

# Ensemble Neural Network with Type-2 Fuzzy Weights Using Response Integration for Time Series Prediction



Fernando Gaxiola, Patricia Melin, Fevrier Valdez and Juan R. Castro

**Abstract** In this paper an ensemble of three neural networks with type-2 fuzzy weights is proposed. One neural network uses type-2 fuzzy inference systems with Gaussian membership functions for obtain the fuzzy weights; the second neural network uses type-2 fuzzy inference systems with triangular membership functions; and the third neural network uses type-2 fuzzy inference systems with triangular membership functions with uncertainty in the standard deviation. Average integration and type-2 fuzzy integrator are used for the results of the ensemble neural network. The proposed approach is applied to a case of time series prediction, specifically in the Mackey-Glass time series.

## 1 Introduction

We are presenting an ensemble with three neural networks for the experiments. The final result for the ensemble was obtained with average integration and type-2 fuzzy integration. The time series prediction area is the study case for this paper, and particularly the Mackey-Glass time series is used to test the proposed approach.

This research uses the managing of the weights of a neural networks using type-2 fuzzy inference systems and due to the fact that these affect the performance of the learning process of the neural network, the used of type-2 fuzzy weights are an important part in the training phase for managing uncertainty.

---

F. Gaxiola (✉) · P. Melin · F. Valdez  
Tijuana Institute of Technology, Tijuana, Mexico  
e-mail: fergaor\_29@hotmail.com

P. Melin  
e-mail: pmelin@tectijuana.mx

F. Valdez  
e-mail: fevrier@tectijuana.mx

J. R. Castro  
UABC University, Tijuana, Mexico  
e-mail: jrcastror@uabc.edu.mx

One type of supervised neural network and its variations is the one that would be of most interest in our study, which is the backpropagation network. This type of network is the most commonly used in the above mentioned areas.

The weights of a neural network are an important part in the training phase, because these affect the performance of the learning process of the neural network.

This conclusion is based on the practice of neural networks of this type, where some research works have shown that the training of neural networks for the same problem initialized with different weights or its adjustment in a different way but at the end is possible to reach a similar result.

The next section presents the basic concepts of neural networks and type-2 fuzzy logic. Section 3 presents a review of research about modifications of the backpropagation algorithm, different management strategies of weights in neural networks and time series prediction. Section 4 explains the proposed ensemble neural network. Section 5 describes the simulation results for the ensemble neural network with average integration and the type-2 fuzzy integrator proposed in this paper. Finally, in Sect. 6, some conclusions are presented.

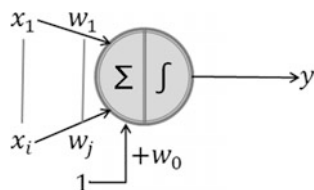
## 2 Basic Concepts

### 2.1 Neural Network

An artificial neural network (ANN) is a distributed computing scheme based on the structure of the nervous system of humans. The architecture of a neural network is formed by connecting multiple elementary processors, this being an adaptive system that has an algorithm to adjust their weights (free parameters) to achieve the performance requirements of the problem based on representative samples [1, 2]. The most important property of artificial neural networks is their ability to learn from a training set of patterns, i.e. they are able to find a model that fits the data [3, 4].

The artificial neuron consists of several parts (see Fig. 1). On one side are the inputs, weights, the summation, and finally the adapter function. The input values are multiplied by the weights and added:  $\sum x_i w_{ij}$ . This function is completed with the addition of a threshold amount  $i$ . This threshold has the same effect as an entry with value  $-1$ . It serves so that the sum can be shifted left or right of the origin.

**Fig. 1** Scheme of an artificial neuron



After addition, we have the function  $f$  applied to the sum, resulting the final value of the output, also called  $y_i$  [5], obtaining the following equation.

$$y_i = f\left(\sum_{i=1}^n x_i w_{ij}\right). \quad (1)$$

where  $f$  may be a nonlinear function with binary output  $+ -1$ , a linear function  $f(z) = z$ , or as sigmoidal logistic function:

$$f(z) = \frac{1}{1 + e^{-z}}. \quad (2)$$

## 2.2 Type-2 Fuzzy Logic

The concept of a type-2 fuzzy set, was introduced by Zadeh (1975) as an extension of the concept of an ordinary fuzzy set (henceforth called a “type-1 fuzzy set”). A type-2 fuzzy set is characterized by a fuzzy membership function, i.e., the membership grade for each element of this set is a fuzzy set in  $[0, 1]$ , unlike a type-1 set where the membership grade is a crisp number in  $[0, 1]$  [6, 7].

Such sets can be used in situations where there is uncertainty about the membership grades themselves, e.g., uncertainty in the shape of the membership function or in some of its parameters [8]. Consider the transition from ordinary sets to fuzzy sets. When we cannot determine the membership of an element in a set as 0 or 1, we use fuzzy sets of type-1 [9–11]. Similarly, when the situation is so fuzzy that we have trouble determining the membership grade even as a crisp number in  $[0, 1]$ , we use fuzzy sets of type-2 [12–17].

