A Cortical Motif for Learning with Vectorized Errors

Qianli Liao[*1,5], Liu Ziyin[*3,4], Yulu Gan[*1,2], Brian Cheung[1,2], Mark Harnett[5,6], and Tomaso Poggio[1,2,5,6]

[1]*Center for Brains, Minds, and Machines, MIT*
[2]*CSAIL, MIT*
[3]*Research Laboratory of Electronics, MIT*
[4]*Physics & Informatics Laboratories, NTT Research*
[5]*McGovern Institute, MIT*
[6]*Department of Brain and Cognitive Sciences, MIT*

February 25, 2026

Abstract

Recent experiments in mouse cortex suggest that instructive/error information is **vectorized**—distributed across neuron-specific signals in distinct neuronal populations—while converging evidence indicates that **both ascending and descending pathways** exhibit experience-dependent synaptic plasticity. These observations motivate learning architectures in which error is carried by a dedicated feedback stream and synapses across streams co-adapt. We introduce **Self-Assembling Motif (SAM)**, a minimal two-stream circuit motif of four classes of plastic connections trained by local heterosynaptic rules, without requiring matched forward–feedback representations. Starting from random connectivity, SAM self-organizes to implement **SGD-like learning** without weight transport or explicit derivatives, extending a line of biologically motivated alternatives to backpropagation. We prove that, under mild stationarity conditions, SAM approximates stochastic gradient descent with a learned positive semidefinite preconditioner, and we show competitive performance across vision benchmarks. SAM makes falsifiable predictions about the relationship between cell-type–specific error signals and bidirectional synaptic plasticity in cortex.

1 Introduction

How cortical circuits assign credit across many synapses remains a central open problem in neuroscience. Gradient-based learning in artificial neural networks provides an existence

proof that neuron-specific error signals can, in principle, support efficient multi-layer learning, but standard backpropagation relies on globally coordinated mechanisms across layers or areas whose cortical implementation is unclear [17, 25].

Two recent experimental trends sharpen the biological constraints on any cortical credit-assignment theory. First, instructive or error-related information in mouse cortex appears to be *vectorized*: learning is guided by structured, neuron-resolved signals carried by specific circuit elements rather than by a single broadcast scalar [1, 7, 8]. Such findings motivate architectures in which error information is routed through dedicated pathways that can deliver heterogeneous, unit-specific teaching signals.

Second, converging evidence indicates that synapses in both ascending and descending pathways are plastic and experience-dependent, including synapses implicated in top–down and cross-stream interactions [1, 4, 5, 7, 9, 19]. In particular, there is substantial evidence that spines in layer 1 (where many "feedback" inputs make their synapses) on both L5 and L2/3 pyramidal neurons are reasonably plastic across the adult mouse cortex, and that they show enhanced plasticity during learning paradigms [1, 7, 10, 13, 18, 24, 27, 28]. This suggests that effective learning circuits should not treat feedback connections as fixed scaffolding and also not as matched one-to-one with the forward connections; instead, forward representations and feedback/teaching routes are likely to co-adapt during learning.

Despite these constraints, many backprop-inspired circuit proposals assume correspondence between forward and feedback representations—often requiring equal dimensionality, or an implicit one-to-one pairing—to deliver neuron-specific error components to appropriate targets [3, 11, 15, 16, 21, 22]. Whether such precise matching can be established and maintained across development and learning seems still uncertain at best, particularly in circuits where feedback pathways differ in size, cell types, and laminar organization [5, 19].

Here we introduce **Self-Assembling Motif (SAM)**, a minimal two-stream motif with four classes of plastic connections trained by local heterosynaptic rules. SAM represents error as a vector propagated through a feedback stream while removing the requirement for matched forward–feedback representations: the forward and feedback pathways may have different widths, and inter-stream mappings are learned through plasticity from initially random connectivity. Starting from random connectivity, SAM self-organizes to support SGD-like learning using only local heterosynaptic plasticity, without weight transport or explicit derivatives, in the spirit of prior biologically motivated proposals [2, 3, 11, 15, 16, 21, 22]. We prove that, under mild stationarity conditions, SAM's updates approximate stochastic gradient descent via a learned positive semidefinite preconditioner. Empirically, SAM performs competitively on standard vision benchmarks [6, 12, 14, 20, 26, 29, 30] and yields falsifiable predictions linking cell-type–specific teaching signals to bidirectional synaptic plasticity across cortical streams [5, 19].

