'90 KACC 1990. 10. 26～27

An attempt to reduce the number of training in the artificial neural network

Akihiro Omae and Shintaro Ishijima
Tokyo Metropolitan Institute of Technology
6-6 Asahigaoka Hino-City Tokyo 191 ,Japan
Tel 0425 83 5111 (Japan)

Abstract

A large number of trainings are requested for the artificial neural network using the backpropagation algorithm. It is shown that one dimensional search technique is effective to reduce the number of trainings through some numerical simulations.

1. Introduction

Artificial neural networks are aimed at advanced information processing of human brain, and they are made up of simple and interconnected processing elements, which work in parallel. For the training of layered type networks, backpropagation algorithm is a typical training algorithm. However, actually, a large number of trainings are requested, it takes considerable time till the training has converged. That would be a serious problem in practical use. So, it is important to reduce the number of training and to shorten the time of training.

Generally, the speed of the training depends on the value of the training rate coefficient. and the value of the coefficient is determined from experience. The backpropagation algorithm can be regarded as a kind of the steepest descent algorithm. One dimensional search technique is known as a standard technique to get the efficient convergence in the steepest descent algorithm. However, there are not so many studies about the effect of one dimensional search technique in the learning process of artificial neural networks. Through some numerical simulations, it is shown that the learning processes have been considerably improved by applying the one dimensional search.

2. Training of the artificial neural network using the backpropagation algorithm

For the training of layered type networks using the backpropagation algorithm, an error function which has to be minimized is calculated by the equation (1).

$$E(t) = \frac{1}{2}\sum_{p}\sum_{j}(y_{jp}(t) - d_{jp})^2 \tag{1}$$

where, t is the number of the training. $y_{jp}(t)$ is the output of the unit j in the output layer, when the input pattern p is given. d_{jp} is the correct answer for the input pattern p to the unit j in the output layer. The weight of the connection is changed by the equation (2).

$$w_{ij}(t+1) = w_{ij}(t) - \eta(t)\frac{\partial E(t)}{\partial w_{ij}(t)} \tag{2}$$

where, $w_{ij}(t)$ is the weight from unit i to unit j at t, and $\eta(t)$ is the training rate coefficient. The training so get as to decrease the summation of the squared error E(t). It is the training to minimize the summation of the squared error between the actual output and the correct answer. The actual output is calculated from input pattern by the network. Altering weights is the change to the direction which decrease the squared error best. So, the backpropagation algorithm can be regarded as a kind of the steepest descent algorithm for squared error E(t) on the space of weights. Generally, although it is hard to draw this state, it is shown in Figure 2.1 by supposing to be able to draw values of the error as contour lines. Where, the inner contour line display the lower value of the error than the outer one. And $\eta(t)$ is fixed as a constant in Figure 2.1.

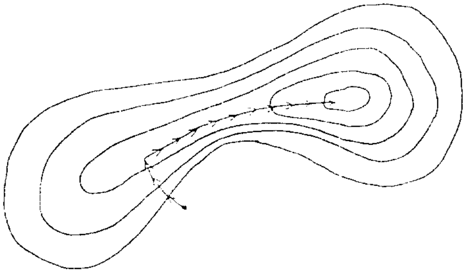

Figure 2.1. The image of the backpropagation learning (The learning rate coefficient is a constant)

3. For the efficient training

The training rate coefficient $\eta(t)$ is a parameter to determine the speed of the training. If a large value is set to $\eta(t)$, the speed of the training would be fast. However, if the value of $\eta(t)$ is too large, the learning process becomes unstable. In order to reduce the number of training, it is need to set to the coefficient $\eta(t)$ as the large value as possible in the range that learning process does not oscillate. However, generally, the value of the coefficient is determined from experience. In many case, it is hard to say that every training is the efficient training. When the value is set to the coefficient, the optimal value of the training rate coefficient is searched by using one dimensional search technique. By setting the optimal value to the coefficient, it is expected to reduce the number of the training until getting the convergence. There is no need to fix the value of the coefficient as a constant. At every time of the training, by searching and setting the value of the coefficient, it is expected to get the efficient training and to reduce the number of training. In every time of the training process, the value of

$\partial E/\partial w$ is calculated. And by using this value, the optimal value of the learning rate coefficient is searched. At every time that a training pattern is given to the artificial neural network, the value of $\partial E_p(t)/\partial w$ is calculated. And after showing all training patterns, the summation of the squared error $\partial E/\partial w$ is calculated. The equation is the equation (3).

$$\frac{\partial E(t)}{\partial w_{ij}(t)} = \sum_{p}^{n} \frac{\partial E_p(t)}{\partial w_{ij}(t)} \qquad (3)$$

If the weights are changing after showing only one training pattern, the weights of the network are over fitted only for one training pattern. And the learning of the network may not converge. By applying the one dimensional search technique, the value of the training rate coefficient $\eta(t)$ is searched. After searching the value, the weights of the connections is changed by using the searched value and equation (2). Image of changing the weight by setting the optimal value to $\eta(t)$ is shown in Figure 3.1.

