A synthesis method for multilayer neural networks having ±1 weights, minimum hidden units and robust binary pattern classification

著者	Nakayama Kenji, Katayama Hiroshi
journal or publication title	Proceedings of theINNS, WCNN'93 Portland
page range	?-190-?-193
year	1993-07-01
URL	http://hdl.handle.net/2297/11899

A SYNTHESIS METHOD FOR MULTILAYER NEURAL NETWORKS HAVING ±1 WEIGHTS, MINIMUM HIDDEN UNITS AND ROBUST BINARY PATTERN CLASSIFICATION

Kenji NAKAYAMA   Hiroshi KATAYAMA

Dept. of Electrical and Computer Eng., Faculty of Tech., Kanazawa Univ.

2-40-20, Kodatsuno, Kanazawa, 920 JAPAN

E-mail: nakayama@haspnn1.ec.t.kanazawa-u.ac.jp

ABSTRACT This paper presents a new synthesis method for multilayer neural networks, which is applied to binary pattern classification. Connection weights can be expressed with ±1, and unit outputs take 1 or 0. The number of hidden units is minimized, while achieving the highest insensitivity to noisy patterns. In the designed multilayer neural network, computational complexity and memory capacity can be drastically saved.

In the proposed method, first, all training patterns are individually represented by a single hidden unit. Second, representative hidden unit, whose input takes large value for many patterns in the same category, is selected. The hidden units for these close located patterns are replaced by this representative unit. Third, the weights connected to one hidden unit are always shifted toward the center of a cluster of the related patterns. This is always carried out when the combination of the patterns is changed. Finally, redundant hidden units, which have too sufficient noise margin, are absorbed by the other unit, whose input tightly satisfies the given noise margin. Simulation results, using many kinds of patterns, demonstrate efficiency of the proposed method.

I INTRODUCTION

Multilayer neural networks are attractive for pattern recognition and classification. Back-propagation (BP) algorithm and its modified version have been successfully applied [1]. However, many problems still remain. For example, there is no general rule to determine the minimum number of the hidden units, with which robustness for noisy patterns can be achieved. In complicated pattern classification, such as the parity problem, training convergence is highly dependent on the initial weights and the parameters. We must try so many times by changing the initial weights and the parameters. Furthermore, reductions in hardware are also important in actual applications. This, however, is rather difficult based on the BP algorithm.

A digital hardware realization is one hopeful approach. In order to simplify digital hardware, it is necessary to decrease the number of units, connections and bits, while maintaining the desired performance. For this purpose, several low-bit learning algorithms have been proposed [2]-[6]. However, efficiency of these methods are rather limited to some extent.

In this paper, a new synthesis method is proposed for multilayer neural networks applied to binary pattern classification. Coefficient weights are expressed with ±1. Unit outputs take only 1 or 0. The number of hidden units is minimized, while achieving robustness for to noisy patterns.

II BINARY PATTERN RECOGNITION

Complete Separation Model

Let training patterns be P(m), which are further expressed by

$$P(m) = \{p_{mi}\}, \quad m=1,2,\ldots,M, \quad i=1,2,\ldots,N \qquad (1)$$

$$p_{mi} = 1 \text{ or } 0$$

All binary patterns up to 2^N can be distinguished by the following N-dimensional hyperplanes.

$$\sum_{i=1}^{N} p^{-}_{mi} - \sum_{i=1}^{N} (p^{+}_{mi} - 1) - \theta = 0 \qquad (2)$$

where, $p^{+}_{mi}=1$, $p^{-}_{mi}=0$, $0 < \theta < 1$

This hyperplane can be replaced by a single layer neural network, having the following connection weights as shown in Fig.1(a). w_{im} and w_{offm} are from the ith input unit and an offset unit, which always outputs 1, to the mth output unit, respectively.

$$w_{im} = \begin{cases} 1, & p_{mi}=1 \\ -1, & p_{mi}=0 \end{cases} \qquad (3a)$$

$$w_{offm} = -N_m \qquad (3b)$$

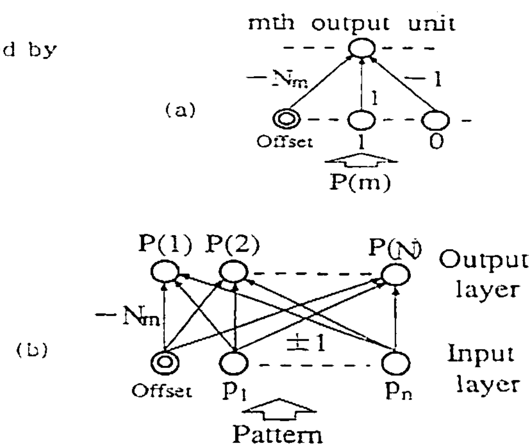

Fig.1 Complete separation model.

