ESANN'1995 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 19-20-21 April 1995, D-Facto public., ISBN 2-9600049-3-0, pp. 141-146

A DETERMINISTIC METHOD FOR ESTABLISHING THE INITIAL CONDITIONS IN THE RCE ALGORITHM*

J.M. Moreno, F.X. Vázquez, F. Castillo, J. Madrenas, J. Cabestany

Dept. d'Enginyeria Electrònica - Universitat Politècnica de Catalunya
Building C4 - Campus Nord - c/Gran Capità s/n - 08034 - Barcelona - Spain

Abstract. The RCE algorithm, which can be considered as belonging to the category of the Region of Influence (ROI) incremental algorithms, is one of the first and most widely used incremental algorithms. It is well known that the performance of this algorithm is limited by constraints affecting the initial parameter settings: order in which input patterns are presented during the training phase and initial radius to be used for the generated hyperspheres. In this paper, we propose a deterministic method for obtaining automatically the suitable initial radius for a given input problem. The method is based on the definition of several functions which give a quantitative measure of the geometrical relations between the different categories defined in the input space. As the simulation results show, this method improves significantly the performance (in terms of number of units generated and generalization capability) of the RCE algorithm.

1. Introduction

In the last years there has been an increasing interest in the research field concerned with the evolutive neural models. These new neural paradigms include algorithms which are capable to determine, either by incremental (network growth) or decremental (unit pruning) principles, the most suitable network topology for solving a particular problem. Hence, the usual and time consuming trial and error process used for determining the proper network structure associated to the classical neural models is avoided by a self-evolving scheme.

Among the different kinds of incremental neural models [1], we shall concentrate in this paper on the Region of Influence (ROI) incremental paradigms, and, in more detail, we shall consider the RCE algorithm [2], which is one of the first and most widely used incremental algorithms.

After reviewing the main limitations associated to this incremental neural algorithm, we shall propose a new deterministic method which is capable of

* This work has been partially funded by Esprit III Project Elena-Nerves 2 (no. 6891) and by spanish CICYT action TIC92-629

ESANN'1995 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 19-20-21 April 1995, D-Facto public., ISBN 2-9600049-3-0, pp. 141-146

determining the proper initial conditions (i.e., the initial value for the radius of the hyperspheres associated with the units generated by the algorithm during the training phase) to guarantee a correct behavior for the classifier system built around the network structure generated by the algorithm.

2. Limitations of the RCE algorithm

The RCE algorithm constructs a network structure composed of two layers. The units constituting the first layer, usually called coding units, cover some areas in the input space by means of hyperspherical influence regions, whose center and radius are determined during the training phase. The units in the second layer, usually referred to as class units, combine the influence regions associated to the units in the first layer, so that each of them can provide the proper approximation for the dominant region corresponding to one of the categories defined in the input space.

The main advantage associated to the RCE algorithm is its simplicity, hence facilitating its eventual implementation, either hardware or software. However, its major drawbacks are related to the way the initial conditions are determined and to the ordering used for presenting the input patterns during the training process.

The second problem is common to most of the incremental neural algorithms, and means that both the solution obtained for the desired discriminant function and the final network organization will strongly depend on the order the input patterns are presented during the training phase, specially at the early stages. A method, based on LVQ-like techniques which search for a proper location of the hyperspheres' centers, has been presented in [3] for overcoming partially the influence of the training order on the performance of the algorithm.

Regarding the initial value for the hyperspheres' radii, it is in fact one of the most critical parameters to be fixed, since it will determine the performance of the final network evolved by the algorithm. Large values for this parameter (large coverage areas) usually imply a smaller number of units, but a poor generalization capability for the final network. On the other hand, smaller values for the initial radius should provide smaller overlapping regions, thus increasing the generalization capabilities, but at the expense of a large number of units. Therefore, there must be a trade-off between the desired network complexity and its ability to generalize correctly on input patterns not present in the training data set.

3. Proposed method

The new method we propose for determining the initial radius to be used by the RCE algorithm is based on the definition of some functions, derived from those used in [4] for improving the performance of PLS incremental algorithms, which provide a quantitative measure of the geometrical relations between the different categories defined in the input space.

ESANN'1995 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 19-20-21 April 1995, D-Facto public., ISBN 2-9600049-3-0, pp. 141-146

Let us consider the problem of separating input patterns $\mathbf{x}=(x_1, x_2, ..., x_n)$ belonging to M different categories in an n-dimensional input space. We define then the category-i centroid, $\underline{C}_i$ as the vector whose components are obtained by calculating the mean value from the components of the vectors which represent the patterns belonging to this category.

