MVA'90 IAPR Workshop on Machine Vision Applications Nov. 28-30,1990, Tokyo

OPTIMAL INPUT SELECTION OF NEURAL NETWORKS BY SENSITIVITY ANALYSIS AND ITS APPLICATION TO IMAGE RECOGNITION

Hiroshi Takenaga, Shigeo Abe, Masao Takatoo, Masahiro Kayama, Tadaaki Kitamura, Yosiyuki Okuyama

Hitachi Research Laboratory, Hitachi, Ltd.
4026 Kuji-cho, Hitachi-shi, Ibaraki-ken, 319-12 Japan

ABSTRACT

This paper describes a method of selecting optimal inputs of neural networks without lowering recognition rate. In general, neural networks learn a recognition algorithm from inputs and their desired outputs. But if some inputs are redundant, namely they are expressed by other inputs fed to the networks or they do not contribute to recognition, their effect on the recognition algorithm, or their sensitivity to outputs is considered to be very small. Therefore, by analyzing input-output sensitivity of neural networks, redundant inputs can be deleted. Repeatedly applying this method, the optimal inputs can be selected. For an experimental neural network for number recognition with 12 feature inputs, the inputs could be reduced to 8 without lowering recognition rate.

INTRODUCTION

A multi-layered neural network is widely used for pattern recognition because a recognition algorithm is easily generated by the back propagation algorithm. But a difficult problem is how to select inputs and the number of hidden neurons optimally.

There are some reports [2][3][4] on how to determine the optimal number of hidden neurons. Regarding the selection of optimal inputs, however, there are no known methods. Also especially for image recognition, the selection of optimal inputs from features or template patterns extracted from a recognition object is very important to achieve both a high speed processing and a high recognition rate.

In this paper, based on analyzing input-output sensitivity of a neural network, we discuss a method of selecting optimal inputs for pattern recognition without lowering recognition rate.

SENSITIVITY OF NEURAL NETWORK

In general, neural networks learn a recognition algorithm from inputs and their desired outputs. But if some inputs are redundant, namely they are expressed by other inputs or they do not contribute to recognition, their effect on the recognition algorithm, or their sensitivity to output is considered to be very small.

Therefore, by analyzing input-output sensitivity of neural networks, redundant inputs can be deleted.

The method derived from a three-layered network can be easily extended to a network with more than four layers. Therefore, input-output sensitivity presented here is discussed on a three-layered network shown in Fig. 1. For the figure relations of input and output are represented as follows:

$$Y_i(1) = X_i(1) = X_i \tag{1}$$

$$X_i(j) = \sum_{r=1}^{n(j-1)} W_{ri}(j-1) Y_r(j-1) + \theta_i(j) \tag{2}$$

$$Y_i(j) = f(X_i(j)) = 1/\{1+\exp(-X_i(j))\} \tag{3}$$

where X_i: an input of the i-th input neuron;

$X_i(j)$: a total input of the i-th neuron in the j-th layer;

$Y_i(j)$: an output of the i-th neuron in the j-th layer;

$W_{ri}(j-1)$: a weight from the r-th neuron in the (j-1)-th layer to the i-th neuron in the j-th layer;

$n(j)$: the number of neurons in the j-th layer; and

$\theta_i(j)$: a bias of the i-th neuron in the j-th layer.

The k-th output neuron in the output layer corresponds to category k and its desired output is 1 for category k and zero for categories other than k.

Fixing inputs other than X_i, sensitivity of output $Y_k(3)$ against input X_i is given by

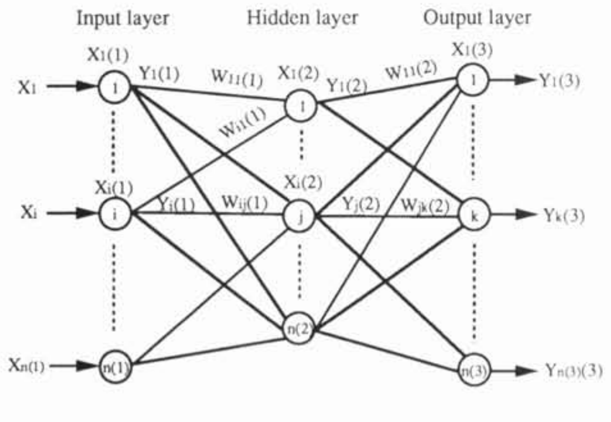


Fig. 1 Three-layered neural network

$$\frac{dY_k(3)}{dX_i} = \frac{dX_k(3)}{dX_i}\frac{dY_k(3)}{dX_k(3)}$$
$$= \left[\sum_{p=1}^{n(2)} W_{ip}(1)W_{pk}(2)\frac{df(X_p(2))}{dX_p(2)}\right]\frac{df(X_k(3))}{dX_k(3)} \quad (4)$$

