Effects of activation functions in multilayer neural network for noisy pattern classification

EFFECTS OF ACTIVATION FUNCTIONS IN MULTILAYER NEURAL NETWORK FOR NOISY PATTERN CLASSIFICATION

Kazuyuki HARA† Kenji NAKAYAMA† ‡
†Graduate School of Nat. Sci. & Tech., Kanazawa Univ.
‡Faculty of Tech., Kanazawa Univ.

2-40-20 , Kodatsuno , Kanazawa , 920 JAPAN
E-mail : nakayama@haspnn1.ec.t.kanazawa-u.ac.jp

ABSTRUCT This paper discusses properties of activation functions in multilayer neural network applied to multi-frequency classification. A rule of thumb for selecting activation functions or their combination is proposed. The sigmoid, Gaussian and sinusoidal functions are employed due to their unique space division properties. Properties of each function and their combinations are discussed based on the internal representation, that is the distributions of the hidden unit inputs and outputs and classification rates with and without noise. The sigmoid function is not effective for a single hidden unit. On the contrary, the other functions can provide good performance. When several hidden units are employed, the sigmoid function becomes useful. However, the convergence speed is still slower than the others. The Gaussian function is sensitive to the additive noise, while the others are rather insensitive. When noise is not included, the Gaussian function is most useful for the convergence rate and the classification accuracy. On the other hand, the additive noise is included, the sigmoid and sinusoidal functions become more effective. These properties are not straight in the combinations. However, their property still remain, and it is possible to select the optimum activation function. This selection also depends on the patterns to be classified.

I INTRODUCTION

Advantage of multilayer neural networks (NNs) trained by the back-propagation (BP) algorithm is to extract common properties, features or rules, which can be used to classify data included in several groups [1]. Especially, when it is difficult to analyze the common features using conventional methods, the supervised learning, using combinations of the known input and output data, becomes very useful.

We studied the multi-frequency signal classification using multilayer neural network[5]-[7]. Since the frequencies are assigned alternately to several groups, it is very difficult to distinguish the waveforms within a short period, and the limited number of samples by conventional methods. The following advantages of the NN over conventional methods were confirmed. The neural network can classify the signals using a small number of samples and a short observation period with which Fourier transform can not classify. The number of calculation is sufficiently smaller than the convolution calculation, required in digital filters.

In the previous work, a sigmoid function was used. However, it is not always optimum. Therefore, properties of activation functions are investigated in this paper. For this purpose, some typical functions are taken into account. They include a sigmoid function, a radial basis function[2] and a periodic function. They will be compared with each other in classifying multi-frequency signals. Effects of noisy signals will be also discussed in the training and classification processes.

As a result, a rule of thumb for selecting the suitable functions and the combination of several kinds of functions will be provided.

II MULTI-FREQUENCY SIGNALS

Multi-frequency signals are defined by

$$x_{pm}(n) = \sum_{r=1}^{R} A_{mr} \sin(\omega_{pr} nT + \phi_{mr}) \quad (1)$$
$$n = 1 \sim N \ , \ \omega_{pr} = 2\pi f_{pr}$$

T is a sampling period. M samples of $x_{pm}(n), m = 1 \sim M$, are included in the group X_p as follows.

$$X_p = \{x_{pm}(n), m = 1 \sim M\}, p = 1 \sim P \quad (2)$$

In one group, the same frequencies are used.

$$F_p = [f_{p1}, f_{p2}, \ldots, f_{pR}] Hz, p = 1 \sim P \quad (3)$$

Amplitude A_{mr} and phase ϕ_{mr} are generated as random numbers, uniformly distributed in following ranges.

$$0 < A_{mr} \leq 1, \quad 0 \leq \phi_{mr} < 2\pi \quad (4)$$

III MULTILAYER NEURAL NETWORK

3.1 Network Structure and Equations

A single-layer neural network is taken into account. N samples of the signal $x_{pm}(n)$ are applied to the input layer in parallel. The nth input unit receives x_{pm}(n). Connection weight from the nth input to the jth hidden unit is denoted w_{nj}. The input and output of the jth hidden unit are given by

$$net_j = \sum_{n=0}^{N-1} w_{nj} x_{pm}(n) + \theta_j \tag{5}$$

$$y_j = f_H(net_j) \tag{6}$$

Letting the connection weight from the jth hidden unit to the kth output unit be w_{jk}, the input and output of the kth output unit are given by

$$net_k = \sum_{j=0}^{J-1} w_{jk} y_j + \theta_k \tag{7}$$

$$y_k = f_O(net_k) \tag{8}$$

The activation function of the output layer is the sigmoid function.

