MATEMATIKA, 2008, Volume 24, Number 2, 141–148

An Improved Fast Training Algorithm for RBF Networks Using Symmetry-Based Fuzzy C-Means Clustering

[1]**Lim Eng Aik &** [2]**Zarita Zainuddin**
[1]Institute of Engineering Mathematics, University Malaysia Perlis, Jejawi, 02600 Arau, Perlis, Malaysia
[2] School of Mathematical Sciences, Universiti Sains Malaysia, 11800 USM, Pulau Pinang, Malaysia
e-mail: [1]ealim@unimap.edu.my

Abstract In fuzzy C-means (FCM) clustering, each data point belongs to a cluster to a degree specified by a membership grade. FCM partitions a collection of vectors in c fuzzy groups and finds a cluster center in each group such that the dissimilarity measure is minimized. This paper presents a training algorithm for the radial basis function (RBF) network using symmetry-based Fuzzy C-means (SFCM) clustering method which is the modified version of FCM clustering method based on point symmetry distance measure. The training algorithm which uses SFCM clustering method to train the network has a number of advantages such as faster training time, more accurate predictions and reduced network architecture compared to the standard RBF networks. The proposed training algorithm has been implemented in the RBF networks created by the *newrb* function of MATLAB which uses gradient based iterative method as learning strategy, therefore the new network will undergo a hybrid learning process. The networks called Symmetry-based Fuzzy C-means Clustering–Radial Basis Function Network (SFCM/RBF) has been tested against the standard RBF network and the networks called standard Fuzzy C-means Clustering (FCM)-RBF network (FCM/RBF) in forecasting. The experimental models has been tested on three real world application problems, particularly in Air pollutant problem, Biochemical Oxygen Demand (BOD) problem, and Phytoplankton problem.

Keywords Fuzzy c-means clustering; SFCM; Radial basis function network; point symmetry distance; forecasting.

1 Introduction

Radial Basis Function (RBF) networks form a class of Artificial Neural Networks (ANNs), which has certain advantages over other types of ANNs, such as better approximation capabilities, simpler network structures and faster learning algorithms. Due to the popularity of RBF networks, there are several researchers who have been working to develop more efficient training algorithms, compared to the standard techniques [1].

RBF networks takes a period of time to train the networks when it comes to a massive number of training data, yet produce a high error due to possible invalid data in the training data. Even though a combination of clustering methods in RBF networks has been proven by Sarimveis et al. [2] to be faster in training, it still produces a larger error. This is due to the standard clustering algorithms which still lack the ability to choose the most accurate

and informative centers. By using Symmetry-Based Fuzzy C-Means (SFCM) clustering method, we are able to improve the accuracy of the problem stated above. As we know, the more accurate the centers are chosen, the more accurate the information that feeds to the train network. This leads to more accurate results.

In this paper, a fast algorithm for training RBF networks which produces high accuracies is presented, which selects the input centers using the SFCM method. The improved of the convergence rate of proposed method is due to the fact that it does not involve the formulation and solution of a nonlinear optimization problem, while it requires only one pass of the training data.

The methodology is illustrated through the application of the experimental models by forecasting the pollutant trend at Forth Worth City, Texas with air quality data from Texas Resource Conservation Commission database, BOD concentration and Phytoplankton growth and death rates, both with data from Yogan [3]. The advantages of the presented learning strategy (SFCM/RBF) are identified and the results are compared with standard RBF networks and FCM/RBF.

2 The Point Symmetry Distance

Obviously, the similarity measure between the patterns or objects is defined by the Euclidean distance, as follows:

$$d(j,k) = \left(\sum_{i=1}^{p} |x_{ji} - x_{ki}|^2 \right)^{1/2} \tag{1}$$

where x_{ji} and x_{ki} are the jth and kth patterns respectively and p is the dimension of the patterns. The jth and kth patterns are said to be more similar to each other, if the value $d(j,k)$ is small enough and close to zero. Otherwise, they are said to be dissimilar. The FCM clustering algorithm that uses the Euclidean distance, can hyper spherical-shaped clusters. However, it fails to detect clusters, which are developed along the principle axes. By applying point symmetry distance [4] to define the similarity measure between two patterns, the above draw back can be overcomed.