Since $df(u)/du = f(u)(1-f(u)) > 0$, curves of the sensitivity, $dY_k(3)/dX_i$, are classified into three types as follows:

(a) If $W_{ip}(1)W_{pk}(2) \geq 0$ holds for all $p = 1, ..., n(3)$, $Y_k(3)$ is a monotone increasing function.

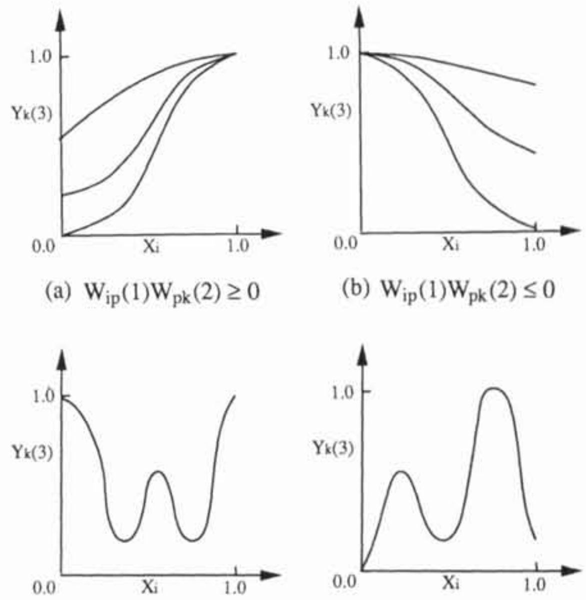
(b) If $W_{ip}(1)W_{pk}(2) \leq 0$ holds for all $p = 1, ..., n(3)$, $Y_k(3)$ is a monotone decreasing function.

(c) If neither (a) nor (b) holds, $dY_k(3)/dX_i$ can be rewritten as follows:

$$\frac{dY_k(3)}{dX_i} = [\sum_{m=1}^{q} W_{im}(1)W_{mk}(2)\frac{df(X_m(2))}{dX_m(2)} +$$
$$\sum_{m=q+1}^{n(2)} W_{im}(1)W_{mk}(2)\frac{df(X_m(2))}{dX_m(2)}]\frac{df(X_k(3))}{dX_k(3)} \quad (5)$$

where $W_{im}(1)W_{mk}(2) \geq 0$ holds for all $m = 1, ..., q$ and $W_{im}(1)W_{mk}(2) \leq 0$ holds for $m = q+1, ..., n(3)$. Namely $Y_k(3)$ is a function composed of monotone increasing functions and monotone decreasing functions.

Therefore, sensitivity curves of output $Y_k(3)$ against input X_i are as shown in Fig. 2. The method of selecting optimal inputs proposed in this paper is to analyze these characteristics and detect redundant inputs for pattern recognition.



(a) $W_{ip}(1)W_{pk}(2) \geq 0$ (b) $W_{ip}(1)W_{pk}(2) \leq 0$

(c) Mixture of $W_{im}(1)W_{mk}(2) \geq 0$ and $W_{im}(1)W_{mk}(2) \leq 0$

Fig. 2 Sensitivity of output Yk(3) against input Xi

ALGORITHM OF OPTIMAL INPUT SELECTION

Fig. 3 shows the algorithm for selecting optimal inputs.

Let

$$(X_{1,1}, ..., X_{i,1}, ..., X_{n(1),1})$$
$$\vdots$$
$$(X_{i,m}, ..., X_{i,m}, ..., X_{n(1),m})$$

be m inputs, all that are used for learning, for category k. To get the sensitivity of output $Y_k(3)$ against input X_i, first fix the inputs other than X_i to representative values, e.g. average values as follows:

$$\widetilde{X_j} = \frac{1}{m}\sum_{r=1}^{m} X_{j,r} \qquad (j \neq i) \qquad (6)$$

Then change input Xi in the range.

