Fuzzy Neural Network for Clustering and Classification

[1]Archana R. Shinde, [2] Prof .D.B Kshirsagar
[1] Student, [2] Professor,
[1] Department Computer Engineering,
[1] S.R.E.S. College of Engineering, Kopargaon,Maharashtra India.

***Abstract* - This paper presents the implementation of Fuzzy Neural Network (FNN) for clustering and classification. Fuzzy neural network combines the advantage of both fuzzy logic and neural network. This paper mainly focuses on implementation of two algorithms. First algorithm is General Fuzzy min max Neural Network (GFMM) training algorithm which combines the processing of supervised and unsupervised data in a single training algorithm. Second algorithm is Data Core Based Fuzzy min max Neural Network (DCFMN) which considers the characteristics of the data and influence of noise simultaneously. These two algorithms provide high accuracy, flexibility, better performance, strong robustness in classification and clustering.**

***Index Terms* - Classification, Clustering, Fuzzy systems, Fuzzy min-max neural networks ,Data core, Overlapped neuron, Pattern classification, Hyperboxes, Hyperbox expansion, Hyperbox contraction, Overlapping neurons, Classifying neurons ,Membership function, Robustness.**

I. INTRODUCTION

Now a days fuzzy logic and neural network are greatly used to develop intelligent systems .The main reason to combine fuzzy logic and neural network is that fuzzy logic have greater ability to handle approximate or uncertain information with the capability of neural networks in learning from processes. This paper presents implementation of two algorithms. One is General Fuzzy min-max Neural Network (GFMM)[4] and second is Data Core Based Fuzzy min-max Neural Network (DCFMN)[7].

General Fuzzy min-max Neural Network (GFMM)[4] processes supervised and unsupervised data in a single training algorithm. In labeled or supervised learning ,which is also referred as a pattern classification problem, labels are provided with input patterns. In unlabeled or unsupervised learning, also called as a cluster analysis problem, the input pattern is unlabeled and we must deal with the splitting a set of input patterns into a number of more or less homogenous clusters with respect to a suitable similar property. Patterns which are similar are allocated to the same cluster, while the patterns which differ are put in different clusters.

In many industrial processes, the measurement data may include noise caused by various kinds of processes which affects system performance and may decrease the accuracy of classification. Therefore, Data Core Based Fuzzy min max Neural Network (DCFMN) [7] algorithm is implemented to improve the performance of classification.

II. RELATED WORK

As we know recently a number of researches were conducted to explore the utilization of fuzzy neural network in clustering and classification. Based on the advantages of combining fuzzy logic and neural network, a fuzzy min–max neural network (FMNN) was proposed by Simpson for classification and clustering [1], [2]. FMNN was based on an aggregation of fuzzy hyperboxes which defined a region in an *n*-dimensional pattern space by its minimum and maximum points. The FMNN was used for classification by creating hyperboxes that belong to different classes. In [3], a new learning algorithm for the Simpson's fuzzy min–max neural network is presented. It overcomes some undesired properties of the Simpson's model: specifically, in it there are neither threshold that bound the dimension of the hyperboxes nor sensitivity parameters. In fact, the classification result does not depend on the presentation order of the patterns in the training set, and at each step, the classification error in the training set cannot increase. The [5] describes an approach to classification of noisy signals using a technique based on the fuzzy ARTMAP neural network (FAMNN).In [6] robust fuzzy neural network (RFNN) sliding-mode control based on computed torque control design for a two-axis motion control system is proposed. The two-axis motion control system is an x-y table composed of two permanent-magnet linear synchronous motors. First, a single- axis motion dynamics with the introduction of a lumped uncertainty including cross-coupled interference between the two-axis mechanisms is derived. Then, to improve the control performance in reference contours tracking, the RFNN sliding-mode control system is proposed to effectively approximate equivalent control of the sliding-mode control method .In [8] a fuzzy wavelet neural network (FWNN) models for prediction and identification of nonlinear dynamical systems are presented. The impressive generalization capability of the presented FWNN models is derived primarily from the use of wavelets and their ability to localize both in time and frequency domains. These models use wavelet functions in the consequent part of fuzzy rules, and with these functions the FWNN models have fast convergence and high precision. In [9], author study the guaranteed cost control problem for stochastic fuzzy systems with multiple time delays and uncertain parameters. In [10] two new learning algorithms for fuzzy min-max neural classifiers are proposed: the adaptive resolution classifier (ARC) and its pruning version(PARC). ARC/PARC generates a regularized min-max network by a succession of hyperbox cuts. The generalization capability of ARC/PARC technique mostly depends on the adopted cutting strategy. By using a recursive cutting procedure (RARC and R-PARC) it is possible to obtain better results.

