

Classification of Multi-frequency Signals with Random Noise Using Multilayer Neural Networks

Kazuyuki HARA

Graduate School of Nat. Sci. & Tech.,

Kenji NAKAYAMA

Faculty of Tech., Kanazawa Univ.

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

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

nakayama@haspnn1.ec.t.kanazawa-u.ac.jp

ABSTRACT Frequency analysis capability of multilayer neural networks, trained by back-propagation (BP) algorithm is investigated. Multi-frequency signal classification is taken into account for this purpose. The number of frequency sets, that is signal groups, is 2~5, and the number of frequencies included in a signal group is 3~5. The frequencies are alternately located among the signal groups. Through computer simulation, it has been confirmed that the neural network has very high resolution. Classification rates are about 99.5% for training signals, and 99.0% for untraining signals. The results are compared with conventional methods, including Euclidean distance with accuracy of about 65%, Fourier transform with accuracy of about 10~30%, and using very high-Q filters with a huge number of computations. The neural network requires only the same number of inner products as the hidden units. Frequency sensitivity and robustness for the random noise are studied. The networks show high frequency sensitivity, namely, the networks have high frequency resolution. Random noise are added to the multi-frequency signals to investigate how does the network cancel uncorrelated noise among the signals. By increasing the number of samples, or training signals, effects of random noise can be cancelled.

I INTRODUCTION

Advantage of multilayer neural networks 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. This application field includes, for instance, pronunciation of English text, speech recognition, image compression, sonar target analysis, stock market prediction and so on [2]-[6]. In this paper, classification performance of the neural networks is discussed based on frequency analysis. Multi-frequency signals are employed for this purpose. Especially, we are interested in super-resolution, which observation interval and the number of samples are very limited. Performances of multilayer neural networks will be investigated based on very limited information. Furthermore, the results are compared with conventional methods in the frequency analysis field, including Euclidean distance, Fourier transform and Filtering methods. Robustness for random noise are also investigated.

II MULTI-FREQUENCY SIGNALS

Multi-frequency signals are defined by

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

M samples of $X_{pm}(n), m = 1 \sim M$, are included in the group X_p as follows.

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

P signal groups, $X_p, p = 1 \sim P$, are assumed.

T is a sampling period. The signals have N samples. In one group, the same frequencies are used.

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

Amplitude A_{mr} and phase ϕ_{mr} are different for each frequency in the same group. They are generated as random numbers, uniformly distributed in following ranges.

$$0 < A_{mr} \leq 1 \quad (4)$$

$$0 \leq \phi_{mr} < 2\pi \quad (5)$$

III MULTILAYER NEURAL NETWORK

A two-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 the sample at nT.

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.

Training and Classification

Sets of signals are categorized into training and untraining data sets, denoted by X_{Tp} and X_{Up} , re-

spectively. Their elements are expressed by $X_{Tpm}(n)$ and $X_{Upm}(n)$, respectively.

$$X_p = [X_{Tp}, X_{Up}] \quad (6)$$

$$X_{Tp} = \{X_{Tpm}(n), m = 1 \sim M_T\} \quad (7)$$

$$X_{Up} = \{X_{Upm}(n), m = 1 \sim M_U\} \quad (8)$$

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

IV SIMULATION

4.1 Multi-frequency signals

Seven kinds of multi-frequency signals are used as shown in Table 1. The number of frequency components are 3 ~ 5, and the signal groups are 2 ~ 5, respectively. In all cases, the frequency components are located alternately between the groups. In Case-2.2 and 3.3, the number of samples is $N=15$ and $N=20$, the sampling period is $T=1/15$ and 0.05 sec, respectively. In other cases, the sampling frequency is 10Hz, that is $T=0.1$ sec. The number of samples is $N=10$. Therefore, the observation interval is 1 sec in all cases.

4.2 Neural Network Classification

Table 1 illustrates simulation results. The training signal set of each group is 200, that is $X_{Tpm}(n), m=1 \sim 200$. 1800 signals are used as untraining signals in each group. Namely, $X_{Upm}(n), m=1 \sim 1800$. In Case-1.1, 1.2 and 2.2, training converged using one hidden unit. Accuracy for X_{Tp} is, therefore 100%. For the untraining signals, the classification rate is around 99%. In other cases, the training did not completely converge. However, classification accuracy is also very high, that is about 99.5%. Thus, highly exact classification can be achieved. Furthermore, improving accuracy rates of Case-3.2 is investigated. Increasing the number of samples n of $X_{pm}(n)$, accuracy improved step by step. 40 samples are needed at least to have 100% accuracy.