Once all the centroids have been calculated, we define the following functions in the entire distribution:

$$J = \sum_{k=1}^{M} \sum_{l=1, l\neq k}^{M} J_{kl} \ , \quad J_{kl} = \frac{1}{N_k} \sum_{i=1}^{N_k} \frac{1}{\left\| \underline{x}_i - \underline{C}_l \right\|} \qquad (1)$$

where:

* J_{kl} : Contribution to the J function from the patterns belonging to the class k with respect to the centroid of class l
* x_i^k : i-th pattern of category k
* J : function whose value is related with the geometrical complexity of the categories' distribution
* N_k : number of patterns belonging to class i

As can be easily deduced from equations (1), the function J gives a quantitative measure of the complexity associated to the distributions of the categories defined in the input space. It means that highly intertwined distributions (input patterns belonging to one class placed close to the centroids of the other classes, thus contributing with large J_{kl} values) should yield large values for the function J, while non-complex distributions should provide small values for this function. The value of J_{kl}, as indicated in equation (1), is weighted by the number of patterns belonging to each class, so as to make it independent of the number of training patterns associated to each category.

Once the value of the J function is calculated, we use for the RCE algorithm an initial radius whose value is given by the following expression:

$$R_i = k + \frac{\alpha}{J} \qquad (2)$$

where:

* k : constant which avoids the problem given by a too small initial radius when the input distributions provide very large values for the J function. Our analyses have provided a value of 0.1 as adequate for this constant.
* α : Proportionality factor which includes the influence of the centroids' dispersion. It is given by the maximum distance between the centroids of the categories in the input space. That is:

$$\alpha = \max\left\{ \left\| \underline{C}_i - \underline{C}_j \right\| \right\}, \qquad \forall i \neq j \quad (3)$$

We can deduce from formula (2) that highly intertwined distributions will give large values for J function, and therefore small values for the initial radius.

ESANN'1995 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 19-20-21 April 1995, D-Facto public., ISBN 2-9600049-3-0, pp. 141-146

Hence, the conflicts given by possible large overlapping regions between hyperspheres associated to different categories can be avoided. On the opposite side, non-complex distributions will provide small values for the J function, meaning a larger value for the initial radius and therefore larger coverage area for the hyperspherical regions whose combination estimates the dominant areas for the input categories.

This method presents however some limitations for certain distributions of the categories defined in the input space. Considering equation (3), it can be deduced that if the input vectors are distributed in such a way that all the categories defined in the input space have the same centroid ($\underline{C}_i=\underline{C}_j \ \forall$ i,j), then the factor α is canceled, meaning that there is no dispersion in the centroids distribution. Therefore, according to equation (2), in this case the value of the initial radius to be used for the hyperspheres associated to the coding units will be equal to the constant k, thus remaining independent of the value given by the function J.

So as to avoid this problem, the dispersion factor α could be redefined as follows:

$$\alpha = \begin{cases} 1 & , \text{if } \underline{C}_i = \underline{C}_j \quad \forall i,j \\ \max\left\{\left\|\underline{C}_i - \underline{C}_j\right\|\right\} & , \text{otherwise} \end{cases} \qquad (4)$$

By defining the factor α in this way, it is possible to make the initial radius, R_i, depend on the value given by the function J, even if there is no dispersion in the distribution of the centroids associated to the input categories.

4. Experimental results

In this section we shall show simulation results obtained by using the RCE algorithm with several real world databases. The results will be presented for the standard RCE algorithm modified by the preprocessing phase presented in [3], for the RCE algorithm with the initial radius determined by our method, and finally for the RCE algorithm modified by both methods (preprocessing phase and initial radius).

The databases used for our test purposes are the following: *Wine Recognition Data* [5], from the *Institute of Pharmaceutical and Food Analysis and Technologies* in Salerno. *Aerial Acoustic Signals*, composed by 4916 patterns, 2377 of them corresponding to aerial acoustic signals given by helicopters, and the rest to signals given by other flying objects. Each input vector consists of 5 components, the fundamental frequency and the level in dB for the first four harmonics. *Pima Indian Diabetes* [6], from the *National Institute of Diabetes, Digestive and Kidney Diseases*. And *Vehicle Silhouettes* [7], from the *Turing Institute*.

The cross-validation method used to test the RCE algorithm against the proposed problems is the Leave-K-out, derived from the Jacknife principle [8]. For this purpose, we have divided the original problems in eleven equal-sized subsets.