The number of output units is equal to that of the signal groups P. The neural network is trained so that a single output unit responds to one of the signal groups.

3.2 Training and Classification

Signals are categorized into training and untraining sets, denoted X_{Tp} and X_{Up}, respectively. Their elements are expressed by $x_{Tpm}(n)$ and $x_{Upm}(n)$, respectively.

The neural network is trained by using x_{Tpm}(n), m = 1 $\sim$ M_T, for the pth group. Here, M_T is the number of the training data. After the training is completed, the untrained signals x_{Upm}(n) are applied to the NN, and the output is calculated. For the input signal x_{Upm}(n), if the pth output y_p has the maximum value, then the signal is exactly classified. Otherwise, the network fails in classification.

IV SELECTION OF ACTIVATION FUNCTIONS

What kinds of activation functions should be selected is very important. At the same time, it is a very difficult problem. In this paper, the following typical functions are selected for the hidden layer.

When binary target can be considered, then the sigmoid function can be used in the output layer.

Sigmoid function:

$$y_j = f_{sig}(net_j) = \frac{1}{1 + e^{-(net_j)}} \tag{9}$$

Sinusoidal function:

$$y_j = f_{sin}(net_j) = \sin(\pi net_j) \tag{10}$$

Gaussian function:

$$y_j = f_{gau}(net_j) = e^{-net_j^2} \tag{11}$$

The input vectors are distributed in a N-dimensional space. Three functions divide the space as follows:

$$f_{sig}(net_j) \begin{cases} > \alpha_+, & net_j > T_{sig} \\ < \alpha_-, & net_j < T_{sig} \end{cases} \tag{12}$$

$$f_{sin}(net_j) \begin{cases} > \alpha_+, & |net_j - (2n\pi + \frac{\pi}{2})| < T_{sin} \\ < \alpha_-, & |net_j - (2n\pi + \frac{3}{2}\pi)| < T_{sin} \end{cases} \tag{13}$$

$$f_{gau}(net_j) \begin{cases} > \alpha_+, & |net_j| < T_{gau} \\ < \alpha_-, & |net_j| > T_{gau} \end{cases} \tag{14}$$

Here, n is integer.

These space division fundamental, and independent to each other. This is an idea behind selecting the above three functions.

Next step of selecting activation functions is how to combine them. It is also highly dependent on the distribution of the input signals, and is very hard to determine before hand. For this reason, both the homogeneous function and the composite functions are investigated.

V SIMULATION OF TRAINING AND CLASSIFICATION WITHOUT NOISE

5.1 Multi-frequency Signals

The number of frequency components is R = 3, and the signal groups is P = 2, respectively. The frequency components are located alternately between the groups as follows: F_1 = [1, 2, 3] Hz for Group 1 (#1) and F_2 = [1.5, 2.5, 3.5] Hz for Group 2 (#2). The sampling frequency is 10 Hz, that is T = 0.1 sec. The number of samples N is 10. Therefore, the observation interval is 1 sec.

5.2 Training and Classification

x_{Tpm}(n), m = 1 $\sim$ 200 and x_{Upm}(n), m = 1 $\sim$ 1800 are used. Simulation results are shown in Table 1. The training converged using three hidden units for all activation functions. In the case of the Gaussian and the sinusoidal function, the training almost converged with one hidden unit. Detailed discussion

will be provided in Sec. 7.

Table 1:Classification rates by three functions[%]

Activation Function	Hidden Unit	Training		Untraining	
		#1	#2	#1	#2
Sigmoid	1	44.5	100	47.9	100
	3	100	100	97.4	100
Sinusoidal	1	86.0	99.0	79.8	99.0
	3	100	100	92.6	100
Gaussian	1	99.5	100	98.1	100
	3	100	100	99.1	99.9

VI SIMULATION USING THREE ACTIVATION FUNCTIONS

6.1 Additive Noise

White noise, denoted *noise(n)*, is generated as random number, and is added to the signal $x_{pm}(n)$. Noisy signal $x'_{pm}(n)$ is given by

$$x'_{pm}(n) = x_{pm}(n) + noise(n) \tag{15}$$

6.2 Training and Classification

The noisy multi-frequency signals are used for training. N is 10 and M is 200 for each group. After training, untraining signals with white noise are applied, and classification rates are evaluated. White noise is uniformly distributed in the range ± 0.5. The results are shown in Table 2. Columns with (A) and (B) list the recognition rates using the training signals without and with white noise, respectively. The NN trained without noise is also used for comparison. From these results, it can be confirmed that training using noisy signals is useful to achieve robustness.

