2nd Hellenic Conf. on AI, SETN-2002, 11-12 April 2002, Thessaloniki, Greece,
Proceedings, Companion Volume, pp. 283-292

A Novel Algorithm for Training Radial Basis Function Networks

K. Koutroumbas[1], A. Pouliakis[2]

[1] Institute of Space Applications and Remote Sensing, National Observatory of Athens, Metaxa and V. Paulou, Palaia Penteli, 152 36 Athens, Greece, Tel. No: +30-1-6138342, Fax No: +30-1-6138343, E-mail: koutroum@space.noa.gr.
[2] Department of Informatics, University of Athens, Panepistimioupolis, T.Y.P.A. Buildings, 15781 Athens, Greece, E-mail: makis@di.uoa.gr

Abstract. In this paper a new method for training Radial Basis Function (RBF) networks, called RBF-MST, is introduced. The novelty of the method is in the way the first layer is constructed. This is carried out using a clustering algorithm that relies on the idea of the Minimum Spanning Tree (MST) and takes into account the classes where the data vectors belong. The second layer nodes are trained using the Least Mean Square (LMS) algorithm. The performance of RBF networks produced by the proposed method is assessed through a real medical application, where the aim is the classification of a cell nucleus as benign or malignant. Finally, the paper concludes with a discussion about ways for improving the performance of the proposed method.

KEYWORDS: Neural Networks, Radial Basis Function networks, Minimum Spanning Tree, Cytological Application.

1 Introduction

Radial Basis Function (RBF) networks is a well known class of neural networks that have drawn significant attention in the recent years. Their architecture is shown in fig. 1. They consist of two layers of nodes. The first layer consists of nodes each one implementing a function of the form

$$y_i = f(||\boldsymbol{x} - \boldsymbol{w_i}||), \quad i = 1, \ldots, k \tag{1}$$

where $\boldsymbol{x}$ is the input vector of the node and $\boldsymbol{w_i}$ its parameter vector. Typical choices for f are

$$y_i = \exp(-\frac{1}{2\sigma_i^2}||\boldsymbol{x} - \boldsymbol{w_i}||^2), \tag{2}$$

where σ_i^2 corresponds to the diameter of the receptive region of the node, and

$$y_i = \frac{\sigma^2}{\sigma^2 + ||\boldsymbol{x} - \boldsymbol{w_i}||^2}. \tag{3}$$

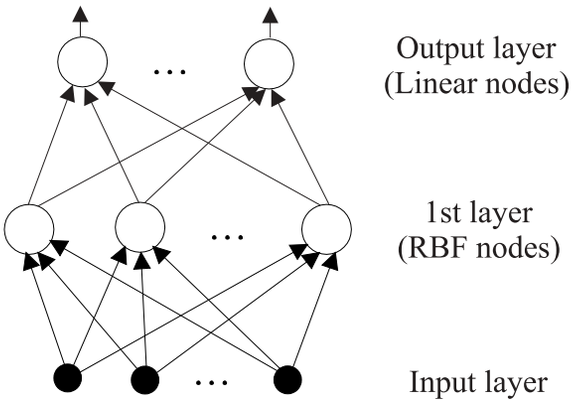

Fig. 1. Architecture of an RBF network

The output layer consists of linear nodes, i.e. nodes that implement the function

$$z_j = \boldsymbol{v_j}^T \boldsymbol{y}, \quad j = 1, \ldots, m, \tag{4}$$

where $\boldsymbol{y}$ is the vector containing the outputs of the k first layer nodes, augmented by an additional coordinate which is set to 1 and $\boldsymbol{v_j}$, the $(k+1)$ dimensional parameter vector of the j-th output node.

RBF networks possess universal approximation capabilities, that is there exists an RBF network with sufficient number of nodes in the first layer that can approximate a given continuous function defined on a compact set $S \subset \mathcal{R}^l$, with arbitrary accuracy (see e.g. [9], [16], [17]).

In the sequel we consider RBF networks in the frame of classification applications. In this case, the number of the output nodes equals to the number of the classes an input vector can be assigned.

One of the major issues in RBF networks is that of *training*, i.e. the determination of the parameters $\boldsymbol{w_i}$, σ_i^2, $i = 1, \ldots, k$ and $\boldsymbol{v_j}$, $j = 1, \ldots, m$. Training is a procedure that embeds to the network knowledge provided by a data set of the form

$$S = \{(\boldsymbol{x_n}, t_n),\ \boldsymbol{x_n} \in \mathcal{R}^l,\ t_n \in \{0, 1, \ldots, m-1\},\ n = 1, \ldots, p\} \tag{5}$$

known as *training set*. $\boldsymbol{x_n}$'s serve as inputs to the network and t_n's as the corresponding desired responses. After the completion of the training, the network should *generalize well*, that is it should be able to respond reliably to inputs $\boldsymbol{x}$ that are not presented in S. The generalization ability of the network is usually assessed via the *test set*, a data set of the form of S that has not been used during training.

