Biol. Cybern. 67, 303–308 (1992)

Biological Cybernetics

Quantitative measurement of two-component pH-sensitive colorimetric spectra using multilayer neural networks

Chii-Wann Lin [1], Joseph C. LaManna [2], and Yashiyasu Takefuji [3]

[1] Department of Biomedical Engineering and [2] Department of Neurology, [3] Department of Electrical Engineering and Applied Physics, Center of Automation and Intelligent System Research, Case Western Reserve University, Cleveland, OH 44106, USA

Received November 18, 1991/Accepted in revised form April 6, 1992

Abstract. The purpose of this research was to develop a noise tolerant and faster processing approach for in vivo and in vitro spectrophotometric applications where distorted spectra are difficult to interpret quantitatively. A PC based multilayer neural network with a sigmoid activation function and a generalized delta learning rule was trained with a two component (protonated and unprotonated form) pH-dependent spectrum generated from microspectrophotometry of the vital dye neutral red (NR). The network makes use of the digitized absorption spectrum between 375 and 675 nm. The number of nodes in the input layer was determined by the required resolution. The number of output nodes determined the step size of the quantization value used to distinguish the input spectra (i.e. defined the number of distinct output steps). Mathematic analysis provided the conditions for which this network is guaranteed to converge. Simulation results showed that features of the input spectrum were successfully identified and stored in the weight matrix of the input and hidden layers. After convergent training with typical spectra, a calibration curve was constructed to interpret the output layer activity and therefore, predict interpolated pH values of unknown spectra. With its built-in redundant presentation, this approach needed no preprocessing procedures (baseline correction or intensive signal averaging) normally used in multicomponent analyses. The identification of unknown spectra with the activities of the output layer is a one step process using the convergent weight matrix. After learning from examples, real time applications can be accomplished without solving multiple linear equations as in the multiple linear regression method. This method can be generalized to pattern oriented sensory information processing and multi-sensor data fusion for quantitative measurement purposes.

Correspondence to: C. LaManna

1 Introduction

It has been shown that a simple linear neuron model with Hebbian-type synaptic modification can perform a principal component analysis (Oja 1982; Rubner and Schulten 1990) and preserve the topological characteristic of the input pattern (Kohonen 1982). The proof for the optimal convergence without local minima has been derived by Kohonen 1974; Sanger 1989; Baldi and Hornik 1989. However, it holds true for linear independent input patterns with linear units only. With merely linear independent patterns, the storage matrix will have crosstalk problems with the encoded information. Under such nonideal conditions, the generalized delta rule (GDR) has been shown by Rumelhart and McClelland (1988) to be a learning rule that can deal with non-orthogonal input patterns. Due to the similarity of the learning objective function to the one in multiple linear regression, this learning process will produce a least square solution with significant components when the input patterns are not linearly independent. This feature detection or extraction property of self-organized behavior using local rules, like Hebb's rule, has been used to explain opponent color processing by Rubner and Schulten (1990). A transformation from input unit coordinates to output pattern coordinates with isomorphism using linear neurons has been used to account for the mapping from the neural space to the conceptual space in Rumelhart's book (1988). Several important issues of this transformation have been addressed in this book, as well, including the effect of damaged units on lower and higher levels of description, and the case of merely linearly independent patterns. Another important issue regarding this transformation, i.e., the capability for quantitative usage of this model, has not yet been elucidated. Unlike the works done by Mayer et al. (1991) and Wythoff et al. (1990) of spectral pattern recognition using neural network as a classifier with discrete index, we used the network to distinguish the similarity between the spectra with continuous output. In this paper, we demonstrate the feasibility of a quan-

titative application with feature detection of a self-organized neural network using a two-component spectrum of the pH sensitive dye, (NR), generated from spectrophotometric data.

Neutral red has been used biologically for intracellular pH measurement because of its activity in the physiological range (pH6–pH8) and because it is relatively nontoxic (LaManna 1987). It has two distinguishing absorption peaks, one characteristic of the acid form of the dye (550 nm) and the other characteristic of the basic form of the dye (450 nm). The ratio of these two absorption peaks follows Henderson-Hasselbalch equation, is proportional to the solution pH and independent of dye concentration or path length (Macdonald et al. 1977). This method has been applied to the measurement of intracellular pH in in-vitro hippocampal brain slices (Sick et al. 1989) and in vivo in brain cortex (LaManna et al. 1984).