Table 2: Classification rates using training signals (A) without and (B) with white noise [%]

Activation Function	Hidden Unit	(A)		(B)	
		#1	#2	#1	#2
Sigmoid	1	47.0	52.9	92.8	28.5
	3	97.3	8.4	82.6	78.0
Sinusoidal	1	80.2	20.9	61.7	87.7
	3	65.9	36.2	79.9	82.7
Gaussian	1	98.2	4.8	71.7	65.9
	3	85.3	46.3	79.8	70.2

6.3 Convergence Rates

Figure 1 shows learning curves obtained using the three hidden units. The NN with the Gaussian function can converge faster than the other. However, the error does not well decreased. The NN with the sinusoidal function can also converge faster. At the same time, the error can be well decreased. A convergence rate using the sigmoid function is slow. However, the error can reach to the same level as in using the sinusoidal function.

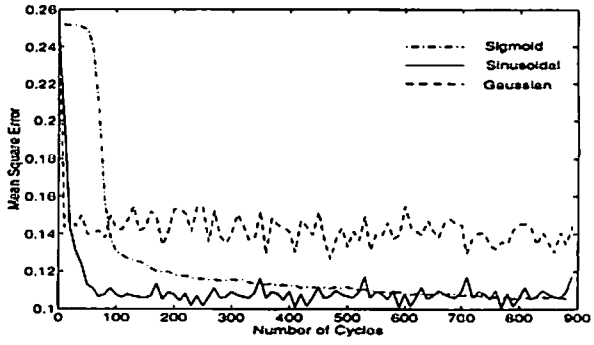

Figure 1: Learning curves

VII Convergence Property Using Single Hidden Unit

7.1 Pure Multi-frequency Signals

The NNs trained without noise are further investigated by hidden unit input and output distribution. Figure 2 illustrates this distribution, using the sigmoid (a1), the sinusoidal (b1) and the Gaussian functions (c1).

In the case of the sigmoid function, the data #1 and the data #2 have to be located the right or left side. This is a fundamental space division property of the sigmoid function. Thus, the network have to adjust the weights, with which the hidden unit input data are completely separated into the right or the left side. The data #2 is concentrated at the edge of the α_+ as shown in Eq.(12), but the data #1 is distributed widely. From this result, the distribution of the hidden unit inputs generated by the multi-frequency signals cannot satisfy the requirements given by Eq.(12).

In the case of the sinusoidal function, the hidden unit inputs of the data #2 locate near one of the peaks and the data #1 distributed widely. The sinusoidal function have large differential coefficient except for the peak. Then the data #2 can be shifted around one of the peaks fast. On the other hand, the data #1 can locate in the region of $f_{sin}(net_j) < \alpha_-$. Therefore, the requirement of the fundamental division property given by Eq.(13) is satisfied by the multi-frequency signals.

In the case of the Gaussian function, the data #2 locate around the peak. Differential coefficients around the peak are large, then, the data #2 can be shifted toward this area very fast. Most of the data #1 are distributed both sides.

From these results, the hidden unit inputs of the multi-frequency signals can be concentrated on a narrow range for one group, and the other is distributed widely for the other group.

