Automatic Clustering of Generalized Regression Neural Network By Similarity Index based Fuzzy C-Means Clustering

Hafizah Husain**, Marzuki Khalid*, and Rubiyah Yusof*
*Center For Artificial Intelligence and Robotics (CAIRO),
Faculty of Electrical Engineering,
Universiti Teknologi Malaysia, Jalan Semarak,
54100 Kuala Lumpur, Malaysia
(*For all correspondence: marzuki@utmkl.utm.my)
**Department of Electrical, Electronics and Systems,
Faculty of Engineering,
UniversitiKebangsaan Malaysia,
43600 UKM Bangi, Selangor, Malaysia

Abstract— In general regression neural networks (GRNN), one drawback is that the number of training vectors is proportional to the number of hidden nodes, thus a large number of training vectors will produce a larger architecture, which is a major disadvantage for many applications. In this paper we proposed an efficient clustering technique referred to as 'similarity index fuzzy c-means clustering'. This technique uses the conventional fuzzy c-means clustering preceded by a technique based on similarity indexing to automatically cluster input data which are relevant to the system. The technique employs a one-pass similarity measures on the data to calculate the similarity index. This index indicates the degree of similarity in which data will be clustered. Similar data then undergoes fuzzy c-means iterative process to determine their cluster centers. We applied the technique for system identification and modeling and found the results to be encouraging and efficient.

Index Terms— similarity measures, fuzzy c-means, optimal clustering.

I. INTRODUCTION

The generalized regression neural network or GRNN was proposed by Donald Specht as an alternative to the well-known back-error propagation training algorithm for feedforward neural networks [1]. It is closely related to the better-known probabilistic neural network (PNN) [2], which has been mainly used for pattern recognition purposes. In the GRNN proposed by Specht in [3], the number of training vectors is proportional to the number of hidden nodes, thus a large number of training vectors will produce a larger architecture, which is a major disadvantage for many applications. One way of reducing the hidden nodes in the GRNN is through the use of clustering techniques, however, there has not been much attempt on such approach. Two of the authors in this paper have attempted an alternative approach in reducing the number of hidden nodes in GRNN as described in [4], in which the hidden nodes are added-on and pruned off during training, however, this does not ensure a judicious structure for many applications.

In this paper, we proposed an automatic clustering technique which we referred to as the "similarity index based fuzzy c-means clustering" technique that can result in a more optimal GRNN structure. The strategy results in a smaller network even with a very large training data. The optimized GRNN is demonstrated on two commonly used benchmark problems in dynamic system identification and modeling.

This paper has been organized as follows. In the next section we briefly describe the original GRNN approach. In Section III we discuss the proposed similarity index based fuzzy c-means clustering technique. Simulation results show that the performance of the proposed strategy applied on two benchmark problems is better than two other existing clustering techniques. This is discussed in Section IV. We conclude the approach in the section that follows.

II. GENERALIZED REGRESSION NEURAL NETWORK (GRNN)

GRNN is based on the estimation of a probability density function (PDF) from observed data samples using Parzen window estimates [5]. Supposing x(vector) and y(scalar) are random variables, and X and Y are measured values, $f(x,y)$ is defined as the joint continuous probability density function. If $f(x,y)$ is known, it is easy to estimate the expected value of y (the regression of y on X) such that:

$$E(y|X) = \frac{\int_{-\infty}^{\infty} y f(X,y)\,dy}{\int_{-\infty}^{\infty} f(X,y)\,dy} \qquad (1)$$

0-7803-8560-8/04/$20.00©2004IEEE

The estimated value $\hat{Y}(x)$ is an exponentially weighted average value of all observed values Y^i given as:

$$\hat{Y}(x) = \frac{\sum_{i=1}^{n} Y^i \exp(-\frac{D_i^2}{2\sigma^2})}{\sum_{i=1}^{n} \exp(-\frac{D_i^2}{2\sigma^2})} \quad (2)$$

where D_i is defined as

$$D_i^2 = (X - X^i)^T (X - X^i) \quad (3)$$

The smoothing parameter, σ controls the generalization factor to determine how close the estimate is made to fit the data. If σ is large, the estimates will be smooth while smaller σ, allows the estimate to closely fit the data.

