Learning Algorithm for the Threshold of Nonmonotonic Neurons Composing a Boltzmann Machine

Takuya HAGA, Fumihiko ISHIDA, Mitsunaga KINJO, Shigeo SATO and Koji NAKAJIMA

Laboratory for Electronic Intelligent Systems,
Research Institute of Electrical Communication,
Tohoku University, Sendai, 980-8577, Japan
Email: takuya-h@nakajima.riec.tohoku.ac.jp

1. Introduction

Learning ability is one of the most attractive features of neural networks. Various models have been proposed such as Back propagation[1] and Boltzmann Machine[2], etc. In these models, monotonic functions like a sigmoid function are used for the activation functions of neurons.

A Deterministic Boltzmann Machine(DBM)[3][4], which is another Boltzmann Machine obtained by applying a mean field theory approximation, also uses monotonic neurons. In the DBM, it has been reported that the learning performance is improved when nonmonotonic neurons are introduced as hidden neurons[5]. These nonmonotonic neurons have a new threshold parameter on which the learning performance depends strongly.

In this paper, we propose a new learning algorithm for the threshold based on the stationary distribution of the Boltzmann Machine with nonmonotonic neurons, and show the advantages of this algorithm by simulations.

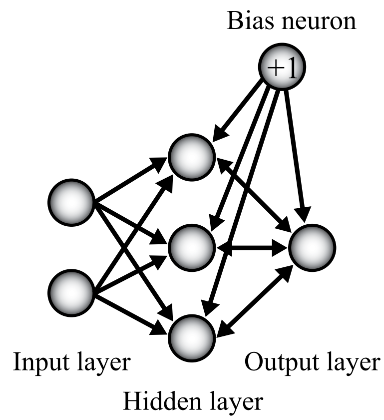


Figure 1: DBM network

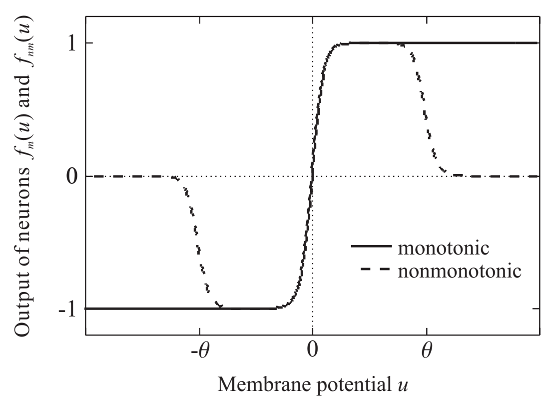


Figure 2: Characteristics of monotonic and nonmonotonic neurons

2. Learning Ability of Nonmonotonic Neurons

The nonmonotonic neurons exhibit good performance not only for associative memory[6][7][8] but also for learning[5]. In this section, we show the learning ability of the nonmonotonic neurons and its dependence on the threshold as DBM. We consider three layer DBM networks including one bias neuron as shown in Fig.1. In the original DBM, all neurons have monotonic characteristics as given by Eq.(1). We introduce nonmonotonic neurons which have end-cut-off characteristic into hidden neurons as given by Eq.(2),

$$f_m(u) = \tanh(\beta u), \tag{1}$$

$$f_{nm}(u) = \frac{1}{2}\{2f_m(u) - f_m(u+\theta) - f_m(u-\theta)\}, \tag{2}$$

where u, β, and θ are the membrane potential of the neuron, the inverse of temperature regulating noise, and the threshold of the nonmonotonic neurons, respectively. These characteristics are shown in Fig.2.