Thus, the space division property of the Gaussian

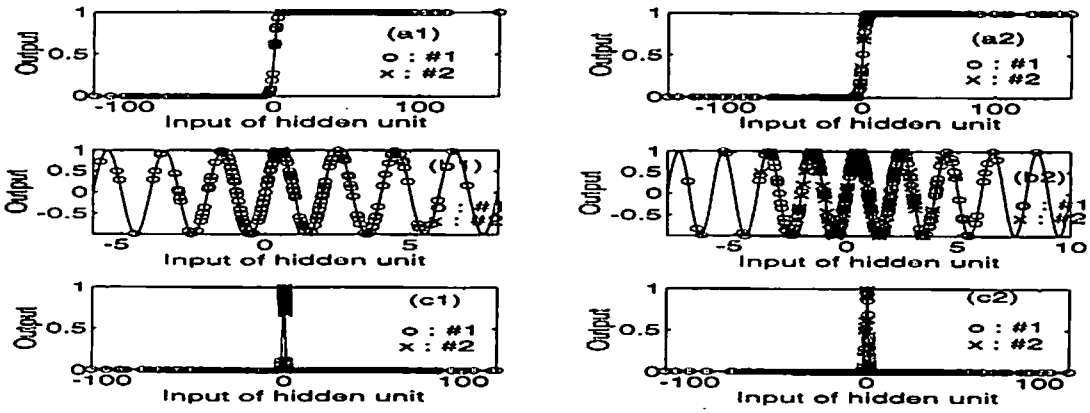

Figure 2: Hidden unit input and output distributions

function is best match with the distribution of the multi-frequency signals. This function can provide the best accuracy as shown in Table 1.

7.2 Noisy Multi-frequency Signals

In Figure 2 , (a2), (b2) and (c2) correspond to the hidden unit inputs and output distributions, in which random noise is added. The network is trained by using the pure multi-frequency signals. After the training, the untrained noisy signals are applied to the NN. The distribution of the hidden unit inputs are easily spread by adding the noise.

In the case of the sigmoid, the data #2 distributed widely. However, the most of the data #2 still remain in its own region. Because it has wide stable regions. This is a reason why it can provide better accuracy than the others.

In the case of the Gaussian, the data #2 distributed over the other region. Because a single peak is very narrow. Then these data easily move over the other group's region. Thus, the accuracy is decreased by adding the noise.

The sinusoidal case, the data #2 also widely distributed. However, the sinusoidal function is a periodic function, having several narrow stable regions. Thus, it can provide higher accuracy than that of the Gaussian function.

VIII Convergence Property Using Several Hidden Units

8.1 Homogeneous Activation Funtions

Figures 3, 5 and 7 show distributions of the hidden unit inputs and outputs. The NNs are trained by using the signals without noise. The sigmoid, the sinusoidal and the Gaussian functions are separately used. For each figure, (a), (b) and (c) correspond to one of the hidden unit. (a1), (b1) and (c1) are the response for the data #1, and (a2), (b2) and (c2) are for the data #2.

From these figures, there are two type of distributions, that is concentrated and dispersed distributions. One of two groups locates at near the peak of the functions and the other is widely spread. The overlap of the distributions between the two groups cause miss classification.

In Fig.3, it is very interesting that the data #2 locate at the middle of the slope. Since this region is not a stable region, it can be expected that accuracy is easily degraded by adding the noise. As shown in Table 2, it is true. The classification rates are 97.3% for the data #1 and 8.4% for the data #2. Accuracy for the data #2 is greatly reduced.

Figures 4, 6 and 8 show distribution of the inputs of the two output units. In these figures, (a) and (b) correspond to the data #1 and the data #2, respectively. The region of overlap of the solid and the doted lines will cause miss classification. We can investigate from these figures, how the hidden units separate the signals into two groups. In the case of the data #2 is applied, there are no overlap. So, the hidden unit input space is well separated. In the case of the data #1 is applied, there are some overlap. These overlaps cause miss classification. These results are consistent with the accuracies shown in Table 1.

From the figures, the input space of the output units are well separated by the sigmoid and sinusoidal function. So, it can be concluded that three hidden units cooperate to make the distribution of the inputs to the output unit to be linearly separable.

8.2 Composite Activation Functions

Three functions can be combined in the same hid-

den layer. This combination is called 'Composite Activation Function' in this paper.

Table 3 shows classification rates using the multi-frequency signals without noise. In this table, the symbols D through J correspond to the combination of the functions.

The combination C, having three Gaussian functions, achieves the best accuracy. The convergence rate is also the fastest among three functions. The combination D, having all activation functions, achieves better accuracy than the others except for C. However, I and J, which include two Gaussian functions, are worse than D.

K through M are compared with E through J. E and F are better than K. Then adding both the sinusoidal and the Gaussian to the sigmoid can improve the performance. H is better than L, but G is worse than L. Then adding the Gaussian to the sinusoidal can improve, while the sigmoid can not do.

In the most of the combinations, the Gaussian achieves better accuracy. Then, property of each function does not appear straightly in the combinations.