2 The network model

The working network is a three-layered, feed-forward model, consisting of an input layer, a hidden layer and an output layer with $N_i = 25$, $N_j = 25$ and $N_k = 8$ nodes, shown as Fig. 1. All the units exhibit real, continuous-valued activities and are bounded between 0 and 1, I and O for input and output. The set of presented patterns is denoted by P^n and they are normalized using the Euclidean vector norm. The output activities of the input layer, o_{Ni}, correspond to the values of presented patterns, i.e. $o_{Ni} = p^n$. Activities of the units of the hidden and output layers are a sigmoidal transformation of the linear summation of their inputs (i) weighted by the synaptic strengths (w), $o = f(iw)$. The convergence criterion is determined by the value of the objective function between the expected patterns (T) and the actual output activities of the output layer (O), $\sum_p \sum_j (t_{pj} - 0_{pj})^2/2$, which is set to 0.01 times the number of the output nodes in the simulation. The weight modification is done according to the GDR following presentation of pattern p, $\Delta_p w_{ji} = \eta * \Psi_{pj} * i_{pi}$, where η is a small positive constant for learning rate, $\Psi_{pj} = (t_{pj} - o_{pj}) f_j'(i * w)$ for the output layer and $\Psi_{pj} = f_j'(i * w) \sum_k \Psi_{pk} * w_{kj}$ for the hidden layer. If the learning rate is small enough, the update of the weight matrix can be performed after presentation of all patterns.

The concept underlying this approach is based on analogy to the hypothetical function of neurons in the brain. Because of the fact that information is carried by the firing frequency instead of membrane potential of the neurons, we use the notation of 1 and 0 for maximum firing frequency and silence of the neurons respectively. This is based on the consideration of limited resources and refractory period in the real situation. The input pattern is normalized to facilitate the convergence toward its principal components. Instead of doing normalization of the weight matrix after each iteration cycle as suggested by Oja (1982), we relax this restriction to let it freely evolve towards convergence. By doing so, we can use the maximum value of the weight matrix as an index for the saturation of learning capacity as shown in Fig. 2.

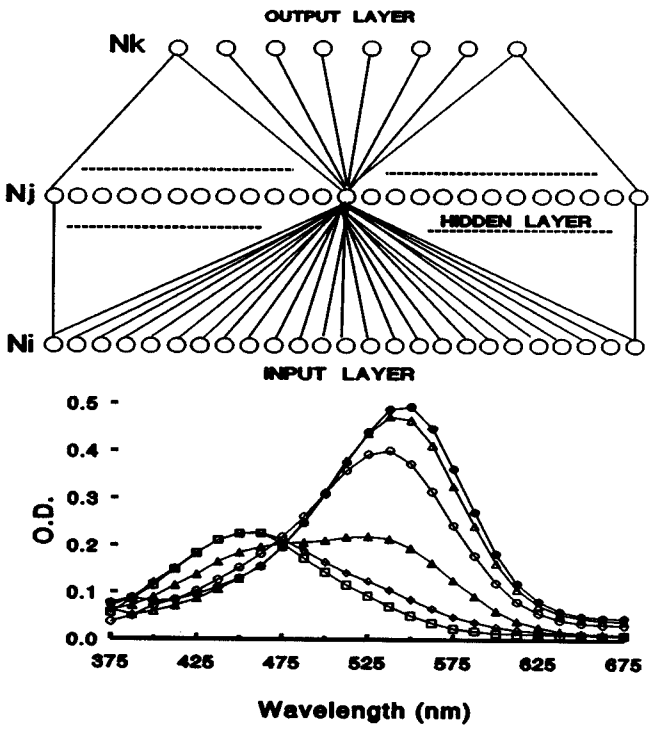

Fig. 1. Artificial neural network model for quantitative spectra measurement. Normalized absorption spectra (375 nm to 675 nm) from Neural Red (NR) are fed into the input layer ($Ni = 25$) for sufficient resolution. Same number of hidden units ($Nj = 25$) are used to transmit the complete information. Nk ($Nk = 8$) dimensional hypercube is set by the number of output nodes. Each binary coded teaching spectrum with known pH value represents a vertex on this hyper cube, to which the neural state tries to converge

