PROCEEDINGS

5th IEEE International Workshop on

# Robot and Human Communication

November 11-14, 1996 Tsukuba

'96

# RO-MAN

IEEE

IEEE Catalog Number : 96TH8179
Library of Congress Number : 96-75514
ISBN : 0-7803-3253-9 (Softbound Edition)
0-7803-3254-7 (Microfiche Edition)

CO-SPONSORED BY
- IEEE Industrial Electronics Society
- The Society of Instrument and Control Engineers
- The Robotics Society of Japan
- The Japan Society of Mechanical Engineers
- New Technology Foundation

# Pattern Classification of Time-series EEG Signals Using Neural Networks

Osamu FUKUDA, Toshio TSUJI and Makoto KANEKO
Faculty of Engineering, Hiroshima University,
Higashi-Hiroshima, Japan

## Abstract

*This paper proposes a pattern classification method of time-series EEG signals using neural networks. To achieve successful classification for non-stationary EEG signals, a new network structure that combines a probabilistic neural network and recurrent neural filters is used. This network is suitable to express statistical and time-varying characteristics of time-series EEG signals. In the experiments, two types of photic stimulation caused by eye opening/closing and by artificial light are used to measure the EEG data. It is shown that the proposed network can achieve high classification performance.*

## 1 Introduction

An EEG signal pattern changes depending on external or internal factors such as photic stimulation, auditory stimulation, and intentions of movements. These factors may be used as an interface in virtual reality and tele-operation devices, or as a communication tool for handicapped persons if the operator's intention can be estimated from the EEG pattern.

Up to the present time, some investigations of EEG pattern classification using neural networks have been carried out in [1][2]. Most of them, however, dealt with research on an automatic diagnosis in clinical medicine, and only few studies were concerned with developing a new interface tool [3]. In case of the pattern classification of EEG signals using back propagation neural networks [4], the networks need a large number of training data, learning iterations, and a large scale of structure. Also, it is very difficult to attain high classification performance.

On the other hand, we had already proposed an EEG pattern classification method [5] utilizing a probabilistic neural network called Log-Linearized Gaussian Mixture Neural Network (LLGMN) [6]. This network can construct the statistic model of the EEG signals through learning and improve classification ability. This method, however, does not consider the time-varying characteristics of the EEG signals, so that classification performance may decrease because of the non-stationality.

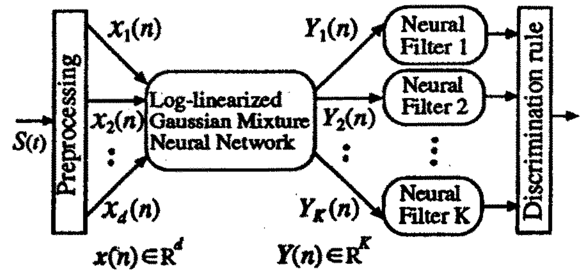



Figure 1: Structure of the proposed network

In order to classify the time-varying EEG signals accurately enough, the present paper proposes a new network structure considering statistical and time-varying characteristics of the time-series EEG signals. To construct the statistical model and optimal filter using neural networks, the LLGMN [6] and recurrent neural filters [7] are used in our approach.

## 2 Method

### 2.1 Network Structure

Figure 1 shows structure of the proposed network. First, the EEG signal $S(t)$ is pre-processed and converted into the input feature vector $x(n) \in \Re^d$ $(n = 1, \cdots, N)$. Next, the LLGMN receives it, and outputs the posteriori probability $Y(n) \in \Re^K$ $(n = 1, \cdots, N)$ of the input feature vector belonging to each class. Then the neural filter modifies this posteriori probability. Finally, the Bayes decision theory is used to determine the specific class.