Table 4 shows classification rates of the network trained using the noisy signals. Training itself did not converge in all cases. This means that the accuracy is not 100% for all combinations of the functions. The network using the homogeneous activation function A and B has higher accuracy than the others. However, C does not achieve better accuracy than the others. Then the homogeneous activation function can not always achieve better accuracy than the composite activation functions.

Table 3: Classification rates using signals without noise

	Combination			Training		Untraining		
	Sig	Sin	Gauss	#1	#2	#1	#2	Ave.
A	3	0	0	100	100	97.4	100	98.7
B	0	3	0	100	100	92.6	100	96.3
C	0	0	3	100	100	99.1	99.9	99.5
D	1	1	1	100	100	100	98.3	99.1
E	2	1	0	99.5	100	96.6	98.4	97.5
F	2	0	1	100	100	97.4	100	98.7
G	1	2	0	93.5	98.5	83.8	97.3	90.6
H	0	2	1	100	100	99.9	97.8	98.9
I	1	0	2	100	100	96.2	99.6	97.9
J	0	1	2	100	100	97.3	98.9	98.1
K	2	0	0	99.0	100	94.0	100	97.2
L	0	2	0	86.0	95.5	86.8	97.3	92.1
M	0	0	2	99.5	98.5	99.4	98.8	99.1

The network using the composite activation function J has higher accuracy, while C and I have worse accuracy than the others. G and H also provide good accuracy. E and F achieve worse accuracy while A provides good one.

K through M are compared with E through J. G and H are better than L. Then adding the sigmoid or the Gaussian to the sinusoidal works well. K is better than E and F. Then adding both the sinusoidal and the Gaussian to the sigmoid does not work well.

The sinusoidal and sigmoid functions achieve good accuracy in the most of the combinations. However, the sinusoidal combination does not always achieve better accuracy. Thus, property of each function is not straight in the combination, as previously discussed in the no additive noise case.

Table 4: Classification rates using signals with noise

	Combination			Training		Untraining		
	Sig	Sin	Gauss	#1	#2	#1	#2	Ave.
A	3	0	0	83.5	86.0	82.6	78.9	80.8
B	0	3	0	84.5	89.0	79.9	82.7	81.3
C	0	0	3	87.0	81.5	79.8	70.2	75.0
D	1	1	1	77.0	92.5	69.1	84.3	77.6
E	2	1	0	88.5	77.0	80.9	67.8	74.4
F	2	0	1	78.5	98.5	63.8	85.9	74.9
G	1	2	0	74.0	92.5	69.4	87.0	78.2
H	0	2	1	79.0	92.5	72.3	84.3	78.3
I	1	0	2	84.0	87.5	73.5	75.9	74.7
J	0	1	2	84.5	82.0	81.0	78.5	79.8
K	2	0	0	91.5	70.5	81.3	69.3	75.3
L	0	2	0	80.3	83.0	79.1	73.6	76.4
M	0	0	2	75.5	85.0	74.6	76.1	75.4

IX CONCLUSIONS

Properties of the activation functions for multi-frequency signal classification has been discussed using multilayer neural network supervised by BP algorithm. The Gaussian function can provide the highest performance for the signals without noise. However, it is sensitive to the additive noise. The sigmoid function is not useful for a single hidden unit. If several hidden units are used, then the sigmoid function becomes useful, and is insensitive to the additive noise. The sinusoidal function is useful for noisy signal.

References

[1] D.E.Rumelhart and J.L.McCelland et al, "Parallel Distributed Processing", MIT Press, 1986.

[2] Philip D. Wasserman, "Advanced Methods in Neural Computing", Van Nostrand Reinhold, pp.147–155, 1993.

[3] G.Veciana and A.Zakhor, "Neural Net-Based Continuous Phase Modulation Receivers", IEEE Transaction on communications, vol.40, No.8, 1992.

[4] J.Karhunen, J.Joutsensalo, "Tracking of sinusoidal frequencies by neural network learning algorithms", IEEE, CH2977-7/91/0000-3177, 1991.

[5] K.Hara and K.Nakayama, "Multi-frequency signal classification using multilayer neural network trained by backpropagation algorithm (in Japanese)", Tech., Rep. IEICE, NC92-75, pp.47–54, 1992.

[6] K.Hara and K.Nakayama, "High resolution of multi- frequencies using multilayer networks trained by back-propagation algorithm", Proc. WCNN'93, Portland Oregon, vol.IV, pp.675–678, 1993.