III. SYSTEM ARCHITECTURE

This paper presents the implementation of a fuzzy neural network for clustering and classification. In this fuzzy neural network two training algorithm are implemented for clustering and classification. First algorithm is General Fuzzy min max Neural Network (GFMM)[4] and second is Data Core Based Fuzzy min max Neural Network (DCFMN)[7]. In GFMM single training algorithm is required to processes labeled and unlabeled input patterns. In this algorithm learning process is completed in a few passes through the data and of placing and adjusting the hyperboxes in the pattern space which is referred as an expansion–contraction process. To retaining all the interesting features, a number of modifications to their definition have been made in order to accommodate fuzzy input patterns in the form of lower and upper bounds, combine the labeled and unlabeled input patterns, and improve the effectiveness of system , but in GFMM new data can be included without retraining of data. DCFMN considers the characteristics of the data and influence of noise simultaneously while performing classification. In DCFMN , membership function has been designed based on the geometric center and data core in a hyperbox. Instead of using the contraction process, overlapped neuron with new membership function based on the data core is proposed and added to neural network to represent the overlapping area of hyperboxes belonging to different classes. DCFMN has strong robustness and high accuracy in classification taking onto account the effect of data core and noise. The general architecture of the proposed system is shown in figure.1

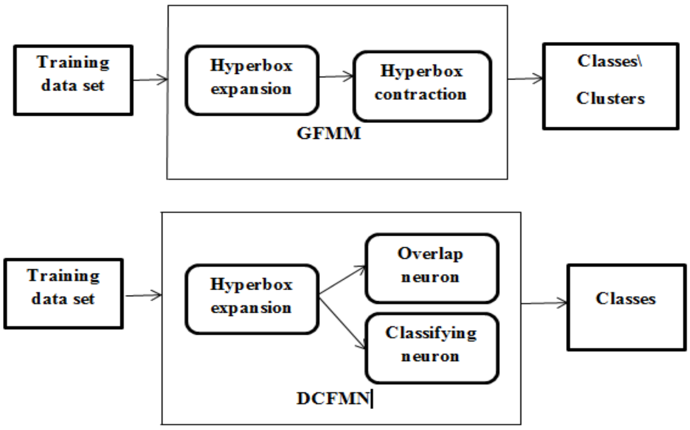

Figure.1 General System Architecture

A. General Fuzzy min max neural network (GFMM) Algorithm

GFMM algorithm consist of mainly four steps which are Initialization, Expansion, Overlap Test, and Contraction.

Basic Terms :

First we see some basic terms that are used in this algorithm

1. Input:

Input provided to the GFMM algorithm is in the form of ordered pair as

$$\{X_{h,}d_h\} \qquad (1)$$

where X_h is the h^{th} input pattern and $d_h \in \{1,2,\ldots,p\}$ is the index of one of the p+1 classes, where d_h =0 means that the input pattern is unlabeled.

2. Fuzzy Hyperbox Membership Function:

Fuzzy Hyperbox Membership Function is very important in fuzzy min max neural network algorithm. The decision whether the given input pattern belongs to a particular class or cluster, thus whether the corresponding hyperbox is to be expanded, depends mainly on the membership value describing the degree to which an input pattern fits within the hyperbox .In GFMM algorithm the membership function of j^{th} hyperbox for h^{th} input pattern is calculated as

$$b_j(X_h)=\min_{i=1..,n}(\min([1-f(x_{hi}^u-w_{ji},\gamma_i)], [1-f(v_{ji}-x_{hi}^l,\gamma_i)])) \qquad (2)$$

$$\text{Where } f(r,\gamma)=\begin{cases}1 & if\ r\gamma>1\\ r\gamma & if\ 0\le r\gamma\le 1\\ 0 & if\ r\gamma<0\end{cases}$$

Where x_{hi}^l and x_{hi}^u is the lower and upper limit of h^{th} input pattern, v_{ji} and w_{ji} is the min and max points of j^{th} hyperbox and threshold function; γ=[$\gamma1,\gamma2,\ldots,\gamma n$]—sensitivity parameters which regulates how fast the membership values decrease.

