Clustering Algorithms for Radial Basis Function Neural Network

Sneha P. Munnoli & A. U. Bapat

Department of Computer Science and Engineering,
Goa College of Engineering, Ponda, Goa
E-mail : pmsneha@rediff.com,

Abstract - **Radial basis function neural networks (RBFNN) form a class of artificial neural networks, which has certain advantages over other types of neural networks including better approximation capabilities, simple network structure, and faster learning. RBFNN training is mainly based on the estimation of various parameters: centres, widths, connecting weights between neurons etc. Clustering algorithms are used in the estimation of centre parameter. We introduce two algorithms i.e. K-means and fuzzy clustering algorithms for RBFNN centre parameter selection.**

Keywords - Radial Basis Function Neural Networks; Fuzzy clustering; K-means clustering; Clustering; Artificial neural network

I. INTRODUCTION

An artificial neural network, often just called a neural network, is a mathematical model inspired by biological neural networks. A neural network consists of an interconnected group of artificial neurons, and it processes information using a connectionist approach to computation. In most cases a neural network is an adaptive system that changes its structure during a learning phase. Neural networks are used to model complex relationships between inputs and outputs or to find patterns in data. In an artificial neural network, simple artificial nodes, called "neurons", "neurodes", "processing elements" or "units", are connected together to form a network which mimics a biological neural network.

There is no single formal definition of what an artificial neural network is. Generally, it involves a network of simple processing elements that exhibit complex global behaviour determined by the connections between the processing elements and element parameters. It can have two learning paradigms; supervised and unsupervised learning.

Radial basis function neural networks (RBFNN) form a class of artificial neural networks, which has certain advantages over other types of neural networks including better approximation capabilities, simple network structure, and faster learning. In order to make use of RBFNN we need to specify some parameters like the hidden unit activation functions, centres, weights, widths etc. RBF networks have been applied in various applications [1,6].

II. RADIAL BASIS FUNCTION NEURAL NETWORK

The idea of Radial Basis Function (RBF) Networks derives from the theory of function approximation. RBF Networks take a slightly different approach. Their main features are:

1. They are two-layer feed-forward networks.
2. The hidden nodes implement a set of radial basis functions (e.g. Gaussian functions).
3. The output nodes implement linear summation functions as in an MLP.
4. The network training is divided into two stages: first the weights from the input to hidden layer are determined, and then the weights from the hidden to output layer.
5. The training/learning is very fast.
6. The networks are very good at interpolation.

Radial basis functions are powerful techniques for interpolation in multidimensional space. A RBF is a function which has built into a distance criterion with respect to a center. RBF networks have two layers of processing: In the first, input is mapped onto each RBF in the 'hidden' layer. In regression problems the output layer is then a linear combination of hidden layer values

representing mean predicted output. The interpretation of this output layer value is the same as a regression model in statistics. In classification problems the output layer is typically a sigmoid function of a linear combination of hidden layer values, representing a posterior probability. The process of finding RBF weights is called network training [5].

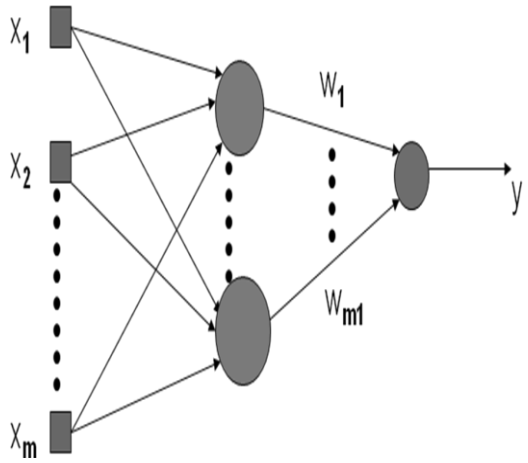

Fig. 1 : Structure of RBFNN

The RBF chosen is usually a Gaussian,

$$h(x)=\exp((-(x-c)^2)/(r^2)) \quad (1)$$

where c is the center and r is the radius. A Gaussian RBF monotonically decreases with distance from the centre. In contrast, a multiquadric RBF which, in the case of scalar input monotonically increases with distance from the centre. Gaussian-like RBFs are local (give a significant response only in a neighbourhood near the centre) and are more commonly used than multiquadric-type RBFs which have a global response.

III. CLUSTERING

Clustering involves the task of dividing data points into homogeneous classes or clusters so that items in the same class are as similar as possible and items in different classes are as dissimilar as possible. Clustering can also be thought of as a form of data compression, where a large number of samples are converted into a small number of representative prototypes or clusters. A loose definition of clustering could be "the process of organizing objects into groups whose members are similar in some way". A cluster is therefore a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters [2, 3].

A. K-Means clustering

K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem [4]. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids shoud be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more.

Finally, this algorithm aims at minimizing an *objective function*, in this case a squared error function. The objective function,

$$J = \sum_{j=1}^{k} \sum_{i=1}^{x} \left\| x_i^{(j)} - c_j \right\|^2 \quad (2)$$

where $\left\| x_i^{(j)} - c_j \right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre c_j, is an indicator of the distance of the *n* data points from their respective cluster centres.

