Learning to Learn with Feedback and Local Plasticity

Anonymous Author(s)

Abstract

Developing effective biologically plausible learning rules for deep neural networks is important for advancing connections between deep learning and neuroscience. To date, local synaptic learning rules like those employed by the brain have failed to match the performance of backpropagation in deep networks. In this work, we employ meta-learning to discover networks that learn using feedback connections and local, biologically motivated learning rules. Importantly, the feedback connections are not tied to the feedforward weights, avoiding any biologically implausible weight transport. It can be shown mathematically that this approach has sufficient expressivity to approximate any online learning algorithm. Our experiments show that the meta-trained networks effectively use feedback connections to perform online credit assignment in multi-layer architectures. Moreover, we demonstrate empirically that this model outperforms a state-of-the-art gradient-based meta-learning algorithm for continual learning on regression and classification benchmarks. This approach represents a step toward biologically plausible learning mechanisms that can not only match gradient descent-based learning, but also overcome its limitations. [1]

1 Introduction

Deep learning has achieved impressive success in solving complex tasks, and in some cases its learned representations have been shown to match those in the brain [18, 10]. However, there is much debate over how well the learning algorithm commonly used in deep learning, backpropagation, resembles biological learning algorithms. Causes for skepticism include the facts that (1) backpropagation ignores the nonlinearities imposed by neurons in the backward pass and assumes instead that derivatives of the forward-pass nonlinearities can be applied, (2) in backpropagation, feedback path weights are exactly tied to feedforward weights, even as weights are updated with learning, and (3) backpropagation assumes alternating forward and backward passes. The question of how so-called credit assignment – appropriate propagation of learning signals to non-output neurons – can be performed in biologically plausible fashion in deep neural networks remains open.

We propose a new learning paradigm that aims to solve the credit assignment problem in more biologically plausible fashion. Our approach is as follows: (1) endow a deep neural network with feedback connections that propagate information about target outputs to neurons at all layers, (2) apply local plasticity rules (e.g. Hebbian or neuromodulated plasticity) to update feedforward synaptic weights following feedback projections, and (3) employ meta-learning to optimize for the initialization of feedforward weights, the setting of feedback weights, and synaptic plasticity levels.

2 Related Work

Some research has investigated alternative algorithms to backpropagation that relax or eliminate the requirement of weight symmetry. A surprising set of results [13, 15], show that random feedback weights are sufficient to induce learning for simple tasks. Another family of methods, known as target propagation, use a reconstruction loss to learn a feedback pathway that approximates the inverse of the feedforward pathway [3]. However, both of these approaches have been found not to scale well to difficult tasks such as ImageNet classification [2]. To some extent, performance can be recovered by

[1] Code available anonymously at https://osf.io/92qhs/?view_only=a7b37d2403a745c6ae4e27953005524b

Submitted to the 2019 NeurIPS NeuroAI workshop. Do not distribute.

permitting sign-symmetry in forward and backward weights [12], but this partially re-introduces the weight symmetry issue and fails to address concerns (1) and (3) above.

Backpropagation-based deep learning notably falls short of human and animal learning in several key respects. In particular, it has difficulty learning from few examples, and learning in online fashion from a stream of data and on multiple tasks. One approach to addressing these issues is meta-learning, in which a network's learning procedure itself is learned in an "outer loop" of optimization. A popular class of such methods is gradient-based meta-learning [4], in which the network initialization is meta-optimized so that batch gradient descent will learn quickly from few examples of a new task. This method has been extended to the continual learning case, in which the "inner loop" optimization consists of many online gradient steps on a potentially nonstationary data distribution [8].

Building on the meta-learning paradigm, another line of research has explored the approach of performing inner-loop updates according to biologically motivated Hebbian learning rules rather than by gradient descent [1, 14]. However, none of these methods fully address the credit assignment problem, in that they either restrict plasticity to output weights or allow plasticity to proceed without any dependence on supervised error signals. Recent work has also considered meta-learning algorithm for learning feedback weights [11]. Their methods, based on node perturbation and RL algorithms, differ substantially from ours, but a comparison or synthesis could prove fruitful.

