EXPLAINING THE LEARNING DYNAMICS OF DIRECT FEEDBACK ALIGNMENT

Justin Gilmer*[*], Colin Raffel[*], Samuel S. Schoenholz[*], Maithra Raghu & Jascha Sohl-Dickstein
Google Brain
{gilmer,craffel,schsam,maithra,jaschasd}@google.com

ABSTRACT

Two recently developed methods, Feedback Alignment (FA) and Direct Feedback Alignment (DFA), have been shown to obtain surprising performance on vision tasks by replacing the traditional backpropagation update with a random feedback update. However, it is still not clear what mechanisms allow learning to happen with these random updates. In this work we argue that DFA can be viewed as a noisy variant of a layer-wise training method we call Linear Aligned Feedback Systems (LAFS). We support this connection theoretically by comparing the update rules for the two methods. We additionally empirically verify that the random update matrices used in DFA work effectively as readout matrices, and that strong correlations exist between the error vectors used in the DFA and LAFS updates. With this new connection between DFA and LAFS we are able to explain why the "alignment" happens in DFA.

1 INTRODUCTION

Deep neural networks have achieved human or superhuman performance on an increasing variety of tasks. These networks are most frequently trained with the back-propagation algorithm (Rumelhart et al., 1985). However, it has been suggested (Bengio et al., 2015b) that the back-propagation algorithm is not biologically plausible, and recently there have been a number of proposals for learning the weights of a neural network in a way which could be more feasibly implemented in the brain (Lillicrap et al., 2014; Nøkland, 2016; Liao et al., 2015; Scellier & Bengio, 2016; Bengio et al., 2015a). One such proposal is feedback-alignment (FA) (Lillicrap et al., 2014), which demonstrated that one can swap out the weight matrices used in a backward pass with fixed random matrices and still achieve comparable performance to standard backpropagation. They also prove that for a linear network with a single hidden layer trained with FA, the feedforward weight matrix will approach the pseudo-inverse of the fixed random weight matrix. Consequently, the gradient updates will become correlated with the correct gradient update from backpropagation.

Motivated by feedback-alignment, a variant called direct-feedback-alignment (DFA) was proposed in (Nøkland, 2016). This method differs from feedback-alignment in that the error signal from the output is directly propagated to each internal layer via a random matrix. DFA was found to achieve comparable performance to FA on the MNIST and CIFAR-10 datasets. DFA is also interesting because it allows for updates to internal matrices to be computed in parallel, which could lead to more efficient training algorithms.

In this work we interpret DFA as a noisy variant to a supervised layer-wise training scheme that we call Linear Aligned Feedback Systems (LAFS). We show that the update equations are very similar between the two training schemes, only differing in the error vectors used to initialize backprop. Finally, we empirically demonstrate that the error vectors used in the LAFS and DFA updates are correlated throughout training.

[*]Work done as a member of the Google Brain Residency Program (g.co/brainresidency)

2 Direct Feedback Alignment

We begin with the update equations for training a k-layer feed forward network with DFA. We use $\mathbf{W}^i$ and $\mathbf{b}^i$ to denote the learned weight matrix and bias vector in layer i, and f to denote the nonlinearity. For all $i \in \{1, \ldots, k\}$ we have

$$\mathbf{a}^i = \mathbf{W}^i \mathbf{h}^{i-1} + \mathbf{b}^i, \quad \mathbf{h}^i = f(\mathbf{a}^i), \tag{1}$$

where $\mathbf{h}^0 = \mathbf{x}$ is the input to the network. For the output we have

$$\hat{\mathbf{y}} = f_y(\mathbf{a}^k), \tag{2}$$

where f_y is the softmax function. We assume a differentiable loss function $L(\mathbf{y}, \hat{\mathbf{y}})$, where y is the target output. The error vector $\mathbf{e}^y$ is defined to be the gradient of L with respect to the logits $\mathbf{a}^k$. DFA creates fixed random matrices $\mathbf{B}^i \in \mathbb{R}^{d \times d_y}$ for $i = 1 \ldots k-1$, where d is the dimension of all the hidden layers, and computes for $i \neq k$

$$\delta \mathbf{a}^i = (\mathbf{B}^i \mathbf{e}^y) \odot f_y'(\mathbf{a}^i), \tag{3}$$

and at layer k, $\delta \mathbf{a}^k = \mathbf{e}^y$. The update to the weight matrix for layer i is then

$$\delta \mathbf{W}^i = -\delta \mathbf{a}^i \left(\mathbf{h}^{i-1}\right)^T = \left((\mathbf{B}^i \mathbf{e}^y) \odot f_y'(\mathbf{a}^i)\right) \left(\mathbf{h}^{i-1}\right)^T. \tag{4}$$

