Error Forcing in Recurrent Neural Networks

A. Erdem Sağtekin (aes10217@nyu.edu)
Center for Data Science, NYU, 60 5th Ave
New York, NY 10011, USA

Colin Bredenberg (colin.bredenberg@mila.quebec)
Mila - Quebec AI Institute, 6666 Rue Saint Urbain
Montréal, QC H2S 3H1

Cristina Savin (csavin@nyu.edu)
Center for Neural Science and Center for Data Science, NYU,
60 5th Ave New York, NY 10011, USA

Abstract

How can feedback improve learning outcomes? Traditionally, feedback signals are thought to directly drive parameter (synaptic) changes. Yet, biophysically, the same signals also affect the activity of neurons. Here, we use this observation to develop a new algorithm termed error forcing (EF) for learning in recurrent neural networks, where feedback influences both synaptic plasticity and the network state. We geometrically contrast our approach with the established teacher forcing framework, and further provide an interpretation of its function from a Bayesian standpoint. EF learning outperforms traditional approaches in scenarios with temporally sparse feedback when the output is weakly constrained by the task. These benefits generalize across tasks and are maintained in a biologically-constrained approximation of error forcing.

Keywords: recurrent neural networks, bio-plausible learning

Introduction

Feedback signals play a dual role in learning: while conventional models use them primarily to induce synaptic changes (Werbos, 1990), they can also guide neural activity toward an optimal manifold for task performance, which in turn guides local synaptic plasticity to update the weights (Manchev & Spratling, 2020). Building on previous control-based methods such as teacher forcing (TF) (Williams & Zipser, 1989), and inspired by effects of feedback signals modulating synaptic weights *and* neuronal activities in the brain, here we propose *error forcing (EF)*, a simple yet effective learning mechanism for recurrent neural networks (RNNs).

Background

Consider the general formulation of a discrete-time RNN with a linear decoder:

$$\mathbf{r}_t = \mathbf{F}_\theta(\mathbf{r}_{t-1}, \mathbf{x}_t), \quad \mathbf{y}_t = \mathbf{W}_\phi \mathbf{r}_t, \tag{1}$$

where $\mathbf{r}_t \in \mathbb{R}^N$ is the hidden state, $\mathbf{x}_t \in \mathbb{R}^{N_x}$ the input, and $\mathbf{y}_t \in \mathbb{R}^{N_y}$ the output. Parameters θ and ϕ define the recurrent dynamics and decoder, respectively. Minimizing the error between network outputs and target $\mathbf{y}_t^*$ requires computing the derivative of the loss with respect to network parameters θ. This is typically achieved by backpropagation through time (BPTT), or online approximations of it, aimed to make learning more computationally efficient or more biologically plausible (Marschall, Cho, & Savin, 2020). BPTT suffers from well-known issues such as vanishing and exploding gradients (Pascanu, Mikolov, & Bengio, 2013), which motivate the development of alternative approaches such as teacher forcing (Doya, 1992).

In its generalized version (Hess, Monfared, Brenner, & Durstewitz, 2023), teacher forcing (TF) pushes neuron activity towards states that would correspond to correct outputs:

$$\tilde{\mathbf{r}}_t := (1-\alpha)\mathbf{r}_t + \alpha\bar{\mathbf{r}}_t, \tag{2}$$

$$\mathbf{r}_t = F_\theta(\tilde{\mathbf{r}}_{t-1}, \mathbf{x}_t), \tag{3}$$

where $\mathbf{r}_t$ denotes natural RNN state, $\bar{\mathbf{r}}_t$ is the target (teacher) state, and their linear interpolation leads to the forced dynamics $\tilde{\mathbf{r}}_t$, with $0 \leq \alpha \leq 1$. When $\alpha = 0$, the method reduces to BPTT. When states are partially forced, the computational graph for the network dynamics changes, which in turn changes the BPTT computations; well-behaved gradients can thus be ensured by a judicious selection of α.

In most practical scenarios, the dimensionality of the output is smaller than network size, which implies a manifold of possible neural states with zero error. To pick one specific target on that manifold, TF uses a minimum-norm mapping from the low-dimensional output space to activity space, given by the decoder pseudoinverse, $\bar{\mathbf{r}}_t^{\text{min}} = \mathbf{W}_\phi^+ \mathbf{y}_t^*$. EF proposes an alternative way to choose this target, as the closest point on the manifold from the current network state.