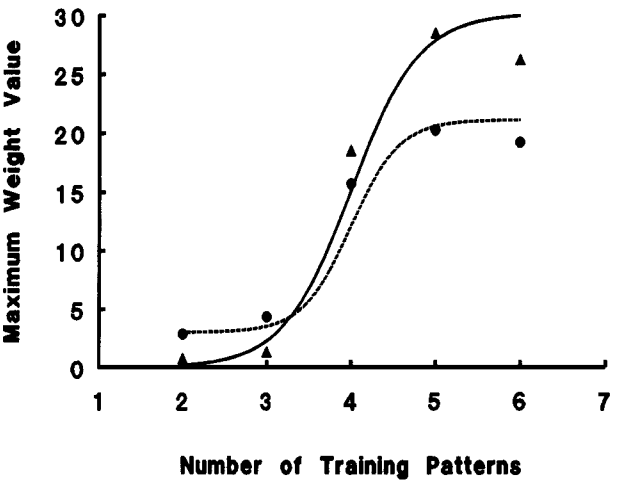

Fig. 2. The maximum total weight at convergence of the weight matrix from the input layer to the hidden layer (*dot*) and the hidden layer to the output layer (*triangle*) as a function of the number of training patterns demonstrating saturation

The number of output nodes is set to allow sufficient states for distinguishing the number of training patterns (spectra) during the training phase. In theory, one bit (node) can distinguish two different states, so we can distinguish 2^n patterns with n nodes in the output layer. Each binary coded expected patterns (T) with a known pH teaching spectra represents a vertex, to which the neural state tries to converge, on the n dimensional hypercube constructed by the output nodes. As the number of output nodes is increased beyond the minimum necessary, the resolution is more and more improved. For example, if one output node is allowed, then we can only train two spectra, pH 4 for 0 and pH 9 for 1. If eight output nodes are allowed, then we can distinguish 256 different spectra with pH 4 and pH 9 are set corresponding to the activity values of 0 and 255 with the expected patterns of (0 0 0 0 0 0 0 0) and (1 1 1 1 1 1 1 1).

The prediction of the pH of an unknown spectra uses the preset number of output nodes to interpret the output activities with a binary weighted method. For example, for an output layer of eight nodes, the output activities are $0 < o_i < 1$ $(i = 0 \ldots 7,)$, the interpreted value will be $2^7 * o_7 + \ldots + 2^0 * o_0$. Therefore, the interpreted value of output activities will be a value between 0 and 255 in this case.

Mathematical analysis of this model is shown in the appendix to provide insight into the convergent behavior.

3. Simulation results

3.1 Feature identification with generalized delta rule

With the similarity of the objective function to the multiple linear regression function, the GDR will evolve to identify and store the principal components within the weight matrix. Using normalized pH 4 and pH 9 spectra generated from microspectrophotometry of NR to train the network with 25 nodes to cover the visible range form 375 nm to 675 nm, we reached a convergency state where the two principal components of NR are identified and stored in the weight matrix of the input to hidden layer as shown in Fig. 3. The geometric appearance of this weight matrix shows that two inverse peaks are present in the connection weights from each hidden node to all the input nodes. This can be accounted for by the analysis derived in the appendix, i.e. by setting output pattern 1, weight increase; output pattern 0, weight decrease. The network can distinguish two states by adjusting the weight matrix toward either positive or negative. The two peak positions correspond to the basic and acid forms of NR at about 450 nm and 550 nm individually.

With the convergent weight matrix form a network trained with the spectra of pH 6 and pH 8, unknown spectra of intermediate pH values (7 and 7.5) can be predicted by the output activity of the hidden and output layers. By looking at the hidden layer activites, the projected output is presented in multiple nodes with order as shown in Fig. 4. After the principal components have been identified by the learning procedure, the subsequent prediction of unknown spectra can be considered as the inner product of the input vector with convergent weight matrix.

