Deep Learning without Weight Transport

Diego Aguado

December 6, 2019

Overview

1. Context: Biologically plausible learning
2. Biologically plausible algorithms
    - Weight Mirrors
    - Kolen-Pollack algorithm
3. Experiments
4. Discussion/Future Work

Weight Transport Problem

The default way to train Neural Networks that achieve state-of-the-art results is using backpropagation. In order to use backprop, training relies on a forward (1) and a feedback pass (2).

$$y_{l+1} = \phi(\mathbf{W}_{l+1}\mathbf{y}_l + \mathbf{b_{l+1}}), \tag{1}$$

$$\boldsymbol{\delta}_l = \phi'(\mathbf{y}_l)\mathbf{W}_{l+1}^T\boldsymbol{\delta}_{l+1}. \tag{2}$$

Weight Transport Problem: Biological interpretation

When giving (1) a biological interpretation in the brain, we can consider

- the input of a given layer, i.e. $\mathbf{y}_l$, vectors of neural firing
- $\mathbf{W}_{l+1}$ synaptic weights
- $\mathbf{b}_{l+1}$ bias currents
- ϕ some non linearlity allowing to model spiking behaviour only after certain threshold is reached

Weight Transport Problem: Biological interpretation

- The forward pass (1) is interpreted then as a brain structure and the feedback pass (2) as another one.
- The feedback pass could be another set of neurons different from the forward ones or the same set for inference through different signals.
- Yet the fact that the same synaptic weights show up in both signals is biologically not plausible.

Weight Transport Problem: Implausibility

In the brain, forward and feedback synapses are physically different. Implausibility arises from the fact that there's no known way of forward and feedback synapses coordinating the same synaptic weights to be the transpose of the other as used in (2).

Background: Feedback Alignment

In order to have an algorithm that could take place in the brain, the authors refer back to the feedback alignment algorithm [3].

$$\boldsymbol{\delta}_l = \phi'(\mathbf{y}_l)\mathbf{B}_{l+1}\boldsymbol{\delta}_{l+1}, \tag{3}$$

$$\Delta\mathbf{W}_{l+1} = -\eta_W \boldsymbol{\delta}_{l+1}\mathbf{y}_l^T. \tag{4}$$

With $\mathbf{B}_{l+1}$ non learning, fixed random weights.
It was shown on [3] that these learning algorithm can match backprop with simple tasks.

Feedback Alignment vs Backprop

Can Feedback Alignment scale to solve more complex tasks?

Enhancing Feedback Alignment

A way to enhance feedback alignment is to have learning feedback weights $\mathbf{B}_l$ without weight transport. In other words, $\mathbf{B}$ should learn the transpose of some other matrix $\mathbf{W}$. How? Asuming $\mathbf{y} = \mathbf{W}\mathbf{x}$

$$\Delta_B = \eta_B \mathbf{x}\mathbf{y}^T \tag{5}$$

Since $\mathbb{E}[\mathbf{x}\mathbf{y}^T] = \mathbb{E}[\mathbf{x}\mathbf{x}^T\mathbf{W}^T] = \mathbb{E}[\mathbf{x}\mathbf{x}^T]\mathbf{W}^T = \sigma^2\mathbf{W}^T$, then (5) will lead $\mathbf{B}$ to be a multiple of $\mathbf{W}^T$.

Weight Mirroring: a Circuit for learning the transpose

Using this motivation, weight mirrors circuit is proposed in [1].
The proposal almost constitutes a new architecture of neural networks and definately a new learning algorithm. This algorithm has two modes:

- Engaged mode:
  Infering and adjusting forward pass weights.
- Mirror mode:
  Adjust feedback pass weights to mimic forward ones.

A biological interpretation can be given to this two modes: engagement of an activity and sleeping/resting/idle in-between.