4.3 Frequency Sensitivity

Frequency sensitivity correspond to frequency resolution. When one of the frequency components is slightly shifted, that is, approaches to another group's, it is necessary for the networks not to respond. It is measured by difference between the sifted frequency signals and not shifted one. Case-1.1 is used. Frequency components are shifted in a

Table 1: classification results by neural network [%]

CASE	Frequency Sets [Hz]	Hidden Units	Accuracy	
			X_{Tp}	X_{Up}
1.1	$F_1 = [1, 2, 3]$	1	100	99.8
	$F_2 = [1.5, 2.5, 3.5]$		100	94.6
1.2	$F_1 = [1, 2, 3]$	1	100	99.7
	$F_2 = [1.1, 2.1, 3.1]$		100	99.7
2.1	$F_1 = [1.1, 5, 2, 2.5, 8]$	5	82.8	77.2
	$F_2 = [1.25, 1.75, 2.25, 2.75, 3.25]$		86.8	82.3
2.2	$F_1 = [1, 2, 3, 4, 5]$	1	100	99.9
	$F_2 = [1.5, 2.5, 3.5, 4.5, 5.5]$		100	99.9
3.1	$F_1 = [1, 2, 3]$	4	99.0	93.5
	$F_2 = [1.33, 2.33, 3.33]$		99.5	98.8
	$F_3 = [1.67, 2.67, 3.67]$		100	99.6
3.2	$F_1 = [1, 2, 3]$	8	98.0	96.3
	$F_2 = [1.2, 2.2, 3.2]$		79.5	75.3
	$F_3 = [1.4, 2.4, 3.4]$		98.5	98.1
	$F_4 = [1.6, 2.6, 3.6]$		89.0	87.2
	$F_5 = [1.8, 2.8, 3.8]$		98.0	87.8
3.3	$F_1 = [1, 4, 7]$	8	99.5	98.4
	$F_2 = [1.5, 4.5, 7.5]$		99.5	98.5
	$F_3 = [2, 5, 8]$		99.0	99.1
	$F_4 = [2.5, 5.5, 8.5]$		99.0	97.2
	$F_5 = [3, 6, 9]$		100	99.6

range of ± 0.5 Hz and only one component is shifted. Figure 1 shows the results. From this results, the network has high frequency sensitivity.

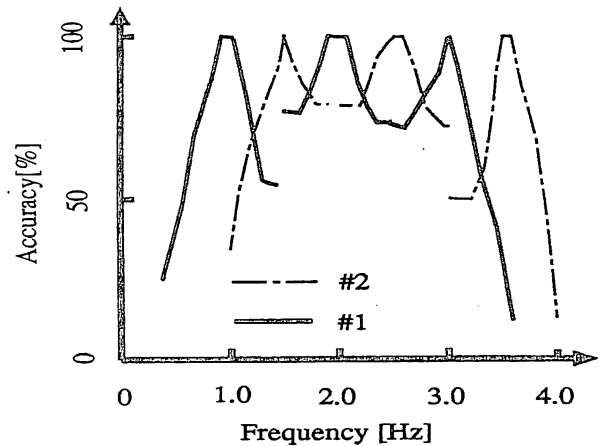


Figure 1: Frequency sensitivity (Case-1.1)

V FREQUENCY ANALYSIS BY CONVENTIONAL METHODS

Fourier transform and filter analysis are very popular in frequency domain analysis. Multi-frequency signal classification using these two methods are carried out and compared with neural networks.

5.1 Euclidean Distance Analysis

Similarity between the training and untraining signals can be evaluated using Euclidean distance in some sense. It is defined by

$$D_{pq}(m, m') = \left\{ \frac{1}{N} \sum_{n=1}^N (X_{Tpm}(n) - X_{Upm'}(n))^2 \right\}^{\frac{1}{2}} \quad (9)$$

In this section, classification based on this distance is investigated. The Euclidean distances between

$X_{Upm}(n)$ and all other training signals are calculated. If the training signal, having the minimum Euclidean distance, is included in X_{Tp} , then $X_{pm}(n)$ is classified into the p th group.

