ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 33-38

ENHANCED UNIT TRAINING FOR PIECEWISE LINEAR SEPARATION INCREMENTAL ALGORITHMS

J.M. Moreno*, F. Castillo, J. Cabestany

Departament d'Enginyeria Electrónica
Universitat Politéctica de Catalunya
c/Jordi Girona Salgado s/n
P.O. Box 30002, 08080 Barcelona, SPAIN

Abstract :In this paper we present some of the problems most commonly found when perceptron-based learning algorithms are used for training the units generated by Piecewise Linear Separation incremental algorithms. Some improvements for the unit training algorithms, based on concepts derived from the Information Theory, are indicated. Finally, a new method is introduced which compares well with the methods based on the measurement of entropy-like functions, and further requires a low computational cost, making it suitable for hardware implementations.

1. Introduction

One of the most promising research areas among the recent advances in the field of the Artificial Neural Network's Theory has been devoted to the study and development of the so called Evolutive or Incremental Neural Networks. In this kind of neural architectures, the network topology need not be guessed in advance, but is determined during the training phase, depending on the problem to be solved.

As was pointed out in [1], there are several types of incremental algorithms, from which we shall consider in this paper the Piecewise Linear Separation (PLS) Incremental algorithms. These algorithms try to perform a classification task by performing a separation of the input data space. The global separation is obtained by combining the individual separations produced by perceptron-like units. In this way, the incremental algorithm begins by training a small network (usually one unit). If the linear decision region generated by this network is not capable of solving the whole classification problem, new units are added in order to obtain better solutions, until the classification problem is totally solved.

In this paper we review some of the methods most frequently used in training the individual units generated by the incremental algorithms. Next, we shall analyze the problems encountered as the network evolves. Finally, we shall

---

*Holder of an FI research Grant under the Generalitat de Catalunya's Educ. Dept.

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 33-38

present some criterions for improving the performance in training individual units.

2. Training the perceptron-like units

The units generated by the PLS incremental algorithm are usually trained by means of the perceptron [2] or the Pocket [3] learning algorithms. Therefore, when the training process is finished, the unit has assigned the set of connection weights which optimally performs the input-output mapping presented as problem to the unit.

However, since the solutions found by these algorithms try to maximize the number of patterns correctly classified by the unit, the separation performed may not be useful for constructing the whole network, in the sense that it may not reduce the complexity of the problem.

An example of such an optimal but not useful solution is depicted in figure 1, which represents the problem of classifying input patterns belonging to two non-linearly separable classes.

In this figure, the solid line corresponds to a possible optimal, in the sense of maximum classification rate stated above, solution found by the perceptron-based learning algorithms. As can be noted, this solution is not useful for the network evolution, since this unit performs no separation of its input data space. Therefore, the next units added to the network by the incremental algorithm will have to solve the same problem, and will probably arrive at the same solution. In this way, a useless network consisting of only redundant units will be generated.

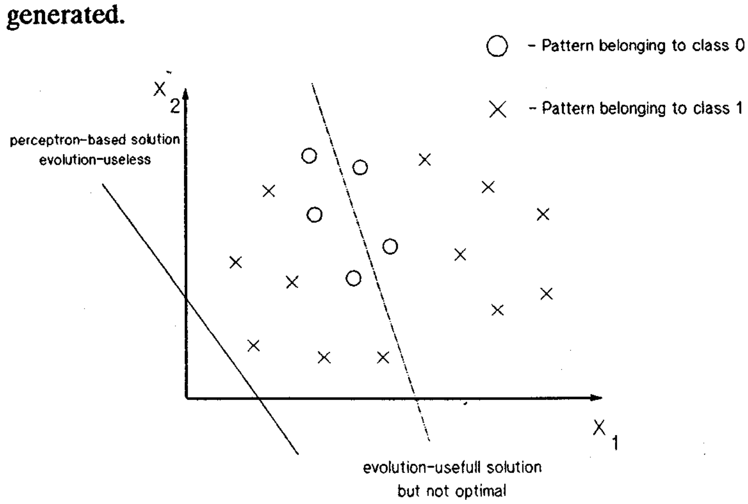

Figure 1 - Optimal and useful solutions

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 33-38

The previous problem would be avoided by a solution like the one depicted as a dotted line in figure 1. This solution performs a true separation in the input data space, and so lower complexity problems are generated for the new units added to the network.

In the next section, we present a review of some criterions used to modify the unit training algorithms so as to obtain suboptimal but evolution-useful solutions.

3. Enhancing individual unit training

As was pointed out in the previous section, some modifications must be introduced in the unit training algorithms in order to avoid redundant solutions for the final network. It is also important in PLS incremental algorithms that each unit performs a "good" separation of the patterns belonging to its training data set, so that the complexity of the problem will be reduced at each step of the incremental algorithm.

Bearing in mind the previous considerations, a criterion has been proposed [4], [5], for improving the quality of the separation performed by each individual unit in its training data set. This method consists on running the standard Pocket algorithm, but considering that the function to be maximized is the following entropy-like function, as opposed to directly maximize the number of input patterns correctly classified by the unit.

$$I = \log C_p^q - \log C_{p'}^{q'} - \log C_{p-p'}^{q-q'}$$

where C is a combinatorial term, and p, p', q, q' are, respectively, the total number of patterns in the set under consideration, the number of patterns belonging to the first class, the number of patterns being classified as belonging to the first class and the number of patterns of the first class which are correctly classified.

The function I can be considered as the difference in mixing entropy before and after the partition has been performed. In this way, we measure the "goodness" of the partition performed by the unit in terms of the value given by I for the separation, and finally consider that the "best" partition corresponds to the set of weights which maximizes I.