## 3 Historical Development

The backpropagation algorithm and its variations are the most useful basic training methods in the area of research of neural networks. When applying the basic backpropagation algorithm to practical problems, the training time can be very high. In the literature we can find that several methods have been proposed to accelerate the convergence of the algorithm [18–21].

There exist many works about adjustment or managing of weights but only the most important and relevant for this research will be considered here [22–25].

Ishibuchi et al. [26], proposed a fuzzy network where the weights are given as trapezoidal fuzzy numbers, denoted as four trapezoidal fuzzy numbers for the four parameters of trapezoidal membership functions.

Ishibuchi et al. [27], proposed a fuzzy neural network architecture with symmetrical fuzzy triangular numbers for the fuzzy weights and biases, denoted by the lower, middle and upper limit of the fuzzy triangular numbers.

Momentum method—Rumelhart, Hinton and Williams suggested adding in the increased weights expression a momentum term  $\beta$ , to filter the oscillations that can be formed a higher learning rate that lead to great change in the weights [5, 28].

Adaptive learning rate—focuses on improving the performance of the algorithm by allowing the learning rate changes during the training process (increase or decrease) [28].

Castro et al. [29], proposed interval type-2 fuzzy neurons for the antecedents and interval of type-1 fuzzy neurons for the consequents of the rules.

Kamarthi and Pittner [30], focused in obtaining a weight prediction of the network at a future epoch using extrapolation. Feuring [31], developed a learning algorithm in which the backpropagation algorithm is used to compute the new lower and upper limits media weights. The modal value of the new fuzzy weight is calculated as the average of the new computed limits.

Recent works on type-2 fuzzy logic have been developed in time series prediction, like that of Castro et al. [32], and other researchers [33, 34].

## 4 Proposed Ensemble Neural Network

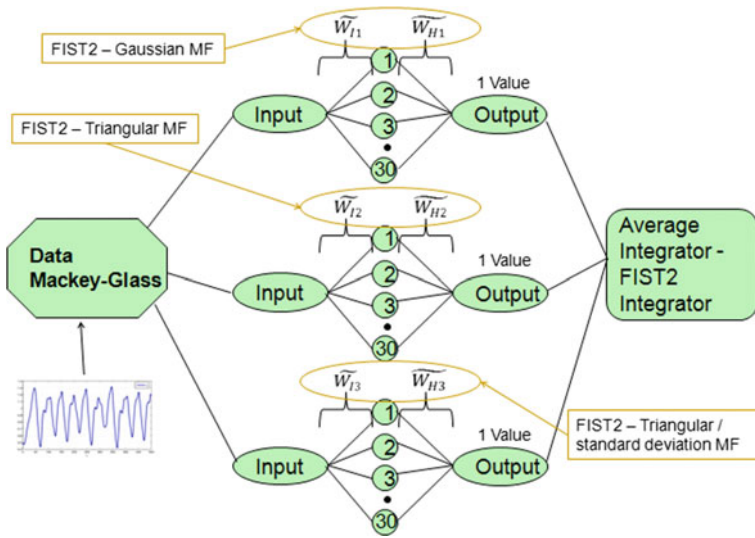
The focus of this work is to use ensemble neural networks with three neural networks with type-2 fuzzy weights to allow the neural network to handle data with uncertainty; we used an average integration approach and type-2 fuzzy integrator for the final result of the ensemble. The approach is applied in time series prediction for the Mackey Glass time series (for  $\tau = 17$ ).

The three neural network works with type-2 fuzzy weights [35], one network works with two-sided Gaussian interval type-2 membership functions with uncertain mean and standard deviation in the two type-2 fuzzy inference systems (FIST2) used to obtain the weights (one in the connections between the input and hidden layer and the other between the hidden and output layer); the other two networks work with triangular interval type-2 membership function with uncertain and triangular interval type-2 membership function with uncertain standard deviation, respectively (see Fig. 2).

We considered a three neural network architecture, and each network works with 30 neurons in the hidden layer and 1 neuron in the output layer. These neural networks handle type-2 fuzzy weights in the hidden layer and output layer. In the hidden layer and output layer of the networks we are working with a type-2 fuzzy inference system obtaining new weights in each epoch of the networks [36–39].

We used two similar type-2 fuzzy inference systems to obtain the type-2 fuzzy weights in the hidden and output layer for the neural network.

The weight managing in the three neural networks will be done differently to the traditional management of weights performed with the backpropagation algorithm



**Fig. 2** Proposed ensemble neural network architecture with interval type-2 fuzzy weights using average integration or type-2 fuzzy integrator

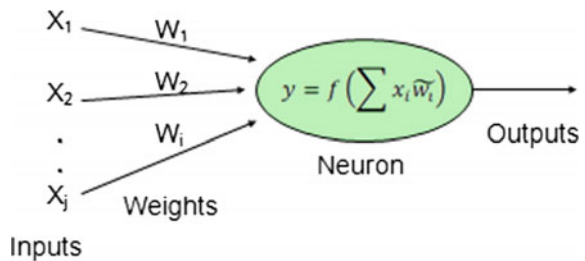
(see Fig. 3); the method works with interval type-2 fuzzy weights, taking into account the change in the way we work internally in the neuron (see Fig. 4) [40].