Since inputs include representative values for category k, $Y_k(3)$ assumes the value 1 at some X_i. Then we call output $Y_k(3)$ the first candidate and the output other than $Y_k(3)$, which outputs the maximum value, the second candidate. Then we investigate sensitivities of both first and second candidates against an input X_i.

If a first candidate sensitivity curve crosses over a second candidate one as shown in Fig. 3(b), this means that input X_i contributes to classification of category k. Thus we judge input X_i is useful for category recognition and the corresponding table entry (Fig. 3(d)) is marked with a cross. On the other hand, in the case shown in Fig. 3(c), we judge input X_i is redundant and a circle is entered in the table. After repeating these procedures for all inputs and categories, we detect the inputs judged as redundant for almost all the categories. Deleting those inputs from the network, we generate the reduced network by learning. Then for the network, the above procedures are repeated for the remaining inputs until no candidate for deletion, which worsens the recognition rate, is detected.

EXPERIMENTAL RESULTS

The developed method is applied to an experimental neural network for number recognition, with 12 feature inputs and 10 outputs. Training data include 20 data for each number in total a set of 200 data, while the database for the evaluation of the error rates consists of a set of about 7500 data. Initial weights of synapses are set by a uniformly distributed random number, with range A=[0, 0.5], B=[-0.5, 0.5] and C=[-0.5, 0].

Fig. 4 shows the ratio of the error rate against the number of the hidden neurons, assuming the error rate of a neural network with eight neurons and the initial weight range of A as 1.0. The ratios of the error rates of the neural networks with six hidden neurons are the lowest among the same initial weights. Therefore for the networks with six hidden neurons and with the initial weight range of A and B, we applied the optimal input selection method as mentioned above.

Table 1 (a) shows the input-output sensitivity of the neural network A, where a circle represents that the feature input is redundant for the corresponding output and a cross represents that the feature is not. Therefore, the features which are redundant for almost all outputs can be deleted. In Table 1 (a), four features, namely 4, 7, 10 and 12, can be deleted. Table 1 (b) shows the input-output sensitivity of the neural network with the remaining eight feature inputs. As no more features can be deleted, eight inputs are judged optimal. Tables 1 (c), (d) and (e) show the results by analyzing the input-output sensitivity repeatedly for the neural network B. From the results four features, namely 2, 4, 10 and 12, can be deleted. In these cases, the ratios of the error rates are improved as shown by the black circle and black triangle in Fig. 4.

We can reduce 12 feature inputs to eight without lowering recognition rate.

CONCLUSION

We proposed the design algorithm for selecting optimal inputs of a neural network without lowering recognition rate. By using the method based on analyzing input-output sensitivity of a network, we could detect redundant inputs for pattern recognition.

The developed method was applied to an experimental neural network for number recognition with 12 feature inputs. It was seen that 12 feature inputs were reduced to eight without lowering recognition rate.

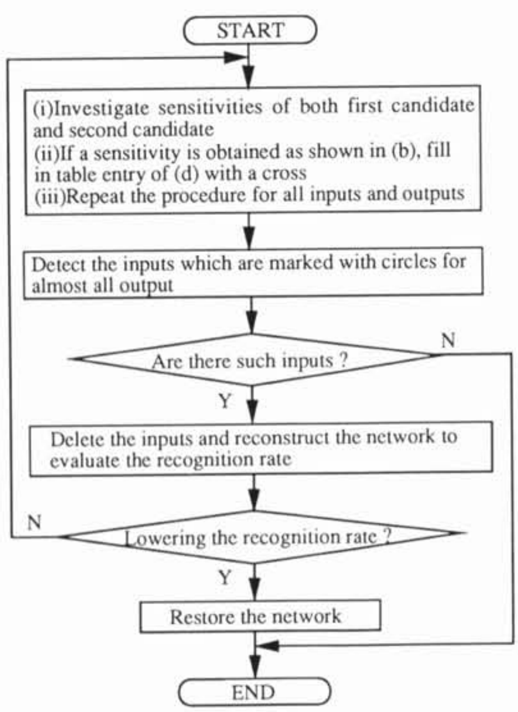
REFERENCES

[1]D. E., Rumelhart, G. E., Hinton, and R. J., Williams,: Learning Internal Representations by Error Propagation, In Parallel Distributed Processing Volume 1, MIT Press, 1986