Several training approaches have been proposed in the literature. One of them lies in the general framework of cost function optimization. In this case a

cost function J which depends on the dissimilarity between t_i and z_i, is adopted. The estimates of the values of the parameters $\boldsymbol{w_i}$, σ_i and $\boldsymbol{v_j}$ are obtained via the optimization of J, using e.g. a gradient descent (or ascent) method (see e.g. [22]). However, this approach exhibits high computational complexity for real world problems [1].

An alternative approach is to choose the parameter vectors $\boldsymbol{w_i}$ in a way representative of the distribution of the data set ([15]). This can be carried out using a clustering algorithm which will reveal the regions in the input space that are dense in data (see e.g. [1], [4], [5], [22]). One such algorithm is described in [12]. It is an implementation of the BSAS clustering algorithm described in [22], where the class information for each vector is utilized.

Other methods for training RBF networks have also been suggested (e.g. [3], [11], [19], [25]). A review of training methods of RBF networks can be found in [7].

In this paper a training scheme, called RBF-MST, is presented that follows the second apporach. Specifically, the number of the first layer nodes, k, as well as the values of the parameters $\boldsymbol{w_i}$ are determined by a clustering algorithm that relies on the idea of the minimum spanning tree (MST). Its innovative aspect is that it takes also into account the class where the data vectors belong. The paramters σ_i^2 are estimated with two methods, giving thus rise to two variants of RBF-MST, namely RBF-MST1 and RBF-MST2. Finally, the parameter vectors $\boldsymbol{v_j}$ of the second layer are determined via the well-known LMS algorithm.

The rest of this paper is organized as follows. In section 2 the description of RBF-MST scheme takes place. In section 3 the proposed scheme is compared to other known schemes in terms of a cytological application. Finally, section 4 contains the concluding remarks and guidelines for future work.

2 Description of the RBF-MST scheme

Before we proceed with the presentation of the RBF-MST scheme, some useful definitions from graph theory are in order (for a more thorough introduction see e.g. [13], [22]). A *graph* $G = (V, E)$ is defined as an ordered pair of a set of *vertices* and a set of *edges* connecting some pairs of vertices. An edge is denoted by (x_i, x_j), where x_i, x_j are the vertices it connects. Usually, a *weight* is assigned to the edges of the graph depending on the application at hand.

$G_1 = (V_1, E_1)$ is called a *subgraph* of G if $V_1 \subseteq V$ and $E_1 \subseteq E$ with the additional restriction that the edges of E_1 connect pairs of vertices of V_1. Clearly, a graph is a subgraph of itself.

A *path* between two vertices x_n and x_q is a sequence of vertices and edges of the form $x_n, (x_n, x_{n_1}), x_{n_1}, (x_{n_1}, x_{n_2}), x_{n_2}, \ldots, (x_{n_{r-1}}, x_q), x_q$. Note that there may be pairs of vertices that are not connected via a path.

A *connected component* of a graph is a subgraph such that for every pair of vertices there exists a path that connects them. If, in addition all pairs of vertices are connected to each other the subgraph is called *complete*.

[1] Other optimization techniques may also be used (see e.g. [14]).

A *spanning tree* is a connected graph (containing *all* the vertices of the graph) that has no loops (i.e., there exists no path connecting a vertex to itself). If the edges of the graph are weighted, we define as the *weight of the spanning tree* the sum of the weights of its edges. A *minimum spanning tree (MST)* is a spanning tree with the smallest weight among all spanning trees connecting the nodes of the graph. An MST of a graph may be obtained using Prim's algorithm or Kruskal's algorithm (see e.g. [6]).

After the above definitions let us proceed with the description of the RBF-MST scheme.

(a) Determination of k and $\boldsymbol{w_i}$'s

- View the data vectors $\boldsymbol{x_n}$ as the vertices of a complete graph.
- Weight each edge $(\boldsymbol{x_n}, \boldsymbol{x_q})$ with the squared Euclidean distance of the two vectors.
- Determine the MST of the graph.
- Remove the edges of it that connect vectors from different classes.
- Represent the resulted connected components (clusters) C_i of the MST by their mean vectors $\boldsymbol{\mu_i}$, $i = 1, \ldots, k$.
- Place k nodes in the first layer of the network, such that each one corresponds to one of the above determined clusters.
- Set $\boldsymbol{w_i}$ equal to $\boldsymbol{\mu_i}$, $i = 1, \ldots, k$.