ESANN'1995 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 19-20-21 April 1995, D-Facto public., ISBN 2-9600049-3-0, pp. 141-146

The training is performed with a training set obtained by joining ten of this subsets, and then the generalization test is performed on the remaining subset. In this way, there may be ten different training-test pairs for each problem. So as to avoid the fluctuations in the network structure given by the order the input vectors are presented during the training phase, two simulations with different random order are performed for each test set, thus giving 20 runs in total for each problem.

In the following, we shall show the results corresponding to the solution of the *Pima Indian Diabetes* and *Vehicle Silhouettes* problems. Similar results have been obtained for the other two problems considered in our test benchmarks. The presented results are divided in two parts: one regarding the network structure generated by the RCE algorithm (mean and variance of the number of units generated), and the other related to its performance when generalizing for input patterns not included in the training set (mean percentages for classification, misclassification and rejection).

Table 1 shows the results obtained when the preprocessing technique presented in [3] is used for the RCE algorithm. In this case, the initial radius is obtained by an heuristic procedure derived from the inspection of the value of the calculated variances for each of the components of the input vectors.

On table 2 we indicate the results given by the RCE algorithm using the LVQ-like preprocessing technique stated above and with the initial radius determined by the method presented in this paper.

Problem	**Network size (# of units)**		**Network performance (%)**		
	Mean	Variance	Correct	Incorrect	Rejected
Diabetes	457	105.9	44.9	15.8	39.3
Vehicle	473	38.3	43.1	14.9	42.0

Table 1. Results given with LVQ-like preprocessing

Problem	**Network size (# of units)**		**Network performance (%)**		
	Mean	Variance	Correct	Incorrect	Rejected
Diabetes	428	162.1	47.0	14.8	38.2
Vehicle	452	50.1	43.9	13.4	42.7

Table 2. Results given by the LVQ-like preprocessing technique and the initial radius determined from the function J

The results given in this case demonstrate a rather big improvement in the behavior of the RCE algorithm, since it is capable to generate more compact structures which are capable also to generalize better on the classification tasks given as input problems. Furthermore, we have demonstrated that, by combining the LVQ-like preprocessing technique and the method for determining the initial radius,

it is possible to circumvent the problems caused by the sensitivity of the RCE algorithm to the initial conditions imposed during the training phase.

5. Conclusions

In this paper we have reviewed some of the problems associated to the RCE algorithm caused by its sensitivity to the initial conditions imposed for the training phase. These problems are due to the influence on the algorithm performance of the order the input patterns are presented during the training phase. Another critical parameter to be considered is the value of the initial radius assigned to the hyperspheres in charge of estimating the dominant regions for the categories defined in the input space.

After considering the limitations associated with the RCE algorithm, we have presented a novel method for determining the initial radius to be used by the RCE algorithm. The exhaustive experimental results obtained by applying the proposed method to real world databases show that the behavior of the RCE algorithm (both in terms of network complexity and generalization capability) is improved, thus making this method suitable for eventual physical implementations. The main advantage provided by this method is given by the fact that no heuristics are required in order to determine the proper initial conditions for the RCE algorithm.

6. References

1. F. Castillo, "Incremental Neural Networks: A Survey". Technical Report. INPG Grenoble, (1991)
2. D.L. Reilly, L.N. Cooper, C. Elbaum, "A Neural Model for Category Learning". Biological Cybernetics, 45, 35-41 (1982)
3. F. Blayo, M. Verleysen, "Setting Initial Conditions for the RCE Model", Proceedings of the first IFIP Working Group 10.6, 31-35 (1992)
4. J.M. Moreno, F. Castillo, J. Cabestany, "Improving Piecewise Linear Separation Incremental Algorithms using Complexity Reduction Methods", Proceedings of the European Symposium on Artificial Neural Networks, ESANN'94, 141-146, (1994)
5. S. Aeberhard, D. Coomans, O. de Vel, "Comparison of Classifiers in High Dimensional Settings", Technical Report no. 92-02, Dept. of Computer Science and Dept. of Mathematics, James Cook University of North Queensland, (1992)
6. J.W. Smith, J.E. Everhart, W.C. Dickson, W.C. Knowler, R.S. Johanes, "Using the ADAP Learning Algorithm to Forecast the Onset of Diabetes Mellitus", Proceedings of the Symposium on Computer Applications and Medical Care, 261-265 (1988)
7. J.P. Siebert, "Vehicle Recognition Using Rule Based Methods", Turing Institute Research Memorandum TIRM-87-018 (1987)
8. B. Efron, "Boostrap Methods: Another look at the Jacknife", The Annals of Statistics, 7(1), 1-26 (1979)