N_m is the number of activated units included in the pattern P(m). w_{OFFm} is expressed with several bits. However, since only one offset unit is required in the input layer, it is trivial in hardware.

When P(m) is applied to the network, the input of the kth output unit, denoted by $x_m(k)$, is determined by the number of activated units included in the following pattern.

$$P(k) \cup P(m) - P(k) \cap P(m) \quad (4)$$

The first term represents the units included in P(k) or P(m), and the second the units included in both patterns, respectively. Therefore, the number of units included in the above pattern is equivalent to a Hamming distance, denoted by H_{mk}, between P(m) and P(k). Then, $x_m(k)$ becomes

$$x_m(k) = -H_{mk} \quad (5)$$

By using the connection weights given by Eq.(3), the following relation is always held.

$$x_m(m) > x_m(k), \quad k \neq m \quad (6)$$

Therefore, by using a output unit for each training patterns, as shown in Fig.1(b), and employing the following winner take all rule, all training patterns can be exactly recognized. Letting the output of the kth output unit for P(m) be $y_m(k)$, it is determined by

$$x_m(k) = \sum_{i=1}^{N} w_{ik} p_{mi} + w_{OFFk} \quad (7a)$$

$$\text{If } x_m(k) = \max_r \{x_m(r)\}, \text{ then } y_m(k) = 1, \text{ otherwise } y_m(k) = 0. \quad (7b)$$

Robustness for Noisy Pattern Recognition

A noisy pattern is generated by adding some noises to P(m), and it is denoted by P'(m). When P'(m) is applied to the network, the input of the kth output unit, denoted by $x_{m'}(k)$, becomes also the Hamming distance between P'(m) and P(k), denoted by $H_{m'k}$.

$$x_{m'}(k) = -H_{m'k} \quad (8)$$

Therefore, the noisy pattern is always recognized as the training pattern, which has the shortest Hamming distance.

Simulation of Noisy Pattern Recognition

Alphabet letters with 16x16 pixels are employed. The BP algorithm with sufficient number of bits [1], and the conventional low-bit learning algorithm, in which the number of bits are gradually decreased during a learning process [2],[3], are used for comparison. Two kinds of evaluations are employed. First, the original pattern, to which the noises are added, is assumed as a correct answer (Original pattern evaluation). Second, the pattern, which has the minimum Hamming distance with the noisy pattern, is assumed as a correct answer (Hamming distance evaluation).

Figure 2 shows the results. The horizontal and vertical axes indicate the number of noises and the number of patterns, which cannot be recognized. From these results, the proposed method can provide robust recognition for noisy binary patterns in both evaluations.

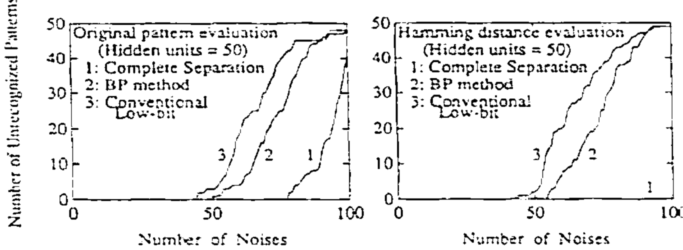

Fig.2 Noisy pattern recognition.

III BINARY PATTERN CLASSIFICATION

Pattern Classification Problem

It is possible to realize a network for binary pattern classification, based on the complete separation model. However, it is impractical when a large number of training patterns are required. Because the same number of hidden units as the training patterns are required.

Local Representation Model

Classification problems can be solved using a two-layer neural network, wiht a smaller number of hidden units than the training patterns. A blockdiagram is shown in Fig.3.

Method I : Let a pattern included in the kth category be $P_k(m)$. All connection weights are initialized to be zero.

Step 1: Apply a pattern $P_k(m)$ to the network. Calculate the input of all hidden units. Let x_{kmax} and $x_{k'max}$ be the maximum input of the hidden units assigned to the kth

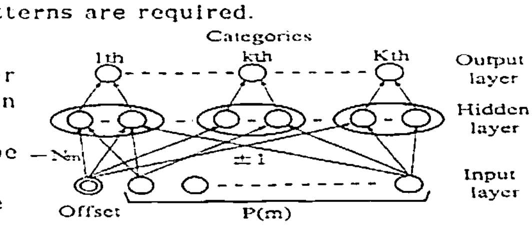

Fig.3 Two-layer classification model (local representation).

category and the other categories, respectively. If they satisfy

$$x_{k\,max} - x_{k'\,max} \gtrsim \alpha, \qquad (9)$$

then no change in hidden unit assignment and connection weights is required. Otherwise, a new hidden unit is assigned to $P_k(m)$. α is a noise margin, which can control robustness for noisy patterns. Step.1 is repeated until Eq.(9) is satisfied for all patterns by increasing hidden units.