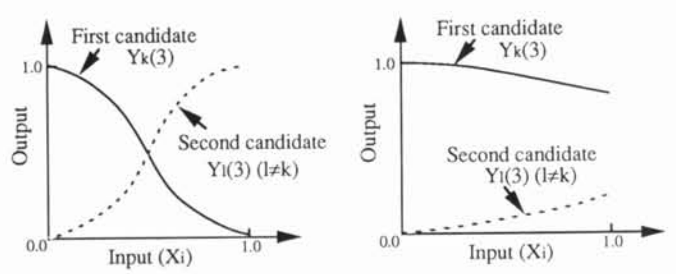
[2]O. Xue et al.: Analyses of the Hidden Units of Back Propagation Model by Singular Value Decomposition(SVD), IJCNN'90-WASH-DC, January 1990

[3]T. Ash: Dynamic Node Creation in Back Propagation Networks, IJCNN'89-WASH-DC, June 1989

[4]M. Kayama: Constructing Optimal Neural Networks by Linear Regression Analysis, accepted to present at Neuro-Niems 90, November 1990



(a) Flowchart



(b) X_i contributes to classification of category k

(c) X_i does not contribute to classification of category k

(d) Table which represents whether an input is redundant or not

Input / Output	X_1	----------	X_i	---------	$X_{n(1)}$
$Y_1(3)$	×		○		×
⋮	⋮		⋮		⋮
$Y_k(3)$	○		○		×
⋮	⋮		⋮		⋮
$Y_{n(3)}(3)$	×		○		○

○ : redundant × : useful

Fig. 3 Algorithm for selecting optimal inputs

Table 1 Input-output sensitivity of number recognition neural network

(a) Neural network with twelve inputs and six hidden neurons and with the initial weight range of A

Output \ Input	1	2	3	4	5	6	7	8	9	10	11	12
0	O	X(8)	X(4)	O	O	X(9)	O	X(6)	X(4)	O	O	O
1	X(8)	O	X(4)	O	O	X(9)	O	X(6)	X(4)	O	O	O
2	O	O	X(6)	O	O	X(3)	O	X(7)	O	O	O	O
3	X(9)	O	X(9)	O	X(9)	O	O	X(2)	X(1)	O	X(5)	O
4	O	O	O	O	X(3)	X(9)	X(6)	X(6)	X(1)	O	X(6)	O
5	O	O	O	O	O	O	O	O	O	O	X(1)	O
6	O	O	O	O	X(2)	O	O	O	O	O	X(4)	O
7	X(4)	O	X(4)	O	X(4)	X(3)	O	X(2)	X(4)	O	O	O
8	X(1)	X(0)	O	O	O	O	O	X(6)	X(4)	X(4)	X(6)	X(0)
9	O	O	O	O	O	X(4)	O	O	O	O	X(5)	O

(b) Neural network with eight inputs remaing where four inputs are deleted from neural network (a)

Output \ Input	1	2	3	5	6	8	9	11
0	O	X(1)	X(4)	X(2)	X(9)	X(8)	X(4)	X(6)
1	X(8)	X(0)	X(4)	X(7)	X(9)	X(8)	X(4)	X(6)
2	O	O	X(4)	X(4)	X(3)	X(7)	O	O
3	X(9)	O	X(9)	X(9)	X(7)	X(2)	X(1)	X(7)
4	O	O	O	X(2)	X(9)	X(8)	X(0)	X(6)
5	X(9)	O	X(1)	X(7)	X(1)	X(6)	X(4)	X(9)
6	O	O	X(4)	X(2)	X(9)	O	X(4)	X(8)
7	X(2)	O	X(4)	X(4)	X(3)	X(2)	X(1)	X(2)
8	X(1)	X(0)	X(4)	O	X(9)	O	X(4)	X(6)
9	O	O	O	O	X(4)	X(8)	X(4)	X(5)