The activation function  $f(-)$  used in this research was the sigmoid function in the neurons of the hidden layer and the linear function in the neurons of the output for the three neural networks.

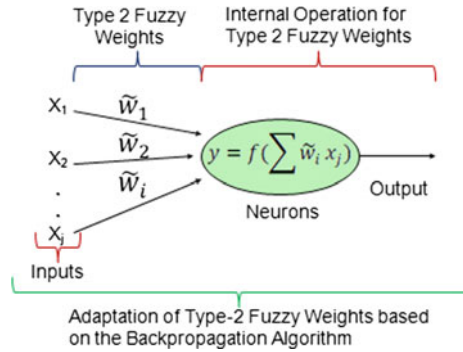
The three neural networks used two type-2 fuzzy inference systems with the same structure (see Fig. 5), which have two inputs (the current weight in the actual epoch and the change of the weight for the next epoch) and one output (the new weight for the next epoch).

In the first neural network, the inputs and the output for the type-2 fuzzy inference systems used between the input and hidden layer are delimited with two Gaussian membership functions with their corresponding range (see Fig. 6); and the inputs and output for the type-2 fuzzy inference systems used between the hidden

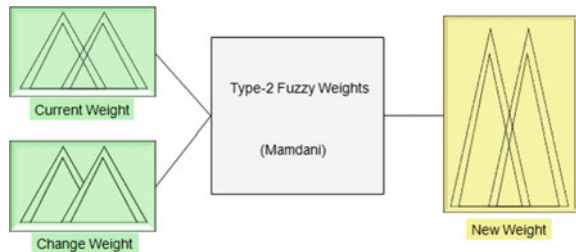
**Fig. 3** Schematic of the management of numerical weights for input of each neuron



**Fig. 4** Schematic of the management of interval type 2 fuzzy weights for input of each neuron



**Fig. 5** Structure of the six type-2 fuzzy inference systems used in the three neural networks



and output layer are delimited with two Gaussian membership functions with their corresponding range (see Fig. 7).

In the second neural network, the inputs and the output for the type-2 fuzzy inference systems used between the input and hidden layer are delimited with two triangular membership functions with their corresponding ranges (see Fig. 8); and the inputs and output for the type-2 fuzzy inference systems used between the hidden and output layer are delimited with two triangular membership functions with their corresponding ranges (see Fig. 9).

In the third neural network, the inputs and the output for the type-2 fuzzy inference systems used between the input and hidden layer are delimited with two triangular membership functions with standard deviation with their corresponding range (see Fig. 10); and the inputs and output for the type-2 fuzzy inference systems used between the hidden and output layer are delimited with two triangular membership functions with uncertainty in the standard deviation with their corresponding ranges (see Fig. 11).

The rules for the six type-2 fuzzy inference systems are the same, we used six rules for the type-2 fuzzy inference systems, corresponding to the four combinations of two membership functions and we added two rules for the case when the change of weight is null (see Fig. 12).

We obtain the prediction result for the ensemble neural network using the average integration and type-2 fuzzy integrator.

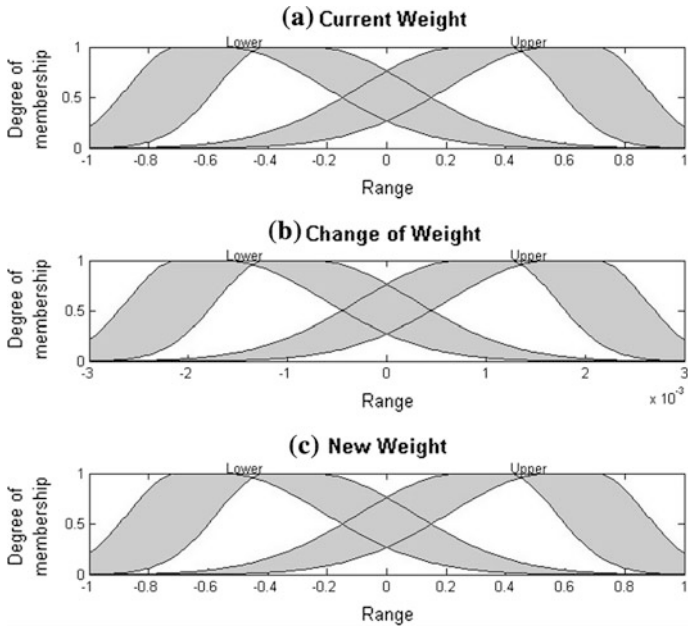


Fig. 6 Inputs (a and b) and output (c) of the type-2 fuzzy inference system used between the input and hidden layer for the first neural network