[7] K.Hara and K.Nakayama, "Classification of multi-frequency signals with random noise using multilayer neural networks", Proc. IJCNN'93, Nagoya Japan, vol.I, pp.601–604, 1993.

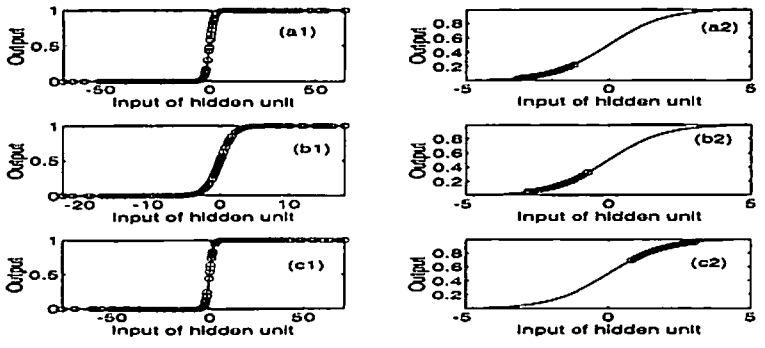

Figure 3: Distribution of sigmoid hidden unit outputs

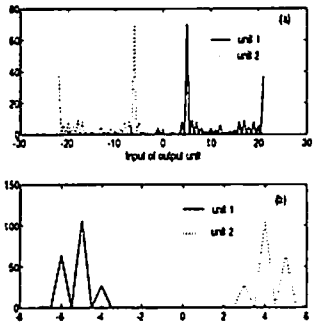

Figure 4: Distribution of output unit inputs

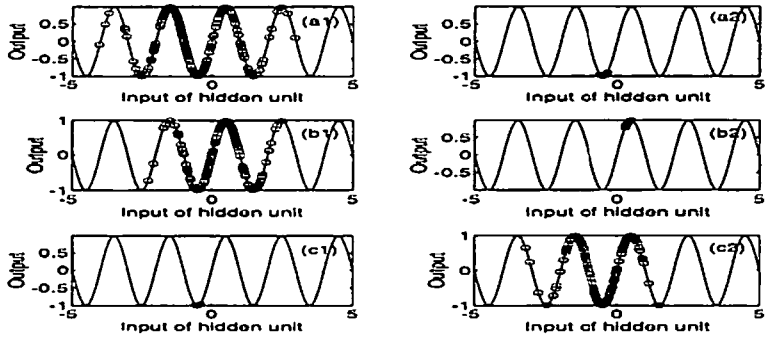

Figure 5: Distribution of sinusoidal hidden unit outputs

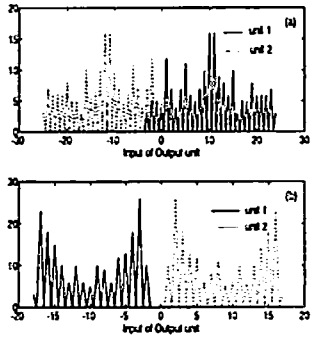

Figure 6: Distribution of output unit inputs

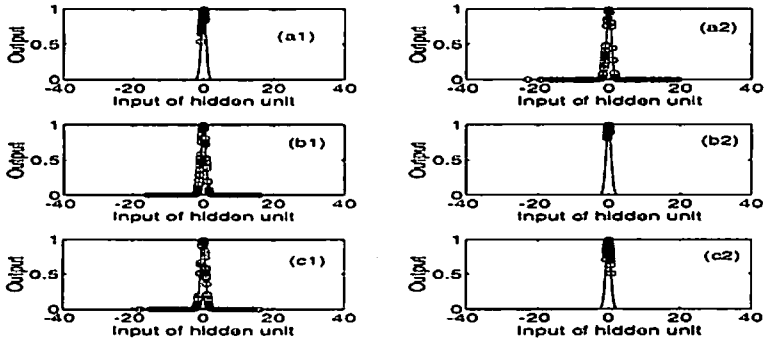

Figure 7: Distribution of Gaussian hidden unit outputs

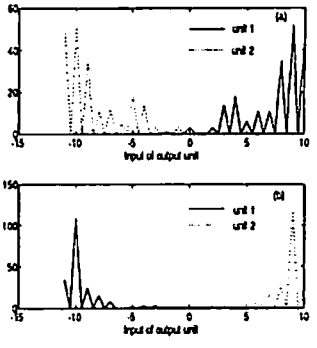

Figure 8: Distribution of output unit inputs