Study on Stochastic Gradient Descent Without Explicit Error Backpropagation with Momentum

1st Shahrzad Mahboubi
Dept. Informatics, *Shonan Institute of Technology*
Fujisawa, Japan
shaa@info.shonan-it.ac.jp

2nd Hiroshi Ninomiya
Dept. Informatics, *Shonan Institute of Technology*
Fujisawa, Japan
ninomiya@info.shonan-it.ac.jp

***Abstract*—This paper describes a novel training algorithm based on the Stochastic Gradient Descent method without explicit error backpropagation (SWDP) with momentum term for faster neural network training.**

***Index Terms*—neural network, training algorithm, stochastic gradient descent, weight difference propagation, momentum**

I. Introduction

Neural Networks (NN) are generally trained with a gradient algorithm based on the error Back Propagation (BP) method. BP is a widely used training algorithm. Various improvement methods have been proposed, such as Stochastic Gradient Descent (SGD) method for learning big data or SGD+Momentum, a faster version of SGD [1]. Recently, the Feedback-Network-Free method was proposed as a method that enables asynchronous parallel learning with light hardware cost, taking into account biological relevance [2]. While this method had allowed training by differential propagation of weights, updating weights had been limited to each sample. To solve this problem, the Stochastic Weight Difference Propagation (SWDP) method, which was SGD without explicit error backpropagation, had been proposed [3]. SWDP updated the weights only by the inner product of the weights of the next layer and their differences. This allowed each neuron in each layer to be trained in parallel. Therefore, backpropagation for each training sample in mini-batch (stochastic) learning was unnecessary, and the computational cost could be reduced. However, since SWDP simplified the learning structure of SGD, it introduced errors compared to the calculation of exact backpropagation, and these errors significantly affected the learning. Therefore, SWDP required more iterations than SGD [3]. This paper proposes a new algorithm as Momentum in Stochastic Weight Difference Propagation (MoSWDP) that improves the disadvantage of SWDP. MoSWDP is expected to suppress the increase in the number of iterations, which is one of the problems of SWDP, and to accelerate the learning process while maintaining the training accuracy. The proposed method is simulated on a benchmark problem and compared with conventional algorithms to show its effectiveness.

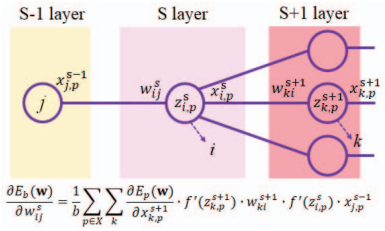

Fig. 1. The gradient in SGD for weight w_{ij}^{s}.

The Japan Society supported this work for the Promotion of Science (JSPS), KAKENHI (23K11267).

II. SWDP

This paper considers the multi-layer feed-forward NN with stochastic training, which is an unconstrained optimization problem to minimize the error function $E_b(\mathbf{w}) = (1/b)\sum_{p\in X} E_p(\mathbf{w})$ with respect to the weight vector $\mathbf{w} \in \mathbb{R}^n$. Where, $E_p(\mathbf{w})$ and $b = |X|$ denote the error of the p^{th} sample and mini-batch size, respectively. Let $\mathbf{x}_p$ and $\mathbf{o}_p$ be the p^{th} input and output vectors, respectively. The relation between the inputs and outputs of the NN is defined as $\mathbf{o}_p = \mathbf{x}_p^{\text{out}} = f_{NN}(\mathbf{w}, \mathbf{x}_p)$. Moreover, let $x_{i,p}^{s}$ be the output of the i^{th} neuron in the s^{th}layer $(1 \le s \le out)$ for the p^{th} sample, and w_{ij}^{s} be the weight from the j^{th} neuron of the $(s-1)^{\text{th}}$layer to the i^{th} neuron of the s^{th}layer, then the input-output relation of the neuron is given by $x_{i,p}^{s} = f(z_{i,p}^{s})$ and $z_{i,p}^{s} = \sum_j w_{ij}^{s} \cdot x_{j,p}^{s-1}$. Note that $s = out$ denotes the output layer. Where $f(z_{i,p}^{s})$ and w_{ij}^{s} denote the activation function and a component of the weight vector $\mathbf{w}$, respectively. The update formula of SGD is defined as $w_{ij}^{s}(t+1) = w_{ij}^{s}(t) - \eta(\partial E_b(\mathbf{w})/\partial w_{ij}^{s})$. Where η, t and $\partial E_b(\mathbf{w})/\partial w_{ij}^{s}$ denote the learning rate, iteration number, and the mini-batch gradient for w_{ij}^{s}, respectively. To update the weights in all layers, SGD requires the inner product of the weight and the gradient, which includes the backpropagation component, for