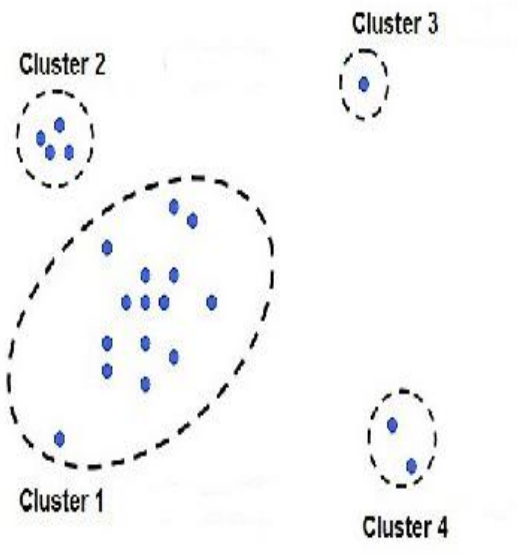

Fig. 2: K-Means clustering

The algorithm is composed of the following steps:

1. Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
2. Assign each object to the group that has the closest centroid.
3. When all objects have been assigned, recalculate the positions of the K centroids.
4. Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centres. The k-means algorithm can be run multiple times to reduce this effect.

B. Fuzzy clustering

In non-fuzzy or hard clustering, data is divided into crisp clusters, where each data point belongs to exactly one cluster. In fuzzy clustering, the data points can belong to more than one cluster, and associated with each of the points are membership grades which indicate the degree to which the data points belong to the different clusters.

Fuzzy clustering methods, however, allow the objects to belong to several clusters simultaneously, with different degrees of membership [3, 4]. In many situations, fuzzy clustering is more natural than hard clustering. Objects on the boundaries between several classes are not forced to fully belong to one of the classes, but rather are assigned membership degrees between 0 and 1 indicating their partial membership. The discrete nature of the hard partitioning also causes difficulties with algorithms based on analytic functional, since these functional are not differentiable.

Fuzzy c-means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters. This method is frequently used in pattern recognition. It is based on minimization of the following objective function:

$$J_m = \sum_{j=1}^{N}\sum_{i=1}^{C} u_{ij}^{m} \left\|x_i - c_j\right\|^2 \qquad 1 \le m < \infty \qquad (3)$$

where m is any real number greater than 1, u_{ij} is the degree of membership of x_i in the cluster j, x_i is the ith of d-dimensional measured data, c_j is the d-dimension center of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the center.

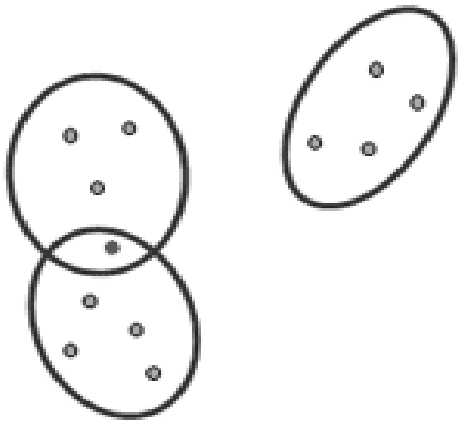

Fig. 3 : Fuzzy clustering

Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership u_{ij} and the cluster centres c_j by:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{\left\|x_i - c_j\right\|}{\left\|x_i - c_k\right\|}\right)^{\frac{2}{m-1}}}, \qquad (4)$$

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}} \qquad (5)$$

This iteration will stop when,

$\max_{ij}\left\{\left|u_{ij}^{(k+1)} - u_{ij}^{(k)}\right|\right\} < \varepsilon$, where ε is a termination criterion between 0 and 1, whereas k is the iteration steps. This procedure converges to a local minimum or a saddle point of J_m. The algorithm is composed of the following steps:

1. Initialize $U=[u_{ij}]$ matrix, $U^{(0)}$
2. At k-step: calculate the centers vectors $C^{(k)}=[c_j]$ with $U^{(k)}$
3. Update $U^{(k)}$, $U^{(k+1)}$
4. If $\| U^{(k+1)} - U^{(k)}\| < \varepsilon$ then STOP; otherwise return to step 2.

In this way, clustering algorithms can be used to find the center parameter of the RBFNN, which is an efficient way of finding the centers. Various other clustering algorithms can also be tried out for further research.

IV. ACKNOWLEDGMENT

This work was carried out as a part of my thesis, and we would like to thank other colleagues of Goa College of engineering for their help and support.

V. CONCLUSION

Radial basis function is a class of artificial neural network which has many advantages over other networks. In order to perform accurately it makes use of some parameters and accurate selection of these parameters increases its performance. We make use of clustering algorithms in order to select the centre parameter of the network and these clustering algorithms are k-means and fuzzy clustering. Similarly other clustering algorithms can also be used and worked on.

VI. REFERENCES

[1] "Introduction of the Radial basis network", by Adrian G. Bors.

[2] "Survey of Clustering Algorithms" by Rui Xu and Donald Wunsch II.

[3] "A survey of fuzzy clustering" by M.S. Yang.

[4] "Data Mining" a text book by Jewei Han

[5] "On the training of radial basis function classifiers", by Musavi, M.T., Ahmed, W., Chan, K.H., Faris, K.B., Hummels, D.M..

[6] "Neural networks and radial basis functions in classifying static speech patterns", by Niranjan, M., Fallside, F.

◈◈◈