Analysis of the Back-Propagation Algorithm with Momentum

V. V. Phansalkar and P. S. Sastry

***Abstract*— In this letter, the back-propagation algorithm with the momentum term is analyzed. It is shown that all local minima of the sum of least squares error are stable. Other equilibrium points are unstable.**

I. Introduction

Back-propagation (BP) [1] is one of the most widely used algorithms for training feedforward neural networks. However, it is seen from simulations that it takes a long time to converge. Consequently, many variants of BP have been suggested [3]. One of the most well-known variants is the back-propagation with momentum terms (BPM). BP can be shown to be a straightforward gradient descent on the least squares error, and it has been shown recently [2] that BP converges to a local minimum of the error. While it is observed that the BPM algorithm shows a much higher rate of convergence than the BP algorithm, at present there does not exist any analysis of the BPM algorithm.

In this letter, we analyze the behavior of the BPM algorithm and show that all local minima of the least squares error are the only locally asymptotically stable points of the algorithm.

Let o_i denote the output of the ith unit upon presentation of pattern $\boldsymbol{x}$. The jth unit is connected to the ith unit by a synaptic strength of u_{ij} and the output of the ith unit is

$$o_i = f_i\left(\sum_j u_{ij} o_j\right) \tag{1}$$

where f_i is the activation function of the ith unit. The usual choices for f_i are the logistic or tanh functions. The desired output of unit i (if it is an output unit), upon presentation of pattern $\boldsymbol{x}$ is $t_i(\boldsymbol{x})$. The objective function for the optimization problem of learning weights is

$$E(\boldsymbol{u}) = \sum_{\boldsymbol{x}} F(\boldsymbol{u}, \boldsymbol{x}) \tag{2}$$

where

$$F(\boldsymbol{u}, \boldsymbol{x}) = \sum_i [o_i(\boldsymbol{x}, \boldsymbol{u}) - t_i(\boldsymbol{x})]^2 \tag{3}$$

is the error on $\boldsymbol{x}$ with weights $\boldsymbol{u}$. The BP algorithm does a gradient descent on $F(\boldsymbol{u}, \boldsymbol{x})$ when the pattern presented is $\boldsymbol{x}$:

$$u_{ij}(n+1) = u_{ij}(n) - \alpha(\delta F/\delta u_{ij})((\boldsymbol{x}(n), \boldsymbol{u}(n)). \tag{4}$$

Similarly, the BPM algorithm can be written as

$$u_{ij}(n+1) = u_{ij}(n) - \alpha\left(\frac{\delta F}{\delta u_{ij}}\right)((x(n), u(n)) + \eta(u_{ij}(n) - u_{ij}(n-1)) \tag{5}$$

where α and η are positive constants. The goal of the algorithm is to minimize $E(\boldsymbol{u})$. It will be assumed that α is small and that all the patterns are presented frequently so that (5) is essentially equivalent to the algorithm

$$u_{ij}(n+1) = u_{ij}(n) - \alpha(\delta E/\delta u_{ij})(u(n)) + \eta(u_{ij}(n) - u_{ij}(n-1)). \tag{6}$$

This is justified if α is small enough. Equation (6) is exactly equivalent to (5) if the number of patterns are finite and the updating is done only after each cycle of presentation of the patterns.

II. Analysis

Algorithm (6) is converted to a state variable form to facilitate analysis. Define v_1 and v_2 as

$$\boldsymbol{v}_1(n) = \boldsymbol{u}(n); \qquad \boldsymbol{v}_2(n) = \boldsymbol{u}(n) - \boldsymbol{u}(n-1). \tag{7}$$

Then (6) can then be rewritten as

$$\boldsymbol{v}_1(n+1) = \boldsymbol{v}_1(n) - \alpha\overline{\nabla} E(\boldsymbol{v}_1(n)) + \eta\boldsymbol{v}_2(n) \tag{8a}$$

$$\boldsymbol{v}_2(n+1) = -\alpha\overline{\nabla} E(\boldsymbol{v}_1(n)) + \eta\boldsymbol{v}_2(n). \tag{8b}$$

Theorem 1: $(\boldsymbol{s}_1, \boldsymbol{s}_2)$ is equilibrium point of (8) iff $\overline{\nabla} E(\boldsymbol{s}_1) = 0$ and $\boldsymbol{s}_2 = 0$.

Proof: It can be verified by direct substitution that if $\overline{\nabla} E(\boldsymbol{s}_1) = 0$ and $\boldsymbol{s}_2 = 0$, then $(\boldsymbol{s}_1, \boldsymbol{s}_2)$ is an equilibrium point of (8).