#### A) Log-Linearized Gaussian Mixture Neural Network

The structure of the LLGMN [6] is shown in Fig. 2. This network is of feedforward type and contains three layers. First, the input feature vector $x(n)$ is

IEEE International Workshop on
Robot and Human Communication
0-7803-3253-9/96 $5.00 ©1996 IEEE

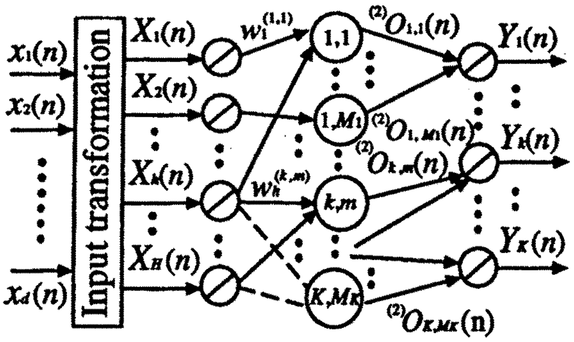



Figure 2: Structure of the LLGMN

transformed into the modified vector $\boldsymbol{X}(n) \in \Re^H (H = 1 + d(d+3)/2)$ in order to represent the probability density function corresponding to each component of the Gaussian Mixture Model (GMM) [8] as a linear combination of $\boldsymbol{X}(n)$:

$$\begin{aligned} \boldsymbol{X}(n) &= [1, x(n)^{\mathrm{T}}, x_1(n)^2, x_1(n)x_2(n), \cdots, \\ &\quad x_1(n)x_d(n), x_2(n)^2, x_2(n)x_3(n), \\ &\quad \cdots, x_2(n)x_d(n), \cdots, x_d(n)^2]^{\mathrm{T}}. \end{aligned} \quad (1)$$

The first layer consists of $H$ units corresponding to the dimension of $\boldsymbol{X}(n)$ and the identity function is used for activation of each unit.

The second layer consists of the same number of units as the total component number of the GMM. Each unit receives the output of the first layer weighted by the coefficient $w_h^{(k,m)}$ and outputs the posteriori probability of each component. The input to the unit $\{k, m\}$ in the second layer, ${}^{(2)}I_{k,m}(n)$, and the output, ${}^{(2)}O_{k,m}(n)$, are defined as

$${}^{(2)}I_{k,m}(n) = \sum_{h=1}^{H} w_h^{(k,m)(1)}O_h(n), \quad (2)$$

$${}^{(2)}O_{k,m}(n) = \frac{\exp[{}^{(2)}I_{k,m}(n)]}{\sum_{k'=1}^{K}\sum_{m'=1}^{M_K} \exp[{}^{(2)}I_{k',m'}(n)]}, \quad (3)$$

where ${}^{(1)}O_h(n)$ denotes the output of the $h$-th unit in the first layer, and $w_h^{(K,M_k)} = 0$ $(h = 1, \cdots, H)$. It should be noted that (3) can be considered as a generalized sigmoid function.

Finally, the third layer consists of $K$ units corresponding to the number of classes, and outputs the posteriori probability of the class $k$ $(k = 1, \cdots, K)$. The relationship between the input and the output in the third layer is defined as

$${}^{(3)}I_k(n) = \sum_{m=1}^{M_k} {}^{(2)}O_{k,m}(n), \quad (4)$$

$$Y_k(n) = {}^{(3)}I_k(n). \quad (5)$$

In the LLGMN described above, the posteriori probability of each class is defined as output of the last layer. Note that the log-linearized Gaussian mixture structure is incorporated into the network by learning the weight coefficients $w_h^{(k,m)}$.

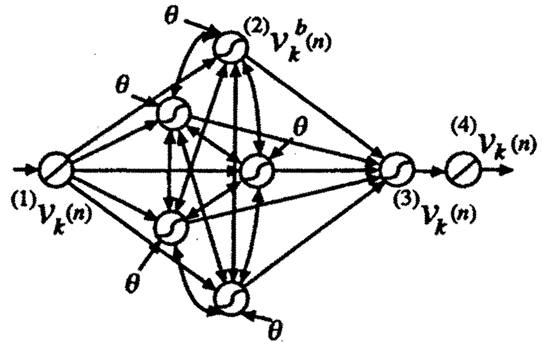



Figure 3: Structure of the NF

### B) Neural Filter

Figure 3 shows the structure of the neural filter (NF) [7]. The unit in the first layer receives the $n$-th outputs $Y_k(n)$ of the LLGMN, and send ${}^{(1)}v_k(n)$ to the second layer. The identity function is used for the activation function in the first layer.