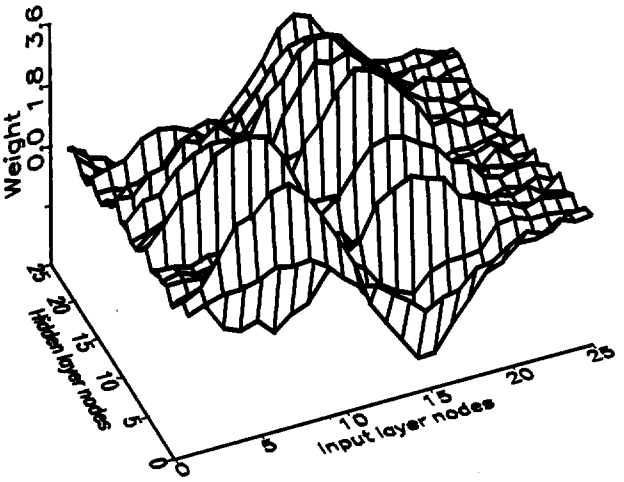

Fig. 3. Two principal components of the NR spectrum are identified and stored in the weight matrix between the input and hidden layer. Two training spectra (pH 4 and pH 9) are used to train the network until convergence (objective function <0.01 time the number of output nodes). The peak positions of these two components correspond to the acid and basic forms of NR at input node # 7 (around 450 nm) and # 15 (around 550 nm). The maximum positive weight is used to monitor the saturation of learning capacity in Fig. 2

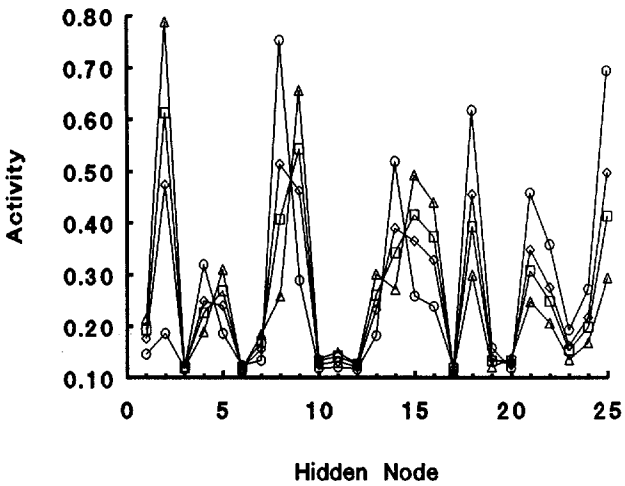

Fig. 4. Two spectra pH 6 (*circle*) and pH 8 (*triangle*) are used to train the network until convergence. With the convergent weight matrix, intermediate pH spectra pH 7 (*diamond*) and pH 7.5 (*square*) are used to demonstrate the ability of this model to identify the pH value of interpolated input spectra with the activities of hidden nodes. This activity value can be considered as the inner product of unknown input vector with the convergent weight matrix

3.2 Calibration curve

By looking at the output activities of the output layer using a binary weighted method after convergence, a calibration curve can be constructed from a limited