For the converse, let $\boldsymbol{v}_1(n+1) = \boldsymbol{v}_1(n)$ and $\boldsymbol{v}_2(n+1) = \boldsymbol{v}_2(n)$ when $\boldsymbol{v}_1(n) = \boldsymbol{s}_1$ and $\boldsymbol{v}_2(n) = \boldsymbol{s}_2$. Using this in (8a), we see that

$$-\alpha\overline{\nabla} E(\boldsymbol{s}_1) + \eta\boldsymbol{s}_2 = 0. \tag{9}$$

By (8b), this implies $\boldsymbol{s}_2 = 0$ and using this fact in (9), we obtain $\overline{\nabla} E(\boldsymbol{s}_1) = 0$. This completes the proof. □

The above theorem shows that the only equilibrium points of (9) are those where $\overline{\nabla} E$ is zero. This is similar to a gradient following algorithm. Next, local stability/instability properties of (9) around an equilibrium point $\boldsymbol{s} = (\boldsymbol{s}_1, \boldsymbol{s}_2)$ are examined. This is done using small signal analysis. To linearize (8) around $\boldsymbol{s}$, the perturbed signals are defined as

$$\epsilon_1 = \boldsymbol{v}_1 - \boldsymbol{s}_1; \qquad \epsilon_2 = \boldsymbol{v}_2 - \boldsymbol{s}_2 = \boldsymbol{v}_2 \qquad (\text{as } \boldsymbol{s}_2 = 0). \tag{10}$$

Then, using linear approximations (where $\nabla^2 E$ is Hessian of E),

$$\begin{aligned}\epsilon_1(n+1) &= \boldsymbol{v}_1(n+1) - \boldsymbol{s}_1\\ &= \epsilon_1(n) - \alpha\overline{\nabla} E(\boldsymbol{s}_1 + \epsilon_1(n)) + \eta\epsilon_2(n)\\ &\approx \epsilon_1(n) - \alpha\nabla^2 E(\boldsymbol{s}_1)\epsilon_1(n) + \eta\epsilon_2(n) \end{aligned}\tag{11a}$$

$$\epsilon_2(n+1) \approx -\alpha\nabla^2 E(\boldsymbol{s}_1)\epsilon_1(n) + \eta\epsilon_2(n). \tag{11b}$$

Thus, the small signal (or linearized) model around $\boldsymbol{s}$ is

$$\begin{bmatrix}\epsilon_1(n+1)\\ \epsilon_2(n+1)\end{bmatrix} = \begin{bmatrix} I - \alpha A & \eta I\\ -\alpha A & \eta I\end{bmatrix}\begin{bmatrix}\epsilon_1(n)\\ \epsilon_2(n)\end{bmatrix} \tag{12a}$$

where $A = \nabla^2 E(\boldsymbol{s}_1)$. In more compact form, (12a) can be written as

$$\epsilon(n+1) = D\epsilon(n). \tag{12b}$$

The following assumptions are made about the behavior of $E(\cdot)$. These assumptions imply that $E(\cdot)$ is well behaved with respect to its Hessian at all points where its gradient is zero. These properties are generic and cases where they fail would be rare.

A1) $\nabla^2 E$ is positive definite at all local minima of E.

Manuscript received July 7, 1992.

The authors are with the Department of Electrical Engineering, Indian Institute of Science, Bangalore 560 012, India.

IEEE Log Number 9207237.

1045–9227/94$04.00 © 1994 IEEE

A2) $\nabla^2 E$ has at least one strictly negative eigenvalue at all points $\boldsymbol{x}$ where $\nabla E(\boldsymbol{x}) = 0$ but $\boldsymbol{x}$ is not a local minimum of E.

The study of stability or instability is done by looking at the eigenvalues of D. It is well known [5] that if the magnitude of the maximum eigenvalue of D is strictly less than unity, then $(\boldsymbol{s}_1, \boldsymbol{s}_2)$ is locally asymptotically stable equilibrium point of (8). Conversely, even if one of the eigenvalues of D has a magnitude strictly greater than unity, then $(\boldsymbol{s}_1, \boldsymbol{s}_2)$ is an unstable equilibrium point of (8).

The following lemma characterizes the eigenvalues of D in terms of the eigenvalues of $A = \nabla^2 E(\boldsymbol{s}_1)$. It should be noted that D has twice the number of eigenvalues of A. Essentially, each eigenvalue of A "splits" to give two eigenvalues of D. All eigenvalues of A are real as A is symmetric, if we assume that $E(\cdot)$ is sufficiently well behaved so that the second partial derivatives exist and the order of differentiation is immaterial.

Lemma 1: If η is an eigenvalue of A, then the two corresponding eigenvalues of D are solutions of the quadratic equation

$$\Theta^2 - \Theta(1 - \alpha\mu + \eta) + \eta = 0. \tag{13}$$

Proof: It can be easily shown that D is invertible for any A, as long as $\eta \neq 0$. Let Θ be any eigenvalue of D. It is nonzero as D is invertible. Let $z = (\boldsymbol{x}, \boldsymbol{y})$ be a (nonzero) eigenvector corresponding to Θ. Then using $Dz = \Theta z$, we obtain

$$\boldsymbol{x} - \alpha A\boldsymbol{x} + \eta\boldsymbol{y} = \Theta\boldsymbol{x} \tag{14a}$$

$$-\alpha A\boldsymbol{x} + \eta\boldsymbol{y} = \Theta\boldsymbol{y}. \tag{14b}$$