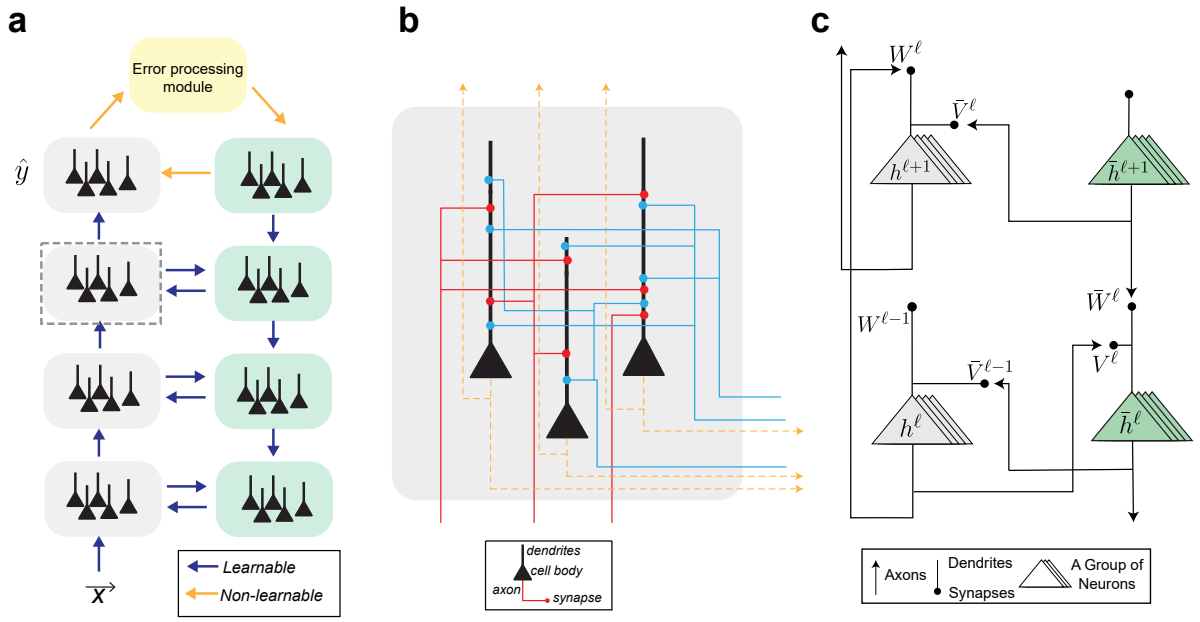

Figure 1: **A synaptic motif for interactions between ascending and descending information streams. (a)** Schematic of the upstream-downstream architecture. The upstream consists of a standard fully-connected neural network with multiple layers, roughly corresponding to a multi-region cortical processing pathway (e.g. V1-V2-V4-IT) [19]. The vector output of the upstream network goes to an error processing module (potentially corresponding to frontal cortices). This module computes a local error signal. This *vector* error signal is sent to the feedback (downstream) pathway, which processes information layer by layer downwards. The orange arrows represent non-learnable (identity) connections. Blue arrows represent learnable connections, each parameterized by a fully-connected weight matrix. **(b)** A detailed view of the smallest unit of the connection motif in panel a, indicated by grey dashed box. **(c)** A mathematical description of the unit in panel b. Each arrow represents full connections between two groups of neurons, parameterized by a weight matrix. Every h is a vector of activations. The inter-stream matrix V allows upstream and downstream layers to have different widths.

2 Results

2.1 A minimal two-stream motif with four plastic pathways

We consider an upstream (feedforward) stream and a downstream (feedback) stream that interact through four classes of synaptic pathways: upstream-to-upstream ($u \to u$), downstream-to-downstream ($d \to d$), upstream-to-downstream ($u \to d$), and downstream-to-upstream ($d \to u$) (Fig. 1), consistent with known feedforward/feedback organization in sensory cortex [5, 19]. Learning is driven by a local heterosynaptic rule of the form

$$\Delta W \;\propto\; S_{\text{post}}\, S_{\text{pre}}^{\top}, \tag{1}$$

where the postsynaptic factor is supplied by feedback/instructive inputs rather than by self-activation alone, consistent with the broader biological motivation for multi-factor plasticity

and feedback-gated learning [17, 25]. The full update rules for the four pathways are given in Appendix A (and summarized in Methods). The motif of the figure is the simplest possible. In reality, interneurons are also likely to be involved. Mechanisms such as burst-dependent synaptic plasticity previously proposed ([23]) may also be involved increasing the effectiveness of SAM.