GFMM Learning Algorithm

1. Initialization : Initially min point Vj and max point Wj of j^{th} hyperbox set to

$$V_j = 0 \ \ and \ \ W_j = 0 \qquad (3)$$

When first input pattern is presented j^{th} hyperbox adjusted and min and max points set to

$$V_j \ = \ x_h^l \ and \ \ W_j = x_h^u \qquad (4)$$

2. Hyperbox Expansion : When the input pattern is presented, find the hyperbox with the highest degree of membership and allowing expansion (if needed) is expanded to include input pattern . Expansion criteria is

$$\forall_{i=1\ldots n}(\max(w_{ji}, x_{hi}^{u}) - \min(v_{ji}, x_{hi}^{l})) \ \leq \theta \qquad (5)$$

And

if $d_h = 0$ then adjust B_j

else

$$\text{if class}(B_j) = \begin{cases} 0 \quad \Rightarrow \text{adjust } B_j \\ \text{class}(B_j) = d_h \\ d_h \quad \Rightarrow \text{ adjust } B_j \\ \quad \text{take} \\ \text{else} \Rightarrow \text{another } B_j \end{cases} \qquad (6)$$

We adjust B_j in following way.

$$v_{ji}^{new} = \min(v_{ji}^{old}, x_{hi}^{l}) \ \text{ for each } i = 1, \ldots, n$$
$$w_{ji}^{new} = \max(w_{ji}^{old}, x_{hi}^{u}) \ \text{ for each } i = 1, \ldots, n$$

If none of the existing hyperboxes can expand to include the input pattern x_h , then a new hyperbox B_k is created, adjusted, and labeled by setting $class(B_k)$= d_h.

The θ is a user-defined parameter that define a bound on the maximum size of a hyperbox and its value significantly affects the effectiveness of the training algorithm.

3. Hyperbox Overlap Test: Due to inclusion of input patterns hyperboxes of different classes are overlapped with each other .So to detect overlapping between hyperboxes overlap test is performed as follows

- If the hyperbox, expanded in the last expansion step ,is not labeled then test for overlapping with all the other hyperboxes. This ensures that all unlabeled hyperboxes do not overlap with any of the other existing ones.
- If the hyperbox B_j expanded in the last expansion step,belongs to one of the existing classes then test for the overlap only with the hyperboxes not being part of the same class as B_j. Notice that this allows to overlap the hyperboxes belonging to the same class.

The full hyperbox overlap test can be therefore summarized as follows.

$$\text{class}(B_j) = \begin{cases} 0 \qquad \Rightarrow \text{ test overlapping with} \\ \qquad all\ other\ hyperboxes \\ else \Rightarrow \ test\ overlapping\ only\ if \\ \qquad class(B_j) \neq class(B_k) \end{cases} \qquad (7)$$

The four cases are being considered (where initially ∂^{old}=1).

Case 1:$v_{ji} < v_{ki} < w_{ji} < w_{ki}$

$$\partial^{new} = \min(w_{ji} - v_{ki}, \partial^{old}).$$

Case 2:$v_{ki} < v_{ji} < w_{ki} < w_{ji}$

$$\partial^{new} = \min(w_{ki} - v_{ji}, \partial^{old}).$$

Case 3:$v_{ji} < v_{ki} \leq w_{ki} < w_{ji}$

$$\partial^{new} = \min(\min(w_{ki} - v_{ji}, w_{ji} - v_{ki}), \partial^{old}).$$

Case 4:$v_{ki} < v_{ji} \leq w_{ji} < w_{ki}$

$$\partial^{new} = \min(\min(w_{ki} - v_{ji}, w_{ji} - v_{ki}), \partial^{old}).$$

If overlap for the i^{th} dimension has been detected (one of the above four cases is valid) and $\partial^{old} - \partial^{new} > 0$, then Δ=i, ∂^{old}= ∂^{new} and $case = l$ (l= {1,2,3,4}− the case for which the smallest overlap was found). If overlap for the i^{th}dimension has not been detected, set Δ=−1 which indicate that the contraction step is not necessary.