The point symmetry distance has been successful used in pattern classification and object recognition [4]. It is a non-metric distance and is defined as follows:

Given N patterns, $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$, i = 1, 2, ..., N and a reference point (cluster center) c, the point symmetry distance between x_i and c is defined as

$$d_s(x_i, c) = \min_{i=1,\ldots,N, i \neq j} \frac{\|(x_j - c) + (x_i - c)\|}{\|x_j - c\| + \|x_i - c\|} \tag{2}$$

A pattern is assigned to a cluster for which its point symmetry distance is minimum.

3 Overview of Fuzzy C-Means Clustering Method

The Fuzzy C-means (FCM) algorithm was originally developed by Dunn [12], and later generalised by Bezdek [9]. To describe the algorithm, we set some notations. The set of all points considered is $X = \{x_1, x_2, \ldots, x_n\}$. We write $u_i : X \to [0,1]$ for the ith cluster, $i = 1, \ldots, c$, and we will use u_{ik} to denote $u_i(x_k)$, i.e. the grade of membership of x_k in

cluster u_i. We use $U = \langle u_{ik} \rangle$ for the matrix of all membership values. The midpoint of u_i is v_i, and is computed according to

$$v_i = \frac{\sum_{k=1}^{n} U_{ik}^m \, x_k}{\sum_{k=1}^{n} U_{ik}^m} \tag{3}$$

A parameter m, $1 \leq m < \infty$, will be used as a weighted exponent, and the particular choice of value m is application dependent [10]. For $m = 1$, it coincides with the FCM algorithm, and if $m \to \infty$, then all u_{ik} values tend to $1/c$ [11]. Membership in fuzzy clusters must fulfil the condition

$$\sum_{i=1}^{c} U_{ik} = 1, \forall \, k = 1, ..., n \tag{4}$$

i.e. for each $x \in X$, the sum of memberships of x in respective c_i must be 1. A typical distance measure $\|\cdot\|$ is the Euclidean distance.

The objective of the clustering algorithm is to select u_i so as to minimize the objective function

$$J = \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m \, \|x_k - v_i\|^2 . \tag{5}$$

The following is the algorithm for FCM clustering:

(i) Fix c and m. Initialise U to some $U^{(1)}$. Select $\varepsilon > 0$ for stopping condition.

(ii) Update midpoint values v_i for each cluster c_i.

(iii) Compute the set $\mu_k = \{i : 1 \leq i \leq c : \|x_k - v_i\| = 0\}$, and update $U^{(\ell)}$ and (3) according to the following: if $\mu_k = \phi$, then $\mu_{ik} = 1 \Big/ \left[\sum_{j=1}^{c} (\|x_k - v_i\| / \|x_k - v_j\|)^2 \right]$, otherwise $\mu_{ik} = 0, \forall i \notin \mu_k$ and $\sum_{i \in \mu_k} \mu_{ik} = 1$. ($\phi \in \{i : 1 \leq i \leq c : \|x_k - v_i\| = 0\}$).

(iv) Stop if $J < \varepsilon$, otherwise go to step 2.

In Step 1, $c(\geq 1)$ is set to a fixed number of clusters. In the rule generation phase, each cluster will be the basis for one rule. Usually we keep c as small as possible in order to keep the number of rules within reasonable bounds. Further, the matrix $U = \langle u_{ik} \rangle$ is to be initialised. A crisp, and even random, partition of X into c subsets can be sufficient to provide a good starting point for the algorithm.

In Step 2, midpoint values v_i are computed, and respective midpoints will of course move towards points with higher membership values in their clusters. Note that a midpoint can coincide with some x_λ. In such a case we will have $u_{i\lambda} = 1$, and then, for all $\xi \neq i$, we will have $u_{\xi\lambda} = 0$.

Step 3 is the core of the algorithm. There, membership values u_{ik} are updated. Note that we must distinguish between cases depending on whether or not midpoints coincide with data points. The variable ℓ denotes the iteration number.

In Step 4, we compute the difference between present and previous matrices of membership values. If the stopping condition is met, we are done.