Engaged Mode

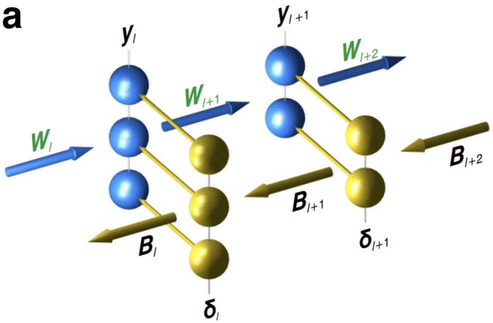

Figure: Engaged Mode

- Uses incoming/sensory input to infer and adjust forward weights.
- Usual forward pass
- Use (3) and (4) to update $\mathbf{W}_l$ (yellow Cross-projection)

Mirror Mode

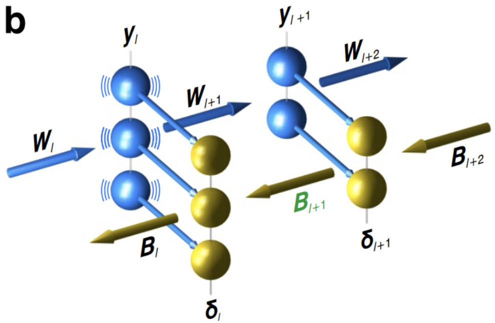

Figure: Engaged More

- Still computes $\mathbf{y}_l$ the usual way,
- Uses cross-projection on (6)
- Uses (7) to update $\mathbf{B}_{l+1}$

Mirror Mode

In mirror mode, error signals are computed in a way that allows $\mathbf{B}_{l+1}$ to mimic $\mathbf{W}_{l+1}^T$. To accomplish the previous, consider

$$\boldsymbol{\delta}_l = \mathbf{y}_l, \tag{6}$$

$$\Delta \mathbf{B}_{l+1} = \eta_B \boldsymbol{\delta}_l \boldsymbol{\delta}_{l+1}^T. \tag{7}$$

This means that the cross-projection "controls" the firing of the feedback path.

Equation (7) can be seen as Hebbian learning.

Hebbian learning states that the change in weight connections between neurons is proportional to the product of activations between neurons.

Mirror weight circuit: Why does it work?

The circuit $\{\mathbf{y}_l, \mathbf{y}_{l+1}, \boldsymbol{\delta}_l, \boldsymbol{\delta}_{l+1}\}$ is the weight mirror, it manages to approximate $\mathbf{W}_l^T$ through $\mathbf{B}_l$.
Why?
To see that (7) actually will follow (5) we recall that

$$\boldsymbol{\delta}_l \boldsymbol{\delta}_{l+1}^T = \mathbf{y}_l \mathbf{y}_{l+1}^T = \mathbf{y}_l \phi(\mathbf{W}_{l+1}\mathbf{y}_l + \mathbf{b}_{l+1})^T \tag{8}$$

Mirror weight circuit: Why does it work?

Under certain conditions about differentiability of ϕ and scale of σ^2:

$$\phi(\mathbf{W}_{l+1}\mathbf{y}_l + \mathbf{b}_{l+1}) \approx \phi'(\mathbf{b}_{l+1})\mathbf{W}_{l+1}\mathbf{y}_l + \phi(\mathbf{b}_{l+1}). \tag{9}$$

Hence

$$\begin{aligned} \mathbb{E}[\Delta\mathbf{B}_{l+1}] &\approx \eta_B \mathbb{E}[\mathbf{y}_l\mathbf{y}_l^T]\mathbf{W}_{l+1}^T\phi'(\mathbf{b}_{l+1})^T + \mathbb{E}[\mathbf{y}_l]\phi(\mathbf{b}_{l+1})^T \\ &= \eta_B \mathbb{E}[\mathbf{y}_l\mathbf{y}_l^T]\mathbf{W}_{l+1}^T\phi'(\mathbf{b}_{l+1})^T \\ &= \eta_B \sigma^2 \phi'(\mathbf{b}_{l+1})\mathbf{W}_{l+1}^T \end{aligned} \tag{10}$$

Hence $\mathbf{B}_{l+1}$ will approximate $\mathbf{W}_{l+1}^T$.

Mirro weight circuit: Why does it work?