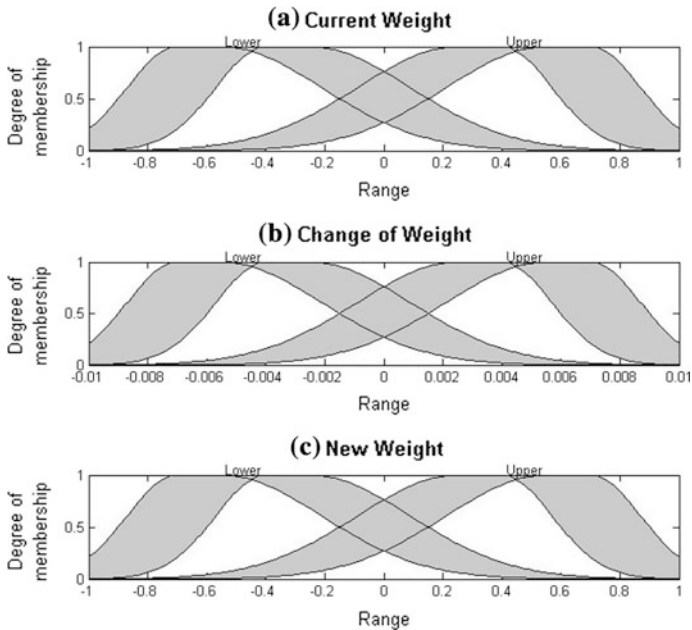
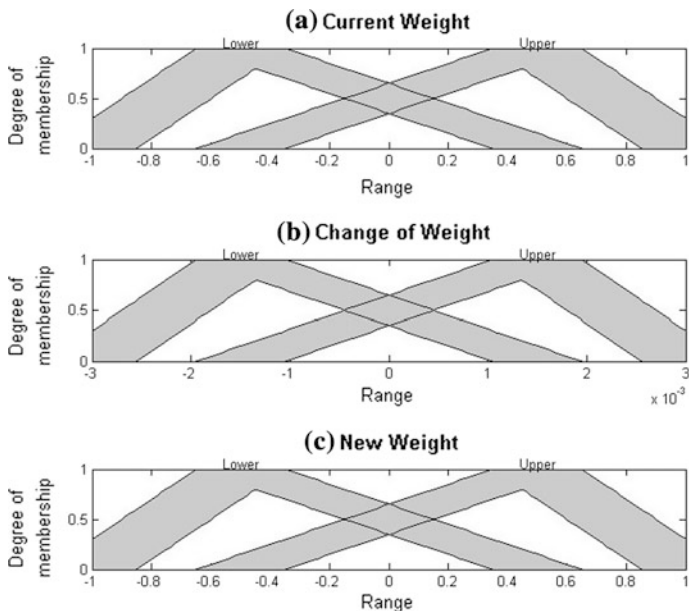
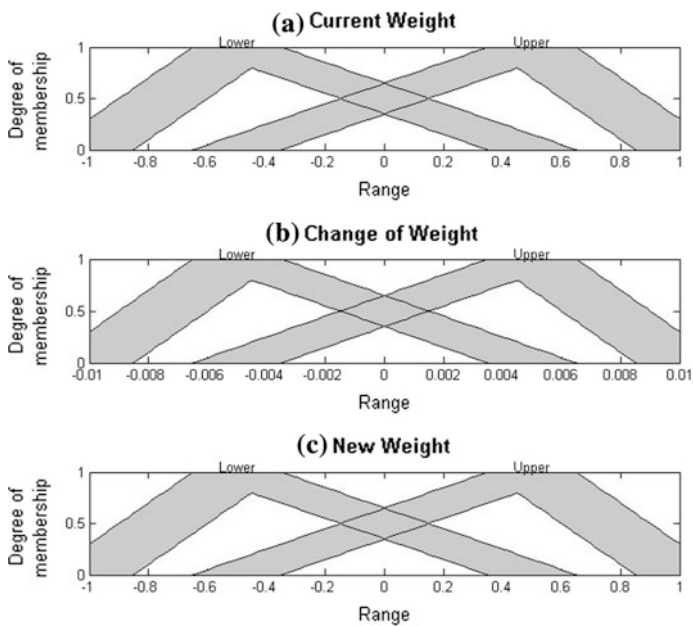


Fig. 7 Inputs (a and b) and output (c) of the type-2 fuzzy inference system used between the hidden and output layer for the first neural network

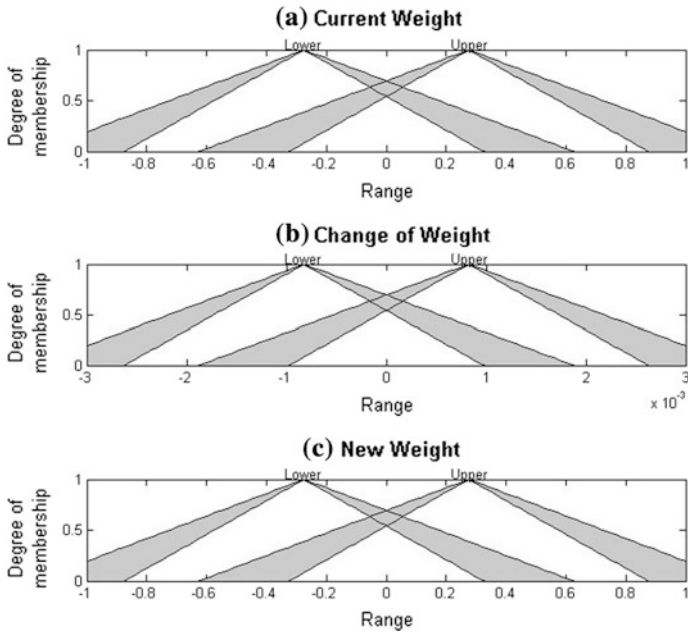


**Fig. 8** Inputs (a and b) and output (c) of the type-2 fuzzy inference system used between the input and hidden layer for the second neural network