Substituting (14b) in (14a) and solving for $\boldsymbol{y}$ (as $\Theta \neq 0$)

$$\boldsymbol{y} = \{(\Theta - 1)/\Theta\}x. \tag{15}$$

Further substituting (15) in (14b), we get

$$A\boldsymbol{x} = \{(\Theta - \eta)(\Theta - 1)/(-\alpha\Theta)\}x. \tag{16}$$

As $(\Theta - \eta)(\Theta - 1)/(-\alpha\Theta)$ is a scalar, $\boldsymbol{x}$ is an eigenvector of A. Let it correspond to the eigenvalue μ. μ is real as A is symmetric. Then, substituting $A\boldsymbol{x} = \mu\boldsymbol{x}$ in (16) and equating the scalars (as $\boldsymbol{x}$ is nonzero, else it is easily seen from (14) that $\boldsymbol{y}$ will also be zero and therefore z),

$$\mu = (\Theta - \eta)(\Theta - 1)/(-\alpha\Theta) \tag{17}$$

which reduces to

$$\Theta^2 - \Theta(1 - \alpha\mu + \eta) + \eta = 0 \tag{18}$$

which completes the proof. □

Jury's criterion [4] is used to check whether a polynomial has roots within or without the unit circle. In our case, this reduces to (considering (18) for different eigenvalues μ of A),

$$\alpha\mu > 0 \tag{19a}$$

$$2 + 2\eta - \alpha\mu > 0 \tag{19b}$$

$$|\eta| < 1. \tag{19c}$$

In general, the BPM algorithm is used with both α and η positive. With this restriction, (19) further simplifies to

$$\mu > 0 \tag{20a}$$

$$2 + 2\eta - \alpha\mu > 0 \tag{20b}$$

$$0 < \eta < 1. \tag{20c}$$

Thus it is sufficiant that all the eigenvalues of A be positive for us to conclude that s is locally asymptotically stable. By our assumption, all local minima satisfy this condition. Thus all the local minima of $E(\cdot)$ are locally asymptotically stable. If even one eigenvalue of A is negative, then s is unstable. In particular, all local maxima of $E(\cdot)$ are unstable. Condition (20b) may be violated if μ is large. But in most cases all the minima which are of interest lie within a bounded set. Thus $\nabla^2 E$ is bounded and therefore if α is sufficiently small, all the local minima are stable. Of course, if $\nabla^2 E$ is bounded, there need not be any restriction on considering minima within a bounded set.

Next, we consider a scalar case where $f(u) = -cu^2/2\,(c > 0)$. Thus $f'(u) = -cu$. The BP and BPM algorithms can be written as

$$u(n+1) = (1 - \alpha c)u(n) \tag{21a}$$

$$u(n+1) = u(n) - \alpha c u(n) + \eta\{u(n) - u(n-1)\}. \tag{21b}$$

It can be seen that the BP algorithm corresponds to a linear first-order discrete time system with a pole at $\tau = (1 - \alpha c)$. Assume that α is small enough so that $\tau > 0$. The BPM algorithm has two poles at

$$\Theta_1 = \frac{1}{2}\{\tau + \eta + [(\tau + \eta)^2 - 4\eta]^{1/2}\} \tag{22a}$$

$$\Theta_2 = \frac{1}{2}\{\tau + \eta - [(\tau + \eta)^2 - 4\eta]^{1/2}\}. \tag{22b}$$

BPM speeds up convergence in this case if $|\Theta_1|$ and $|\Theta_2|$ are less than τ. It can easily be seen that choosing a negative value for η will make $|\Theta_1|$ greater than τ. Thus, a positive value for η is necessary to accelerate convergence, which justifies using only a positive value of η.

III. Conclusions

It is shown in this letter that the stable points of the BPM algorithm are the local minima of the least squares error. Other equilibrium points are unstable. It is also shown by a simple example that if the momentum term is negative, the speed of convergence goes down. This analysis does not prove that BPM will converge to one of the local minima. But it can be easily shown that for small values of α (and $|\eta| < 1$) BP and BPM have essentially the same behavior over a finite time interval. Thus, if BP converges to a local minima u, BPM will be within a small enough neighborhood of u if small enough α is used. One can then use the fact that local minima are locally asymptotically stable to prove that BPM converges to u.

Similar techniques can be used to analyze algorithms with higher order memory. That is, where the "momentum term" depends not only on $u(n-1)$ but also on $u(n-2), \cdots, u(n-N)$ for some N.

References

[1] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1, in *Foundations*. Cambridge, MA: M.I.T. Press, 1986.

[2] C. M. Kuan and K. Hornik, "Convergence of learning algorithms with constant learning rates," *IEEE Trans. Neural Networks*, vol. 2, no. 5, pp. 484–490, 1991.

[3] J. A. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation.* Redwood City, CA: Addison Wesley, 1991.

[4] B. C. Kuo, *Discrete Data Control Systems.* Englewood Cliffs, NJ: Prentice-Hall, 1970.

[5] A. M. Ostrowski, *Solutions of Equations in Euclidean and Banach Spaces.* New York: Academic, 1973.