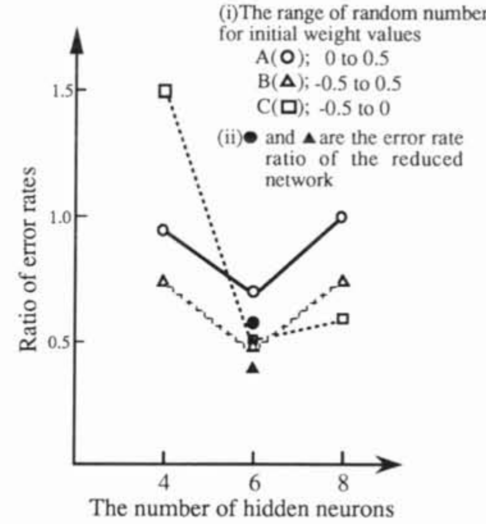


Fig.4 Ratio of error rates plotted against the number of hidden neurons

(c) Neural network with twelve inputs and six hidden neurons and with the initial weight range of B

Output \ Input	1	2	3	4	5	6	7	8	9	10	11	12
0	X(1)	X(1)	X(4)	O	X(2)	X(9)	O	X(6)	X(4)	X(4)	X(6)	O
1	X(8)	O	O	O	X(7)	X(5)	O	X(6)	X(4)	O	X(5)	O
2	O	O	X(8)	O	O	O	O	X(7)	O	O	O	O
3	X(9)	O	X(9)	O	X(4)	O	O	X(2)	X(7)	O	O	X(7)
4	O	O	O	O	X(7)	X(3)	X(1)	X(6)	X(1)	O	X(6)	O
5	O	O	O	O	O	O	O	O	O	O	X(9)	O
6	O	O	O	O	O	O	O	X(6)	X(4)	O	X(1)	O
7	X(8)	X(2)	X(1)	O	X(4)	X(3)	O	X(2)	X(4)	O	X(3)	O
8	X(1)	X(0)	O	O	O	X(6)	O	X(6)	X(4)	X(9)	X(6)	O
9	O	X(4)	O	O	O	O	X(1)	X(6)	X(4)	O	X(5)	X(1)

(d) Neural network with eleven inputs remaing where four inputs are deleted from neural network (c)

Output \ Input	1	2	3	5	6	7	8	9	10	11	12
0	X(1)	O	X(4)	O	X(9)	X(1)	X(6)	X(4)	O	X(6)	O
1	X(8)	O	X(4)	X(2)	X(9)	O	X(6)	X(4)	O	X(6)	O
2	O	O	X(6)	O	O	O	X(7)	O	O	O	O
3	X(9)	O	X(9)	X(9)	X(7)	O	X(2)	X(5)	O	O	O
4	X(8)	O	O	X(3)	X(9)	X(5)	X(6)	X(1)	O	X(6)	O
5	O	O	O	O	X(1)	O	X(1)	X(9)	O	X(1)	O
6	O	O	O	X(2)	X(9)	O	O	O	O	O	O
7	X(3)	O	X(4)	X(4)	X(3)	O	X(2)	X(1)	O	O	O
8	X(4)	O	O	O	O	O	X(6)	X(4)	X(9)	O	O
9	O	O	O	O	X(4)	X(5)	O	O	O	X(5)	O

(e) Neural network with eight inputs remaing where four inputs are deleted from neural network (d)

Output \ Input	1	3	5	6	7	8	9	11
0	X(8)	O	O	X(9)	O	X(6)	X(4)	X(6)
1	X(8)	X(4)	X(7)	X(9)	O	X(6)	X(4)	X(6)
2	O	X(4)	X(4)	X(3)	O	X(7)	X(1)	O
3	X(8)	X(9)	X(9)	X(7)	O	X(2)	X(5)	X(2)
4	X(8)	O	X(3)	X(9)	X(6)	X(6)	X(0)	X(6)
5	O	O	O	X(1)	O	X(6)	X(4)	X(1)
6	X(8)	X(4)	X(2)	X(9)	O	O	X(4)	X(4)
7	X(8)	X(4)	X(4)	X(3)	O	X(2)	X(4)	X(2)
8	X(0)	O	X(0)	X(9)	O	O	O	O
9	O	O	X(3)	X(4)	X(5)	X(6)	O	X(5)

O: redundant input

X: not redundant input

( ): represents the category of the second candidate