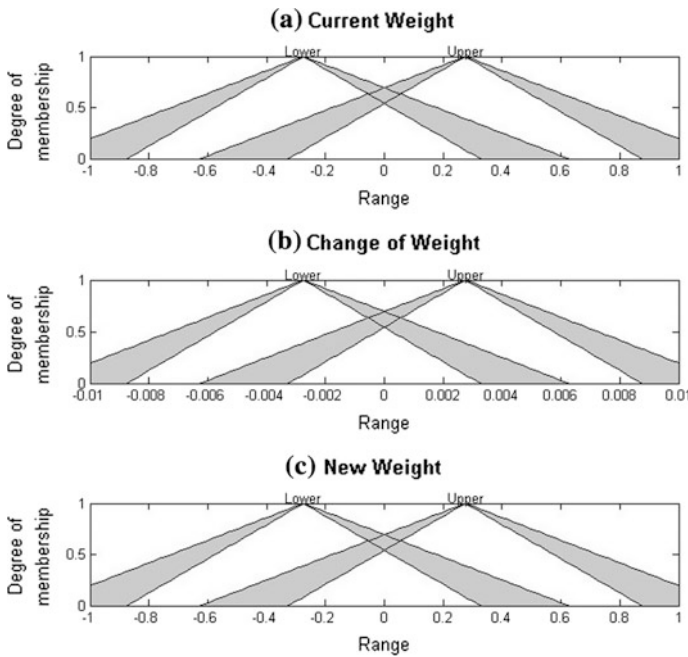


**Fig. 9** Inputs (a and b) and output (c) of the type-2 fuzzy inference system used between the hidden and output layer for the second neural network





**Fig. 10** Inputs (a and b) and output (c) of the type-2 fuzzy inference system used between the input and hidden layer for the third neural network



**Fig. 11** Inputs (a and b) and output (c) of the type-2 fuzzy inference system used between the hidden and output layer for the third neural network

1. (Current\_Weight is lower) and (Change\_Weight is lower) then (New\_Weight is lower)
2. (Current\_Weight is lower) and (Change\_Weight is upper) then (New\_Weight is lower)
3. (Current\_Weight is upper) and (Change\_Weight is lower) then (New\_Weight is upper)
4. (Current\_Weight is upper) and (Change\_Weight is upper) then (New\_Weight is upper)
5. (Current\_Weight is lower) then (New\_Weight is lower)
6. (Current\_Weight is upper) then (New\_Weight is upper)

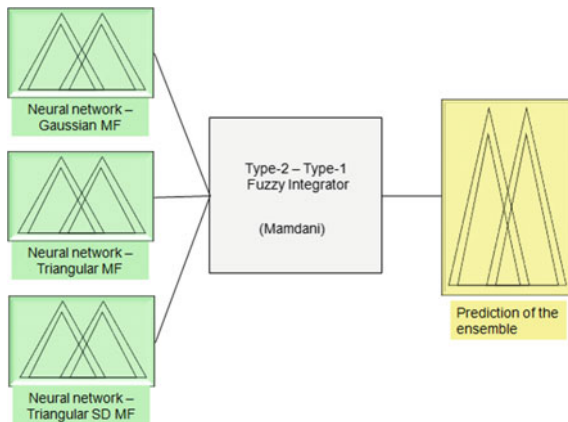
**Fig. 12** Rules of the type-2 fuzzy inference system used in the six FIST2 for the neural networks with type-2 fuzzy weights

The average integration is performed with the Eq. 3 (prediction of the neural network with FIST2 Gaussian MF: NNGMF, prediction of the neural network with FIST2 triangular MF: NNTMF, prediction of the neural network with FIST2 triangular SD MF: NNTsdMF, number of neural networks in the ensemble: #NN, and prediction of the ensemble: PE).

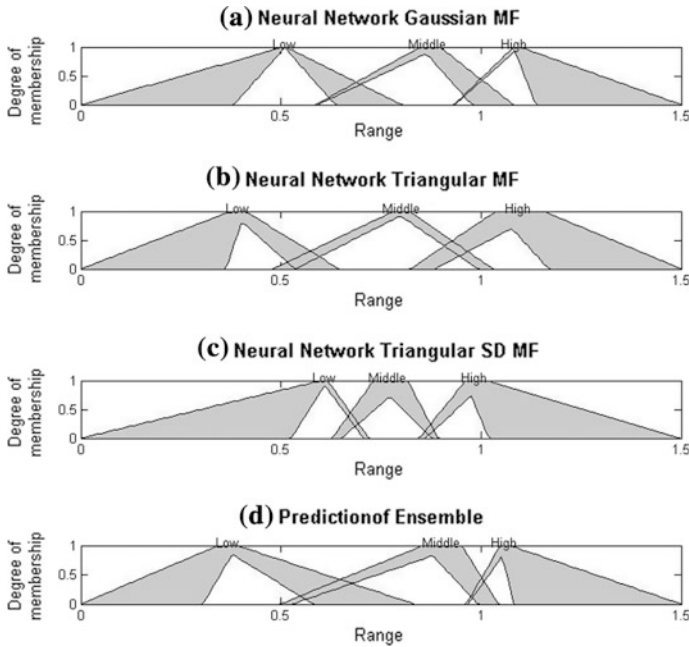
$$PE = \frac{NNGMF + NNTMF + NNTsdMF}{\#NN} \tag{3}$$