4. Hyperbox Contraction: If Δ>0 then only the Δ_{th} dimensions of the two hyperboxes are adjusted. For contraction again four cases are being considered.

Case 1: $v_{j\Delta} < v_{k\Delta} < w_{j\Delta} < w_{k\Delta}$

$$v_{k\Delta}^{new} = w_{j\Delta}^{new} = \frac{v_{k\Delta}^{old} + w_{j\Delta}^{old}}{2}$$

or alternatively ($w_{j\Delta}^{new} = v_{k\Delta}^{old}$).

Case 2: $v_{k\Delta} < v_{j\Delta} < w_{k\Delta} < w_{j\Delta}$

$$v_{j\Delta}^{new} = w_{k\Delta}^{new} = \frac{v_{j\Delta}^{old} + w_{k\Delta}^{old}}{2}$$

or alternatively ($v_{j\Delta}^{new} = w_{k\Delta}^{old}$).

Case 3: $v_{j\Delta} < v_{k\Delta} \le w_{k\Delta} < w_{j\Delta}$

$if\ w_{k\Delta} - v_{j\Delta} < w_{j\Delta} - v_{k\Delta}$

$then\ v_{j\Delta}^{new} = w_{k\Delta}^{old}$ otherwise ($w_{j\Delta}^{new} = v_{k\Delta}^{new}$)

Case 4: $v_{k\Delta} < v_{j\Delta} \le w_{j\Delta} < w_{k\Delta}$

$if\ w_{k\Delta} - v_{j\Delta} < w_{j\Delta} - v_{k\Delta}$

$then\ w_{k\Delta}^{new} = v_{j\Delta}^{old}$ otherwise ($v_{k\Delta}^{new} = w_{j\Delta}^{old}$).

B. Data Core Based Fuzzy min max Neural Network (DCFMN) Algorithm:

In DCFMN a new membership function for classifying the neuron of DCFMN is defined in which the noise, the geometric center of the hyperbox, and the data core are considered.

1. CN Membership Function: For considering the influence of noise and the density of data in the hyperbox, an improved hyperbox membership function of CN is defined as follows:

$$b_j(x_h) = \min_{i=1,..,n}(\min(f(x_{hi} - w_{ji} + \varepsilon, c_{ji}), f(v_{ji} + \varepsilon - x_{hi}, c_{ji}))) \quad (8)$$

where ε is a parameter representing noise, c is difference between the data core in the hyperbox and the geometric center of the corresponding hyperbox, and f is the ramp threshold function which is defined as.

$$f(r,c) = \begin{cases} e^{-r^2 \times (1+c) \times \lambda}, & r > 0, c > 0 \\ e^{-r^2 \times (1-c) \times 1/\lambda}, & r > 0, c < 0 \\ 1, & r < 0 \end{cases}$$

2. OLN Membership Function: An OLN is created when there is an overlapping area between two hyperboxes that belong to two different classes V', W' nodes in OLN represent the minimum and maximum points of the overlapped area. The activation function used for OLN is given by

$$d_{o,q}(x_h) = g(v'_o, w'_o, x_h, y_q)$$

$$= \begin{cases} \frac{1}{n} \sum_{i=1}^{n} (1 - |x_{hi} - y_{qi}|), \forall_{i=1,..n} w_{oi} > (x_{hi}) > v_{oi} \\ 0, \end{cases} \quad (9)$$

where o = 1, 2, . . . , l is the index of overlapped node, q =1, 2 is the index of hyperbox which has the same overlap, and y_q is the data core of each overlapped hyperbox.

DCFMN Learning Algorithm

DCFMN algorithm consist of three steps as follows:

1. Hyperbox Expansion: Identify the expandable hyperboxes and expand them. For the hyperbox to be expanded, the following expansion criteria must be met:

$$\forall_{i=1,..,n, j=1,..,m}(\max(w_{ji}, x_{hi}) - \min(v_{ji}, x_{hi})) \le \theta \quad (10)$$

If the expansion constraint condition is met in DCFMN, the minimum and maximum points of the hyperbox are adjusted as follows:

$$v_{ji}^{new} = \min\left(v_{ji}^{old}, x_{hi}^{l}\right) \quad for\ each\ i = 1, \dots, n$$
$$w_{ji}^{new} = \max\left(w_{ji}^{old}, x_{hi}^{u}\right) \quad for\ each\ i = 1, \dots, n$$

2. Hyperbox Overlap Test : Due to inclusion of input patterns two hyperboxes are overlapped with other. So detect overlapping between hyperbox of different classes overlap test is performed. Assuming that $\alpha^{old} = 1$ initially, the four test cases and the corresponding minimum overlap value for the i^{th} dimension is as follows.