If biases are small enough or under assumption of bias blocking and same activation function accross layers then, $\mathbf{B}_{l+1}$ will be a positive scalar approximate of $\mathbf{W}_{l+1}^{T}$.

$$\mathbb{E}[\Delta \mathbf{B}_{l+1}] \approx \eta_B \sigma^2 \phi'(0) \mathbf{W}_{l+1}^{T} \tag{11}$$

Mirror weight circuit: Two modes

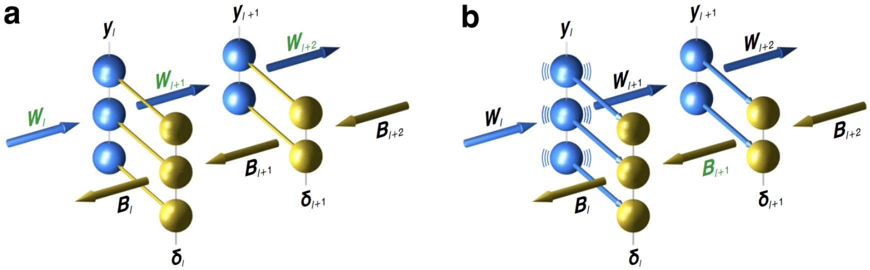

Figure: Mode Comparisson, source [1]

Kolen-Pollack

Kole-Pollack: Convergence through weight decay.

Kolen-Pollack: Convergence through weight decay.

Kolen and Pollack proposed an algorithm in [2] whose convergence relies on a weight decay mechanism.

$$\Delta W(t) = A(t) - \lambda W(t), \tag{12}$$

$$\Delta B(t) = A(t) - \lambda B(t). \tag{13}$$

We notice the weight decay, λ, is identical for two different signals as well as the adjustment $A(t)$.

Kolen-Pollack: Convergence through weight decay.

With time, these signals will converge if $0 < \lambda < 1$:

$$W(t+1) - B(t+1) = (1-\lambda)^{t+1}[W(0) - B(0)] \tag{14}$$

But transporting changes is also not biologically plausible. Authors propose also propose in [1] a circuit to implement KP's method without transport.

A circuit for KP's algorithm

Considering (5), the update rule for $\mathbf{B}_{l+1}$ on this algorithm relies on the input and error signal:

$$\Delta\mathbf{B}_{l+1} = -\eta\mathbf{y}_l\boldsymbol{\delta}_{l+1}^T. \tag{15}$$

So now, agreeing in learning rate and weight decay, we see,

$$\Delta\mathbf{W}_{l+1} = -\eta_W\boldsymbol{\delta}_{l+1}\mathbf{y}_l^T - \lambda\mathbf{W}_{l+1}, \tag{16}$$

$$\Delta\mathbf{B}_{l+1} = -\eta_W\mathbf{y}_l\boldsymbol{\delta}_{l+1}^T - \lambda\mathbf{B}_{l+1}, \tag{17}$$

which is

$$\Delta\mathbf{B}_{l+1}^T = -\eta_W\boldsymbol{\delta}_{l+1}\mathbf{y}_l^T - \lambda\mathbf{B}_{l+1}^T. \tag{18}$$

We see convergence of $\mathbf{B}_{l+1}$ to $\mathbf{W}_{l+1}^T$ following (14).

A circuit for KP's algorithm

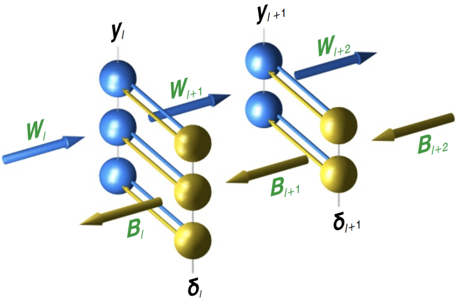

Figure: Circuit for KP's algorithm, source [1].

- Single computing mode
- Gold cross-projections, $\boldsymbol{\delta}_l$, allow $\mathbf{W}_l$ to perform forward pass update.
- Blue cross-projections, $\mathbf{y}_l$, allow $\mathbf{B}_l$ to perform feedback pass update.

Experiments

Experiments: Algorithm comparisson

Weight mirror (WM), Kolen-Pollack (KP), plain feedback alignment (FA), backpropagation and sign-symmetry (SS)[4] are compared on a complex visual task: ImageNet. The previous, in order to compare with recent biologically plausible algorithms.