4 The Proposed SFCM/RBF Training Method

An RBF network can be considered as a three layer network. The input nodes pass the input values to the connection arcs. The internal units form a single layer of L RBF nodes (the Gaussian function was used in this work), which have localized response functions in the input space. The hidden node responses are weighted and the output nodes are simple summations of the weighted responses. The formulation of the training algorithm involves a set of input-output pairs [*x(i), y(i)*], *i = 1,...,K*, where $x(i)$ is the N-dimensional input vector, $y(i)$ is the corresponding target or desired M-dimensional output vector and K is the number of training examples. The set of input-output examples is the information base, which is used to determine the values of the unknown parameters, i.e. the hidden node centers and radii and the connection weights between the hidden and the output layer.

An approach to simplify the original input data set is the crucial step in developing a successful RBF networks model [5]. The innovation in this work is the proposed algorithm which is used for selecting the most significant input centers, based on the symmetry based Fuzzy C-Means clustering method. The rest of the network parameters are calculated using standard methods.

SFCM algorithm is the modified version of FCM algorithm from Section 3, but the modification is only done in Step 2 and Step 4 of FCM algorithm while the rest of the steps of SFCM algorithm remain the same as FCM algorithm. Modification of Step 2 of the FCM algorithm is done by added a minimum-value criterion while finding the cluster centers using point symmetry distance from Section 2. Meanwhile, for Step 4 of FCM algorithm, two extra stopping criterions are added. The SFCM algorithm is presented in Section 4.

The SFCM algorithm [8] is presented as follows:

Step 1: Initialization. Randomly choose K data points from the data set to initialize K cluster centers, $c_1, c_2, \ldots c_k$.

Step 2: Fine-Tuning. For pattern $\underline{x}$, find the cluster centers nearest it in the symmetrical sense. That is, find the cluster centers $k*$ nearest to the input pattern $\underline{x}$ using the minimum-value criterion:

$$k* = Arg \min_{k=1,\ldots,K} d_s(x, c_k) \tag{6}$$

where the point symmetry distance $d_s(\underline{x}, \underline{c_k})$ is computed by (2). If the point symmetry distance $d_s(\underline{x}, \underline{c_k})$ is smaller than a pre-specified parameter θ, then assign the data point $\underline{x}$ to the $k*$-th cluster. Otherwise, the data point is assigned to the cluster center $k*$ using the following criterion:

$$k* = Arg \min_{k=1,\ldots,K} d(x, c_k) \tag{7}$$

where $d(\underline{x}, \underline{c_k})$ is the Euclidean distance between the input pattern and the cluster center c_k.

Step 3: Updating. Compute a new membership matrix U using (8) and then compute the new centers of the K clusters using (9). The update rule is:

$$U_{ij} = \frac{1}{\sum\limits_{k \in S_k(t)} (d_{ij}/d_{kj})^2} \tag{8}$$

$$c_k(t+1) = \frac{\sum\limits_{i \in S_k(t)} U_{ij} x_i}{\sum\limits_{i \in S_k(t)} U_{ij}} \tag{9}$$

where $S_k(t)$ is the set whose elements are the patterns assigned to the kth cluster at time t.

Step 4: Continuation. If $J < \theta$ or there are no patterns change categories, or the number of iterations has reached a pre- specified maximum number, then stop. Otherwise, go to Step 2.

Here we use the value of $\theta = 0.05$ given by Su & Chau [4].

5 Results

The proposed methodology was tested by forecasting the pollutant trend at Forth Worth City, Texas [6], BOD concentration and Phytoplankton growth rate and death rate [3]. The experimental result for SFCM/RBF, standard RBF network and FCM/RBF on the pollutant trend at Forth Worth City, Texas, BOD concentration and Phytoplankton growth rate and death rate, will be shown at the end of this section. For the pollutant trend at Forth Worth City problem, the training set consists of 480 sets of air data and the test set comprises of 72 sets of air data which both were taken from hourly air data. Meanwhile, for BOD concentration problem and Phytoplankton growth and death rates problem, both training set consists of 100 sets of data and the test set comprises of 100 sets of data [3].