The second layer consists of $B$ units. Each unit receives the $n$-th output of the first layer and the $(n$-1$)$-th output of the second layer. Also, each unit in this layer has the bias input $(\theta = 1)$. The fully interconnected units keep the internal representation so that the time history of the input data can be taken into consideration. The input to the unit $b$ in the second layer, ${}^{(2)}r_k^b(n)$, and the output, ${}^{(2)}v_k^b(n)$, are defined as

$${}^{(2)}r_k^b(n) = \sum_{a=1}^{B} {}^{(2,2)}u_k^{a,b(2)}v_k^a(n-1)$$

$$+{}^{(1,2)}u_k^{b(1)}v_k(n) + {}^{(\theta)}u_k^b, \quad (6)$$

$$^{(2)}v_k^b(n) = g({}^{(2)}r_k^b(n)), \quad (7)$$

where ${}^{(2,2)}u_k^{a,b}$, ${}^{(1,2)}u_k^b$ and ${}^{(\theta)}u_k^b$ denote the weight coefficients between the $a$-th and the $b$-th unit in the second layer, between the unit in the first layer and the $b$-th unit in the second layer, and between the bias input and the $b$-th unit in the second layer, respectively. The activation function $g(x)$ is the sigmoid function defined as $g(x) = 1/(1+\exp(-x))$. The unit in the third layer is connected to all the units in the second layer, and the relationship between the input and the output of the unit is defined as

$$^{(3)}r_k(n) = \sum_{b=1}^{B} {}^{(2,3)}u_k^b \, {}^{(2)}v_k^b(n), \quad (8)$$

$$^{(3)}v_k(n) = g({}^{(3)}r_k(n)), \quad (9)$$

where ${}^{(3)}r_k(n)$ and ${}^{(3)}v_k(n)$ denote the input and the output in the third layer, and ${}^{(2,3)}u_k^b$ denotes the weight coefficient between the $b$-th unit in the second layer and two units in the third layer.

The identity function is used as the activation function in the fourth layer, and the output is defined as

$$^{(4)}v_k(n) = {}^{(3,4)}u_k \, {}^{(3)}v_k(n), \quad (10)$$

where ${}^{(3,4)}u_k$ denotes the weight coefficient between the third layer and the fourth layer. Note that the weight coefficient ${}^{(3,4)}u_k$ functions as a gain.

## 2.2 Learning Algorithm

If the teacher signal is given only to the output unit in the NF, the error may back-propagate from the NF to the LLGMN, so that the learning is performed for both the networks at the same time. However, the appropriate error back-propagation between the NF and the LLGMN can not be guaranteed because of the complexity of the network structure.

Therefore, we introduce the following two step learning schedule that divides the learning into the LLGMN and the NF. First, the LLGMN is trained using the training data in order to construct the statistical model of the training data. Then another set of the input feature vector $x(n)$ is given and the LLGMN outputs the posteriori probability $Y_k(n)$ $(k = 1, \cdots, K)$. Next, each NF is trained using this output data and the teacher signal $T_k(n)$ $(k = 1, \cdots, K)$ given for each output unit in order to construct a kind of the optimal filter.

### A) Learning Rule of the LLGMN

Now, let us consider the supervised learning with the teacher vector $T(n) = (T_1(n), \cdots, T_k(n), \cdots, T_K(n))^{\mathrm{T}}$ for the $n$-th input vector $x(n)$. When the teacher provides perfect classification, $T_k(n) = 1$ for the particular class $k$ and $T_k(n) = 0$ for all the other classes. As an energy function for the network, we use

$$\begin{aligned} J &= \sum_{n=1}^{N} J_n \\ &= -\sum_{n=1}^{N}\sum_{k=1}^{K} T_k(n) \log Y_k(n), \end{aligned} \quad (11)$$

and the learning is performed to minimize it, that is, to maximize the likelihood. For $x(n)$, the weight modification $\Delta w_h^{(k,m)}$ of the corresponding weight $w_h^{(k,m)} (h = 1, \cdots, H)$ is defined as

$$\Delta w_h^{(k,m)} = -\eta_1 \frac{\partial J_n}{\partial w_h^{(k,m)}}, \quad (12)$$

in a sequential learning scheme, and

$$\Delta w_h^{(k,m)} = -\eta_1 \sum_{n=1}^{N} \frac{\partial J_n}{\partial w_h^{(k,m)}}, \quad (13)$$

in a collective learning scheme. Here, $\eta_1 > 0$ is the learning rate.