Case 1: $v_{ji} < v_{ki} < w_{ji} < w_{ki}$

$$\alpha^{new} = \min(w_{ji} - v_{ki}, \alpha^{old}).$$

Case 2: $v_{ki} < v_{ji} < w_{ki} < w_{ji}$

$$\alpha^{new} = \min(w_{ki} - v_{ji}, \alpha^{old}).$$

Case 3: $v_{ji} < v_{ki} \le w_{ki} < w_{ji}$

$$\alpha^{new} = \min(\min\left(w_{ki} - v_{ji}, w_{ji} - v_{ki}\right), \alpha^{old}).$$

Case 4: $v_{ki} < v_{ji} \le w_{ji} < w_{ki}$

$$\alpha^{new} = \min(\min\left(w_{ki} - v_{ji}, w_{ji} - v_{ki}\right), \alpha^{old}).$$

If $\alpha^{old} - \alpha^{new} > 0$, then there is an overlap for the i^{th} dimensions, and let $\Delta = i$. Otherwise, the test stops and set $i^{th} = 0$, and nothing is done.

3. Adding OLN - If an overlap between hyperboxes of different classes exists, the process of adding OLN is performed as follows.

Case1: $v_{ji} < v_{ki} < w_{ji} < w_{ki}$

$$v'_{m+oi} = v_{ki} \ , w'_{m+oi} = w_{ji}$$

Case 2: $v_{ki} < v_{ji} < w_{ki} < w_{ji}$

$$v'_{m+oi} = v_{ji} \ , w'_{m+oi} = w_{ki}.$$

Case 3: $v_{ji} < v_{ki} \le w_{ki} < w_{ji}$

$$v'_{m+oi} = v_{ki} \ , w'_{m+oi} = w_{ki}$$

Case 4: $v_{ki} < v_{ji} \le w_{ji} < w_{ki}$

$$v'_{m+oi} = v_{ji} \ , w'_{m+oi} = w_{ji}$$

where $o = (1, 2, \dots, l)$ is index of neuron to be added to the network, and V' and W' are used to store the minimum and maximum points of the hyperboxes.

IV. IMPLEMENTATION DETAILS

This system mainly presents implementation of two algorithm, GFMM and DCFMN and our objective is to implements these two algorithm for standard datasets iris and wine .GFMM is four step algorithm which are Initialization, Expansion, Overlap Test, and Contraction which are successfully completed and gives desired output.

Figure 2.shows browsing of dataset as input for our system .Iris dataset is browsed which contains 150 records of three classes .All input patterns are labeled.

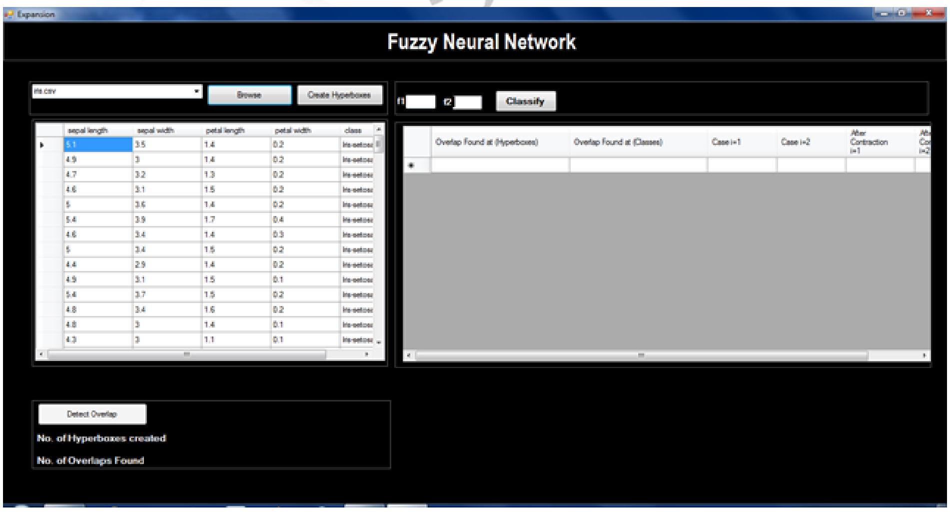

Figure 2: Browsing dataset

Figure 3. shows that number of hyperboxes created of different classes , here user defined parameter θ set 0.1 and number of hyperboxes created is 139. As we increase value of θ from 0.1 to 0.9 ,number of hyperboxes created is decreases.