The experiment was implemented by using the *newrb* function because it represents the general form of a RBF network. Furthermore, the proposed clustering method has been implemented by using MATLAB's function. Gaussian basis function has been used for both networks with other parameters such as spread was set to default value, so that the performance of the proposed network can be evaluated effectively [8]. Performance of SFCM/RBF, standard RBF network and FCM/RBF in this experiment has been measured by comparing the computation time taken for training with number of iteration taken for convergence and the Root Mean Squared Error (RMSE) to measure how well both networks approximates the chosen functions and it is given by

$$RMSE = \sqrt{\frac{\sum\limits_{i=1}^{n} (t_i - y_i)^2}{n}}$$

where t is target output, y_i is actual output, and n is total number of data.

The number of centers chosen for both FCM clustering method and SFCM clustering method are based on [6, 7, 8] where the taken number of centers can perform the best result compared to other number of centers. Here we chose the number of center for both clustering method as 417 for air pollutant problem, 53 for Phytoplankton problem and 13 for BOD problem. The database of air quality monitored at Forth Worth city, Texas of United States, and data from [3] were selected to test the developed SFCM/RBF network model. For air pollutant problem, the type of pollutant monitored includes carbon monoxide, nitric oxide, nitrogen dioxide, ozone, and oxides of nitrogen. For experimental purposes, hourly updated air quality data obtained from Texas Natural Resource Conservation Commission's

homepage has been used to predict the trend of interested pollutants for Nitric Oxide, Nitrogen Dioxide and Oxides of Nitrogen. While for Phytoplankton problem, growth rate and death rate have been used as the interested values. As for the BOD problem, the BOD concentration has been taken as the interested value.

Results from Table 1, Table 2 and Table 3 have shown that SFCM/RBF networks approximate the chosen functions very well and it outperform the Standard RBF network and FCM/RBF network in the experiments. Our results in Table 1, Table 2 and Table 3 do not included the number of iterations due to *newrb* function in MATLAB by default generates number of iteration which contain the same amount as the number of center generated from clustering method.

From Table 1, SFCM/RBF network surpasses the standard RBF and FCM/RBF network in terms of accuracy, learning speed and the network architecture by using training set which consists only 417 centers compared to 480 centers because significant data have been chosen as center successfully. This means that, it is possible to find a number of centers such that it will provide a network with reduced complexity, faster training time yet improved accuracy.

From Table 2, result shows that SFCM/RBF network once again outperform both the standard RBF and FCM/RBF network in term of accuracy. Even using only half of total 100 centers, it was able to perform such satisfying result. Finally, result from Table 3 shows that SFCM/RBF network are able to show better accuracy and training time compare with both FCM/RBF network and Standard RBF network even with the least number of center which is only 13 centers.

Table 1: Performance for SFCM/RBF network, FCM/RBF network and Standard RBF network prediction results for air pollutant problem.

Method	NO		NO_2		NOx	
	CPU time (s)	RMSE	CPU time (s)	RMSE	CPU time (s)	RMSE
FCM/RBF	51.609	0.4819	52.719	0.3741	51.828	0.4804
SFCM/RBF	51.478	0.1601	51.540	0.2118	51.368	0.3732
Standard RBF network	93.281	0.1875	93.890	0.3909	94.218	0.4776

Generally, both SFCM/RBF network and Standard RBF network performed well in the experiments. The SFCM/RBF network is superior in terms of learning speed and the architecture of the network but it requires a proper value of number of centers for determining the number of input centers.

Reduced number of training set take less computation time and it means shorter training time compared to actual number of training set. As we can see from Table 1, the computation time is reduced to about 40%.

The results above show that less centers for training would use less computation time but produce larger errors compared to actual training set, because there is a significant loss of information which is represented by the original data especially when we try to reduce it into smaller sizes. However, due to the SFCM clustering method able to provide more accurate

Table 2: Performance comparison for SFCM/RBF network, FCM/RBF network and Standard RBF network prediction results for Growth rate and Death rate in Phytoplankton problem.