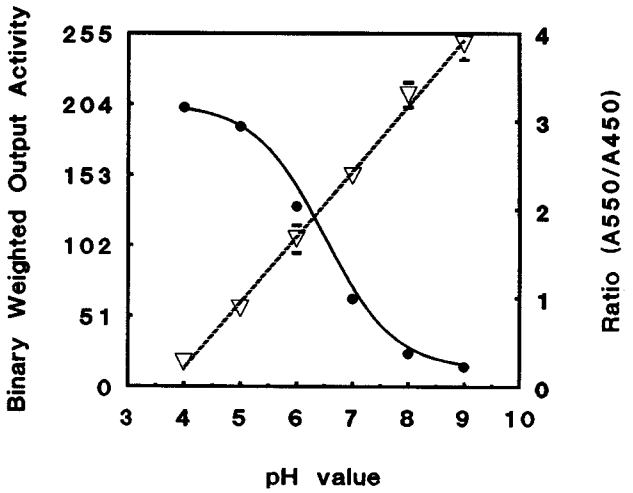

Fig. 5. Calibration curve (*dotted line*) constructed with the output activities at the output layer with known pH spectra after training to convergence. With eight output nodes, pH 4 and pH 9 correspond to 0 and 255, using binary coded method. This result is compared to the calibration curve from the ratio calculation (*solid line*) of the multiple linear regression method which we currently used to detect the change of spectra. This network method maps the input spectra to the output activity from pH 4 to pH 9 linearly, rather than just the range of pH 6 to pH 8 as in ratio method

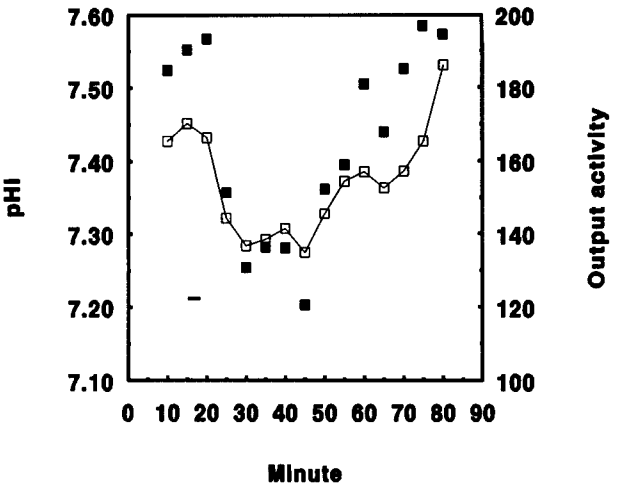

Fig. 6. A trained network with convergent weight matrix is used to predict unknown spectra from neutral red stained hippocampal brain slice in response to 5 min of anoxia and recovery. The result produced by the activity value (white square) of neural network is compared with the ratio calculation method (black square). The intracellular pH (pHi) became more acid after 5 min of anoxia and then recovered to initial level. The activity value of the network follows a similar pattern

number of teaching spectra with known pH values. Simulation with different numbers of teaching spectra gives the average activity value at each specific pH. These values are then plotted vs pH to given a calibration curve. The calibration curve constructed from the absorption ratio (A550/A450) determined by a standard multiple linear regression method is plotted for purposes of comparison. As shown in Fig. 5, this new method maps the input spectra to the output activities of the output nodes linearly from pH 4 to pH 9, rather than the range of pH 6 to pH 8 in the ratio method.

3.3 Intracellular pH measurement of 5 min anoxic response of hippocampal brain slice

In hippocampal slices, anoxia results in intracellular acidification due to lactic acid accumulation. Spectra generated from microspectrophotometry with five minutes anoxic response of a hippocampal brain slice are used to test the feasibility of this model for quantiative measurement (Fig. 6). A convergent weight matrix is used to give the projected activity output value corresponded to its input spectrum in one step of process. This activity output value can then be transformed to pH value by applying the calibration curve derived above. The result is compared with pHi reading from the ratio calculation of standard method as shown in Fig. 6. We currently use this method to find the optimal amplitude of two gaussian distribution curves with peak positions at 550 nm and 450 nm to represent the two forms of NR in the solution. The ratio value is calculated by the amplitude value at 550 nm (acid form) divided by the amplitude value at 450 (basic form) of the reconstructed spectrum which is the linear summation of the two gaussian curves. The transient response of the pHi determined using neural network is similar to the direct ratio method especially in the recovery phase where the recovery slope is our primary interest.

3.4 Noise effect

After convergence of the weight matrix, random noise is added to the training spectrum as a percentage of its maximum amplitude to test the effect of noise on its output activities. The outcome is compared to the result of adding zero noise in units of percentage change. Figure 7 shows that the network can tolerate up to 20% noise with the variation of outcome limited to within 4%.