Based on the original GRNN method, the number of hidden nodes is therefore proportional to the number of observed data samples. A large X will result in a large number of hidden nodes. In this case, we identify a representative for a group of input-output data that is necessary to optimize the GRNN.

Through clustering, X^i will then represent the values of the kernel centers, C^j, where j=1,….,p and p is the number of clusters. And thus substituting this into (3) yields

$$D_j^2 = (X - C^j)^T (X - C^j) \quad (4)$$

Therefore, by substituting (4) in (2), the estimate output $\hat{Y}(x)$ is defined as

$$\hat{Y}(x) = \frac{\sum_{i=1}^{p} Y^j \exp(-\frac{D_j^2}{2\sigma^2})}{\sum_{i=1}^{p} \exp(-\frac{D_j^2}{2\sigma^2})} \quad (5)$$

where Y^j is the kernel centers for the output data.

III. SIMILARITY INDEX FUZZY C-MEANS CLUSTERINGTECHNIQUE

Fuzzy C-means clustering technique in [6] has shown that input-output data pair can be grouped into clusters. A center of each cluster will then be determined to represent a compact description of data that constitute strong intrinsic similarity with partitioned samples in disjoint subset. However, the selection of centers is restricted among densely populated region while forfeiting the less populated area.

In our approach, we first use a simple similarity measurement to calculate the similarity index among the input data. The index is compared to the similarity threshold level, γ_{th} to determine whether the data is in the vicinity of each other. The data which are in the vicinity of each other will be clustered and will be further trained using the conventional fuzzy c-means method to select the center for its cluster. This approach thus effectively reduces the training time compared to the conventional fuzzy c-means method. Fuzzy c-means method is considered because it shows good performance in reducing clustering problems such as local minima, center redundancy and dead center. Its self-organizing capability to merge homogeneous clusters gives an added advantage.

A. Data Clustering using the Similarity Index Technique

A simple one-pass similarity measure process is carried out to indicate the similarity index, γ_{in} between the data. This index determines the degree of similarity based on the neighborhood function according to its Euclidean squared distance using the equation below

$$\gamma_{in}^{ij} = exp(-\left\| x^i - x^j \right\|^2) \quad (6)$$

where x is the input vector, for i and j=1,2,…,m.

If $\gamma_{in} \leq \gamma_{th}$, where γ_{th} is the threshold level of similarity index, it indicates that both data is not within vicinity of each other and thus regarded as 'dissimilar'. If the similarity index of only one datum lies above the threshold level, then it is unique. This datum will be assigned as the representation of that particular cluster. Figure 1 illustrates how the clustering algorithm is implemented.

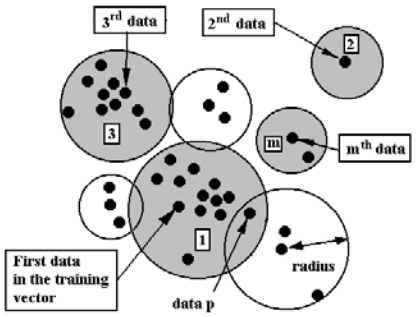

Fig. 1. Example of clustering using the similarity index based fuzzy c-means clustering technique

The similarity indices of the first data and other training vector are calculated. Indices higher than the threshold level, γ_{th} will be considered as similar and thus forming Cluster 1 as shown in Fig. 1. Data p belongs to Cluster 1 since it is recognized as being similar to the first data. The center of the cluster is then decided by using fuzzy c-means, which is

discussed in the next section. The radius of the cluster is inversely proportionate to the threshold level; therefore the radius of the cluster can be estimated by varying the threshold level.

B. Fuzzy C-Means Algorithms (FCM)

Fuzzy c-means algorithm uses the reciprocal of distances to decide the cluster centers and this representation reflects the distance of a feature vector from the cluster center and the similarities between the data. It is an iterative algorithm used to divide N number of data $\boldsymbol{x}^j$ into $\boldsymbol{c}$ cluster sets by finding the degrees of membership $\mu_{jk} \in [0,1]$ and cluster centers $\underline{v}^k$ to minimize the objective function J,

$$J = \sum_{j=1}^{N} \sum_{k=1}^{c} \left(\mu_{jk}\right)^m \left\| x^j - \underline{v}^k \right\|^2 \qquad (7)$$