In a conventional DBM, a learning rules for connection

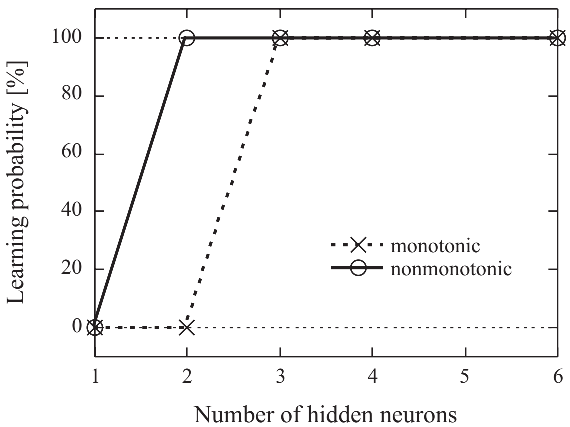


Figure 3: Learning results for the 3-parity problem with 3-X-1 networks

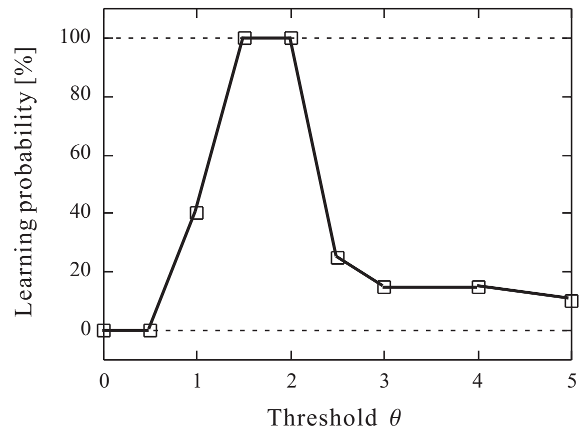


Figure 4: θ dependence of learning performance for the 3-parity problem with 3-2-1 networks

weights are given by

$$\Delta w_{ij} = \epsilon\{\langle V_i V_j\rangle_{clamped} - \langle V_i V_j\rangle_{free}\}, \qquad (3)$$

where w_{ij}, ϵ, and V_i are the connection weight between the i-th neuron and the j-th neuron, a positive constant, and the output of the i-th neuron, respectively. $\langle V_i V_j\rangle_{clamped}$ and $\langle V_i V_j\rangle_{free}$ are a time average of correlation when the visible units are clamped to an environmental distribution and that when network is running freely, respectively. We apply this learning rule to the case of nonmonotonic neurons as well as monotonic neurons.

Figure 3 shows the simulation results for the 3-parity problem as a function of the number of hidden neurons. From the results, we confirm that the nonmonotonic neurons improve the learning performance compared with the monotonic neurons. Figure 4 shows the relation between the learning probability and the threshold θ for the 3-parity problem. The learning performance depends on the threshold θ strongly and there are some optimum values.

3. Stationary Distribution of Boltzmann Machine with Nonmonotonic Neurons

The Boltzmann Machine minimizes the distance between an environmental distribution and a stationary distribution of the Boltzmann Machine, which is given by Boltzmann distribution

$$p(\boldsymbol{x}) = \frac{1}{Z}\exp\{-\beta E(\boldsymbol{x})\}, \qquad (4)$$

$$Z = \sum_{\boldsymbol{x}} \exp\{-\beta E(\boldsymbol{x})\}, \qquad (5)$$

where the vector $\boldsymbol{x}$ represents the state of the Boltzmann Machine and Z is the partition function. $E(\boldsymbol{x})$ is an energy of the state $\boldsymbol{x}$ defined by

$$E(\boldsymbol{x}) = -\frac{1}{2}\sum_{ij} w_{ij} x_i x_j, \qquad (6)$$

where x_i is a state of the i-th neuron and takes on two values $\{+1, -1\}$ stochastically.

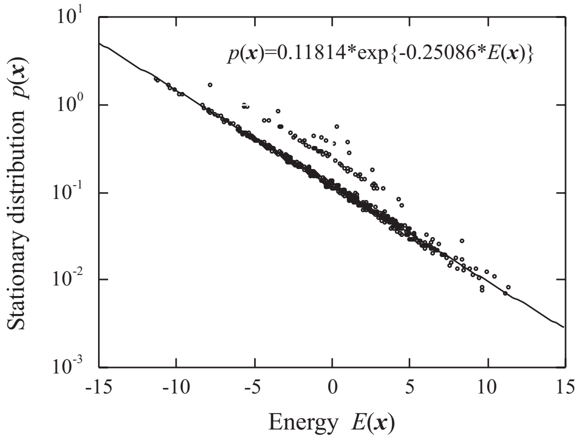
To study a learning algorithm for the threshold θ, we investigate a stationary distribution of a Boltzmann Machine with nonmonotonic neurons. We consider the network with symmetric recurrent connections $w_{ij} = w_{ji}$, and self recurrent connection w_{ii} is assumed to be 0. Each neuron takes $x_i = +1$ with probability $g(u_i)$ and $x_i = -1$ with probability $1 - g(u_i)$,

$$g(u) = \frac{1}{1+\exp(-2\beta u)} - \frac{\alpha}{1+\exp\{-2\beta(u+\theta)\}} - \frac{\alpha}{1+\exp\{-2\beta(u-\theta)\}} + \alpha. \qquad (7)$$

