecasts of the proposed method were given in Figures 5 and 6, respectively when the length of the test set is 7 and 15 for Series 2.

Similarly, obtained results for Series 3 were summarized in Tables 5 and 6, respectively.

Table 5. Comparison of Series 3 forecasts for test set in terms of RMSE and MAPE criteria when $n_{test}=7$

Methods	RMSE	MAPE
ARIMA	1058	0.0144
ES	1057	0.0144
MLP-ANN	920	0.0128
SC	1396	0.0200
FF	1083	0.0153
FTS-N	765	0.0105
Proposed Method	924	0.0109

Table 6. Comparison of Series 3 forecasts for test set in terms of RMSE and MAPE criteria when $n_{test}=15$

Methods	RMSE	MAPE
ARIMA	1130	0.0150
ES	1130	0.0150
MLP-ANN	1096	0.0146
SC	1360	0.0189
FF	1146	0.0156
FTS-N	917	0.0121
Proposed Method	943	0.0118

According to all tables, the best results are obtained from our proposed method. Moreover, proposed method gives the empirical probability distributions of forecasts. For example, the probability distribution for the first forecast of Series 1 when length of test set is 15 is given in Table 7. It is noted that the forecast is the mean of this distribution.

Table 7. The probability distribution for the first forecast of Series 1 when $n_{test}=15$

Value	Probability
51355.402	0.02
52504.219	0.34
53340.522	0.05
53959.400	0.08
54775.578	0.01
55139.474	0.01
55564.950	0.23
56662.976	0.12
57962.177	0.05
58414.179	0.09
Mean	54696.35

5. Discussions and Conclusions

In this paper, a probabilistic fuzzy time series method was introduced. The proposed method was applied for Istanbul stock exchange BIST100 data by using three different time series selected from the entire data. According to findings, the new method can produce better forecasts than other methods. The empirical probability distributions can be estimated for forecasts in the proposed method. The proposed fuzzy time series method is the first method which produces probabilities for the forecasts in the literature. It can be said that there is a weakness of the method about computation time. Nevertheless, the proposed method can complete to forecast a time series in a suitable time. Besides, one of the most important properties of the proposed method is to produce probabilistic results.

REFERENCES

- [1] Singh, P., 2015, A brief review of modelling approaches based on fuzzy time series, *Int. J. Mach. Learn. & Cyber.*, Doi: 10.1007/s13042-015-0332-y.
- [2] Huarng, K., and Yu, H.K., 2006, The application of neural networks to forecast fuzzy time series, *Physica A*, 363, 481-491.
- [3] Aladag, C.H., Basaran, M.A., Egrioglu, E., Yolcu, U., and Uslu, V.R., 2009, Forecasting in high order fuzzy time series by using neural networks to define fuzzy relations, *Expert Systems with Applications*, 36, 4228-4231.
- [4] Egrioglu, E., Aladag, C.H., Yolcu, U., Uslu, V.R., and Basaran, M.A., 2009a, A new approach based on artificial neural networks for high order multivariate fuzzy time series, *Expert Systems with Applications*, 36, 10589-10594.
- [5] Aladag, C.H., Egrioglu, E., and Yolcu, U., 2010, Forecast combination using artificial neural networks, *Neural Processing Letters*, 32, 269-276.
- [6] Yu, T.H.K., and Huarng K.H., 2010, A neural network-based fuzzy time series model to improve forecasting, *Expert Systems with Applications*, 37, 3366-3372.
- [7] Huarng K.H., and Yu T.H.K., 2012, Modelling fuzzy time series with multiple observations, *International Journal of Innovative Computing, Information and Control*, 8(10), 7415-7426.
- [8] Yolcu, U., Aladag, C.H., and Egrioglu, E., 2013, A new linear & nonlinear artificial neural network model for time series forecasting, *Decision Support System Journals*, 54, 1340-1347.
- [9] Aladag, C.H., 2013, Using multiplicative neuron model to establish fuzzy logic relationships, *Expert Systems with Applications*, 40 (3), 850 – 853.
- [10] Bezdek, J.C., *Pattern recognition with fuzzy objective function algorithms*, Plenum Press, NY, 1981.
- [11] Yadav, R.N., Kalra, P.K., and John, J., 2007, Time series prediction with single multiplicative neuron model, *Applied Soft Computing*, 7, 1157-1163.
- [12] Kennedy, J., and Eberhart, R.C., In: *Particle Swarm Optimization*, IEEE International Conference on Neural Network, 1942-1948, 1995.
- [13] Yolcu, U., Yolcu, O.C., Aladag, C.H., and Egrioglu, E., 2014, An enhanced fuzzy time series forecasting method based on artificial bee colony algorithm, *Journal of Intelligent and Fuzzy Systems*, 26(6), 2627-2637.
- [14] Song Q., and Chissom B.S., 1993b, Forecasting enrollments with fuzzy time series - Part I., *Fuzzy Sets and Systems* 54, 1-10.
- [15] Turksen, B., 2008, Fuzzy function with LSE, *Applied Soft Computing*, 8, 1178-1188.
- [16] Bas, E., Egrioglu, E., Aladag, C.H., and Yolcu, U., 2015, A fuzzy time series network to forecast nonlinear time series, *Applied Intelligence*, 43 (2), 343-355.