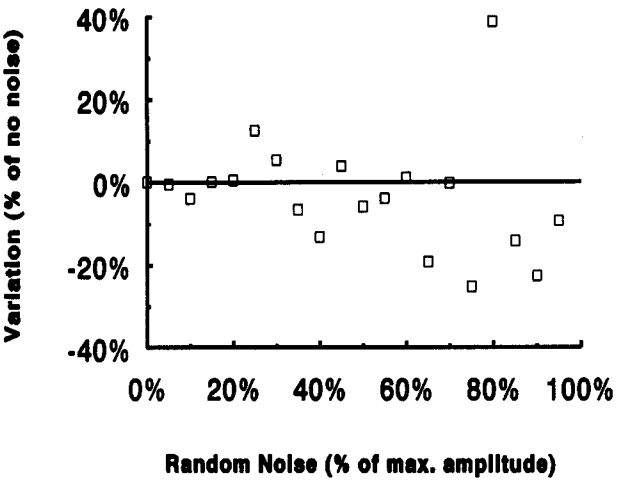

Fig. 7. Random noise added as a percentage of the maximum amplitude to every specturm showed that this network can tolerate up to 20% noise with a variation of output less than 4% as compared to the zero noise added spectrum

4 Discussion

By considering that information is carried by frequency coding instead of an all or none single pulse in the central nervous system and that the limited resource of neurons can provide for activation, we used 0 and 1 for the notation of silent and maximum firing frequency in our model. At the input layer, this is coded by the Euclidean vector norm to normalize the input pattern to values between 0 and 1. After this layer, coding is done by a sigmoidal function which transforms the input of each node to an output between 0 and 1. As shown in the appendix, for a two-layered model, this function guarantees convergence with the weight matrix monotonically increasing or decreasing, while for a three-layered model, as long as there are retrograde error signals from layer above, this function also helps the convergence.

In this stimulation, we demonstrated that with a simple local interactive learning rule, like GDR, the eigenvectors (principal components) of the incoming information are identified and stored in the weight matrix (synaptic junction), after a successful learning procedure. This convergent weight matrix can then be used to predict unknown input patterns with the output activities showing the correlation between input patterns. This is different from the model proposed by Oja (1982) in which the weight matrix is normalized after every iteration. We relaxed this constraint by analogy to the neurons where synaptic junctions are increased after a heavy stimulus. The connecting strength will grow with stimulus and becoming saturated until new synaptic junctions are needed for faster processing. By monitoring the maximum connecting strength in the weight matrix, we can determine the maximum weight level after training for a specified number of teaching patterns (shown as Fig. 2). This global information can be used as an index to determine the optimal capacity of the network or when it becomes necessary to enroll parallel networks dynamically.

In conclusion, this model (Fig. 1) has been shown to be capable of quantitative measurement for spectral recognition and prediction. It also presents a new method for spectrophotometric measurement where quantitative information (concentration) depends on the amplitude of spectra. This model might be generalized for application to multisensory data fusion and pattern-oriented sensory information processing.

Appendix: Mathematical analysis

1. Two-Layered model

Without a hidden layer, the guaranteed convergence of gradient descent of total error with respect to the weight change, $\partial E_p/\partial w_{ji}$, can be directly formulated. Let the measured error of pattern p be E_p, then

$$E_p = \sum_p \sum_j (t_{pj} - o_{pj})^2/2 \tag{1}$$

where,

$$o_{pj} = f_j(\text{net}_{pj}) \tag{2}$$

$$\text{net}_{pj} = \sum_i w_{ji} * o_{pi} \tag{3}$$

and the function f is a sigmoid function,

$$f(x) = 1/(1+\exp(-x)), \quad 0 < f(x) < 1 \tag{4}$$

To analyze the property of $\partial E_p/\partial w_{ji}$, we can write from chain rule that

$$\partial E_p/\partial w_{ji} = (\partial E_p/\partial \text{net}_{pj}) * (\partial \text{net}_{pj}/\partial w_{ji}) \tag{5}$$

where we can derive from (3)

$$\partial \text{net}_{pj}/\partial w_{ji} = \partial\left(\sum_k w_{jk} * o_{pk}\right)\Big/\partial w_{ji} = O_{pi} \tag{6}$$