This involved usage of Convolutional, BatchNorm and RELU layers.

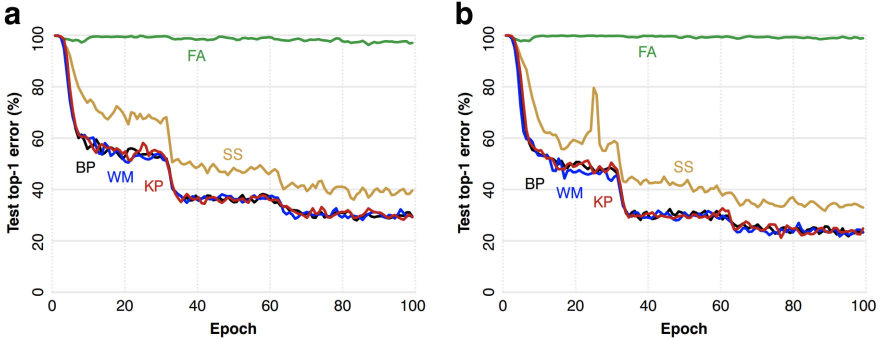

Figure: a) ResNet-18, b) ResNet-50. Source [1]

Experiments: Learning the transpose

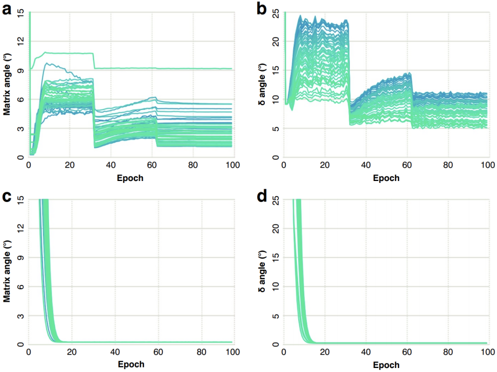

Figure: Angles of Matrices and deltas, WM, KP. Source [1]

Experiments: Learning the transpose

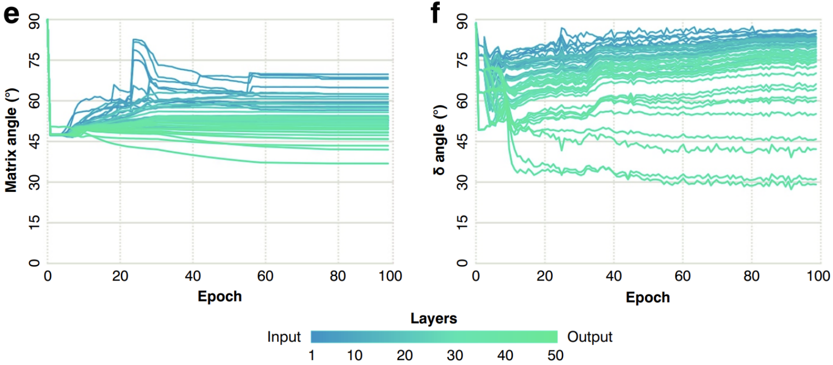

Figure: Angles of Matrices and deltas, SS. Source [1]

Discussion/Future Work

- One advantage of KP over Weight Mirrors is the single operation mode.
- Convergence of **B** on KP depends on the the weight-decay λ.
- More research needs to be done in non-convolutional layers.

[1] Mohamed Akrout, Collin Wilson, Peter Humphreys, Timothy Lillicrap, and Douglas B Tweed. Deep learning without weight transport. In *Advances in Neural Information Processing Systems*, pages 974–982, 2019.

[2] John F Kolen and Jordan B Pollack. Backpropagation without weight transport. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, volume 3, pages 1375–1380. IEEE, 1994.

[3] Timothy P Lillicrap, Daniel Cownden, Douglas B Tweed, and Colin J Akerman. Random synaptic feedback weights support error backpropagation for deep learning. *Nature communications*, 7:13276, 2016.

[4] Will Xiao, Honglin Chen, Qianli Liao, and Tomaso Poggio. Biologically-plausible learning algorithms can scale to large datasets. *arXiv preprint arXiv:1811.03567*, 2018.