The structure of the type-2 fuzzy integrator consists of three inputs: the prediction for the neural network with type-2 fuzzy weights using Gaussian membership functions (MF), triangular MF and triangular MF with uncertainty in the standard deviation; and one output: the final prediction of the integration (see Fig. 13)

We used three triangular membership functions in the inputs and output for the type-2 fuzzy integrator (T2FI) and the range is established in the interval for 0–1.5 (see Fig. 14). The footprint and positions of the membership functions are established empirically.



**Fig. 13** Structure of the type-2 fuzzy integrator



**Fig. 14** Structure of the type-2 fuzzy integrator

In the type-2 fuzzy integrator we utilized 30 rules, 27 for the combination of the three inputs with “and” operator and there are also 3 rules using the “or” operator (see Fig. 15).

## 5 Simulation Results

The results for the experiments for the ensemble neural network with average integration (ENNAI) are shown on Table 1 and Fig. 16. The best prediction error is of 0.0346, and the average error is of 0.0485.

We presented 10 experiments of simulations for the ensemble neural network with the average integration and the type-2 fuzzy integrator, but the average error was calculated considering 30 experiments with the same parameters and conditions. The results for the experiments for the ensemble neural network with type-2 fuzzy integrator (ENNT2FI) are shown on Table 2. The best prediction error is of 0.0265, and the average error is of 0.0561.

We show in Table 3 a comparison for the prediction for the Mackey-Glass time series between the results for the monolithic neural network (MNN), the neural

1. (NNGMF is Low) and (NNTMF is Low) and (NNTsdMF is Low) then (PE is Low)
2. (NNGMF is Low) and (NNTMF is Low) and (NNTsdMF is Middle) then (PE is Low)
3. (NNGMF is Low) and (NNTMF is Low) and (NNTsdMF is High) then (PE is Low)
4. (NNGMF is Low) and (NNTMF is Middle) and (NNTsdMF is Low) then (PE is Low)
5. (NNGMF is Low) and (NNTMF is Middle) and (NNTsdMF is Middle) then (PE is Middle)
6. (NNGMF is Low) and (NNTMF is Middle) and (NNTsdMF is High) then (PE is Middle)
7. (NNGMF is Low) and (NNTMF is High) and (NNTsdMF is Low) then (PE is Low)
8. (NNGMF is Low) and (NNTMF is High) and (NNTsdMF is Middle) then (PE is Middle)
9. (NNGMF is Low) and (NNTMF is High) and (NNTsdMF is High) then (PE is Middle)
10. (NNGMF is Middle) and (NNTMF is Low) and (NNTsdMF is Low) then (PE is Middle)
11. (NNGMF is Middle) and (NNTMF is Low) and (NNTsdMF is Middle) then (PE is Middle)
12. (NNGMF is Middle) and (NNTMF is Low) and (NNTsdMF is High) then (PE is Middle)
13. (NNGMF is Middle) and (NNTMF is Middle) and (NNTsdMF is Low) then (PE is Middle)
14. (NNGMF is Middle) and (NNTMF is Middle) and (NNTsdMF is Middle) then (PE is Middle)
15. (NNGMF is Middle) and (NNTMF is Middle) and (NNTsdMF is High) then (PE is Middle)
16. (NNGMF is Middle) and (NNTMF is High) and (NNTsdMF is Low) then (PE is Middle)
17. (NNGMF is Middle) and (NNTMF is High) and (NNTsdMF is Middle) then (PE is Middle)
18. (NNGMF is Middle) and (NNTMF is High) and (NNTsdMF is High) then (PE is High)
19. (NNGMF is High) and (NNTMF is Low) and (NNTsdMF is Low) then (PE is High)
20. (NNGMF is High) and (NNTMF is Low) and (NNTsdMF is Middle) then (PE is High)
21. (NNGMF is High) and (NNTMF is Low) and (NNTsdMF is High) then (PE is High)
22. (NNGMF is High) and (NNTMF is Middle) and (NNTsdMF is Low) then (PE is High)
23. (NNGMF is High) and (NNTMF is Middle) and (NNTsdMF is Middle) then (PE is Middle)
24. (NNGMF is High) and (NNTMF is Middle) and (NNTsdMF is High) then (PE is Middle)
25. (NNGMF is High) and (NNTMF is High) and (NNTsdMF is Low) then (PE is High)
26. (NNGMF is High) and (NNTMF is High) and (NNTsdMF is Middle) then (PE is High)
27. (NNGMF is High) and (NNTMF is High) and (NNTsdMF is High) then (PE is High)
28. (NNGMF is Low) or (NNTMF is Low) or (NNTsdMF is Low) then (PE is Low)
29. (NNGMF is Middle) or (NNTMF is Middle) or (NNTsdMF is Middle) then (PE is Middle)
30. (NNGMF is High) or (NNTMF is High) or (NNTsdMF is High) then (PE is High)