2.2 Vectorized error is represented by a self-learning feedback stream

In SAM, the instructive signal is a *vector* propagated through the downstream stream, initialized at the output as $\bar{h}^L = \epsilon(\hat{y})$ and transmitted to lower levels to gate synaptic updates. This architecture supports neuron-resolved vector signals without requiring matched forward–feedback representations, aligning with the idea that biologically plausible learning may rely on structured feedback pathways rather than exact weight transport [16, 17, 25].

To test whether vectorization is functionally necessary, we compared SAM to an ablated variant in which the instructive signal is collapsed to a scalar modulatory factor shared across neurons within a layer (Methods). Collapsing the vector signal substantially degrades learning, consistent with the idea that neuron-specific error components provide essential credit-assignment structure.

2.3 Bidirectional plasticity enables self-assembly and robust learning

Because SAM does not assume pre-established correspondences between streams, effective learning requires that cross-stream mappings and feedback routes co-adapt with forward representations. This is also compatible with increasing evidence that synapses implicated in top-down or feedback circuitry can be experience-dependent [1, 4, 7, 9]. We therefore tested the necessity of plasticity in each pathway by selectively freezing subsets of synapses (Fig. 2).

We observe three robust trends. First, updating only the feedforward pathway W yields limited improvement and tracks a linear baseline, indicating that cross-stream plasticity is needed to learn nonlinear relations. Second, among inter-stream connections, making the downstream-to-upstream pathway $\bar{V}$ plastic provides the largest single gain, suggesting that learning benefits strongly from adaptive routing of vectorized teaching signals into upstream populations. Third, allowing all four classes of connections $(W, \bar{W}, V, \bar{V})$ to be plastic recovers the strongest performance and most reliably approaches the behavior of SGD-trained networks [3, 11, 16] (Table 1; Fig. 2). Together, these results mirror emerging biological evidence that synapses mediating ascending, descending, and cross-stream interactions exhibit experience-dependent plasticity [4, 5, 9, 19].

3 Theory

3.1 SAM approximates preconditioned stochastic gradient descent

SAM updates the feedforward weights using a heterosynaptic, three-factor rule in which the presynaptic factor is the upstream activity and the postsynaptic factor is supplied by a feedback/instructive signal, following the general motivation for alternatives to backprop that avoid exact weight transport [3, 11, 15–17, 25]. Writing the upstream activations at layer ℓ as h^ℓ and the feedback-stream activities as $\bar{h}^\ell$, the feedforward synapses obey an update of the generic form

$$\Delta W^\ell \;=\; \eta\, \bar{h}^\ell\, (h^{\ell-1})^\top \;-\; \eta\gamma\, W^\ell, \tag{2}$$

where η is a learning rate and γ is a weight-decay coefficient.

Key idea. If the feedback stream delivers an *approximately linear* transformation of the true backprop error at each layer, then the local rule in Eq. (2) becomes a *preconditioned* version of the SGD gradient [17, 25]. In SAM, this linear transformation is itself learned through plastic inter-stream connectivity, yielding an adaptive, data-dependent preconditioner.

Theorem 1 (Informal: preconditioned-SGD equivalence)**.** *Assume (i) the inter-stream mappings vary slowly relative to the feedforward weights (a stationarity/separation-of-timescales condition), (ii) the learned feedback-to-upstream map has full row rank so that error components span the relevant subspace, and (iii) the feedback-stream activity $\bar{h}^\ell$ is a stable linear transform of the layerwise backprop error signal δ^ℓ up to higher-order terms. Then the SAM update on the feedforward weights can be written as*

$$\Delta_{\mathrm{SAM}} W^\ell \;=\; H^\ell\, \Delta_{\mathrm{SGD}} W^\ell \;-\; \eta\gamma\, W^\ell \;+\; o(1). \tag{3}$$

where H^ℓ is a learned positive semidefinite matrix (a preconditioner) determined by inter-stream connectivity.

Form of the preconditioner. In the simplest (and empirically relevant) regime, the preconditioner takes the form

$$H^\ell \;\approx\; \bar{V}^\ell (\bar{V}^\ell)^\top, \tag{4}$$

where $\bar{V}^\ell$ denotes the learned mapping that routes feedback-stream signals into the upstream population at layer ℓ.[1] Equation (4) makes two points explicit: (i) $H^\ell \succeq 0$ by construction, and (ii) the "effective learning rate structure" for W^ℓ is set by cross-stream plasticity.

