A Geometric Interpretation of Feedback Alignment

Andreas Stöckel, Terrence C. Stewart, Chris Eliasmith | {astoecke, tcstewar, celiasmith}@uwaterloo.ca
Centre for Theoretical Neuroscience, University of Waterloo | http://ctn.uwaterloo.ca/

Motivation ①

Feedback alignment (FA;②) is a **biologically plausible supervised learning** method derived from backpropagation. While competetive for shallow networks, **FA fails to solve certain tasks** and has issues with training deep networks.

We present a geometric interpretation of FA that may help researchers to **better understand its limitations**.

Background ②

- **Backpropagation** assigns an error $\vec\delta^\nu$ to each layer ν. This error is propagated to previous layers $\nu - 1$ by transposing the connection weight matrix W^ν.

$$\vec\delta^{\mathrm{out}} = (\underbrace{\vec y}_{\text{Network output}} - \underbrace{\vec t}_{\text{Target}})^T$$

$$\vec\delta^\nu = \underbrace{f'(\vec x^\nu)}_{\text{Activation function differential}} \odot \left(W^{\nu+1}\right)^T \vec\delta^{\nu+1}$$

$$\Delta W^\nu = -\kappa \vec a^{\nu-1} \left(\vec\delta^\nu\right)^T$$

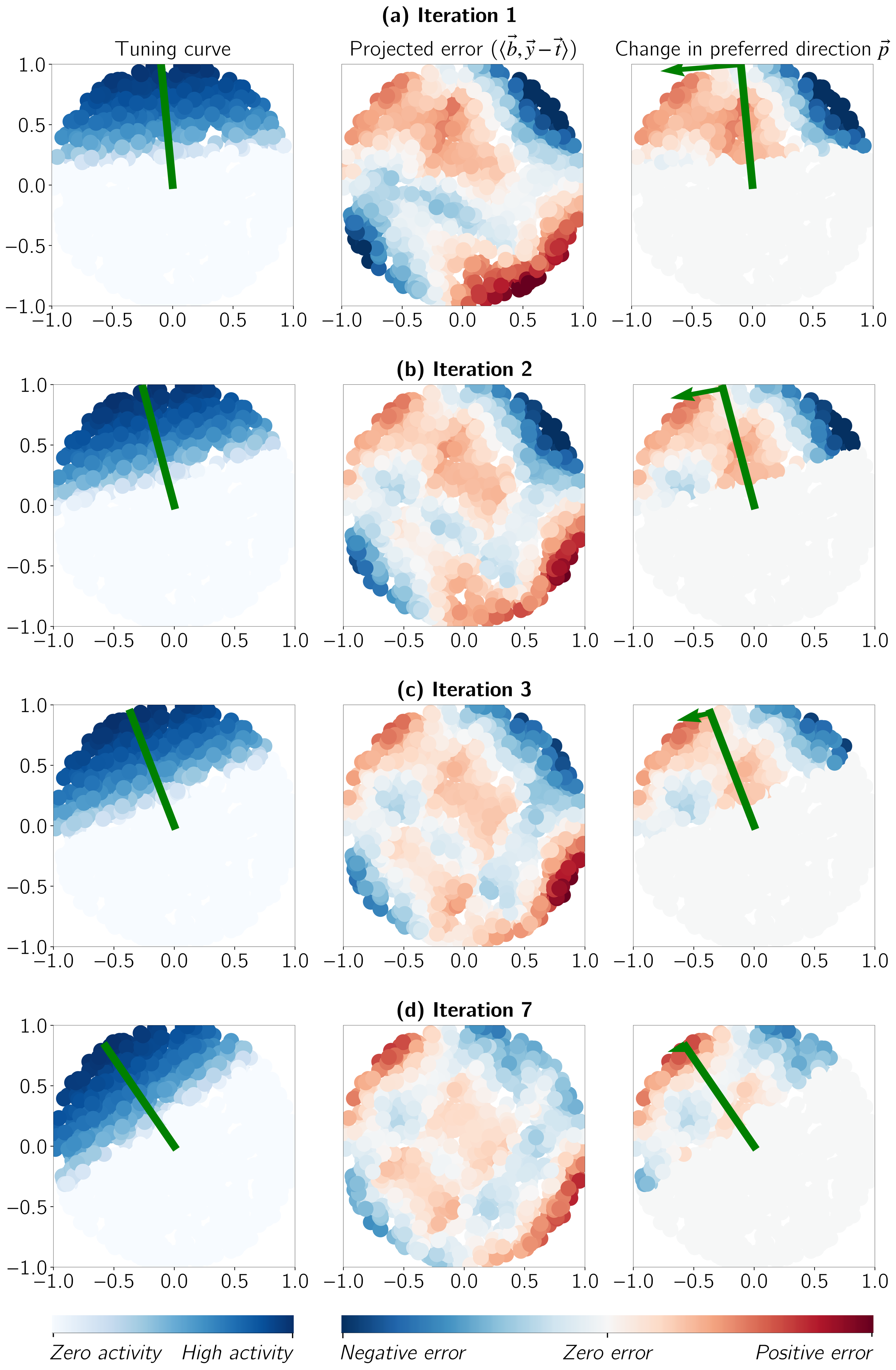
Having access to W^ν as feedback weights is biologically implausible.