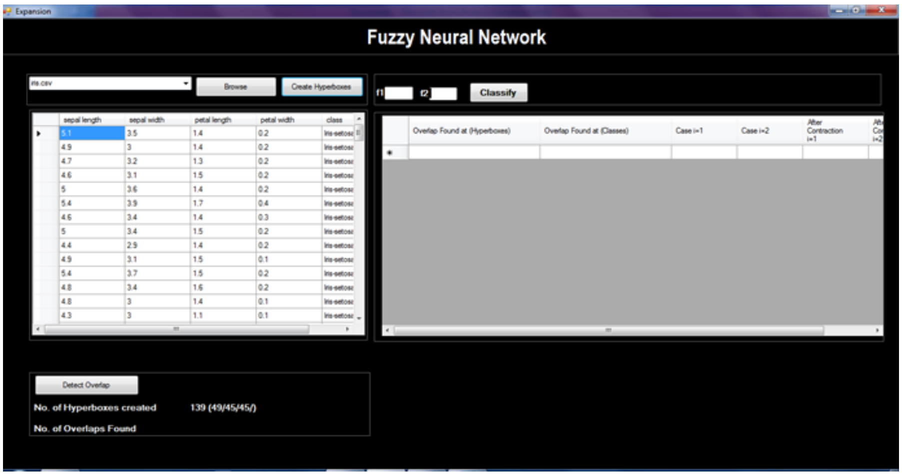

Figure 3: Number of Hyperboxes created.

Figure 4. Shows overlapping found between hyperboxes and contraction of hyperboxes performed to remove overlapping of hyperboxes of different classes.

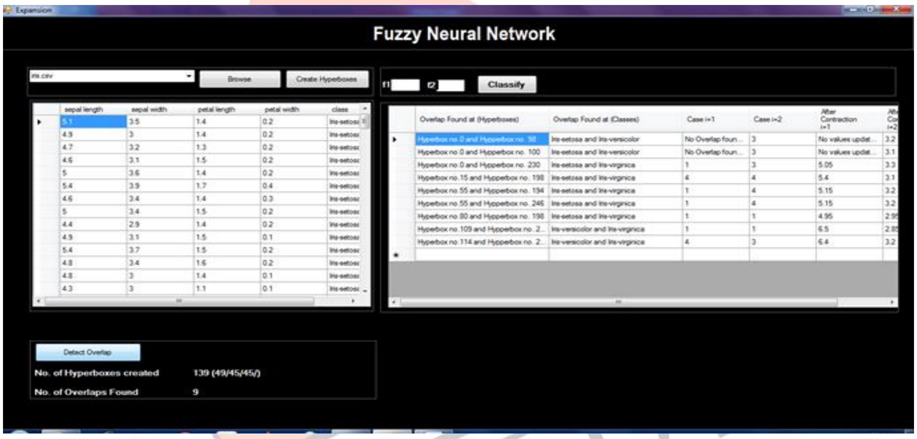

Figure 4: Overlap test and contraction

Up to this step training of algorithm is completed , now testing is performed by presenting input pattern to the system and system classify that input pattern correctly and displaying class of that input pattern which shown in figure 5. Misclassification rate of GFMM algorithm is very less.