Sketch of derivation. Under the above assumptions, the feedback factor can be expressed as $\bar{h}^\ell \approx \bar{V}^\ell \delta^\ell$ (up to higher-order terms). Substituting into Eq. (2) gives

$$\Delta W^\ell \;\approx\; \eta\, (\bar{V}^\ell \delta^\ell)\, (h^{\ell-1})^\top - \eta\gamma W^\ell.$$

[1] The exact definition of $\bar{V}^\ell$ and the conditions under which Eq. (4) holds are given in Appendix A.

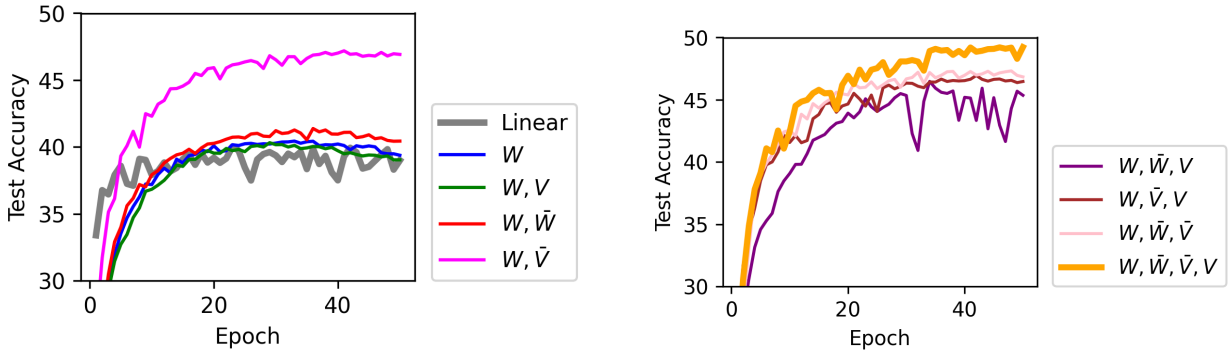

Figure 2: **Ablation studies.** Ablation study on roles of $\bar{V}$, $\bar{W}$ and V on CIFAR-10 [12]. Here, we make a subset of all interconnections that are not trainable. The legend labels indicate trainable components. "Linear" refers to a linear network trained with SGD. We see that (1) making $\bar{V}$ plastic is more important than making other connections plastic, and (2) making everything plastic significantly improves the performance further.

Since the SGD gradient has the standard form $\Delta_{\text{SGD}} W^\ell \propto \delta^\ell (h^{\ell-1})^\top$, we obtain Eq. (3) with a learned left-multiplication by H^ℓ induced by inter-stream routing. The $o(1)$ term collects the approximation error from imperfect stationarity and residual nonlinear coupling between streams. A limitation of the present analysis is that the convergence result relies on stationarity assumptions for inter-stream mappings; characterizing the full coupled dynamics in biophysical terms remains a goal for future work.

Mechanistic interpretation. The learned matrix H^ℓ acts as a data- and time-dependent preconditioner: SAM does not merely approximate SGD, but does so with an *adaptive geometry* set by experience-dependent inter-stream synapses. Biologically, this links improvements in credit assignment to measurable learning-dependent changes in cross-stream connectivity, and predicts that manipulations that disrupt $\bar{V}^\ell$ plasticity should specifically reduce the "conditioning" benefits of learning even when feedforward plasticity remains intact [1, 4, 7, 9]. Consistent with this view, $\bar{V}^\ell (\bar{V}^\ell)^\top$ develops structured correlations and effectively low-rank spectra during learning (Fig. S1a).

A full statement, proof, and the precise assumptions are provided in Appendices A and D.

4 Methods

Model architecture. We use a two-stream network with an upstream (feedforward) pathway and a downstream (feedback) pathway connected by four classes of synapses ($u \to u$, $d \to d$, $u \to d$, $d \to u$), consistent with canonical feedforward/feedback organization in cortex [5, 19]. The downstream pathway may be wider than the upstream pathway, consistent with anatomical asymmetries observed in some systems [5].

Table 1: **Performance of the proposed method compared with SGD and biologically plausible algorithm baselines.** Bold denotes the best performing algorithm, and italics denote the best runner-up algorithm. SGD and the SAM are comparable, while significantly outperforming existing biologically plausible learning algorithms. Baselines include Feedback Alignment [16], Weight Mirroring [3], and the Kolen–Pollack algorithm [11]. For each experiment, we conduct five runs.