Clearly, each connected subgraph contains vectors only from a single class.

(b) Determination of σ_i^2's

According to the way σ_i^2's are estimated, we have two variants of the RBF-MST scheme. Specifically, in the first variant, called RBF-MST1, the parameter σ_i^2 of the i-th first layer node is set equal to the variance of the vectors of C_i around $\boldsymbol{\mu_i}$.

In the second variant, called RBF-MST2, the parameter σ_i^2 is set equal to the squared Euclidean distance between $\boldsymbol{\mu_i}$ and its closest mean vector among the mean vectors of the rest C_r's, $r = 1, \ldots, k$, $r \neq i$.

After the determination of $\boldsymbol{w_i}$'s and σ_i^2's, we present each $\boldsymbol{x_n} \in S$ to the network, we compute the output of the first layer $\boldsymbol{y_n}$ and we form the set

$$S' = \{(\boldsymbol{y_n}, t_n),\ \boldsymbol{y_n} \in \mathcal{R}^{k+1},\ t_n \in \{0, 1, \ldots, m-1\},\ n = 1, \ldots, p\}, \qquad (6)$$

where the $k+1$ coordinate of all $\boldsymbol{y_n}$'s is set equal to 1.

(c) Determination of $\boldsymbol{v_j}$'s

They are determined via the LMS learning rule based on the data set S' (see eg. [22], [24]).

In contrast to the methods that rely on cost function optimization for the determination of k and $\boldsymbol{w_i}$'s, RBF-MST requires no multiple runs for the determination of the proper values of k and $\boldsymbol{w_i}$'s. This is a consequence of the fact

that the minimum spanning tree of a graph is (under broad conditions) unique. The main computational burden of RBF-MST is in the determination of the minimum spanning tree.

3 Experimental results

In this section the performance of the proposed algorithm is assesed through a real cytological application. The aim of this application is the assignment of a gastric cell nucleus to one of the following pathological classes: *ulcer, gastritis, inflammatory displasia, true displasia, cancer.* The cells were obtained from brushing cytology smears taken from patients during endoscopy. Each smear contained about 100 cells. For each cell the measurements of 26 characteristic attributes were taken, forming a 26-th dimensional vector that characterizes the corresponding cell. The 26 attributes are divided into two categories, as follows:

1. *Geometric characteristics:* area, circularity, major axis, minor axis, perimeter, formArea, formPerimeter, nuclear contour index, contour ratio, roundness factor, nucleus diameter and mean radius [8],[21],[20].
2. *Textural characteristics:* ([18], [2]):

   a. Nucleus run length matrix: short run, long run, grey level, distribution.

   b.Nucleus histogram: mean, variance and standard deviation.

   c.Nucleus coocurence matrix: maximum, entropy and inertia.

   d.Nucleus differences histogram: mean, variance, contrast and entropy.

The selection of the features was based on factors that affect the cytologist decision during screening. For example texture provides an indication of the DNA activation, and changes in the nuclear size and shape reflect alterations related to the behavior of the cells during mitosis. The data set used in this paper can be downloaded from http://www.di.uoa.gr/~makis/projects/CCS/CCS.html.

The whole data set consists of 13300 vectors (nuclei) extracted from 120 patients. The number of measured nuclei for each patient was ranging from 35 to 183 with a mean value of 120. This fact indicates the problems that cytologists face during everyday practice. 2920 of the cells belong in the class *"cancer" (C)*, 370 in the class *"true displasia" (TD)*, 6550 in the class *"ulcer" (U)*, 3150 in the class *"gastritis" (G)* and 310 in the class *"inflammatory displasia" (ID)*. The identification of the class of each cell was made by two experienced cytologists and confirmed by the histological examination of biopsies and/or surgical specimens. The most difficult to identify cases are the displasias, which have not clear cut characteristics. Thus, it is expected that these classes will lead the classifiers to erroneous classifications.

15% of the available data were used as training set and 70% as a test set, using stratified random selection, and thus preserving the original class distribution in the two datasets. All vector components are linearly scaled in order to lie in the range $[-1, 1]$.