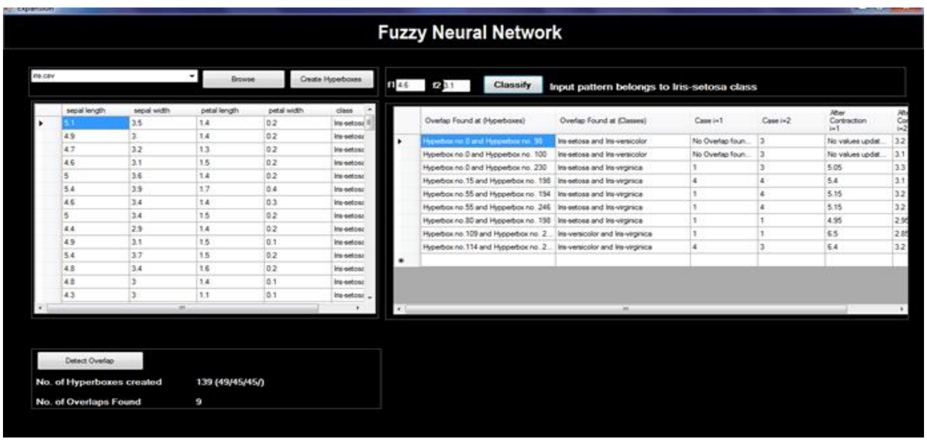

Figure 5: Classify input pattern

V. CONCLUSION

This paper presents the implementation of fuzzy neural network for clustering and classification which uses two training algorithm for clustering and classification. These two training algorithm is based on fuzzy logic and neural network. First algorithm is General Fuzzy min max Neural Network which combines the processing of supervised and un supervised input data within a single training algorithm and provide better performance. Second algorithm is Data Core Based Fuzzy min max Neural Network which considers the characteristics of the data and influence of noise simultaneously and provide strong robustness against noise and better performance in classification

VI. ACKNOWLEDGMENT

I would like to take this opportunity to express my profound gratitude and deep regard to my Project Guide Prof. D. B. Kshirsagar, for his exemplary guidance, valuable feedback and constant encouragement throughout the duration of the project. His valuable suggestions were of immense help throughout my project work. His perceptive criticism kept me working to make this project in a much better way. Working under him was an extremely knowledgeable experience for me.

REFERENCES

[1] P. K. Simpson, “Fuzzy min-max neural networks-part I: Classification,” IEEE Trans. Neural Netw., vol. 3, no. 5, pp. 776–786, Sep. 1992.

[2] P. K. Simpson, “Fuzzy min-max neural network-part II: Clustering,” IEEE Trans. Fuzzy Syst., vol. 1, no. 1, pp. 32–45, Feb. 1993.

[3] M. Meneganti, F. S. Saviello, and R. Tagliaferri, “Fuzzy neural networks for classification and detection of anomalies,” IEEE Trans. Neural Netw., vol. 9, no. 5, pp. 846–861, Sep. 1998.

[4] B. Gabrys and A. Bargiela, "General fuzzy min-max neural network for clustering and classification," IEEE Trans. Neural Netw., vol. 11, no. 3, pp. 769-783, May 2000.

[5] D. Chralampidis , T. Kasparis, and G. Meorgiopoulos, "Classification of noisy signals using fuzzy ARTMAP neural networks " , IEEE Trans. Neural Netw., vol. 12, no. 5, pp. 1023-1036, Sep. 2001.

[6] F. J. Lin and P. H. Shen, "Robust fuzzy neural network sliding-mode control for two-axis motion control system" , IEEE Trans. Ind. Electron., vol. 53, no. 4, pp. 1209-1225, Aug. 2006.

[7] Huaguang Zhang, Jinhai Liu , Dazhong Ma,and Zhanshan Wang , "Data Core-Based Fuzzy Min-Max Neural Network for Pattern Classification", IEEE Trans. vol. 22, no. 12, Dec. 2011.

[8] S. Yilmaz and Y. Oysal, “Fuzzy wavelet neural network models for prediction and identification of dynamical systems ,” IEEE Trans. Neural Netw., vol. 21, no. 10, pp. 1599–1609, Oct. 2010.

[9] H. G. Zhang, C. Y. Wang, and D. Liu, “Delay-dependent guaranteed cost for uncertain stochastic fuzzy systems with multiple time delays,” IEEE Trans. Syst., Man Cybern., Part B, vol. 38, no. 1, pp. 126–140, Jan. 2008.

[10] A. Rizzi, M. Panella, and F. M. F. Massciloi, Adaptive resolution min max classifiers,IEEE Trans. Neural Netw., vol. 13, no. 2, pp.402414, Mar. 2002.[28] M. Meneganti, F. S. Saviello.