	CIFAR10	MNIST	Fashion MNIST	Chest MNIST	Path MNIST	SVHN	STL10
SGD	*50.36 ± 0.16*	*98.44 ± 0.09*	*96.80 ± 0.05*	**73.26 ± 0.14**	**70.11 ± 0.10**	*78.20 ± 0.13*	*41.00 ± 0.07*
FA	47.85 ± 0.20	97.80 ± 0.19	95.75 ± 0.20	70.44 ± 0.27	69.90 ± 0.10	78.03 ± 0.15	39.29 ± 0.23
WM	49.35 ± 0.13	98.40 ± 0.23	96.00 ± 0.25	70.14 ± 0.08	70.00 ± 0.15	78.08 ± 0.15	39.48 ± 0.19
KP Algorithm	48.16 ± 0.09	98.45 ± 0.09	96.13 ± 0.07	71.03 ± 0.10	69.51 ± 0.05	78.20 ± 0.05	40.32 ± 0.11
SAM (ours)	**51.91 ± 0.21**	**98.85 ± 0.10**	**97.23 ± 0.08**	**73.21 ± 0.11**	**70.23 ± 0.15**	**78.50 ± 0.07**	**41.50 ± 0.10**

Learning rule and ablations. All synapses are updated by local heterosynaptic rules (Appendix A). For the *scalar-error ablation*, the layerwise feedback vector is replaced by a shared scalar modulatory factor (e.g., the mean or norm of the feedback vector within the layer), holding all other components fixed.

For *plasticity ablations*, we selectively freeze subsets of the four connection classes while keeping W plastic.

Datasets and training protocol. We evaluate on standard vision benchmarks and report accuracy averaged across multiple runs (Table 1); datasets include CIFAR-10 [12], MNIST [14], Fashion-MNIST [26], SVHN [20], STL-10 [6], and MedMNIST subsets such as ChestMNIST/PathMNIST [29, 30]. Full hyperparameters are in Supplementary Methods.

5 Discussion

SAM makes three testable predictions: (i) instructive/error signals should be vector-valued at the population level and causally gate plasticity at targeted synapses [1, 7]; (ii) synapses mediating cross-stream interactions should exhibit learning-linked plasticity in both directions [4, 5, 9]; and (iii) efficient learning should not require layerwise one-to-one matching between forward and feedback representations, but should instead be compatible with mismatched pathway widths and learned inter-stream mappings [5, 19]. Across diverse vision benchmarks, SAM matches or exceeds standard biologically motivated baselines and is competitive with SGD (Table 1), motivating its consideration as a cortical learning motif [17, 25]. These predictions can be operationalized as a circuit-level "compatibility test" (Table 2). Many backprop-inspired models effectively assume a layerwise correspondence be-

Table 2: **Circuit-level falsifiable criteria for cortical credit assignment.** We summarize four pathway types between an upstream (u) and downstream (d) stream: $u \to u$, $d \to d$, $u \to d$, $d \to u$. A learning proposal specifies both (i) whether each pathway exists at appreciable strength (connectivity) and (ii) whether it is experience-dependent (plasticity). SAM predicts that all four pathways are present and plastic; several prior schemes correspond to strict subsets [3, 11, 16, 17].

	Connectivity				Plasticity			
	$u \to u$	$d \to d$	$u \to d$	$d \to u$	$u \to u$	$d \to d$	$u \to d$	$d \to u$
SGD	✓	✗	✗	✗	✓	✗	✗	✗
Feedback Alignment [16]	✓	✓	✗	✗	✓	✗	✗	✗
Weight Mirroring [3]	✓	✓	✗	✗	✓	✓	✗	✗
SAM	✓	✓	✓	✓	✓	✓	✓	✓

tween forward and feedback representations, which typically implies restricted inter-stream connectivity (often $u \leftrightarrow d$ mappings fixed or absent) and/or restricted plasticity in feedback pathways [16, 21, 22]. In contrast, SAM predicts that all four pathway types are both present and experience-dependent. This yields an eight-criterion experimental program—four connectivity tests and four plasticity tests—that can distinguish SAM-like learning from feedback-alignment-like alternatives in identified circuits [16, 17, 25].

Experimental identification and distinction from feedback-alignment family. A practical way to distinguish SAM-like circuits from feedback-alignment-like mechanisms is to test, separately, (a) the existence of each of the four pathway classes and (b) whether synapses in each class are plastic over learning [5, 19]. In feedback alignment, neuron-specific teaching depends on a fixed (or non-plastic) feedback scaffold and typically presumes a one-to-one correspondence between upstream and downstream representations [16, 21]. In SAM, by contrast, the inter-stream mappings are learnable: the circuit predicts measurable, learning-dependent changes in cross-stream synapses ($u \to d$ and $d \to u$) that accompany improvements in behavioral performance or task accuracy [1, 4, 7, 9].