- **Feedback alignment** (FA) [1] replaces W^ν with random feedback weights B^ν

$$\vec\delta^\nu = f'(\vec x^\nu) \odot B^\nu \vec\delta^{\nu+1}.$$

- Direct feedback alignment (DFA) sends the output error $\vec\delta^{\mathrm{out}}$ to each layer

$$\vec\delta^\nu = f'(\vec x^\nu) \odot B^\nu \vec\delta^{\mathrm{out}}.$$



▲ **Figure 1 Geometric interpretation of direct feedback alignment in a 2D space for a single neuron.** Coloured circles correspond to random samples in the input space. Green lines correspond to the preferred direction (encoder) $\vec p$. Input weights W^{in} are trained using DFA, output weights W^{out} are optimised using least squares.

Methods & Observations ③

- **Weight normalisation.** For each neuron i, we split input weights into a bias β_i, gain α_i, and a preferred direction $\|\vec p_i\| = 1$ (Figure 2)

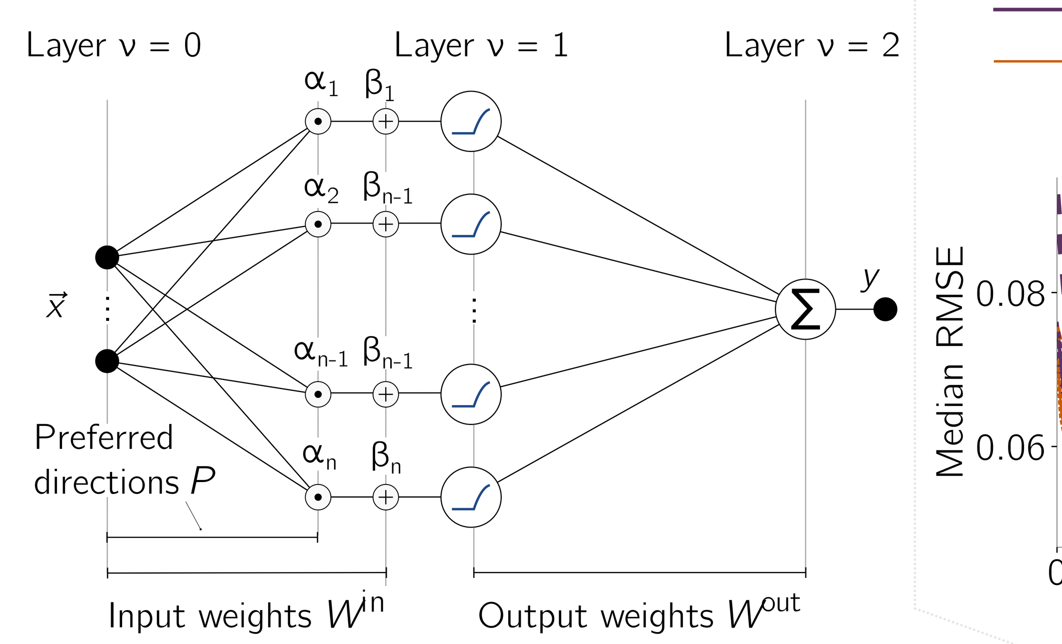
$$a_i = f(\alpha_i \langle \vec p_i, \vec x \rangle + \beta_i).$$

- **Preferred direction update.** Given input $\vec x$ and target $\vec t$, the preferred directions $\vec p$ move into the direction of the average input weighted by the back-projected output error $\left(\vec b_i\right)^T \left(\vec y - \vec t\right)$. (Figures 1, 3)