and we define

$$\partial E_p/\partial \text{net}_{pj} = -\Psi_{pj} \tag{7}$$

Using the chain rule, we can have

$$\Psi_{pj} = -(\partial E_p/\partial o_{pj}) * (\partial o_{pj}/\partial \text{net}_{pj}) \tag{8}$$

from (1), we can have

$$\partial E_p/\partial o_{pj} = -(t_{pj} - o_{pj}) \tag{9}$$

from (4), the derivative of the sigmoid function is

$$\begin{aligned}\partial o_{pj}/\partial \text{net}_{pj} &= f_j'(\text{net}_{pj}) \\ &= f_j(\text{net}_{pj}) * (1 - f_j(\text{net}_{pj}))\end{aligned} \tag{10}$$

putting (9) and (10) into (8), we get

$$\Psi_{pj} = (t_{pj} - o_{pj}) * o_{pj} * (1 - o_{pj}) \tag{11}$$

putting (11) into (5), we have the explicit function for the gradient descent of total error with respect to the weight change as

$$\partial E_p/\partial w_{ji} = -(t_{pj} - o_{pj}) * o_{pj} * (1 - o_{pj}) * o_{pi} \tag{12}$$

The corresponding discrete version of this equation can be written as

$$\Delta E_p/\Delta w_{ji} = -(t_{pj} - o_{pj}) * o_{pj} * (1 - o_{pj}) * o_{pi} \tag{13}$$

According to GDR, the modification of weight strength is done by

$$\begin{aligned}\Delta_p w_{ji} &= \eta * \Psi_{pj} * o_{pi} \\ &= \eta * (t_{pj} - o_{pj}) * o_{pj} * (1 - o_{pj}) * o_{pi}\end{aligned} \tag{14}$$

The binary case of expected can be seen using (13) and (14). At each iteration cycle of learning phase with teaching pattern p, the weight change is calculated with (14) and the total measured error is calculated using (13).

1.1. If $t_{pj} = 1$, from (13) and (14), we can have

$$\Delta_p w_{ji} = \eta * (1 - o_{pj}) * o_{pj} * (1 - o_{pj}) * o_{pi} > 0$$

because $0 < o_{pj}, o_{pi} < 1$

$$\Delta E_p/\Delta w_{ji} = -(1 - o_{pj}) * o_{pj} * (1 - o_{pj}) * o_{pi} > 0$$

By setting the expected pattern to be 1, a positive weight change and a negative gradient descent total error energy function with respect to weight change will be produced.

This means that the measured error energy, E_p, decreases within each iteration cycle.

1.2. If $t_{pj} = 0$, from (13) and (14), we can have

$$\Delta_p w_{ji} = \eta * (1 - o_{pj}) * o_{pj} * (1 - o_{pj}) * o_{pi} < 0$$

because $0 < o_{pj}, o_{pi} < 1$

$$\Delta E_p / \Delta w_{ji} = -(1 - o_{pj}) * o_{pj} * (1 - o_{pj}) * o_{pi} > 0$$

Setting the expected pattern to 0 will produce a negative weight change and positive gradient descent total error energy function with respect to weight change. This means that the measured error energy, E_p, decreases within each iteration cycle.

2. Three-layered model

With another layer k as an output layer, the layer j acts as hidden layer. The objective function is the same as (1), except the notation is changed from j to k. For the output layer k, the analysis is the same as for two layer model. The same conclusion can be made in both cases of $t_k = 1$ and $t_k = 0$.

The error back propagation procedure calculates and transports the error (ME) for the node j of the hidden layer.

$$\mathrm{ME}_{pj} = f'_{pj}(\mathrm{net}_{pj}) * \sum_k w_{kj} * (t_{pk} - o_{pk})$$

$$= o_{pj} * (1 - o_{pj}) * \sum_k w_{kj} * (t_{pk} - o_{pk}) \quad (15)$$

The summation term on the right side of (15) represents the retrograde signal from, k, the layer above, back to the node, j.