### B) Learning Rule of the NF

The energy function for the $k$-th NF is defined as

$$E^k = \sum_{n=1}^{N} E_n^k = \sum_{n=1}^{N} \frac{1}{2}({}^{(4)}v_k(n) - T_k(n))^2, \quad (14)$$

where $T_k(n)$ is the teacher signal for the output of the $k$-th NF. The learning is performed to minimize this sum of the square errors. The weight coefficients from the fourth layer to the second layer are modified using the error back propagation learning. On the other hands, in the first and second layers, the weight coefficients are modified using the error back-propagation through time [4] because of the interconnection in the second layer. According to this algorithm, the weight modification $\Delta{}^{(2,2)}u_k^{a,b}$ in the second layer is defined as

$$\Delta{}^{(2,2)}u_k^{a,b} = -\eta_2 \sum_{n=1}^{N} \frac{\partial E_n^k}{\partial{}^{(2,2)}u_k^{a,b}}, \quad (15)$$

$$\frac{\partial E_n^k}{\partial^{(2,2)}u_k^{a,b}} = \sum_{p=n-4}^{n} {}^{(2)}\delta_k^b(p)^{(2)}v_k^a(p-1), \qquad (16)$$

where $\eta_2 > 0$ denotes the learning rate. In the present paper, the history of the input data is considered back to four steps. Also the weight modification $\Delta^{(1,2)}u_k^b$ can be calculate in the same way.

The generalized error ${}^{(2)}\delta_k^b(p)$ is the sensitivity of the square error $E_n^k$ to the input ${}^{(2)}r_k^b(p)$, which is defined as the following recursive calculation:

$${}^{(2)}\delta_k^b(p) = g'({}^{(2)}r_k^b(p))^{(2)}\varepsilon_k^b(p), \qquad (17)$$

$${}^{(2)}\varepsilon_k^b(p-1) = \sum_{a=1}^{B} {}^{(2,2)}u_k^{a,b(2)}\delta_k^a(p), \qquad (18)$$

Note that, the sensitivity of the square error $E_n^k$ to the $n$-th output ${}^{(2)}v_k^b(n)$ is calculated as

$${}^{(2)}\varepsilon_k^b(n) = {}^{(2,3)}u_k^{b(3)}\delta_k(n), \qquad (19)$$

where ${}^{(3)}\delta_k(n)$ is the sensitivity of the square error $E_n^k$ to the input ${}^{(3)}r_k(n)$, and ${}^{(2)}\varepsilon_k^b(n)$ denotes the sensitivity of the square error $E_n^k$ to the output ${}^{(2)}v_k^b(n)$. Also, $g'(x)$ is the derivative of the sigmoid function.

# 3 Experiments

## 3.1 Experimental Apparatus [5]

To examine the use of the EEG signals as a human interface tool, simple and handy electroence phalograph (IBVA, Random ELECTRONICS DESIGN) is used. This enables us to measure EEG signals in usual environments. The experimental system consists of the head band, the transmitter, and the receiver.

The transmitter is attached to the head band. The time-series EEG signals measured from the electrodes are digitized by the A/D converter (the sampling frequency =120Hz, quantization = 8bits) after they are amplified and filtered out through high-pass (3Hz) and low-pass (40Hz) analogue filters. The size of the transmitter is quite compact (93mm × 51mm × 25mm). The personal computer, which is connected to the receiver, collects the data. The surface electrodes are located at Fp1 and Fp2 that are specified by the International 10-20 Electrode System. The noise in the EEG signals can be removed significantly by the bipolar derivation between the two electrodes at Fp1 and Fp2.