3 Method

In our model, a network propagates an input $\mathbf{x}$ forward through a neural network $\hat{f}(\cdot;\theta)$, receives a target signal $\mathbf{y}$ from the environment, and propagates a function $g(\mathbf{y})$ back to its neurons. The output of g is an update to the activations of the network. Subsequently, the network undergoes synaptic plasticity according to a local learning rule that adjusts a synaptic weight w based on the previous weight value, the presynaptic activity a, and the postsynaptic activity b resulting from feedback. In some experiments we allow plasticity only in the final N network layers (varying N).

We may take a to be the pre or post-feedback presynaptic activations. The post-feedback case corresponds to a model in which neural activations are updated directly with feedback and Hebbian-style plasticity ensues. The pre-feedback case requires error signals to be propagated without affecting the neural activations used in feedforward computation. This approach has the advantage of avoiding possible disruption to the feedforward computation, though it may be more difficult to implement. Possible biological implementations include a segregated dendrites model (see [6]), or feedback through neuromodulatory signals at postsynaptic sites, with weight updates that are proportional to presynaptic and neuromodulatory activity, but not postsynaptic activity – this effect has been directly observed in the Drosophila mushroom body, see [7]. In our simulations we use Oja's learning rule: $w \leftarrow w + \alpha(ab - b^2 w)$, wher α is a plasticity coefficient [16].

We use linear feedback connections with one ReLU nonlinearity. Concretely, the activation x_i at the output of layer W_i with corresponding feedback matrix G_i is set to $\text{ReLU}[(1-\beta_i)x_i + \beta_i G_i \mathbf{y}]$, where β_i controls the "strength" of feedback. Note that $\beta_i = 0$ corresponds to pure unsupervised Hebbian learning in layer i, while $\beta_i = 1$ corresponds to supervised learning.

3.1 Meta-learning procedure

The description above specifies how a network in our model learns in its "lifetime." However, to create a network that effectively learns using the above procedure, we employ meta-learning. In particular, for each of our benchmark tasks (described below) we simulate an entire learning episode and test input, evaluate the performance on the test input, and backpropagate through the entire learning procedure (see [4, 8]). The meta-learned parameters are the initial weights θ and feedback function g, as well as the plasticity coefficients for each plastic weight and each layer's β coefficient, which controls the balance of supervised vs. unsupervised learning occurring in that layer.

3.2 Universiality

We are able prove that sufficiently wide and deep neural networks using the above learning procedure can approximate any learning algorithm. A learning algorithm, for our purposes, maps a set of training examples $\{(\mathbf{x},\mathbf{y})_k\}$ and a test input $\mathbf{x}^\star$ to a predicted output $\hat{\mathbf{y}}^*$.

Theorem. For any learning rule $f_{\text{target}}(\{(\mathbf{x},\mathbf{y})_k\},\mathbf{x}^\star)$, there exists a deep ReLU network feedforward function $\hat{f}(\cdot;\theta)$ and feedback function $g(\mathbf{y})$ such that $\hat{f}(\mathbf{x}^\star;\theta') \approx f_{\text{target}}(\{(\mathbf{x},\mathbf{y})_k\},\mathbf{x}^\star)$. Here

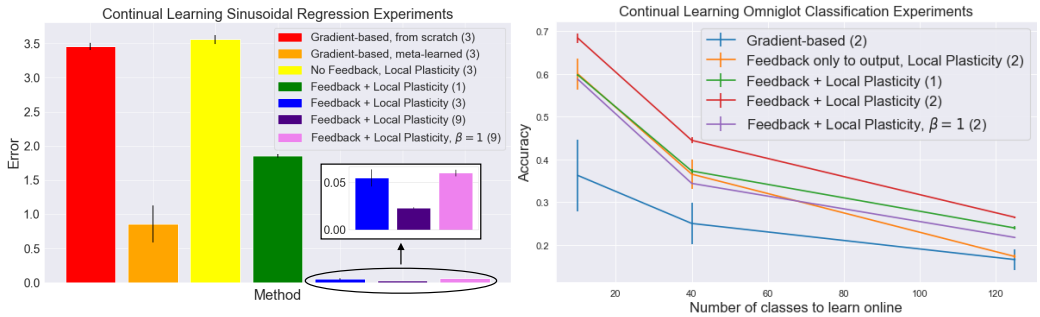

Figure 1: Performance of models on regression and one-shot classification experiments. For regression we show performance after learning all 400 classes. Two networks were meta-trained for each model; error bars indicate max and min performance. Numbers in parentheses show the number of plastic layers, starting with the output.