In Case-1.1 and 1.2, accuracy is 98 % and 61.5 %, respectively. Thus, the Euclidean distance method is impractical to distinguish the frequencies located close each other, such as Case-1.2.

5.2 Fourier Transform Analysis

Fourier transform of a discrete-time signal X_{pm} is given by

$$G_{pm}(e^{j\omega T}) = \sum_{n=0}^{N-1} X_{pm}(n)e^{-j\omega nT} \quad (10)$$

Classification is carried out as follows:

Let $|G_{pm}(e^{j\omega T})|$ be $A_{pm}(f)$ for convenience.

Rule 1: If $A_{pm}(f_{pi}) > A_{pm}(f_{qi})$ for all q ($q \neq p$) and $i = 1 \sim R$, then $X_{pm}(n)$ is classified into the p th group.

Rule 2: If $A_{pm}(f_{pi}) > A_{pm}(f_{qi})$ at more than half of $i = 1 \sim R$ for all q , then $X_{pm}(n)$ is classified into the p th group.

For all cases, classification rates using Rule 1 are not high. At the best case, that is Case-3.3, accuracy rate is less than 60 %. By Rule 2, accuracy rate can be improved little, but they are still low rate.

In case of limited number of samples and observation interval as like Case-1.2, 2.1, 3.1 and 3.2, Fourier transform classify the signals with very low accuracy compare with neural networks. Ability of Fourier transform is further investigated by increasing the number of samples. This method requires about 60 samples and 100 samples for Case-1.1 and 1.2, respectively, in order to achieve the same resolution as the neural networks. Furthermore, it requires complex coefficients as shown in Eq.(10).

5.3 Filter Analysis

Frequency component extraction is also possible using digital filters with real coefficients. This method use same number of filters as groups. Each filters need to extract only the group frequency components, so very high-Q filters are required. In this simulation, very long impulse response having 4000 samples was used. Accuracy rate is 100 % for all cases. However, this method requires a huge number of computations compared with the neural network.

In Case-1.1, 1.2 and 2.2, training was converged with one hidden unit. Compare these two based on the calculation, it is same as that the filter can distinguish the signals into a group with just one output sample. From investigation, it was impossible.

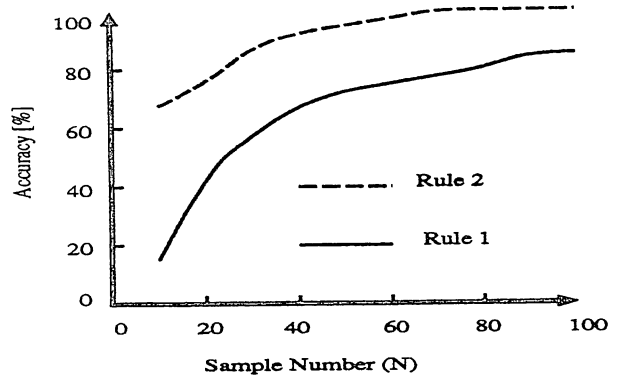


Figure 2: Sample number and accuracy of Fourier transform

VI ROBUSTNESS FOR RANDOM NOISE

Another interesting point of neural network performances is robustness for random noise, that is uncorrelated interference.

It can be expected that the neural network can be insensitive after training using a large number of signals including uncorrelated interference. Case-1.1 is used for this investigation. Random noise is added to both the training and the untraining signals.

6.1 Training signals with and without Noise

The multi-frequency signals with and without noise are used for training the neural network. The number of samples in $X_{Tpm}(n)$, N is 10, and the number of training signals, M is 200 for each group. A single hidden unit is used. After training, untraining signals with random noise are applied, and classification rates are evaluated. Random noise is uniformly distributed in the range 0.1 or 0.5. The results are shown in Table 2. Signal(A) and (B) indicate the training signals without and with random noise, respectively. From these results, noisy training signals are useful to achieve robustness. Also, performance is highly related to the noise level.

Table 2: Classification rates using training signals without(A) and with(B) random noise

Noise Amplitude	(A)	(B)
± 0.1	95.0 %	98.0 %
± 0.5	71.0 %	85.4 %

6.2 Effects of Increasing Training Samples

Random numbers are uncorrelated to each other. However, this can be held for a large number of samples. In order to guarantee uncorrelation among noise samples, the number of samples of $X_{Tpm}(n)$, N , or the number of training signals, M , is increased.