In this method, the resulting number of hidden units is highly dependent on the order of applying the patterns. It will be used for comparison with the newly proposed method later.

IV HIDDEN UNIT MINIMIZATION UNDER SPECIFIED NOISE MARGIN

In this section, a new synthesis method for minimizing the number of hidden units, while achieving robust classification of noisy patterns, is proposed. This method consists of the following three phases. A two-layer neural network, as shown in Fig.3, is also employed.

Phase A: Initial Grouping of Patterns

(1) The complete separation model, described in Sec. II, is initially used.
(2) Apply a training pattern $P_k(m)$ in the kth category to the network. Calculate the input of all hidden units.
(3) Count the number of the hidden units, assigned to the kth category, whose input exceed $x_{k'\,max}$, which is the maximum input of the hidden unit, assigned to the other categories. Let this number for the applied pattern $P_k(m)$ be N_{km}. Find the training pattern, whose N_{km} is the maximum in the kth category. It is denoted by $P_k(m_i)$.
(4) The training patterns, whose hidden unit input exceeds $x_{k'\,max}$, when $P_k(m_i)$ is applied, are included in a sub-group named G_{ki}. It also includes $P_k(m_i)$ itself.
(5) The training patterns included in G_{ki} and their hidden units are removed. After that, (2)～(4) are repeated for the other patterns in the kth category.
(6) (2)～(5) are repeated for all categories. When the other categories are dealt with, the training patterns and their hidden units, removed in the previous category, are restored.

Phase B: Connection Weight Optimization

In the previous phase, all patterns are divided into sub-groups. A single hidden unit is newly assigned to each sub-group. The connection weights from the input layer to each hidden unit are determined so as to locate at the center of the patterns included in the same sub-group.

Let the number of the patterns included in G_{ki} be N_{ki}. Furthermore, it is assumed that the jth input unit is activated n_{kij} times when these patterns are applied to the network. The connection weights w_{kji} from the jth input unit to the ith hidden unit for the kth category, as shown in Fig.4, are determined as follows:

$$\text{If } n_{kij} \geqq \beta N_{ki} \text{ then } w_{kji} = 1 \qquad (10a)$$
$$\text{If } n_{kij} < \beta N_{ki} \text{ then } w_{kji} = -1 \qquad (10b)$$
$$0 < \beta < 1$$

Optimum value for β dependents on each problem. In our experience, $\beta \fallingdotseq 0.6$ can be widely used.

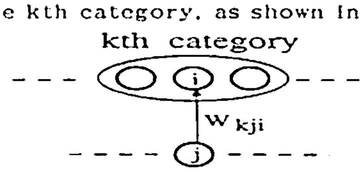

Fig.4 Connection from input layer to kth cluster in hidden layer.

Phase C: Adjusting Hidden Unit Assignments

The network, having the connection weights determined in Phase B, does not always satisfies the noise margin given by Eq.(9). Therefore, the noise margin for all patterns are checked again, and the hidden unit assignments are further adjusted so as to tightly satisfy the noise margin.

When $P_k(m)$ is applied to the network, the hidden unit inputs are defined as follows:

$\vec{x}_{km}$: The input of the hidden unit, in which $P_k(m)$ is included.
x_{kn}: The input of the other hidden units included in the kth category.
$x_{k'\,max}$: The maximum input for the hidden units assigned to the other categories.

The hidden unit assignments are adjusted as follows:

(a) If $x_{km} - x_{k'\,max} \gtrsim \alpha > x_{kn} - x_{k'\,max}$, then any modification is not required.
(b) If $x_{km} - x_{k'\,max} > x_{kn} - x_{k'\,max} \geqq \alpha$, then $P_k(m)$ is moved to the hidden unit, whose input is the minimum value of x_{kn}.
(c) If $x_{kn} - x_{k'\,max} \gtrsim \alpha > x_{km} - x_{k'\,max}$, then $P_k(m)$ is moved to the hidden unit, whose input is the minimum value of x_{kn}.
(d) If $x_{km} - x_{k'\,max}$, $x_{kn} - x_{k'\,max} < \alpha$, then a new hidden unit is assigned to $P_k(m)$.

After Phase B, all training patterns are randomly applied to the network, and the hidden unit assignments are adjusted until all patterns satisfy the condition (a). When the pattern combination in the hidden unit is changed, the connection weights are always optimized following Phase B.

V SIMULATION

Binary Pattern Classification Problems

The following classification problems have been simulated.

(1) Capital and small alphabet letters are classified into their own groups.

(2) Capital and small alphabet letters are randomly classified into two groups.