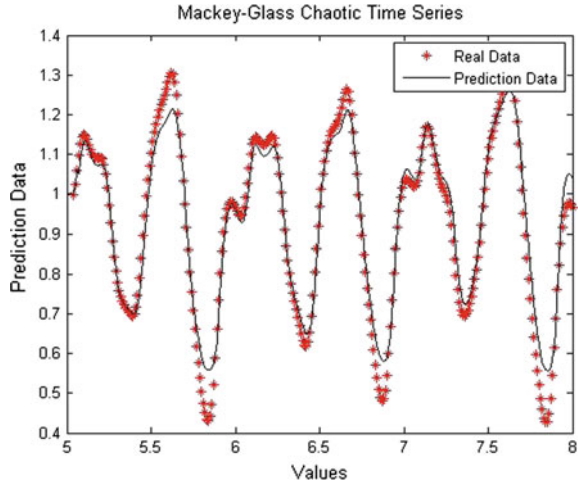
Fig. 15 Rules for the type-2 fuzzy integrator

Table 1 Results for the ensemble neural network with average integration for Mackey-Glass time series

No.	Prediction error
E1	0.0350
E2	0.0496
E3	0.0553
E4	0.0375
E5	0.0428
E6	0.0523
E7	0.0623
E8	<b>0.0346</b>
E9	0.0437
E10	0.0372
Average	0.0485

network with type-2 fuzzy weights (NNT2FW), the ensemble neural network with average integration (ENNAI) and the ensemble neural network with type-2 fuzzy integrator (ENNT2FI).

**Fig. 16** Graphic of real data again prediction data of ENNAI for the Mackey-Glass time series



**Table 2** Results for the ensemble neural network with the type-2 fuzzy integrator for time series Mackey-Glass

No.	Prediction error
E1	0.0290
E2	0.0429
E3	0.0539
E4	<b>0.0265</b>
E5	0.0428
E6	0.0488
E7	0.0340
E8	0.0555
E9	0.0306
E10	0.0666
Average	0.0561

**Table 3** Comparison results for the Mackey-Glass time series

Method	Prediction error
MNN [35]	0.0530
NNT2FW [35]	0.0390
ENNAI	0.0346
ENNT2FI	0.0265

## 6 Conclusions

In the experiments, we observe that using an ensemble neural network with average integration and type-2 fuzzy integrator, we can achieve better results than the monolithic neural network and the neural network with type-2 fuzzy weights for the

Mackey-Glass time series. The ensemble with type-2 fuzzy integrator presents better results in almost all the experiments than the optimization with PSO.

## REFERENCES

1. M. Cazorla, F. Escolano, Two Bayesian methods for junction detection. *IEEE Trans. Image Process.* **12**(3), 317–327 (2003)
2. G. Martinez, P. Melin, D. Bravo, F. Gonzalez, M. Gonzalez, Modular neural networks and fuzzy Sugeno integral for face and fingerprint recognition. *Adv. Soft Comput.* **34**, 603–618 (2006)
3. O. De Wilde, The magnitude of the diagonal elements in neural networks. *Neural Netw.* **10**(3), 499–504 (1997)
4. P.A. Salazar, P. Melin, O. Castillo, A new biometric recognition technique based on hand geometry and voice using neural networks and fuzzy logic, in *Soft Computing for Hybrid Intelligent Systems* (2008), pp. 171–186
5. V.V. Phansalkar, P.S. Sastry, Analysis of the back-propagation algorithm with momentum. *IEEE Trans. Neural Networks* **5**(3), 505–506 (1994)
6. O. Castillo, P. Melin, *Soft Computing for Control of Non-linear Dynamical Systems* (Springer, Heidelberg, Germany, 2001)
7. P. Melin, O. Castillo, *Hybrid Intelligent Systems for Pattern Recognition Using Soft Computing* (Springer, Heidelberg, 2005), pp. 2–3
8. L.A. Zadeh, Fuzzy sets. *J. Inf. Control* **8**, 338–353 (1965)
9. M. Okamura, H. Kikuch, R. Yager, S. Nakanishi, Character diagnosis of fuzzy systems by genetic algorithm and fuzzy inference, in *Proceedings of the Vietnam-Japan Bilateral Symposium on Fuzzy Systems and Applications*, Halong Bay, Vietnam (1998), pp. 468–473
10. W. Wang, S. Bridges, *Genetic Algorithm Optimization of Membership Functions for Mining Fuzzy Association Rules* (Department of Computer Science Mississippi State University, 2000)
11. J.S.R. Jang, C.T. Sun, E. Mizutani, *Neuro-fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence* (Prentice Hall, 1997)
12. O. Castillo, P. Melin, *Type-2 Fuzzy Logic Theory and Applications* (Springer, Berlin, 2008), pp. 29–43
13. J. Castro, O. Castillo, P. Melin, An interval type-2 fuzzy logic toolbox for control applications, in *FUZZ-IEEE* (2007), pp. 1–6
14. J. Castro, O. Castillo, P. Melin, A. Rodriguez-Diaz, Building fuzzy inference systems with a new interval type-2 fuzzy logic toolbox. *Trans. Comput. Sci.* **1**, 104–114 (2008)
15. D. Hidalgo, O. Castillo, P. Melin, Type-1 and type-2 fuzzy inference systems as integration methods in modular neural networks for multimodal biometry and its optimization with genetic algorithms, in *Soft Computing for Hybrid Intelligent Systems* (2008), pp. 89–114
16. D. Sanchez, P. Melin, Optimization of modular neural networks and type-2 fuzzy integrators using hierarchical genetic algorithms for human recognition, in *IFSA World Congress*, Surabaya, Indonesia, OS-414 (2011)
17. R. Sepúlveda, O. Castillo, P. Melin, A. Rodriguez, O. Montiel, Experimental study of intelligent controllers under uncertainty using type-1 and type-2 fuzzy logic. *Inf. Sci.* **177**(11), 2023–2048 (2007)
18. T.G. Barbounis, J.B. Theocharis, Locally recurrent neural networks for wind speed prediction using spatial correlation. *Inf. Sci.* **177**(24), 5775–5797 (2007)
19. T. Gedeon, Additive neural networks and periodic patterns. *Neural Netw.* **12**(4–5), 617–626 (1999)