For comparison, in traditional backpropagation

$$\delta \mathbf{a}^i = \nabla_{\mathbf{h}^i} L \odot f_y'(\mathbf{a}^i) = \left(\mathbf{W}^i\right)^T \delta \mathbf{a}_{i+1} \odot f_y'(\mathbf{a}^i). \tag{5}$$

3 Linear Aligned Feedback Systems

When training a feed forward network with LAFS, we first append auxiliary output matrices to each of the internal hidden layers and define an auxiliary loss at each layer. For each layer $i \neq k$ we introduce an auxiliary output defined as

$$\hat{\mathbf{y}}^i = f_y(\left(\mathbf{B}^i\right)^T \mathbf{h}^{i-1}), \tag{6}$$

where $\mathbf{B}^i$ are fixed random readout matrices. This is similar to the concept of linear probes introduced in Alain & Bengio (2016). Layer k is kept the same as for a feed forward network,

$$\hat{\mathbf{y}}^k = f_y(\mathbf{W}^k \mathbf{h}^{k-1} + \mathbf{b}^k). \tag{7}$$

Each layer then has a loss defined as

$$L^i = L(\hat{\mathbf{y}}^i, \mathbf{y}). \tag{8}$$

When training the weights of the LAFS model we train the weights in layer i to optimize the auxiliary loss L^i. Note that gradients are isolated within each layer – that is we do not consider how changing the weights in layer i affects the loss in layer $j > i$. This corresponds to layer-wise training, where layer i predicts the target by mapping the features $\mathbf{h}^{i-1}$ from layer $i-1$ to $\mathbf{h}^i$ followed by feeding $\mathbf{h}^i$ through the random readout matrix $\mathbf{B}^i$. The key insight behind this work is that the derived update equations for the LAFS network are very similar to the update equations for DFA. In particular, the update to weight matrix $\mathbf{W}^i$ is calculated as

$$\delta \mathbf{W}^i := \nabla_{\mathbf{W}^i} L^i = \left((\mathbf{B}^i \mathbf{e}^i) \odot f_y'(\mathbf{a}^i)\right) \left(\mathbf{h}^{i-1}\right)^T \tag{9}$$

where $\mathbf{e}^i$ is the gradient of the i'th loss with respect to the auxiliary logits at the i'th layer $\left(\mathbf{B}^i\right)^T \mathbf{h}^{i-1}$.

Comparing Equations 4 and 9 the connection between DFA and LAFS is clear. The only difference is the error vector which initializes the backprop update. For a randomly initialized network with small weight variance, the initial predictions in the LAFS model will be roughly uniform, due to the fact that the softmax will map a vector of logits with small norm to an approximate uniform distribution over the targets. In such initialization schemes, e_i will be very similar to e_k, and as a result the DFA update will be correlated with the LAFS update. As training progresses, all layers are optimized to predict the same targets, and as a result these correlations can be expected to persist through training. We verify this empirically in the following section, by plotting the angles between e^i and e^y during training.

4 Results

We trained the same feed forward architecture on MNIST with both DFA and LAFS, with learning rate 0.1, $\tanh$ activations, and 3 hidden layers each with 800 nodes. In Figure 1 we look at the accuracy of the auxiliary readouts at each layer for both DFA and LAFS on the training set. Although the random matrices $\mathbf{B}^i$ are not used explicitly as readout matrices in the DFA model, we find that using the projections of the intermediate layers through the $\mathbf{B}^i$ obtains good accuracy. We also look at the angles between $\mathbf{e}^i$ and $\mathbf{e}^y$ in both DFA and LAFS and find that they are well aligned throughout training, which we believe is the primary condition for DFA to train properly.