In the sequel we consider the two-class problem, where the classes C, TD and the classes ID, U, G are unified, respectively. The cells of the first class are characterized as *malignant* while the cells of the second class are characterized as *benign*. As in the standard cytological practice, it is not expected to obtain good classification results among the five classes as displasias usually indicate a transition from a healthy to a pathological status.

Table 1. Results of the two variants of RBF-MST on the training and the test set. The first and the third line of the table contain the results of the two variants when all clusters produced by the clustering algortihm are used, while the remaining two lines contain the results of the two variants when the single element clusters are removed

	Accuracy on training set	**Accuracy on test set**
RBF-MST1, all clusters	98.05%	96.12%
RBF-MST1, all non single element clusters	97.59%	96.67%
RBF-MST2, all clusters	97.84%	96.43%
RBF-MST2, all non single element clusters	96.99%	96.32%

The results presented in table 1, show that the two variants of the RBF-MST discriminate very well among the two cell classes. This is an indication that there are clear distinctions between the two classes, in terms of the 26 attributes that were used, despite the fact that the two types of displasia are difficult to identify correctly. This happens because the number of the available data from displasias is very small compared to that of the other three classes. In particular, the RBF-MST1 variant gives slightly better results than the other variants, when the single element clusters are not taken into account.

In table 2, the confusion matrices of the test set for the two variants when all clusters and all except the single element clusters are presented.

From this table, it is shown that the above variants perform more or less the same kind of errors. This is an indication that they give almost equivalent results.

As a means of comparison to previous work, we present here the results of three classifiers that are frequently met in practice. These are: (a) the RBF classifier described in [12], (b) the k-nearest neighbor classifier and (c) the multilayer perceptron (see e.g. [22]). The RBF classifier gives 96.24% classification accuracy on the test set [2]. The k-nearest neighbor classifier gives 95.25% classification accuracy on the test set, for $k = 1, 3, 5, 9, 13$. The multi-layered perceptron has been trained with backpropagation in [10], using a subset of the dataset that we

[2] It should be noted that in this case, the training set was 30% of the whole data set.

Table 2. Confusion matrices of the two variants of RBF-MST on the training and the test set when (a) all clusters are taken into account and (b) all clusters with more than one element are taken into account. The (1,2) element of the above confusion matrices corresponds to the number of cells that belong in the class *malignant* and have been identified as *benign*. The opposite holds for the (2,1) element. For example, in the matrix of RBF-MST1 when all clusters are considered, 229 malignant cells have been classified as benign and 132 benign cells have been classified as malignant

	Confusion matrix	
RBF-MST1, all clusters	2074	229
	132	6875
RBF-MST1, all non single element clusters	2127	176
	134	6873
RBF-MST2, all clusters	2101	202
	130	6877
RBF-MST2, all non single element clusters	2098	205
	138	6869

used here (about 11000 cells). Specifically, 30% of the vectors of this set formed the training set and the remaining 70% the test set, such that the original class distribution is preserved in the two datasets. The results of that method were similar to the ones obtained here ($95.7\% - 97.3\%$ accuracy on the test set).

Clearly, RBF-MST exhibits almost the same performance with the RBF classifier described in [12]. Also, RBF-MST performs slightly better than k-nearest neighbor algorithms and, in addition, the resulted RBF network, in parallel implementation, responds much quicker than k- nearest neighbor. Finally, the RBF-MST performs equally well with the multilayer perceptron. However, it does not require multiple runs with different number of hidden layer nodes, as is the case with the back-propagation like schemes that are used for training multilayer perceptrons.

4 Concluding remarks

In this paper a new scheme, called RBF-MST, is described, which is suitable for training of RBF neworks. The new scheme does not rely on the idea of cost function optimization and thus, it does not require multiple runnings to determine the best possible solution, as is the case of other famous classifiers such as the multilayer perceptrons. Actually, the minimum spanning tree of a graph on which the proposed scheme relies is (under broad conditions) unique, thus the algorithm is applied once in the data set.

The proposed scheme exhibited very satisfactory performance on a real cytological application, which compares well to the performance of other well-known classifiers, such as multi-layered perceptrons and the k-nearest neighbor. Of course, in order to get furhter insight on the behavior of the algorithm, various data sets with different characteristics have to be used.

One possible extension for RBF-MST is to use covariance matrices Σ_i instead of σ_i^2's in the first layer nodes of the RBF network, since they describe more accurately the way the points of a cluster are spread around its center.

Also, it would be interesting to focus on a property that is inherit to the proposed algorithm: the produced clusters need not be of spherical shape. In this case, their representation by their mean is not the best choice. One way to face this problem is to use cluster validity indices (see e.g. [22]), to take an idea of the shape of a cluster. If for example an elongated cluster is produced, more vectors may be employed for its representation.