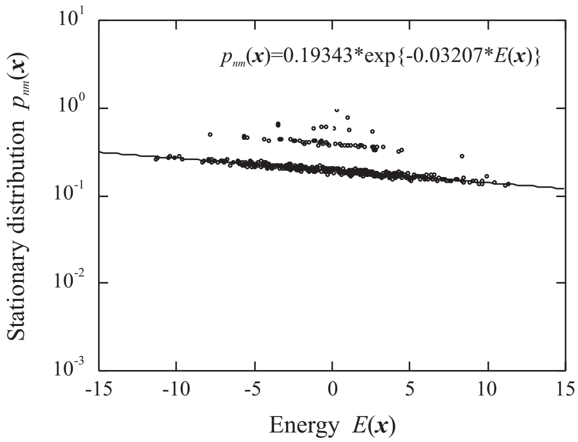
When α is equal to 0, this is the same as the original Boltzmann Machine. When α is equal to 0.5, each neuron has nonmonotonic firing probability, which exhibits end-cut-off characteristic.

It is too complex to calculate the stationary distribution of the Boltzmann Machine with nonmonotonic neurons, so we investigate it by numerical simulations. In the simulations, the connection weights w_{ij} are initialized with random values in the range [-1.0,1.0] and clamped, the state transition times are 10^7 and we investigate the last 10^6 times. The number of neurons and β are set to 10 and 0.25, respectively.

Figure 5(a) shows the results for the case of the monotonic neurons ($\alpha = 0$). Some points are not fitted to



(a)



(b)

Figure 5: Stationary distributions of the Boltzmann Machines with (a)monotonic neurons and (b)nonmonotonic neurons. The solid lines are obtained by fitting and their coefficients are shown

the approximate line. It seems that, this is due to get trapped in local minima because we do not apply a simulated annealing. It can be confirmed that the stationary distribution is given by Boltzmann distribution like Eq.(4). Figre 5(b) shows same results for the case of nonmonotonic neurons ($\alpha = 0.5$), where θ is 3.0. Boltzmann distribution is also obtained, but it has the different coefficient from Eq.(4). Let us define this new coefficient as β'. Figure 6 shows β' as a function of θ. As the probability in Eq.(7) is symmetry with respect to θ, the coefficient β' should be also symmetry with respect to θ. From these results, the stationary distribution of the Boltzmann Machine with nonmonotonic neurons is given

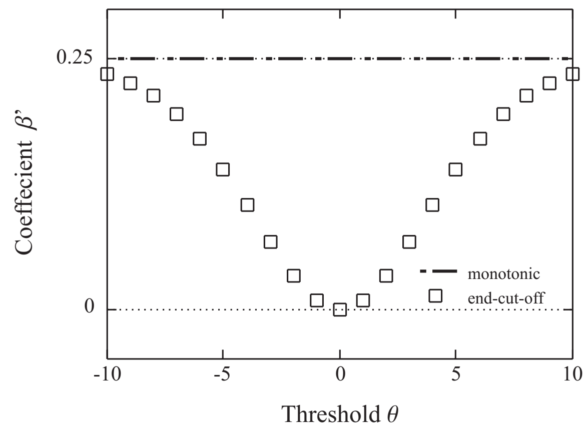


Figure 6: Relation between β' and θ

by Boltzmann distribution written as

$$p_{nm}(\boldsymbol{x}) = \frac{1}{Z} \exp\{-\beta'(\theta)E(\boldsymbol{x})\}. \tag{8}$$

4. Learning Algorithm for Threshold θ

We consider a general situation where neurons are divided into two parts, visible neurons and hidden neurons. Let $\boldsymbol{x}_V$ and $\boldsymbol{x}_H$ denote the states of visible neurons and hidden neurons, respectively. The Boltzmann Machine minimizes the Kullback divergence between $q(\boldsymbol{x}_V)$ which is an environmental distribution to be learned and $p(\boldsymbol{x}_V)$,

$$G = \sum_{\boldsymbol{x}_V} q(\boldsymbol{x}_V) \log \frac{q(\boldsymbol{x}_V)}{p_{nm}(\boldsymbol{x}_V)}, \tag{9}$$

where

$$p_{nm}(\boldsymbol{x}_V) = \sum_{\boldsymbol{x}_H} p_{nm}(\boldsymbol{x}). \tag{10}$$

The learning algorithm for the threshold θ of the nonmonotonic neurons can be obtained by differentiating the Kullback divergence with respect to θ,

$$\Delta\theta = -\eta \frac{\partial G}{\partial \theta}, \tag{11}$$

where η is a positive constant. Therefore, the learning algorithm is given by

$$\Delta\theta = \eta \frac{\partial \beta'(\theta)}{\partial \theta} \left[\langle E \rangle_{free} - \langle E \rangle_{clamped} \right], \tag{12}$$

where E is the energy given by Eq.(6).