20. M. Meltser, M. Shoham, L. Manevitz, Approximating functions by neural networks: a constructive solution in the uniform norm. *Neural Netw.* **9**(6), 965–978 (1996)
21. D. Yeung, P. Chan, W. Ng, Radial basis function network learning using localized generalization error bound. *Inf. Sci.* **179**(19), 3199–3217 (2009)
22. D. Casasent, S. Natarajan, A classifier neural net with complex-valued weights and square-law nonlinearities. *Neural Netw.* **8**(6), 989–998 (1995)
23. S. Draghici, On the capabilities of neural networks using limited precision weights. *Neural Netw.* **15**(3), 395–414 (2002)
24. R.S. Neville, S. Eldridge, Transformations of Sigma-Pi Nets: obtaining reflected functions by reflecting weight matrices. *Neural Netw.* **15**(3), 375–393 (2002)
25. J. Yam, T. Chow, A weight initialization method for improving training speed in feedforward neural network. *Neurocomputing* **30**(1–4), 219–232 (2000)
26. H. Ishibuchi, K. Morioka, H. Tanaka, A fuzzy neural network with trapezoid fuzzy weights, fuzzy systems, in *IEEE World Congress on Computational Intelligence*, vol. 1 (1994), pp. 228–233
27. H. Ishibuchi, H. Tanaka, H. Okada, Fuzzy neural networks with fuzzy weights and fuzzy biases, in *IEEE International Conference on Neural Networks*, vol. 3 (1993), pp. 160–165
28. M.T. Hagan, H.B. Demuth, M.H. Beale, *Neural Network Design* (PWS Publishing, Boston, 1996), p. 736
29. J. Castro, O. Castillo, P. Melin, A. Rodríguez-Díaz, A hybrid learning algorithm for a class of interval type-2 fuzzy neural networks. *Inf. Sci.* **179**(13), 2175–2193 (2009)
30. S. Kamarthi, S. Pittner, Accelerating neural network training using weight extrapolations. *Neural Netw.* **12**(9), 1285–1299 (1999)
31. T. Feuring, Learning in fuzzy neural networks, in *IEEE International Conference on Neural Networks*, vol. 2 (1996), pp. 1061–1066
32. J. Castro, O. Castillo, P. Melin, O. Mendoza, A. Rodríguez-Díaz, An interval type-2 fuzzy neural network for chaotic time series prediction with cross-validation and Akaike test, in *Soft Computing for Intelligent Control and Mobile Robotics* (2011), pp. 269–285
33. N. Karnik, J. Mendel, Applications of type-2 fuzzy logic systems to forecasting of time-series. *Inf. Sci.* **120**(1–4), 89–111 (1999)
34. R. Abiyev, A type-2 fuzzy wavelet neural network for time series prediction. *Lect. Notes Comput. Sci.* **6098**, 518–527 (2010)
35. F. Gaxiola, P. Melin, F. Valdez, O. Castillo, Interval type-2 fuzzy weight adjustment for backpropagation neural networks with application in time series prediction. *Inf. Sci.* **260**, 1–14 (2014)
36. O. Castillo, P. Melin, A review on the design and optimization of interval type-2 fuzzy controllers. *Appl. Soft Comput.* **12**(4), 1267–1278 (2012)
37. P. Melin, *Modular Neural Networks and Type-2 Fuzzy Systems for Pattern Recognition* (Springer, 2012), pp. 1–204
38. H. Hagrass, Type-2 fuzzy logic controllers: a way forward for fuzzy systems in real world environments, in *IEEE World Congress on Computational Intelligence* (2008), pp. 181–200
39. R. Sepúlveda, O. Castillo, P. Melin, O. Montiel, An efficient computational method to implement type-2 fuzzy logic in control applications, in *Analysis and Design of Intelligent Systems using Soft Computing Techniques* (2007), pp. 45–52
40. M.D. Monirul Islam, K. Murase, A new algorithm to design compact two-hidden-layer artificial neural networks. *Neural Netw.* **14**(9), 1265–1278 (2001)