where m>1 is the parameter that determines the overlap factor of the clusters. The number of clusters, $\boldsymbol{c}$ determines the number of rules that will be used to form the premise part of the if-then rules in the fuzzy system. $\boldsymbol{x}^j$ for j= 1,2,..,N is the input-output training data pairs and $\underline{v}^k = [v_1^k, v_2^k, \ldots, v_n^k]^T$ for k=1,2,..,c are the cluster centers. μ_{jk} for j= 1,2,..,N and k=1,2,..,c is the degrees of membership of $\boldsymbol{x}^j$ in the kth cluster while $\left\| x^j - \underline{v}^k \right\|^2$ is the Euclidean norm. By minimizing the objective function, a more optimal cluster centers will then be resulted. More information on this technique can be found in [7].

III. SIMULATION EXAMPLES

In order to show the effectiveness of the proposed technique, we simulated the technique on two benchmark problems and compared it with two other existing clustering techniques, namely, the conventional fuzzy c-means clustering technique and the Kohonen self-organizing map clustering algorithm [8]. All these techniques can be used to determine the kernels of the GRNN. Two examples are discussed as follows.

A. Example 1

The efficiency of the optimized GRNN is simulated on the gas furnace data of Box and Jenkins [9]. This data consists of 296 input-output measurements where the input is the gas flow rate into the furnace and the output is the CO_2 concentration in outlet gas. It is frequently used as a benchmark example for testing identification algorithms.

The input variables are chosen as u(t), u(t-1), y(t) and y(t-1) while the output is y(t+1). All 296 of the observed data samples were used to calculate the centers and the number of clusters using the proposed approach. Various similarity threshold values, γ_{th} resulting in various number of clusters were observed. The calculated centers and the number of clusters were used to define the parameters of GRNN.

The result for GRNN-based modeling of Box-Jenkins Furnace data using the three different clustering techniques is shown in Figure 2 for 145 clusters. Figure 2 (a) compares the observed output with the output of the GRNN using the 3 clustering techniques as mentioned and Figure 2 (b) shows the modeling error of the trained data.

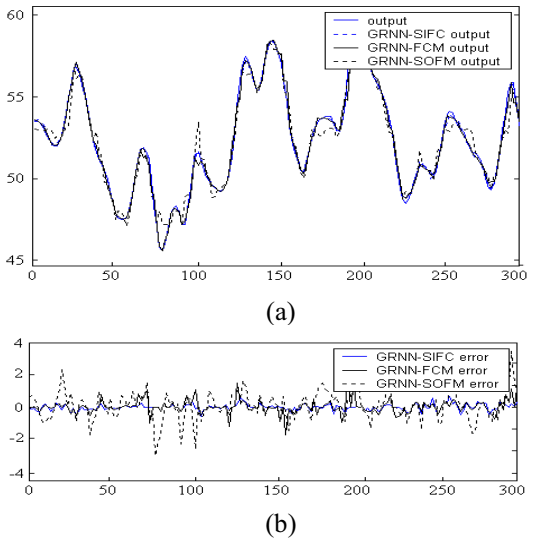

Fig. 2. Box-Jenkins Data (a) training results and (b) modeling error based on the GRNN modeling using the 3 clustering techniques.

Table 1 compares the mean squared error of the GRNN modeling performance based on the 3 clustering techniques identified.

Table 1. MSE results based on three different clustering techniques

Compression Methods	MSE
Proposed	0.0703
Conventional Fuzzy C-means	0.4358
Self-Organizing Map	0.5740

B. Example 2

Another commonly used benchmark problem in testing identification algorithm is the Mackey-Glass delay differential equation. The equation was proposed as a model for the production of white blood cells (WBC) [10] that produces a chaotically evolving continuous dynamic system. The Mackey-Glass model for the production of WBC is given by the following equations

$$\frac{dx}{dt} = F(x_\tau) - Bx = \frac{0.2x(t-\tau)}{1+x^{10}(t-\tau)} - 0.1x(t) \qquad (8)$$

where x is the density of the circulating WBC, B=0.1 is the random WBC destruction rate and the function F is the *current* flux of new WBC into the blood in

response to the demand created at a time τ in the *past* (Note, the notation x_τ stands for x(t - τ), that is, the *x* value τ unit time in the past).