The weight change according to GDR is

$$\Delta_p w_{ji} = \eta * \mathrm{ME}_{pj} * o_{pj} \quad (16)$$

$$\partial E_p / \partial w_{ji} = f'_j(\mathrm{net}_{pj}) * \left(\sum_k \Psi_{pk} * w_{kj} \right) * o_{pi}$$

$$= -o_{pj}(1 - o_{pj}) * \left(\sum_k (t_{pk} - o_{pk}) * o_{pk} * (1 - o_{pk}) * w_{kj} \right) * o_{pi} \quad (17)$$

The corresponding discrete version of this equation can be written as

$$\Delta E_p / \Delta w_{ji} = -o_{pj}(1 - o_{pj}) * \left(\sum_k (t_{pk} - o_{pk}) * o_{pk} * (1 - o_{pk}) * w_{kj} \right) * o_{pi} \quad (18)$$

The case of binary expected patterns and retrograde signal in (15), (16) and (18) can be discussed, because of the sigmoidal function, $0 < o_{pj,pk} < 1$ then

2.1. If $t_{pk} = 1$ and $\sum_k (t_{pk} - o_{pk}) * w_{kj} > 0$ then we can have $\mathrm{ME}_{pj} > 0$ and $\Delta_p w_{jl} > 0$

$\Delta E_p / \Delta w_{ji} < 0$
we have $\Delta E_p < 0$.

2.2. If $t_{pk} = 1$ and $\sum_k (t_{pk} - o_{pk}) * w_{kj} < 0$ then $\mathrm{ME}_{pj} < 0$ and $\Delta_p w_{ji} < 0$

$\Delta E_p / \Delta w_{ji} > 0$

we can have $\Delta E_p < 0$

2.3. If $t_{pk} = 0$ and $\sum_k (t_{pk} - o_{pk}) * w_{kj} > 0$ then $\mathrm{ME}_{pj} > 0$ and $\Delta_p w_{ji} > 0$

$\Delta E_p / \Delta w_{ji} < 0$

we can have $\Delta E < 0$.

2.4. If $t_{pk} = 0$ and $\sum_k (t_{pk} - o_{pk}) * w_{kj} < 0$ then $\mathrm{ME}_{pj} < 0$ and $\Delta_p w_{ji} < 0$

$\partial E_p / \partial w_{ji} > 0$

we can have $\Delta E < 0$.

These four cases demonstrate that the measured error energy, E_p, must decrease within each iteration cycle.

Acknowledgements. This study was supported by The National Institute of Neurological Disorders and Stroke, grants NS-22077; National Heart, Lung, and Blood Institute, Grant HL-42215.

References

Baldi P, Hornik K (1989) Neural networks and principal component analysis: learning from examples without local minima. Neural Network 2:53–58

Kohonen T (1974) An adapative associative memory principle. IEEE Comput 23:444–445

Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biol Cybern 43:59–69

LaManna JC, McCracken KA (1984) The use of neutral red as an intracellular pH indicator in rat brain cortex in vivo. Analyt Biochem 142:117–125

LaManna JC (1987) Intracellular pH determination by absorption spectrophotometry. Metab Brain 2:167–182

Macdonald VW, Keizer JH, Jobsis FF (1977) Spectrophotometric measurements of metabolically induced pH changed in frog skeletal muscle. Arch Biochem 184:423–430

Meyer B, Hansen T, Nute D, Albersheim P, Darvili A, York W, Sellers J (1991) Identification of the ^{1}H-NMR spectra of complex oligosaccharides with artificial neural networks. Science 251:542–544

Oja E (1982) A simplified neuron model as a principal component analyzer. J Math Biol 15:267–273

Rubner J, Schulten K (1990) Development of feature detectors by self-organization a network model. Biol Cybern 62:193–199

Rumelhart DE, McClelland JL (1988) The PDP research group: parallel distributed processing. MIT Press, Cambridge

Sanger TD (1989) Optimal unsupervised learning in a single-layer linear feedforward neural network. Neural Network 2:459–473

Sick TJ, Whittingham TS, LaManna JC (1989) Determination of intracellular pH in the in vitro Hippocampal slice preparation by transillumination spectrophotometry of neutral red. J Neurosci M 27:25–34

Wythoff BJ, Levine SP, Tomellini SA (1990) Spectra peak verification and recognition using a multilayered neural network. Analyt Chem 62:2702–2709