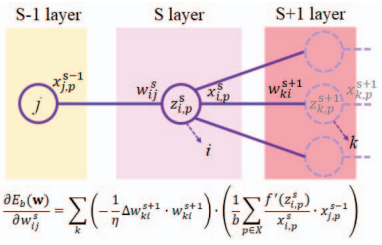

Fig. 2. The gradient in SWDP for weight w_{ij}^{s}.

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00241

all p in b. Figure 1 shows the required components for the gradient of SGD. On the other hand, SWDP has the same update formula as SGD only for the output layer. For the other layers, the weights are updated only by the inner product of the weights and the difference of weights [3]. The components required for the gradient of the SWDP are shown in Figure 2. Therefore, the update formula for w_{ij}^s of SWDP when the s^{th}layer is not the output layer is $w_{ij}^s(t+1) = w_{ij}^s(t) + \sum_k \left(\Delta w_{ki}^{s+1} \cdot w_{ki}^{s+1}\right) \cdot \left((1/b)\sum_{p\in X}(f'(z_{i,p}^s)/x_{i,p}^s) \cdot x_{j,p}^{s-1}\right)$. SWDP reduces the computational complexity in the backward calculations of the training to $1/b$ compared to SGD. However, since SWDP simplifies the learning structure of SGD, it introduces errors compared to the calculation of exact back-propagation, and these errors significantly affect the learning. Therefore, SWDP requires more iterations than SGD [3].

III. MoSWDP

This paper proposes MoSWDP for accelerating the training of SWDP. The idea of accelerating by introducing momentum is commonly used for algorithms based on BP (SGD), and one of the well-known methods is SGD+Momentum. The update formula for the SGD+Momentum for w_{ij}^s is $w_{ij}^s(t+1) = w_{ij}^s(t) + \mu v_{ij}^s(t) - \eta(\partial E_b(\mathbf{w})/\partial w_{ij}^s)$. Where $\mu v_{ij}^s(t) = \mu(w_{ij}^s(t+1) - w_{ij}^s(t)) = \mu\Delta w_{ij}^s(t+1)$ is the momentum and $0 \le \mu \le 1$ is the momentum coefficient. In this study, to derive the MoSWDP, let $w_{ij}^s(t+1) = w_{ij}^s(t) + v_{ij}^s(t+1)$ be the update formula for w_{ij}^s. where, $v_{ij}^s(t+1) = \Delta w_{ij}^s(t+1) = w_{ij}^s(t+1) - w_{ij}^s(t) = \mu v_{ij}^s(t) - \eta(\partial E_b(\mathbf{w})/\partial w_{ij}^s)$. In MoSWDP, $\partial E_b(\mathbf{w})/\partial w_{ij}^s$ is defined as in SGD+Momentum when s^{th}layer neurons are output layer. Therefore, the update vector Δw_{ij}^s is updated as $\Delta w_{ij}^s(t+1) = \mu\Delta w_{ij}^s(t) - \eta\partial E_b(\mathbf{w})/\partial w_{ij}^s = \mu\Delta w_{ij}^s(t) + \eta/b\sum_{p\in X}(d_{i,p} - x_{i,p}^{out}) \cdot x_{j,p}^{s-1}$, when the cross-entropy error function is used. On the other hand, if the s^{th}layer is not the output layer, the gradient $\partial E_b(\mathbf{w})/\partial w_{ij}^s$ is given as (1) [3].