Table 1: Frequency ranges used in the classification experiments

| $d$ | Frequency ranges (Hz) | | | | | |
|---|---|---|---|---|---|---|
| 2 | 0~8 | 9~35 | - | - | - | - |
| 3 | 0~8 | 9~20 | 21~35 | - | - | - |
| 4 | 0~8 | 9~12 | 13~20 | 21~35 | - | - |
| 5 | 0~4 | 5~8 | 9~12 | 13~20 | 21~35 | - |
| 6 | 0~2 | 3~4 | 5~8 | 9~12 | 13~20 | 21~35 |
| | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ |

$d$ : Dimension of the input vector

## 3.2 Experimental Conditions

The time-series EEG signals are measured under the following two conditions.

[1] Photic stimulation by opening and closing eyes

Subjects are seated in a well-lighted room. First, the time-series EEG signals are measured during both eye opening and closing (60 seconds for each). The measured signals are used as training data. Next, subjects are asked to switch their eye states alternatively according to the pseudo-random series for 450 seconds.

[2] Photic Stimulation by the artificial light

Subjects are seated in a dark room, and open their eyes. An artificial flash light (xenon, illuminating power: 0.176[J]) is set at the distance of 50 cm apart from their eyes. The light turning on and off with the frequency 4Hz is used as the artificial photic stimulation.

The power spectral density function of the measured time-series EEG signals is estimated using FFT for every 128 sampled data. The function is divided into several ranges (from 0 to 35Hz). The frequency bands of this range are determined based on the clinical use of the brain wave ($\delta, \theta, \alpha, \beta$). Time-series of the mean values of the power spectral density function within each frequency ranges are calculated and normalized between [0, 1] in each range. Thus, the multi-dimensional data ($x_1$, $x_2$, $\cdots$, $x_d$) are obtained and used as the input feature vector to the networks. Here $d$ denotes the number of the frequency ranges. The frequency ranges used in the experiments are shown in Table 1.

# 4 Results

In the experiments, the time-series EEG signals are measured from five subjects. The second layer of the LLGMN consists of six units (three for each class) corresponding to the total component number of the GMM, and the second layer of the NF consists of eight units.

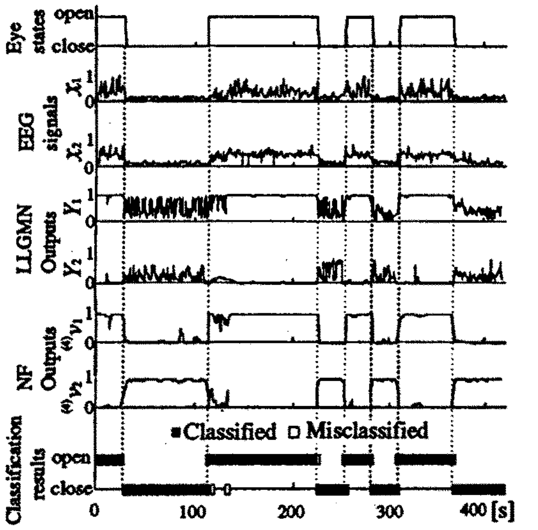



Figure 4: An example of the classification result

Table 2: Classification results of the eye states

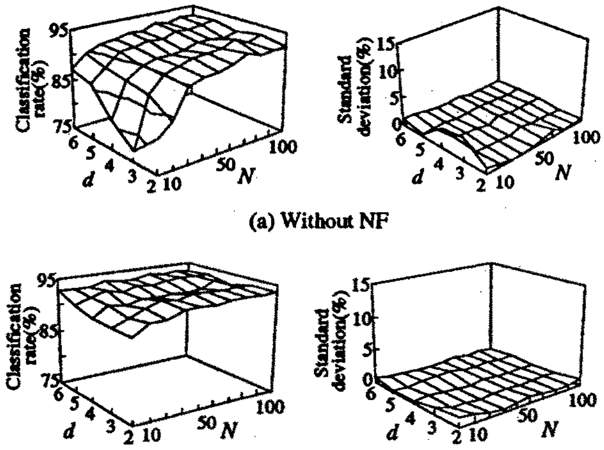
| | Subject | A(male) | B(male) | C(male) | D(male) | E(female) |
|---|---|---|---|---|---|---|
| Without NF | Classificationrate(%) | 91.1 | 83.3 | 88.6 | 81.3 | 93.2 |
| | Standarddeviation(%) | 0.4 | 0.6 | 1.4 | 1.1 | 0.6 |
| Proposed net | Classificationrate(%) | 94.5 | 90.4 | 93.8 | 89.7 | 93.4 |
| | Standarddeviation(%) | 0.3 | 0.8 | 1.1 | 0.5 | 0.1 |