Show the Figure 3.1. At first, the descent $\partial E/\partial w$ at initial state (at A) is calculated. On the calculated direction (from A to B) , the optimal value of the coefficient is searched. While increasing the value of the $\eta(t)$, going down on the curved surface. On the calculated direction, the weights are changed to the direction which is decrease the squared error E(t) best. At the deepest point (at B) on the direction (from A to B), new descent (from B to C) is calculateed at that point (at B). From B, go down to the deepest point (to C) on new direction (from B to C) which is calculated at B. By repeating the same process, it is expected to go down to the deepest point (to D) of the curved surface. And it is expected to be decreased squared error to the minimum value, and reduce the number of the training.

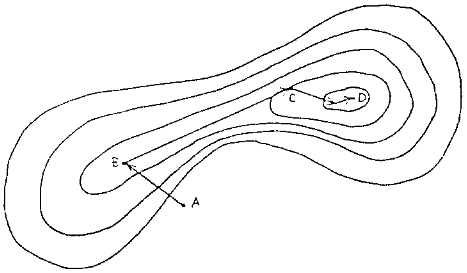

Figure 3.1. The image of the backpropagation learning with searching the optimal value of the learning rate coefficient

4. Numerical simulation

Here, some examples of the numerical simulations is shown. These are the exclusive-or and the coding. The layered type network which has no hidden layers can not learn the exclusive-or. So, it is popular example of learning on the network which has hidden layers. The coding problem can be seemed a kind of the information compression. The number of training until the training has converged depends on the initial state of the network. To compare the number of the training, the network start the training from the same initial state, and the number of training until the training has converged.

The learning of the exclusive-or

The network we used has three layers. It has two units in the input layer, and a unit in the output layer. Two units are in the hidden layer. After showing four exclusive-or training patterns, changing the weights of the connections. This process is counted as one training. The training is repeated until the summation of squared error E(t) is decreased less than 10^{-2}. In Figure 4.1, the number of the training is shown in the case of searching the optimal coefficient and non-searching (The value of the coefficient is fixed as constant).

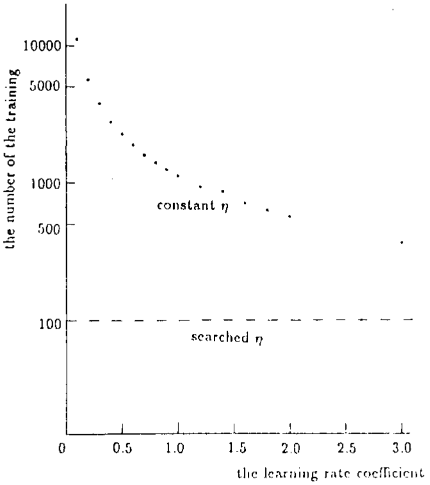

Figure 4.1. The number of the training (ex-or)

The learning of the coding

The network we used has three layers. It has four units in the input layer, and four units in the output layer. To learn this problem, it is requested to make network whose number of units in input and output layer is same. If input and output pattern is N bit pattern, units in hidden layer need more than $log_2 N$. The network we used has two units in hidden layer. A input and a output pattern is expressed in four bits. Training patterns are composed of four sets, (0001) (0010) (0100) (1000). Each of them is given to the network as both input and output. The training is repeated until the summation of squared error E(t) is decreased less than 2.0×10^{-2}. In Figure 4.2, the number of the training is shown in the case of searching the optimal coefficient and non-searching (The value of the coefficient is fixed as constant).

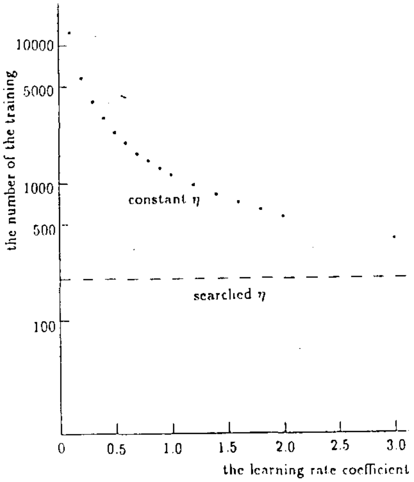

Figure 4.2. The number of the training (coding)

5. Conclusion

By applying the one dimensional search technique to the training of the neural network, it has shown to improve the learning process by searching the optimal value of the learning rate coefficient. It is important to study the effectiveness of this algorithm for more complex class of problems such as the modeling problem, motion control problems and so on.

References

[1]D.E.Rumelhart, J.L.McClelland and PDP Research Group, Parallel Distributed Processing, MIT Press, 1986.
[2]Toshio FUKUDA, Takashi KURIHARA, Masatoshi TOKITA, and Toyokazu MITSUOKA, Position and Force Hybrid Control of Robotic Manipulator by Neural Network (1st Report, Application of Neural Servo Controller to Stabbing Control),1990.