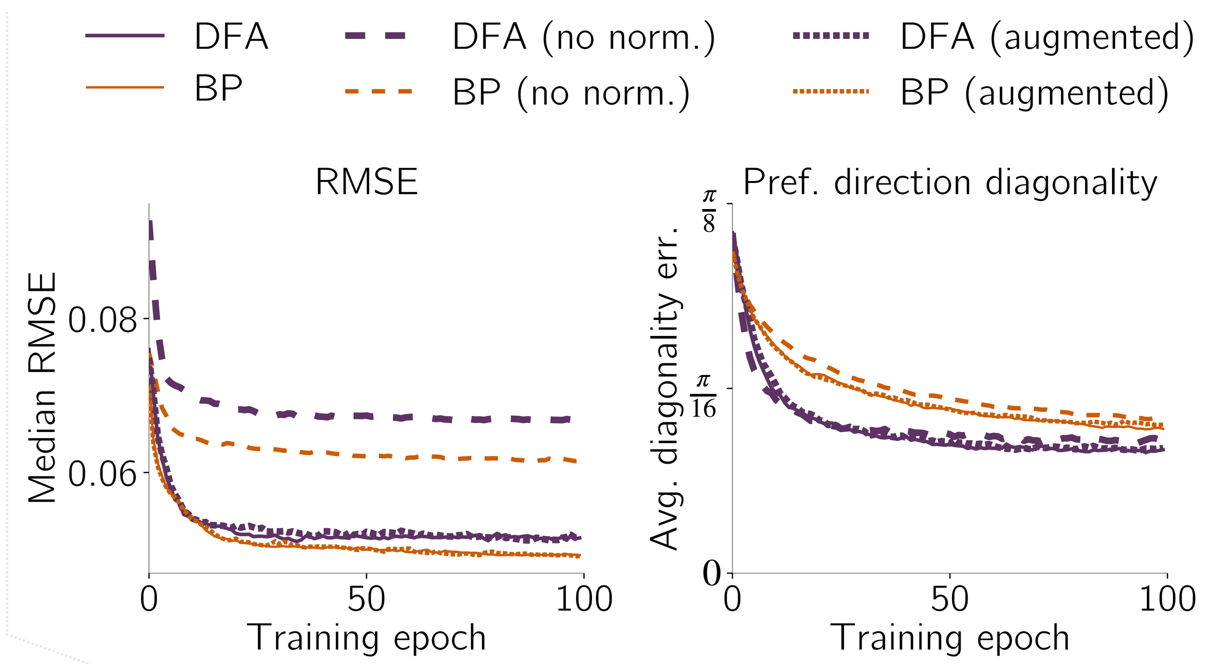
- **Augmented gradient.** The preferred direction vector can be kept normalised ("homeostasis") by augmenting the update rule [2] (Figure 4)

$$\Delta \vec w_i^\nu = -\kappa \alpha_i^\nu \left(a_i^{\nu-1} \vec\delta^{\mathrm{out}} - \frac{\left(\vec b_i\right)^T \vec a^{\nu-1} \delta_i^{\mathrm{out}}}{\left(\alpha_i^\nu\right)^2} \right).$$

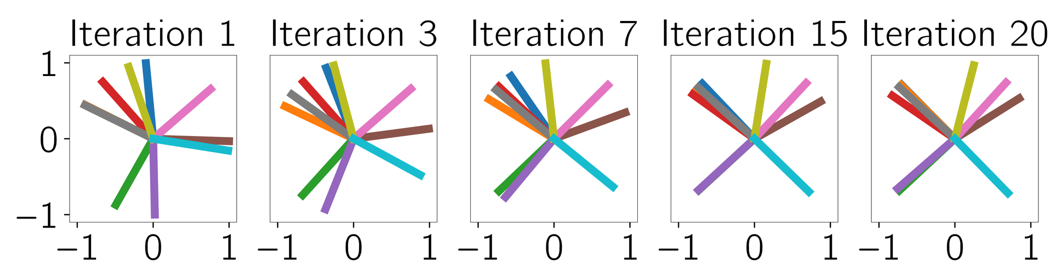
Direct Feedback Alignment greedily optimises the network by sensitising neurons to regions in the input space with large output errors.



▲ **Figure 2 Network topology used in our experiments.** The input weights are separated into gain α, bias β, and normalised preferred direction vector $\vec p$.



▲ **Figure 4 Learning multiplication with feedback alignment (DFA) and backpropagation (BP).** 20 hidden neurons; W^{out} optimised using least squares.



▲ **Figure 3 Preferred direction vectors over time for 20 neurons when training the network to compute multiplication.** Over time, the vectors align with the diagonals, which is locally optimal. [3]

References

[1] Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016, November). Random synaptic feedback weights support error backpropagation for deep learning. *Nature Communications*.
[2] Salimans, T., & Kingma, D. P. (2016). Weight Normalization: A Simple Reparameterization to Accelerate Training of Deep Neural Networks. In *Advances in Neural Information Processing Systems.*
[3] Gosmann, J. (2015). *Precise multiplications with the NEF* (Tech. Rep.). Waterloo, ON: Centre for Theoretical Neuroscience.