The NF is trained using $N = 168$ data according to the pseudo-random series for 180 seconds ($\eta_2 = 0.001$). Then the ratio of the correct classification to 422 data, which are not used in the learning, is computed.

## 4.1 EEG Classification of the Eye States

Figure 4 shows the classification result of the proposed network (subject A), where the input feature vector is two dimensional data for two classes shown in Table 1 ($d = 2, H = 6, K = 2$). The LLGMN is trained using $N = 100$ data (50 for each class). In the figure, the timing of switching eye states, the EEG signals ($x_1, x_2$), the output of the LLGMN ($Y_1, Y_2$), the output of the NF (${}^{(4)}v_1, {}^{(4)}v_2$) and the classification results are shown. It can be seen that the NF makes the output of the LLGMN considerably smooth. In this case, the LLGMN achieves considerably high performance with 95.3 percent of the classification rate. A few misclassified data are observed immediately after switching eye states.

Table 2 shows classification results for five subjects. The mean values and the standard deviations of the classification rate for 30 kinds of initial weights, which are randomly chosen, are shown. As can be inspected, the proposed network achieves considerably high classification performance for all the subjects.




(a) Without NF

(b) Proposed network

$d$ : Number of the input vector $N$ : Number of the training data

Figure 5: Effect of the training data on classification results of the eye states

Next, we examine the changes of the classification rates with the number of training data $N$ and the dimension of the input vector $d$ taken from Table 1. For each input vector, the number of training data $N$ are changed from 10 to 100. The networks are trained for fifty sets of the training data ($N = 10, 20, \cdots, 100$, $d = 2, 3, \cdots, 6$). Then the ratio of the correct classification to 422 data, which are not used in learning, is computed.

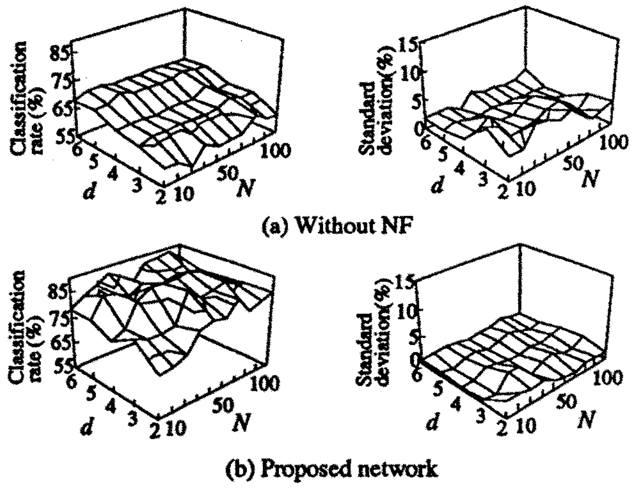
Figures 5 show the mean values and the standard deviations of the classification rate for ten kinds of the initial weights. Although both the networks can achieve high classification rate for large number of training data, the difference becomes clear as the number of the training data decreases. The proposed network keeps the classification rate high even for small sample size of the training data.