$$\frac{\partial E_b(\mathbf{w})}{\partial w_{ij}^s} \simeq \sum_k \frac{\partial E_b(\mathbf{w})}{\partial w_{ki}^{s+1}} w_{ki}^{s+1} \cdot \left(\frac{1}{b}\sum_{p\in X}\frac{f'(z_{i,p}^s)}{x_{i,p}^s} x_{j,p}^{s-1}\right). \tag{1}$$

where the gradient of the $(s+1)^{\text{th}}$layer $(\partial E_b(\mathbf{w})/\partial w_{ki}^{s+1}) = -(1/\eta)\left(\Delta w_{ki}^{s+1}(t+1) - \mu\Delta w_{ki}^{s+1}(t)\right)$. Substituting the gradient of the $(s+1)^{\text{th}}$layer into (1), a gradient with momentum shown in (2) is obtained.

$$\frac{\partial E_b(\mathbf{w})}{\partial w_{ij}^s} = -\frac{1}{\eta}\sum_k w_{ki}^{s+1}\left(\Delta w_{ki}^{s+1}(t+1) - \mu\Delta w_{ki}^{s+1}(t)\right) \cdot \frac{1}{b}\sum_{p\in X}\frac{f'(z_{i,p}^s)}{x_{i,p}^s} x_{j,p}^{s-1}. \tag{2}$$

Therefore, when the s^{th}layer is other than the output layer, the update formula of the weight w_{ij}^s of the proposed method MoSWDP is obtained from (3).

$$w_{ij}^s(t+1) = w_{ij}^s(t) + \mu\Delta w_{ij}^s(t) + \sum_k w_{ki}^{s+1}\left(\Delta w_{ki}^{s+1}(t+1) - \mu\Delta w_{ki}^{s+1}(t)\right) \cdot \frac{1}{b}\sum_{p\in X}\frac{f'(z_{i,p}^s)}{x_{i,p}^s} x_{j,p}^{s-1}. \tag{3}$$

Since momentum is used in MoSWDP to update the weights, it is expected to accelerate the SWDP.

IV. Simulation Results

The performance of the proposed method, MoSWDP, was compared with SGD, SGD+Momentum, and SWDP on the 8×8 pixels MNIST dataset [3]. The mini-batch size was $b = 1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20$, and 32. The termination conditions were set to $E(\mathbf{w}) < 10^{-3}$, and the maximum iteration counts $t_{max} = 500,000$, $\eta = 0.01$, and $\mu = 0.85$. From Figure 3, it can be seen that MoSWDP achieved the same training accuracy as SWDP. On the other hand, Figure 4 shows that the proposed method accelerated SWDP.

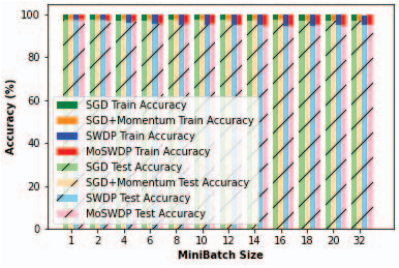

Fig. 3. Relationship between Mini-batch size and accuracy.

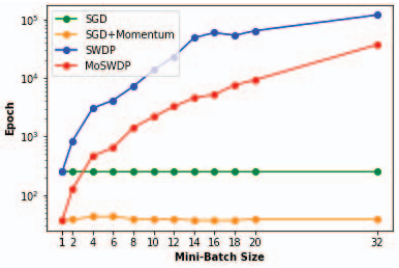

Fig. 4. Relationship between mini-batch size and epoch.

V. Conclusion

This paper proposes MoSWDP to improve SWDP. Computer experiments demonstrated the effectiveness of the proposed method. In the future, the proposed method should be verified for its effectiveness on other benchmark problems.

References

[1] I. Goodfellow, et al., "*Deep Learning*", MIT Press, 2016.

[2] F. Lin, "Supervised Learning in Neural Networks: Feedback-Network-Free Implementation and Biological Plausibility", *IEEE Trans. on NN and Learn. Sys.*, pp. 1–11, 2021.

[3] S. Mahboubi, et al., "Weight Difference Propagation for Stochastic Gradient Descent Learning", *Proc. ICCGI 2023*, *IARIA*, pp.12–17, 2023.