References

[1] E. Abs, R. B. Poorthuis, D. Apelblat, K. Muhammad, M. B. Pardi, L. Enke, D. Kushinsky, D. L. Pu, M. F. Eizinger, K.-K. Conzelmann, I. Spiegel, and J. J. Letzkus. Learning-related plasticity in dendrite-targeting layer 1 interneurons. *Science*, 361 (6406):eaat4799, 2018. doi: 10.1126/science.aat4799.

[2] Nasir Ahmad, Marcel A. J. van Gerven, and Luca Ambrogioni. Gait-prop: A biologically plausible learning rule derived from backpropagation of error. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/7ba0691b7777b6581397456412a41390-Abstract.html.

[3] Mohamed Akrout, Collin Wilson, Peter Conway Humphreys, Timothy P. Lillicrap, and Douglas B. Tweed. Deep learning without weight transport. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[4] Victoria M. Bajo, Fernando R. Nodal, David R. Moore, and Andrew J. King. The descending corticocollicular pathway mediates learning-induced auditory plasticity. *Nature Neuroscience*, 13(2):253–260, 2010.

[5] Farran Briggs. Role of feedback connections in central visual processing. *Annual Review of Vision Science*, 6(1):313–334, 2020.

[6] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

[7] Guy Doron, Jiyun N. Shin, Naoya Takahashi, Moritz Drüke, Christina Bocklisch, Salina Skenderi, Lisa de Mont, Maria Toumazou, Joshua Ledderose, Michael Brecht, Richard Naud, and Matthew E. Larkum. Perirhinal input to neocortical layer 1 controls learning. *Science*, 370(6523):eaaz3136, 2020. doi: 10.1126/science.aaz3136.

[8] Valerio Francioni, Vincent D. Tang, Enrique H. S. Toloza, Norma J. Brown, and Mark T. Harnett. Vectorized instructive signals in cortical dendrites during a brain-computer interface task. *bioRxiv*, January 2025. doi: 10.1101/2023.11.03.565534. URL https://www.biorxiv.org/content/10.1101/2023.11.03.565534v2. Version 2 posted January 13, 2025.

[9] Min Fu and Yi Zuo. Experience-dependent structural plasticity in the cortex. *Trends in Neurosciences*, 34(4):177–187, 2011. doi: 10.1016/j.tins.2011.02.001.

[10] Anthony J. G. D. Holtmaat, Joshua T. Trachtenberg, Linda Wilbrecht, Gordon M. G. Shepherd, Xiao-Qing Zhang, Graham W. Knott, and Karel Svoboda. Transient and persistent dendritic spines in the neocortex in vivo. *Nature*, 435(7045):1020–1025, 2005. doi: 10.1038/nature03720.

[11] John Kolen and Jordan Pollack. Back-propagation without weight transport. In *Proceedings of the IEEE International Conference on Neural Networks (ICNN)*, volume 3, pages 1375–1380. IEEE, 1994.

[12] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

[13] Matthew E. Larkum. A cellular mechanism for cortical associations: an organizing principle for the cerebral cortex. *Trends in Neurosciences*, 36(3):141–151, 2013. doi: 10.1016/j.tins.2012.11.006.

[14] Yann LeCun. The MNIST database of handwritten digits, 1998. Available at http://yann.lecun.com/exdb/mnist/.

[15] Qianli Liao, Joel Leibo, and Tomaso Poggio. How important is weight symmetry in backpropagation?, 2015.

[16] 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(1):13276, 2016.

[17] Timothy P. Lillicrap, Adam Santoro, Luke Marris, Colin J. Akerman, and Geoffrey Hinton. Backpropagation and the brain. *Nature Reviews Neuroscience*, 21(6):335–346, 2020.

[18] Hiroshi Makino and Takaki Komiyama. Learning enhances the relative impact of top-down processing in the visual cortex. *Nature Neuroscience*, 18(8):1116–1122, 2015. doi: 10.1038/nn.4061.

[19] Nikola T. Markov, Mária Ercsey-Ravasz, David C. Van Essen, Kenneth Knoblauch, Zoltán Toroczkai, and Henry Kennedy. Anatomy of hierarchy: feedforward and feedback pathways in macaque visual cortex. *Journal of Comparative Neurology*, 522(1):225–259, 2014.

[20] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[21] Arild Nøkland. Direct feedback alignment provides learning in deep neural networks. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

[22] Arild Nøkland and Lars H. Eidnes. Training neural networks with local error signals. In *Proceedings of the International Conference on Machine Learning*, pages 4839–4850. PMLR, 2019.