In addition, one may perform the basic clustering MST algorithm (see e.g. [22]) and then utilize the class information. It is expected that this approach will reveal the natural clustering structure of the data set. The class information will be used for further refinement.

Finally, other clustering techniques may be used instead of the proposed one for the determination of number of the first layer nodes as well as their parameter vectors.

References

1. Anderberg M.R. *Cluster Analysis for Applications*, Academic Press, 1973.
2. J. P. A. Baak, *Manual of Quantitative Pathology in Cancer Diagnosis and Prognosis*. Springer-Verlag, Berlin Heidelberg, Germany, first ed., 1991.
3. Chen S., Cowan C.F.N., Grant P.M. "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Transactions on Neural Networks*, Vol. 2, pp. 302–309, 1991.
4. Everitt B. *Cluster Analysis*, 2nd ed., Halsted Press, 1981.
5. Hoppner F., Klawonn F., Kruse R., Runkler T., *Fuzzy cluster analysis*, John Wiley and sons, 1999.
6. Horowitz E., Sahni S. *Fundamentals of Computer Algorithms*, Computer Science Press, 1978.
7. Hush D.R., Horne B.G. "Progress in supervised neural networks," *IEEE Signal Processing Magazine*, Vol. 10(1), pp. 8–39, 1993.
8. "Introduction to cytometry and histometry," tech. rep., 1st COMETT *International Cource on Microscope Imaging in Biology and Medicine*, Grenoble, France, Dec. 1991.
9. Kalouptsidis N. *Signal Processing Systems, Theory and Design*, John Wiley, 1997.
10. Karakitsos P., Botsoli-Stergiou E., Pouliakis A., Tzivras M., Archimandritis A., Liossi A., Kyrkou K., "Potential of the Back Propagation Neural Network in the Discrimination of Benign from Malignant Gastric Cells", *Analytical Quantitative Cytology and Histololy*, 1996, 18:3: 245-250.
11. Karayiannis N.B., Mi G.W. "Growing radial basis neural networks. Merging supervised and unsupervised learning with network growth techniques," *IEEE Transactions on Neural Networks*, Vol. 8(6), pp. 1492–1506, 1997.
12. Koutroumbas K., Paliouras G., Karkaletsis V., and Spyropoulos C.D., "Comparison of Computational Learning Methods on a Diagnostic Cytological Application", *Proc. of the European Symposium on Intelligent Technologies, Hybric Systems and Their Implementation on Smart Adaptive Systems EUNITE01*, Tenerife, Spain, pp. 500-508, 2001.

13. Liu C.L. *Introduction to Combinatorial Mathematics*, McGraw-Hill, 1968.
14. Luenberger D. G., *Linear and nonlinear programming*, Addison Wesley, 1984.
15. J. Moody, C. J. Darken, "Fast learning in networks of locally tuned processing units", *Neural Computation*, Vol. 6(4), pp. 281-294,1989.
16. Park J., Sandberg I.W. "Universal approximation using radial basis function networks," *Neural Computation*, Vol. 3(2), pp. 246–257, 1991.
17. Park J., Sandberg I.W. "Approximation and radial basis function networks," *Neural Computation*, Vol. 5(2), pp. 305–316, 1993.
18. I. Pitas, *Digital Image Processing Algorithms*. Prentice-Hall, first ed., 1993.
19. Platt J. "A resource allocating network for function interpolation," *Neural Computation*, Vol. 3, pp. 213–225, 1991.
20. J. C. Russ, *The Image Processing Handbook*. Boka Raton: CRC Press, IEEE Press, second ed., 1995.
21. M. Sonka, V. Hlavac, and R. Boyle, *Image processing analysis and machine vision*. Chapman&Hall, Cambridge, Great Britain, first ed., 1994.
22. S. Theodoridis, K.Koutroumbas, *Pattern Recognition*, Academic Press, 1998.
23. Widrow B., Hoff M.E., Jr. "Adaptive switching circuits," *IRE WESCON Convention Record*, pp. 96–104, 1960.
24. Widrow B., Lehr M.A. "30 years of adaptive neural networks: Perceptron, madaline, and backpropagation," *Proceedings of the IEEE*, Vol. 78(9), pp. 1415–1442, 1990.
25. Yingwei L., Sundararajan N., Saratihandran P. "Performance evaluation of a sequential minimal RBF neural network learning algorithm," *IEEE Transactions on Neural Networks*, Vol. 9(2), pp. 308–318, 1998.