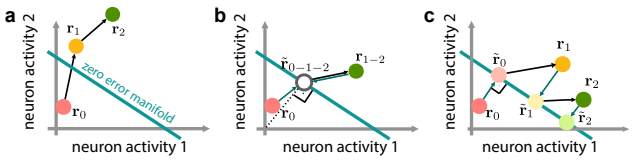

Figure 1: Geometric explanation of differences between no forcing (a), teacher forcing (b), and error forcing (c); $\alpha = 1$.

Approach

Geometric perspective: It is not *a priori* clear why one should use the minimum-norm solution, given the existence of infinitely many alternatives. In fact, TF quenches network network variability outside of the decoder manifold and can induce large perturbations to the network state, both of which potentially disrupt learning. To understand why, consider a toy RNN example with 2 hidden neurons and a 1-dimensional output. For visualization, assume constant target states $\mathbf{y}^*$ (Figure 1, cyan line). When states are not forced, the RNN runs freely (Figure 1a). With full forcing ($\alpha = 1$), using the minimum-norm solution (white circle), the RNN is driven to the teacher state, and the next step follows Eq. 3. Since the output is constant, the same teacher state is used at each step, causing the RNN to repeat this pattern (Figure 1b), limiting the exploration of the neural space during learning. In

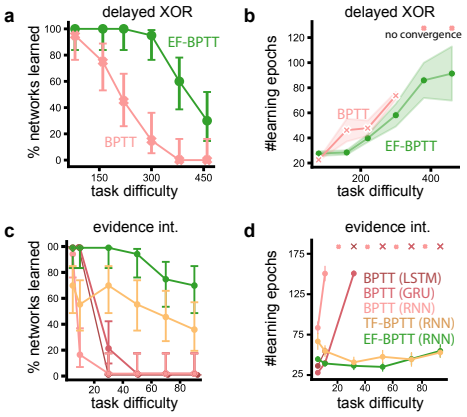

Figure 2: Comparison of task performance (a,c) and learning speed (b,d) for EF-BPTT relative to other approaches for delayed XOR and cued response evidence integration; $\alpha = 0.5$.

contrast, EF simply selects the teacher state such that $\bar{\mathbf{r}}_t$ is the orthogonal projection of $\mathbf{r}_t$ onto the manifold of possible optimal responses (Figure 1c):

$$\tilde{\mathbf{r}}_t = \mathbf{r}_t + \alpha \mathbf{W}_\phi^+ \mathbf{e}_t, \tag{4}$$

where $\mathbf{e}_t = \mathbf{y}_t^* - \mathbf{y}_t$. This lets the RNN explore the N dimensional space. We call this mechanism EF-BPPT, when BPTT is used for synaptic updates.

Bayesian perspective: If we replace the deterministic RNN dynamics (Eq.1) with a Gaussian state-space model (by the addition of gaussian noise with variance $\sigma^2\eta_t$ for latents and $\sigma_o^2\eta_o$ for output), we are also able to show that greedily inferring the optimal state space correction $\tilde{\eta}_{t|t}$ given target output $\mathbf{y}_t$ via *maximum a posteriori* filtering produces nearly identical state space dynamics as the deterministic case, while still allowing for BPTT learning. The stochastic EF dynamics are given by:

$$\tilde{\mathbf{r}}_{t|t} = \mathbf{r}_{t|t-1} + \mathbf{W}_\phi^\top \left(\mathbf{W}_\phi \mathbf{W}_\phi^\top + \frac{\sigma_o^2}{\sigma^2} \mathbf{I} \right)^{-1} \mathbf{e}_t \tag{5}$$

This perspective connects EF to the Extended Kalman Filter, and allows for theoretically grounded stochastic EF training.