Method	Growth rate, Gr		Death Rate, Dr	
	CPU time (s)	RMSE	CPU time (s)	RMSE
FCM/RBF	0.203	0.1297	0.203	0.1630
SFCM/RBF	0.203	0.0455	0.187	0.0783
Standard RBF network	1.031	0.5443	1.000	0.6106

Table 3: Performance comparison for SFCM/RBF network, FCM/RBF network and Standard RBF network prediction results for BOD concentration in BOD problem.

Method	BOD concentration	
	CPU time (s)	RMSE
FCM/RBF	0.094	0.4502
SFCM/RBF	0.078	0.4481
Standard RBF network	1.313	0.4642

center for the network, the accuracy and learning speed of the network improve and even outperform the standard RBF itself. Furthermore, a large training set does not guarantee desirable accuracies because it might contain much invalid data that could jeopardize the desired accuracy, not mentioning the size of network it would create and the time taken for training.

There is no denial on the learning speed of the SFCM/RBF network, but it comes with a hefty compensation for the accuracy if the proper value of number of centers is not selected. As the number of centers for the network becomes lesser and it results much simpler network architecture and faster training time. Although the three models provide good results, the network structure, learning speed and accuracy of the SFCM/RBF network is superior compared to the standard RBF network and FCM/RBF network.

6 Conclusion

Experiments and a real world problem have been simulated in this paper, where we applied on a real case study on forecasting for air pollution problem. The performance of these network has been compared to the case using the training time and Root Mean Squared Error (RMSE) as the criteria for performance measurement.

Results from the experiments and case studies show that the SFCM/RBF network is better than the standard RBF in the context of learning speed and network architecture. SFCM/RBF network also better than FCM/RBF in the context of accuracy. It is possible to improve the accuracy of the proposed network by using statistical methods to choose the best value of number of center to be used. As conclusion, the proposed network is

far more superior to the standard RBF network when it comes to learning speed, network architecture and accuracy.

Since self-organized selection of centers which can be performed by clustering algorithms to select meaningful centers for the hidden nodes has been used, it would be interesting if the network would be tested with noisy training data to verify the efficiency of the chosen clustering algorithm.

References

[1] J. Moody & C. Darken, *Fast learning in networks of locally tuned processing units,* Neural Computing, 1(2)(1989), 281-294.

[2] H. Sarimveis, A. Alexandridis & G. Bafas, *A fast training algorithm for RBF networks based on subtractive clustering,* Neurocomputing, 51(2003), 501-505.

[3] Yogan S/O Jaya Kumar, *Neuro-Fuzzy system*, Master Dissertation, Universiti Sains Malaysia, Pulau Pinang, 2005.

[4] Mu-Chun Su & Chien-Hsing Chou, *A modified version of k-means algorithm with a distance based on cluster symmetry,* IEEE Trans. Pattern Analysis and Machine Intelligent, 23, No. 6(2001), 674-680.

[5] W.Z. Lu, W.J. Wang, X.K. Wang, S.H. Yan & J.C. Lam, *Potential assessment of a neural network model with PCA/RBF approach for forecasting pollutant trends in Mong Kok urban air, Hong Kong*, Journal of Environmental Research, 96(2004), pp. 79-87.

[6] Calen Woi, *Radial Basis Function Neural Networks,* Dissertation submitted for the degree of Master of Science in Mathematics, USM, Pulau Pinang, 2005.

[7] Zarita Zainuddin, Lim Eng Aik & Calen Woi, *Subtractive Clustering for training Radial Basis Function Networks,* ICOQSIA2005 Conference Proceedings, UUM, Malaysia, 2005.

[8] Lim Eng Aik, *Modified Clustering Algorithms for Radial Basis Function Networks,* Master Dissertation, Universiti Sains Malaysia, Pulau Pinang, 2006.

[9] J.C. Bezdek, *Cluster validity with fuzzy sets,* Journal of Cybernetics 4(1974), 58-73.

[10] J.C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms,* Plenum, New York, 1981.

[11] H. Choe & J.B. Jordan, *On the Optimal Choice of Parameters in a Fuzzy C-means Algorithm,* Proc. IEEE International Conference on Fuzzy Systems, San Diego, 1992, 349-354.

[12] J.C. Dunn, *A Fuzzy Relative of ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters,* Journal of Cybernetics 3(1973), 32-57.