(3) 50 random patterns with 50-50% white and black pixels are randomly classified into two groups.

Distributions of Hidden Units

Figure 5 shows the hidden unit distributions. Figures (a) and (b) show the results obtained by Method I in Sec. III, and the proposed method including Phases A through C. The training patterns were applied to the network in 3000 different orders. The noise margin α was set to 4.

The distribution of the number of hidden units, obtained through Method I, widely spreads. On the contrary, the proposed method can concentrate them in the narrow interval at the lower region. Furthermore, it is very easy to find the optimum network within a few searching steps.

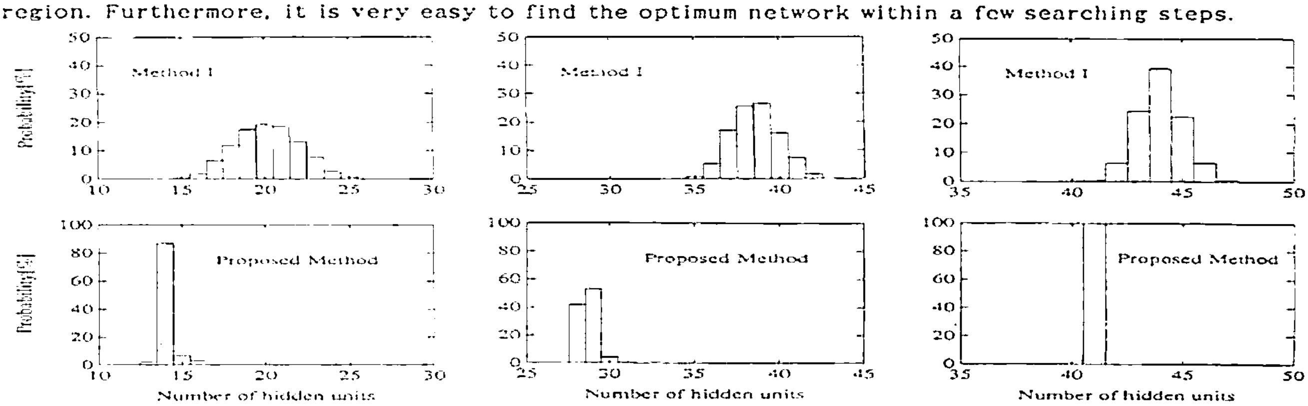

Fig.5 Distribution of hidden units. (a), (b) and (c) show classification problems (1), (2) and (3), respectively.

Accuracy for Noisy Pattern Classification

Another feature of the proposed method is to provide robust classification for noisy patterns. The noise performance has been simulated, by changing the noise margin α, and using the minimum hidden units, for the classification problem (2). Figure 6 shows the simulation results. Insensitivity to noisy patterns can be improved by increasing the noise margin α and the number of hidden units.

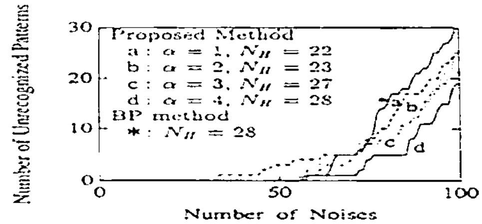

Fig.6 Noisy pattern classification.

Furthermore, the BP algorithm has been investigated using the same number of the hidden units for α=4. The number of unrecognized patterns by BP algorithm is increased from that by the proposed method.

VI CONCLUSIONS

A new synthesis method has been proposed for multilayer neural networks. The connection weights can be expressed with ± 1, and the unit outputs 1 or 0. The minimum hidden units can be obtained, under the given noise margin. This method can be applied to arbitrary binary pattern classification problems.

REFERENCES

[1] D.E.Rumelhart and J.T.McClelland, Parallel Distributed Processing, MIT Press, 1986.

[2] K.Nakayama, S.Inomata and Y.Takeuchi,"A digital multilayer neural network with limited binary expressions", Proc. IJCNN'90, San Diego, pp. II-587-592, June 1990.

[3] K.Nakayama, et al,"Reductions in number of bits for digital realization of multilayer neural networks (in Japanese)", IEICE Trans. vol.J73-D-II, No.8, pp.1336-1345, August 1990.

[4] K.Nakayama and H.Katanama, "A low-bit learning algorithm for digital multilayer neural networks applied to pattern recognition", Proc. IJCNN'91 Singapore, pp.1867-1872, Nov. 1991.

[5] S.Sakaue et al, "Reduction of necessary precision for the learning of pattern recognition", Proc. IJCNN'91 Singapore, pp.1795-1800, Nov. 1991.

[6] K.Nakayama and H.Katayama,"A hybrid multilayer neural network for binary pattern classification and its low-bit learning algorithm", Proc. IJCNN'92 Baltimore, pp.888-893, June 1992.