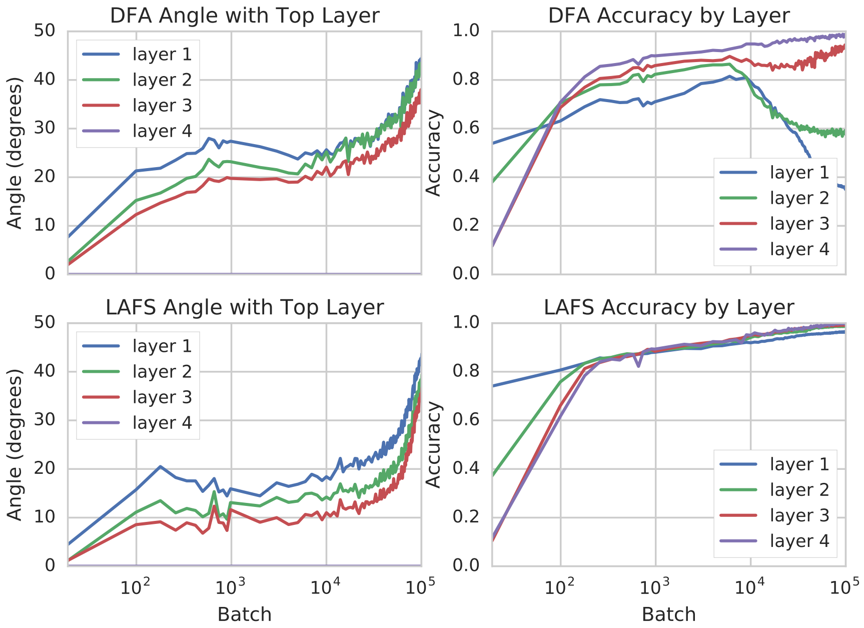

Figure 1: DFA performs like a noisy version of LAFS. *Left:* The angle in degrees between the error vector in the top layer with the intermediate layers. *Right:* The accuracy of the readout in the intermediate layers. Best viewed in color. Note the log scale of the x-axis.

5 Discussion

Training a model with LAFS is closely related to training stacked linear systems, where the i'th linear system is trained on the output of the $i-1$ system. It is important that the backprop update of each system goes through the non-linearity f, we believe this is why the model is able to perform better than linear models. The LAFS update rule may be limited in its expressive power as there is no training signal which captures interactions between layers, this limitation must then also exist for the DFA update rule. We question whether or not LAFS and DFA can be expected to outperform training single hidden layer networks, and leave this for future work.

Acknowledgments

We would like to thank Sergey Ioffe, Wojtek Czarnecki, Ben Poole, and Arild Nkland for helpful discussions. We also thank Alex Irpan and Azalia Mirhoseini for their helpful comments on this manuscript.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Yoshua Bengio, Asja Fischer, Thomas Mesnard, Saizheng Zhang, and Yuhai Wu. From stdp towards biologically plausible deep learning. In *Deep Learning Workshop, International Conference on Machine Learning (ICML)*, 2015a.

Yoshua Bengio, Dong-Hyun Lee, Jorg Bornschein, Thomas Mesnard, and Zhouhan Lin. Towards biologically plausible deep learning. *arXiv preprint arXiv:1502.04156*, 2015b.

Qianli Liao, Joel Z Leibo, and Tomaso Poggio. How important is weight symmetry in backpropagation? *arXiv preprint arXiv:1510.05067*, 2015.

Timothy P. Lillicrap, Daniel Cownden, Douglas B. Tweed, and Colin J. Akerman. Random feedback weights support learning in deep neural networks. *arXiv preprint arXiv:1411.0247*, 2014.

Arild Nøkland. Direct feedback alignment provides learning in deep neural networks. In *Advances In Neural Information Processing Systems*, pp. 1037–1045, 2016.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning internal representations by error propagation. Technical report, DTIC Document, 1985.

Benjamin Scellier and Yoshua Bengio. Towards a biologically plausible backprop. *arXiv preprint arXiv:1602.05179*, 2016.