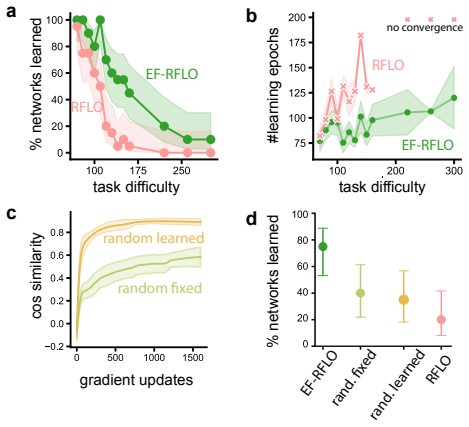

Figure 3: Task performance and learning speed for variants of EF-RFLO for delayed XOR.

Numerical Results

EF-BPTT: We compared EF-BPTT with standard BPTT and TF-BPTT on two tasks which require filtering past inputs and mapping the result into a one-dimensional output at particular moments in time— delayed XOR and a version of evidence accumulation (Liu, Smith, Mihalas, Shea-Brown, & Sümbül, 2021). In both tasks, the target reporting was sparse in time (by using an explicit report cue as a network input, and relatively long delays), which makes learning difficult. For each task and learning mechanism, we further varied task difficulty (by increasing the delay) and measured the fraction of networks (out of 20 runs) that successfully learn to perform the task with no errors (Figure 2a,c) and the number of epochs required for convergence (Figure 2b,d), with the optimal α determined via grid search. Overall, EF-BPTT outperformed the other methods by converging faster and achieving higher success rates. Vanilla RNNs with EF-BPTT also outperformed more complex architectures engineered in service of well-behaved gradients (GRUs and LSTMs, Figure 2c,d) and trained with BPTT, suggesting that EF may be useful for reverse engineering neural dynamics supporting cognitive tasks with long temporal horizons.

Bio-plausible EF: To test EF in a more biologically realistic setting, we replaced the BPTT parameter updates with random feedback local online (RFLO) learning (Murray, 2019), a biologically plausible online approximation of BPTT (same general experimental setup). Error forcing improved the performance and convergence of local learning rules (Figure 3a,b). Beyond the plausibility of parameter updates, EF itself requires knowledge of the network readout for the forcing matrix (Eq.4). To enforce locality, we replaced this idealized solution with a random forcing matrix —either fixed or learnable. With fixed projections, the decoder aligned with the

feedback synapses; when learned, both the projection and decoder aligned, resulting in higher similarity (Figure 3c) and still retain improved task performance, suggesting that EF biological plausibility can be achieved with minimal cost in terms of the quality of learning.

Discussion

In this work, we introduced *error forcing* as a geometrically and probabilistically interpretable method to stabilize and improve learning in RNNs. We demonstrated its benefits in tasks involving long-time dependencies and showed that bioplausible approximations of EF benefit local learning rules, suggesting that the brain could leverage synergistic state-weight updates to both learn and improve instantaneous task performance.

Acknowledgments

This work is supported by the National Science Foundation under NSF Award No. 1922658 and a Google faculty award.

References

Doya, K. (1992). Bifurcations in the learning of recurrent neural networks. In *Proceedings of the 1992 ieee international symposium on circuits and systems (iscas)* (pp. 2777–2780). IEEE.

Hess, F., Monfared, Z., Brenner, M., & Durstewitz, D. (2023). Generalized teacher forcing for learning chaotic dynamics. *arXiv preprint arXiv:2306.04406*.

Liu, Y. H., Smith, S., Mihalas, S., Shea-Brown, E., & Sümbül, U. (2021). Cell-type–specific neuromodulation guides synaptic credit assignment in a spiking neural network. *Proceedings of the National Academy of Sciences*, *118*(51), e2111821118. doi: 10.1073/pnas.2111821118

Manchev, N., & Spratling, M. (2020). Target propagation in recurrent neural networks. *Journal of Machine Learning Research*, *21*(7), 1–33.

Marschall, O., Cho, K., & Savin, C. (2020). A unified framework of online learning algorithms for training recurrent neural networks. *Journal of Machine Learning Research*, *21*(135), 1–34.

Murray, J. M. (2019). Local online learning in recurrent networks with random feedback. *eLife*, *8*, e43299.

Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In *International conference on machine learning* (pp. 1310–1318).

Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, *78*(10), 1550–1560.

Williams, R. J., & Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, *1*(2), 270–280.