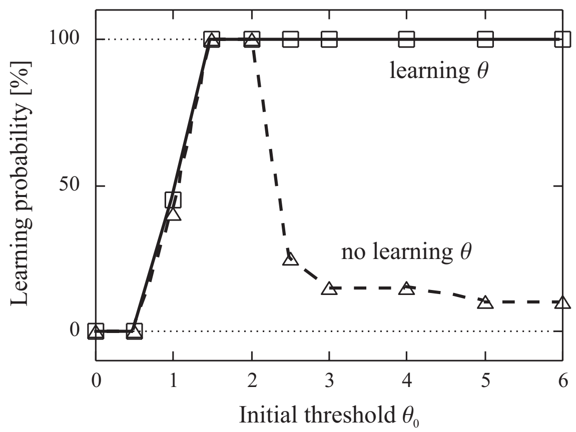


Figure 7: Comparison of learning performance for the 3-pairty problem with 3-2-1 networks

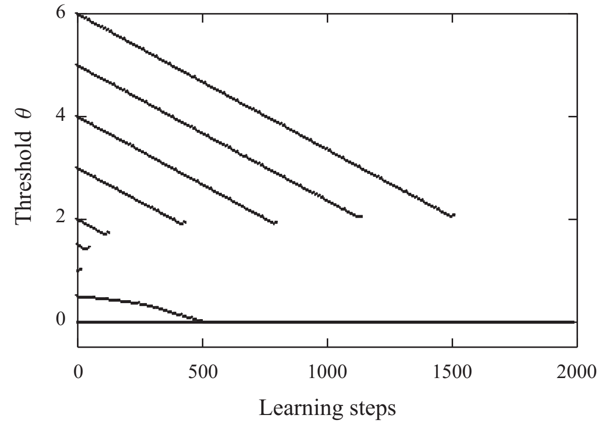


Figure 8: Trace of the threshold θ

5. Simulation Results

To confirm the advantages of the algorithm by simulations, we simplify Eq.(12) like the following,

$$\Delta\theta = \eta \mathrm{sgn}(\theta) \left[\langle E \rangle_{free} - \langle E \rangle_{clamped} \right]. \qquad (13)$$

We apply this algorithm to the DBM network appeared in section 2. We investigate for the 3-parity problem only with two hidden neurons. This is because only nonmonotonic networks can learn as shown in Fig.3. Figure 7 and Fig.8 show the learning probability as a function of initial value of θ and the trace of θ, respectively. From these results, we see that when initial θ is large, θ decreases and converges, that is, the learning algorithm works well. While when initial θ is small, the learning doesn't work well. It seems that this is because we introduced nonmonotonic neurons into only hidden neurons, while the algorithm is derived under the assumption that all neurons are nonmonotonic. Numerical simulations show the learning algorithm for the threshold θ is effective.

6. Conclusion

We have proposed the new learning algorithm for the threshold of nonmonotonic neurons based on the stationary distributions, and confirmed by simulations that this algorithm has advantages when initial value of the threshold is large. We have investigated the stationary distributions by simulations, but it should be proved theoretically. This is our future subject.

References

[1] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Internal Representations by Error Propagation," Parallel Distributed Processing, Vol. 1, pp. 318-362, 1986.

[2] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A Learning Algorithm for Boltzmann Machines," Cognitive Science, Vol. 9, pp. 147-169, 1985.

[3] C. Peterson, and J. R. Anderson, "A Mean Field Theory Learning Algorithm for Neural Networks," Complex Systems, Vol. 1, pp. 995-1019, 1987.

[4] G. E. Hinton, "Deterministic Boltzmann Learning Performs Steepest Descent in Weight-Space," Vol. 1, pp. 143-150, 1989.

[5] M. Kinjo, S. Sato, and K. Nakajima, "Hardware Implementation of a DBM Network with Nonmonotonic Neurons," IEICE Transactions on Information and Systems, Vol. J85-D-II, No. 3, pp. 558-567, March, 2002.

[6] M. Morita, "Associative Memory with Nonmonotone Dynamics," Neural Networks, Vol. 6, pp. 115-126, 1993.

[7] S. Yoshizawa, M. Morita, and S. Amari, "Capacity of Associative Memory Using a Nonmonotonic Neuron Model," Neural Networks, Vol. 6, pp. 167-176, 1993.

[8] H. Yanai, and S. Amari, "Auto-Associative Memory with Two-Stage Dynamics of Nonmonotonic Neurons," IEEE Transactions on Neural Networks, Vol. 7, No. 4, pp. 803-815, July, 1996.