Another criterion [6] proposes the use of a similar entropy-like function to measure the quality of the partition generated by each unit in its input data set. This function is given by:

$$J = - \frac{w_{(+1)}}{m_{(+1)} + 1} - \frac{w_{(-1)}}{m_{(-1)} + 1}$$

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 33-38

Where w and m denote, respectively, the number of correctly and wrongly classified patterns in the side of the hyperplane indicated by their index.

As in the previous method, the best partition will be obtained by running the Pocket algorithm for a maximum entropy function, which also gives information about the quality of the partition performed by the unit in its input data set.

The method we propose in this paper is not based on the evaluation of entropy-like functions, whose main drawback is the computational cost associated to the operations performed to obtain the separation quality indicator. The method consists of running the Pocket algorithm on the input training set associated to each unit, but we don't require that the unit performs the best possible classification on the input data set, but rather, require it to classify correctly patterns of only one particular class on one side of the separating hyperplane. Figure 2 depicts the principle on which this method is based.

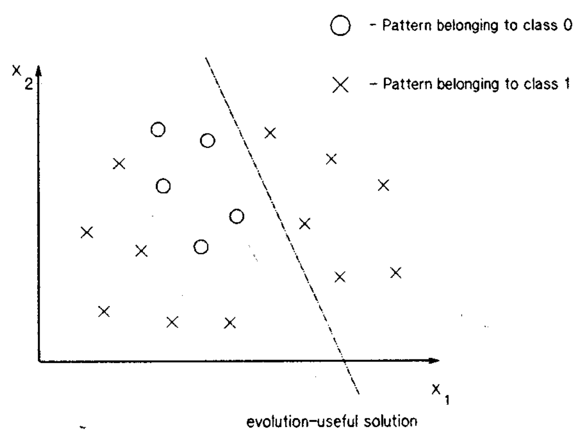

Figure 2 - Principle of the proposed method

As can be seen, this method allows for a step by step simplification of the original problem by eliminating from the training data set patterns belonging to one class, and so the complexity of the classification problem is reduced as the incremental algorithm evolves.

In order to compare the three methods proposed above, two of the most representative PLS incremental algorithms have been selected: the Neural Trees algorithm [4] and the Upstart algorithm [7]. Several runs of these algorithms have been carried out on the following sequence of problems:

* Separation of input data patterns belonging to two bidimensional,

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 33-38

concentric distributed classes.
* Separation of input data patterns belonging to two bidimensional, normal distributed classes.
* Separation of input data patterns belonging to two bidimensional reflected spirals.
* Random boolean function of eight inputs.
* Two or more clumps problem for a sequence of eight bits.

Since the results are similar for both incremental algorithms, we present here only the results obtained for the Neural Trees algorithm. These results, presented in table 1, are give as mean number of units generated by the algorithm.

As can be seen, the results compare well (even better results are obtained for the Upstart algorithm) with the results obtained when using the Pocket algorithm modified for maximizing the entropy functions. The worst results, as may be expected, correspond to the spiral classification problem, due to the large amount of units generated by this method when the classes to be separated are intertwined.

4. Conclusions

In this paper we have investigated the problems which may appear when the PLS incremental algorithms use the classical perceptron or Pocket learning algorithms for unit training. As a result, the need for suboptimal but evolution-useful solutions have been pointed out.

Some of these suboptimal solutions can be found by modifying the unit training algorithms so as to maximize entropy-like functions, which determine the quality of the separation performed by each unit in its input data set.

Problem	I-method	J-method	New method
Concentric classif.	6.59	6.86	28.68
Gaussian classif.	6.32	6.59	10.77
Spiral classif.	16.41	39.36	16.41
Random boolean function	4.41	5.31	6.5
Two or more clumps	5.05	6.32	9.14

Table 1 - Simulation results for the Neural Trees algorithm

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 33-38

Finally, a new method has been presented, which consists of running the Pocket algorithm on the input data set associated to each unit, but trying to have on one side of the separating hyperplane only patterns belonging to one class, which must also be correctly classified. As can be deduced from the simulation results, this method compares well with the methods based on the evaluation of entropy functions, except for the problems in which the classes to be separated are complexly intertwined. In addition, this method has the advantage of the low computational cost associated to the evaluation of the criterion for obtaining the best weight set for each unit.

Acknowledgments :

This work has been partially funded by ESPRIT III Project ELENA-Nerves 2 (no. 6891).

5. References

1. F. Castillo: Incremental Neural Networks: A Survey. Technical Report. INPG Grenoble. (1991)

2. D.E. Rumelhart, J.L. McClelland: Parallel Distributed Processing: Explorations in the Microstructure of Cognition. MIT Press. (1986)

3. S.I. Gallant: Optimal Linear Discriminants. Proc. of the 8th. Intl. Conf on Pattern Recognition. Vol. 2, 849-854. Paris. (1986)

4. J.A. Sirat, J.P. Nadal: Neural Trees: A New Tool for Classification. Technical Report. Laboratoires d'Electronique Philips. (1990)

5. J.P. Nadal, G. Toulouse: Information Storage in Sparsely-Coded Memory Nets. Network, Vol. 1, 61-74. (1990)

6. S. Knerr, L. Personnaz, G. Dreyfus: A new Approach to the Design of Neural Network Classifiers and its Application to the Automatic Recognition of Handwritten Digits. Proc. of IJCNN 91, 91-96, Seattle. (1991)

7. M. Frean: The Upstart Algorithm: A Method for Constructing and Training Feedforward Neural Networks. Neural Computation 2, 198-209. (1990)