Robustness for noisy patterns is investigated in the same way as in 6.1. Two combinations, that is ($N=20$, $M=200$) and ($N=10$, $M=400$) are evaluated. Data with underline indicate increased information. Table 3 shows simulation results in both cases. Classification rates can be greatly increased. Large N can increase uncorrelation among noise samples, at the same time, signal information. On the contrary, large M can only increase uncorrelation among noise samples. Signal information is not increased.

Connection weights from the input layer to the single hidden unit are shown in Figure 2. They are obtained by using training signals without(a) and with(b) noise. By adding random noise, connection weight distribution is modified in order to cancel noise effects, while maintaining classification capability.

From these results, it can be concluded that the multilayer neural network inherently hold capability of removing effects of uncorrelated interference. Furthermore, robustness to noise depends on a signal to noise ratio, and independence among the noise samples.

Table 3: Classification rates using $N=20$ or $M=400$

noise amplitude	$N=20$, $M=200$	$N=10$, $M=400$
± 0.1	99.9 %	98.7 %
± 0.5	97.2 %	94.2 %

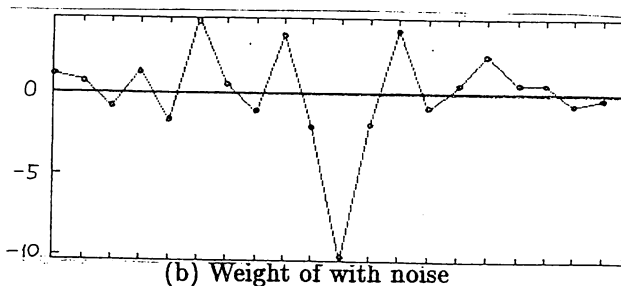
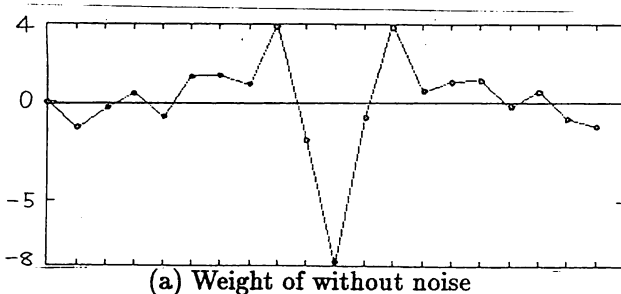


Figure 3: Example of the network weights

VII CONCLUSIONS

Frequency resolution capability of the neural network has been discussed. Comparing conventional

methods, including Euclidean distance, Fourier transform and filtering methods, the multilayer neural networks are superior to them in the following points. First, the neural network can resolve the multi-frequency signals with very high accuracy using limited information. Second, computational requirements are very small. Robustness for random noise can be obtained by additive training using noisy signals. The network is modified so as to suppress noise effects, while precisely extracts frequency components.

References

- [1] D.E.Rumelhart and J.L.McCelland et al, Parallel Distributed Processing, MIT Press, 1986.
- [2] T.J.Sejnowski and C.R.Rosenberg, "Parallel networks that learn to pronounce English text", Univ. of California, San Diego, 1987.
- [3] J.L.Elman and Zipper, "Learning the hidden structure of speech", Tech. Rep. 8701, Univ. of California, San Diego, 1987.
- [4] G.W.Cottrell, R.Munro and D.Zipser, "Image compression by backpropagation: An example of extension programming", Tech., Rep. 8702, Univ. of California, San Diego, 1987.
- [5] R.P.Gorman and T.J.Sejnowski, "Analysis of hidden units in a layered network trained to classify sonar targets", Neural Networks, vol.1, pp.75-90 1988.
- [6] T.kimoto et al., "Stock market prediction system with modular neural networks", Proc. IJCNN'90, San Diego, pp.I-1-6, 1990
- [7] A.V.Oppenheim and R.W.Shafer, Digital Signal Processing, Prentice-Hall, Inc., 1975.
- [8] K.Hara and K.Nakayama, "Multi-frequency signal classification using multilayer neural network trained by backpropagation algorithm", Tech., Rep. IEICE, NC92-75, pp.47-54, 1992.
- [9] K.Hara and K.Nakayama, "Superresolution of multi-frequency signals using multilayer neural network supervised by backpropagation algorithm", Proc. ICNN'93, San Francisco, pp.III-1993.