$\theta' = \theta_k$, $\theta_0 = \theta$, and $\theta_{k+1} = \theta_k + \Delta_{\theta_k}(\mathbf{y}, \mathbf{x})$, where $\Delta_\theta(\mathbf{y}, \mathbf{x})$ is computed following feedback according to a local learning rule at each synapse, either Hebb's rule or Oja's rule.

Proof. Unfortunately the full proof must be omitted from this submission for brevity. It borrows techniques from [5], which proved a similar universality result for gradient-based meta-learning in the non-online batch learning case (in which the entire dataset is available at once). The feedforward network initialization and feedback weights are chosen so that the weight updates losslessly encode the training data in early layers of the network in such a way that it can be processed in an arbitrary way (i.e. to simulate f_{target}) by downstream layers. The proof deviates from [5] in at least one major respect: in the online, continual case, the ability to choose the feedback weights is essential. Indeed, one can indeed show that there are some reasonable f_{target} which gradient-based learning (where feedback weights are tied to feedforward weights) cannot approximate.

4 Experiments

We build off the experimental protocol of [8], evaluating our approach on a regression task and a classification task, all requiring online continual learning. We use the same architectures for the regression and classification tasks as [8] (in short, a nine-layer fully connected network for regression and an six convolutional layer + two fully connected layer for classification).

Incremental Sine Waves: The regression problem is as follows: in each training episode, ten sine functions are sampled randomly, parameterized by amplitude in $[0.1, 5]$ and phase in $[0, \pi]$. In each episode, forty size-32 batches of $(\tilde{x}, y)$ pairs from the first function are presented, then forty from the second, and so on. The input $\tilde{x}$ contains both the function input x and the index k of the function being used. The network is tasked with outputing y for a new $\tilde{x}$. Evaluation occurs on new episodes with sine functions not used in meta-training. Meta-training is performed for 20,000 episodes.

Split-Omniglot: The dataset is split into meta-training and meta-testing classes. During an episode, k examples from one class are presented, followed by k from the next, up to a total of N classes. The model is tested on unseen examples from the classes in the episode. We choose $k = 1$, $N = 20$ to consider the one-shot continual learning case. Feedback to output activations is clamped to their target values, but feedback weights to earlier layers are meta-learned. Evaluation occurs on episodes with classes not used in meta-training. Meta-training is performed for 40,000 episodes.

Experimental Protocol: We evaluate our method in a number of ways: (1) We compare its performance to a gradient-based meta-learner with the same architecture (we also allow its plasticity coefficients to be meta-learned to permit fair comparison). (2) We vary the number of plastic layers in the network. In particular, the case in which only the output weights are plastic serves as a control to indicate whether the feedback propagation to earlier layers is indeed helping learning. (3) We perform ablation experiments to discern the significance of the learned feedback weights. In particular, we experiment with disallowing feedback altogether but maintaining Hebbian plasticity throughout the network, and with clamping all β parameters to 1 to prevent the network from performing Hebbian unsupervised learning along with feedback-modulated updates. (4) We compare using pre or post-feedback presynaptic activations for plasticity updates, corresponding to the two scenarios (with or without dendritically segragated or neuromodulator-carried learning signals) described above.

5 Results

Experimental outcomes are shown in Figure 1. We find that the architecture with meta-learned feedback and local plasticity significantly outperforms an architecturally equivalent gradient-based meta-learner on both the regression and classification tasks. Ablation experiments show that feedback in addition to local plasticity is necessary to enable learning, and that feedback to earlier layers aids performance beyond what can be achieved with feedback only to output layers. Interestingly, we also find (data not shown) that networks invariably learn β values between 0 and 1, and that networks with all β fixed at 1 perform worse. This result indicates that a mix of unsupervised Hebbian and supervised feedback-modulated learning is beneficial. We additionally examined the correlation between weight updates in the feedback network and updates that *would* be computed by gradient descent. We find (data not shown) that the average correlation between the two increases from early to late layers but remains weak (< 0.1) throughout, and is even negative at some stages in the learning process on the regression task. This phenomenon suggests that the meta-learned feedback network learns in a manner that is qualitatively different from gradient-based learners.