Based on the time series data generated for 1000 data points in [10], 500 is used as training (for data compression process) while another 500 is used for testing. The input variables are chosen as [x(t-18), x(t-12), x(t-6) and x(t)] while the output is x(t+6). Various similarity threshold values, γ_{th} resulting in various number of hidden nodes is observed. The MSE for the GRNN using all observed samples as the hidden nodes is 0.005 for smoothing factor, σ =0.1.

The result for the prediction of 500 test data for Mackey-Glass WBC production with chaotic behavior is shown in Figure 3. Figure 3 (a) compares the observed output with the output of GRNN with various data compression techniques while Figure 3 (b) shows the modeling error of the tested data.

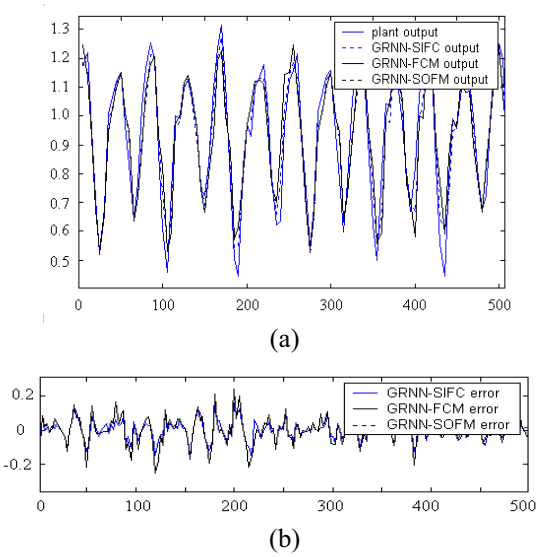

Fig. 3. Mackey-Glass Time Series Data
(a) training results (b) modeling error

Table 2 shows the mean squared error of the proposed method in comparison with the conventional fuzzy c-means and self-organizing clustering for 121 hidden nodes.

Table 2. Results based on three different clustering techniques used on the GRNN.

Compression Methods	MSE
SIFC	0.0062
FCM	0.0077
SOFM	0.0084

IV. CONCLUSIONS

We proposed a more efficient clustering technique to determine a more optimal GRNN structure. This rapid learning algorithm is suitable for on-line dynamic GRNN-based modeling. Compared to the clustering strategies based on fuzzy c-means and self-organizing map, the proposed technique is able to model more accurately as shown in the simulated results on two dynamic time series.

REFERENCES

[1] Specht, D.F.,Shapiro, P.D., *Generalization accuracy of probabilistic neural networks compared with backpropagation networks,* Neural Networks, 1991., IJCNN-91-Seattle International Joint Conference on , Vol 1, pp. 887 –892, July 1991

[2] Specht, D.F., *Probabilistic neural networks for classification, mapping, or associative memory Neural Networks*, 1988., IEEE International Conference on, vol. 1, pp. 525 –532, July 1988

[3] Specht D.F., *A General Regression Neural Network*, IEEE Transaction On Neural Network s, Vol. 2, No. 6, pp 568-576, November 1991

[4] Teo L.S.,Khalid M., and Yusof R., *Adaptive GRNN for the Modeling Of Dynamic Plants*, Proceedings of the 2002 IEEE International Symposium on Intelligent Control, pp 217-222, October 2002

[5] Parzen E., *On Estimation Of A Probability Density Function And Mode*, Ann. Math Statistics, Vol. 33, pp 1065-1076, 1962

[6] Razaz, M., *A fuzzy C-means clustering placement algorithm*, Circuits and Systems, 1993., ISCAS '93, 1993 IEEE International Symposium on, vol.3, pp: 2051 –2054, 3-6 May 1993

[7] Bobrowski, L.; Bezdek, *J.C., C-means Clustering With The l_1 and l_∞ Norms*, Systems, Man and Cybernetics, IEEE Transactions on , Volume: 21 Issue: 3, pp: 545 -554, May-June 1991

[8] Kohonen T., *The Self-Organizing Map,* Proceedings of the IEEE, International Conference on Knowledge-Based Intelligent Electronic Systems, pp. 33-40, April 1998

[9] Box, G.E.P, and Jenkins, G.M.,*Time Series Analysis, Forecasting And Control*, Holden Day, B.G., San Francisco, 1970

[10] Mackey M.C., and Glass J., Oscillation And Chaos In Physiological Control Systems, Journal Of Science, pp. 197-287, 1997