[23] Alexandre Payeur, Jordan Guerguiev, Friedemann Zenke, Blake A. Richards, and Richard Naud. Burst-dependent synaptic plasticity can coordinate learning in hierarchical circuits. *Nature Neuroscience*, 24:1010–1019, 2021. doi: 10.1038/s41593-021-00870-x.

[24] S. Tattikota, K. K. A. Cho, Y. Liu, W. Hu, V. Barrera, J. L. Stein, and W.-B. Gan. Pyramidal neurons in different cortical layers exhibit distinct dynamics and plasticity of apical dendritic spines. *Frontiers in Neural Circuits*, 11:88, 2017. doi: 10.3389/fncir.2017.00088.

[25] James C. R. Whittington and Rafal Bogacz. Theories of error back-propagation in the brain. *Trends in Cognitive Sciences*, 23(3):235–250, 2019.

[26] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[27] Tonghui Xu, Xinzhu Yu, Andrew J. Perlik, Willie F. Tobin, Jonathan A. Zweig, Kelly Tennant, Theresa Jones, and Yi Zuo. Rapid formation and selective stabilization of synapses for enduring motor memories. *Nature*, 462(7275):915–919, 2009. doi: 10.1038/nature08389.

[28] Guang Yang, Feng Pan, and Wen-Biao Gan. Stably maintained dendritic spines are associated with lifelong memories. *Nature*, 462(7275):920–924, 2009. doi: 10.1038/nature08577.

[29] Jiancheng Yang, Rui Shi, and Bingbing Ni. MedMNIST classification decathlon: A lightweight AutoML benchmark for medical image analysis. In *IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 191–195, 2021.

[30] Jiancheng Yang, Rui Shi, Dong Wei, Ziyin Liu, Lina Zhao, Bin Ke, Hanspeter Pfister, and Bingbing Ni. MedMNIST v2: A large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023.

Supplementary Items

A Circuit and Self-assembly

We propose a circuit architecture that is grounded in a simple, biological synaptic motif (Fig. 1). It is based on two interacting streams of information flow: an ascending stream representing forward pathways and a descending stream representing feedback pathways, mirroring cortical connectivity between and within areas. These streams are defined by a set of synaptic weights, which obey local heterosynaptic plasticity rules. Unlike the standard computer implementation of gradient descent – backpropagation – our model does not require the explicit computation of derivatives, and does not require weight symmetry between forward and feedback pathways.

Notation. Let θ be the parameters of the model. We will use $\Delta f(\theta)$ to denote the difference of the quantity $f(\theta)$ after one step of the learning algorithm: $\Delta f(\theta) := f(\theta_{t+1}) - f(\theta_t)$. Any vector without a transpose superscript is treated as a column vector (e.g. $\nabla F(\theta)$ is a column vector for a scalar function F).

We assume that each pathway is organized in layers indexed with $\ell \in \{1, ..., L\}$, where L is the depth of the pathway in terms of modules. Let $h^\ell \in \mathbb{R}^{d_\ell}$ denote the neuron's cell body electric potential (i.e., before applying nonlinearity) of the ℓ-th layer upstream pathway and $\bar{h}^\ell \in \mathbb{R}^{\bar{d}_\ell}$ the downstream pathway. For reasons that will become clear, the upstream is also referred to as the forward pathway, and the downstream is the feedback pathway. Note that it is, in general, the case that the two pathways have different numbers of neurons, and so d_ℓ is not necessarily equal to $\bar{d}_\ell$. The simplest such biological motif is shown in Figure **??**, where four synaptic connection matrices connect two consecutive layers of the two pathways. Here, $W^\ell \in \mathbb{R}^{d_{\ell+1} \times d_\ell}$ is the connection from the ℓ-th layer upstream pathway to the $\ell+1$-th layer upstream pathway, $\bar{W}^\ell \in \mathbb{R}^{\bar{d}_\ell \times \bar{d}_{\ell+1}}$ the connection from $\ell+1$-th layer downstream to the ℓ-th layer downstream. $V^\ell \in \mathbb{R}^{\bar{d}_\ell \times d_\ell}$ and $\bar{V}^\ell \in \mathbb{R}^{d_{\ell+1} \times \bar{d}_{\ell+1}}$ are inter-stream connections: V goes from the downstream to the upstream, and $\bar{V}$ from the upstream to the downstream. The algorithm's Pseudo-code is in Appendix D.