6 Discussion

This work demonstrates that meta-learning procedures can optimize for neural networks that learn using local plasticity rules and feedback connections. Several follow-up directions could be pursued. First, meta-learning of this kind is computationally expensive, as the meta-learner must backpropagate through the network's entire training procedure. In order to scale this approach, it will be important to find ways to meta-train networks that generalize to longer lifetimes than were used during meta-training, or to explore alternatives to backprop-based meta-training (e.g. evolutionary algorithms, or the recently developed implicit gradient-based meta learning algorithm [17]).

Future work could also increase the biological plausibility of the method. For instance, in the present implementation the feedforward and feedback + update passes occur sequentially. However, a natural extension would enable them to run in parallel. This requires ensuring (through appropriate meta-learning and/or a segregated dendrites model [6]) that feedforward and feedback information do not interfere destructively. Third, the meta-learning procedure in this work optimizes for a precise feedforward and feedback weight initialization. Optimizing instead for a distribution of weight initializations or connectivity patterns would better reflect the stochasticity in synapse development. Another direction is to apply meta-learning to understand biological learning systems (see [9] for an example of such an effort). Well-constrained biological learning models meta-optimized in this manner might show emergence of learning circuits used in biology and even suggest new ones.

References

[1] J. Ba, G. E. Hinton, V. Mnih, J. Z. Leibo, and C. Ionescu. Using fast weights to attend to the recent past. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4331–4339. 2016.

[2] Sergey Bartunov, Adam Santoro, Blake Richards, Luke Marris, Geoffrey E Hinton, and Timothy Lillicrap. Assessing the scalability of biologically-motivated deep learning algorithms and architectures. In *Advances in Neural Information Processing Systems*, pages 9368–9378, 2018.

[3] Yoshua Bengio. How auto-encoders could provide credit assignment in deep networks via target propagation. *arXiv preprint arXiv:1407.7906*, 2014.

[4] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In International Conference on Machine Learning, 2017.

[5] Chelsea Finn and Sergey Levine. Meta-learning and universality: Deep representations and gradient descent can approximate any learning algorithm. *arXiv preprint arXiv:1710.11622*, 2017.

[6] Jordan Guerguiev, Timothy P Lillicrap, and Blake A Richards. Towards deep learning with segregated dendrites. *ELife*, 6:e22901, 2017.

[7] Toshihide Hige, Yoshinori Aso, Mehrab N Modi, Gerald M Rubin, and Glenn C Turner. Heterosynaptic plasticity underlies aversive olfactory learning in drosophila. *Neuron*, 88(5):985–998, 2015.

[8] Khurram Javed and Martha White. Meta-learning representations for continual learning. *Advances in Neural Information Processing Systems*, 2019.

[9] Linnie Jiang and Ashok Litwin-Kumar. Models of heterogeneous dopamine signaling in an insect learning and memory center. *bioRxiv*, page 737064, 2019.

[10] Alexander JE Kell, Daniel LK Yamins, Erica N Shook, Sam V Norman-Haignere, and Josh H McDermott. A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy. *Neuron*, 98(3):630–644, 2018.

[11] Benjamin James Lansdell, Prashanth Prakash, and Konrad Paul Kording. Learning to solve the credit assignment problem. *arXiv preprint arXiv:1906.00889*, 2019.

[12] Qianli Liao, Joel Z Leibo, and Tomaso Poggio. How important is weight symmetry in backpropagation? In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[13] 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.

[14] Thomas Miconi, Jeff Clune, and Kenneth O Stanley. Differentiable plasticity: training plastic neural networks with backpropagation. *arXiv preprint arXiv:1804.02464*, 2018.

[15] Arild Nøkland. Direct feedback alignment provides learning in deep neural networks. In *Advances in neural information processing systems*, pages 1037–1045, 2016.

[16] Erkki Oja. Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15(3):267–273, 1982.

[17] Aravind Rajeswaran, Chelsea Finn, Sham Kakade, and Sergey Levine. Meta-learning with implicit gradients. *arXiv preprint arXiv:1909.04630*, 2019.

[18] Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014.