## 4.2 EEG Classification of the Artificial Photic Stimulation

Next, the pattern classification experiments are carried out under the artificial photic stimulation. The experimental results for five subjects are shown in Table 3. The dimensions of the input vector $d = 2, 6$ and the number of the training data $N = 50, 100$ are used in the learning procedure. The ratio of the correct classification to 422 data, which are not used in learning, is computed. Compared to the classification

Table 3: Classification results of the artificial photic stimulation

| Number of the learning data | | $N=50$ | | $N=100$ | |
|---|---|---|---|---|---|
| Dimension of the input vector | | $d=1$ | $d=5$ | $d=1$ | $d=5$ |
| subject A (male) | Classification rate(%) | 83.2 | 84.5 | 84.4 | 85.4 |
| | Standard deviation(%) | 1.5 | 0.6 | 1.4 | 1.1 |
| subject B (male) | Classification rate(%) | 89.8 | 92.3 | 92.3 | 93.1 |
| | Standard deviation(%) | 0.9 | 0.4 | 1.1 | 0.7 |
| subject C (male) | Classification rate(%) | 63.8 | 66.8 | 79.5 | 81.1 |
| | Standard deviation(%) | 1.0 | 5.2 | 0.5 | 2.6 |
| subject D (male) | Classification rate(%) | 78.0 | 78.9 | 80.2 | 80.3 |
| | Standard deviation(%) | 2.4 | 1.7 | 0.2 | 0.1 |
| subject E (male) | Classification rate(%) | 75.6 | 77.3 | 82.4 | 83.1 |
| | Standard deviation(%) | 1.2 | 0.8 | 0.9 | 0.6 |




$d$ : Number of the input vector $N$ : Number of the training data

Figure 6: Effect of the training data on classification results of the artificial photic stimulation

result of the eye states, the classification rates of the artificial photic stimulation decrease. The classification rates tend to improve with increase of the number of training data from $N=50$ to $N=100$ and the dimension of the input vector from $d=2$ to $d=6$. Also, the standard deviations of the classification rates tend to decrease.

Finally, we examine the changes of the classification rates depending on the number of the training data $N$ and the dimension of the input vector $d$. Figure 6 shows the classification result (subject A), where effect of the NF on classification results is clear. Thus, with the use of NF we can improve the classification ability of the network significantly.

## 5 Conclusion

The present paper proposes a new neural network structure that combines the LLGMN and the NF. To examine the classification ability of the proposed network, EEG pattern classification experiments have been performed. The proposed network realized considerably high classification ability for the time-series EEG signals.

Our future research will be directed to develop a technique to incorporate dynamic statistical model into the neural network. This work was supported in part by Tateisi Science and Technology Foundation.

## References

[1] F. Y. Wu, J. D. Slater, L. S. Honih and R. E. Ramsay, "A neural network for event-related potential diagnosis," Computer in biology and machine, 23, pp. 251-264, 1993.

[2] N. Schaltenbrand, R. Lengelle and J. P. Macher, "Neural network model : Application to automatic analysis of human sleep," Computer and biomedical research, 26, pp. 157-171, 1993.

[3] A. Hiraiwa, K. Simohara and Y. Tokunaga, "EMG pattern analysis and classification by neural network," IEEE International Conference on Syst., Man and Cybern., p. 1113, 1989.

[4] D. E. Rumelhart, J. L. McClelland and R. J. Williams, "Learning Internal Representations by Error propagation," in *Parallel Distributed Processing vol. I*, pp. 318-362, MIT Press, 1986.

[5] O. Fukuda, T. Tsuji and M. Kaneko, "Pattern Classification of EEG Signals Using a Log-Linearized Gaussian Mixture Neural Network," IEEE International Conference on Neural Networks, pp. 2479-2484, 1995.

[6] T. Tsuji, H. Ichinobe, O. Fukuda and M. Kaneko, "A Maximum Likelihood Neural Network Based on a Log-Linearized Gaussian Mixture Model," IEEE International Conference on Neural Networks, pp. 1293-1298, 1995.

[7] J. T. H. Lo, "Synthetic Approach to Optimal Filtering," IEEE Trans. Neural Networks, 5, pp. 803-811, 1994.

[8] J. H. Wolfe, "Pattern clustering by multivariate mixture analysis," *Multivariate Behavioral Res.*, vol. 5, pp. 329-350, 1970.