The computation of the two pathways takes place sequentially, where the upstream computation is performed first and eventually used for inference, and the downstream computation takes place after inference and is used for synaptic weight updates:[2]

$$h^{\ell+1} = W^\ell D_u^\ell h^\ell \tag{5}$$

$$\bar{h}^\ell = \bar{W}^\ell D_d^{\ell+1} \bar{h}^{\ell+1}, \tag{6}$$

where D matrices are diagonal matrix that functions as the neuron nonlinearity and may be different for upstream (D_u, corresponding to h) and downstream (D_d, corresponding to $\bar{h}$) pathways and for different layers. The first layer of $h^1 := x$ is the input data, and $h^L := \hat{y}$ is

[2]This sequential ordering is consistent with, for example, the observation that inactivating V1 also leads to the inactivation of V2, while deactivating V2 has no effect on the activations of V1 (Anderson & Martin, 2009).

the output of the network. The input to the downstream pathway is an error signal obtained from an objective function F: $\bar{h}^L := \epsilon(\hat{y}) = -\nabla_{\hat{y}} F(\hat{y})$, which is computed from the output of the upstream pathway. The upstream propagates this signal in the reverse direction. One example of F is the MSE loss, where $F(\hat{y}) = \|y - \hat{y}\|^2$, where y is the correct label.

For the first layer, D is identity (i.e., no nonlinearity on the input). For later layers, we choose D_u to be a diagonal zero-one matrix:

$$D_u^\ell = \text{diag}(\mathbb{1}_{(h^\ell)_1}, ..., \mathbb{1}_{(h^\ell)_d}), \tag{7}$$

where $\mathbb{1}_x = 1$ if $x > 0$ and is zero if $x \leq 0$, corresponding to a ReLU nonlinearity. For D_d, we focus on the simplest case where $D_d = I$ is the identity matrix.

Another choice of D_d is the Jacobian of a ReLU nonlinearity based on forward activations, which requires $V = \bar{V} = I$ (namely a one-to-one correspondence between upstream and downstream neurons). With this choice of D_d, V and $\bar{V}$, our algorithm reduces to: 1. standard backpropagation, if $\bar{W}$ is hardwired to W throughout training. 2. Feedback Alignment (Lillicrap et al., 2016), if $\bar{W}$ is random and fixed. 3. the KP-algorithm (Kolen & Pollack, 1994), if W is learned using our learning rule, which converges to the exact gradient descent provably.

Heterosynaptic Learning: our local learning rule takes the general form of $\Delta W \propto S_{post} * S_{pre}$, where the change ΔW of a synapse is proportional to the product of postsynaptic signal S_{post} and presynaptic signal S_{pre}. This is similar to Hebbian learning, but we have extra emphasis that S_{post} refers to the signals sent by feedback synapses to the postsynaptic neuron instead of those arising from self-activation due to forward synapses. Feedback signals are informative in learning while self-activation signals only lead to non-informative arbitrary self-strengthening.

In most situations, S_{pre} represents the presynaptic potential on an axon, mathematically taking the form of $D_{pre} * h_{pre}$, where D_{pre} is the nonlinearity and h_{pre} is the electric potential of the presynaptic cell body. The nature of S_{post} on the other hand is an open question. There are at least two possibilities: (1) S_{post} is simply proportional to the postsynaptic potential h_{post} and is not affected by postsynaptic neuron's activity (2) S_{post} is proportional to the postsynaptic potential modulated by postsynaptic neuron's activity, namely $S_{post} \propto D_{post} * h_{post}$. This makes the learning rule $\Delta W \propto (D_{post} h_{post}) * (D_{pre} h_{pre})$, which is even more symmetric.

This simple rule guides the learning of all four directions of synapses in our model:

$$\Delta W^\ell = \eta_W (D_u^{\ell+1} (\bar{V}^\ell D_d^{\ell+1} \bar{h}^{\ell+1})) (D_u^\ell h^\ell)^T - \gamma W^\ell \qquad \propto (D_{W post} S_{W post})(D_{W pre} S_{W pre})^T, \tag{8}$$

$$\Delta \bar{W}^\ell = \eta_{\bar{W}} (D_d^\ell (V^\ell D_u^\ell h^\ell)) (D_d^{\ell+1} \bar{h}^{\ell+1})^T - \gamma \bar{W}^\ell \qquad \propto (D_{\bar{W} post} S_{\bar{W} post})(D_{\bar{W} pre} S_{\bar{W} pre})^T, \tag{9}$$

$$\Delta V^\ell = \eta_V (D_d^\ell \bar{h}^\ell)(D_u^\ell h^\ell)^T - \gamma V^\ell \qquad \propto (D_{V post} S_{V post})(D_{V pre} S_{V pre})^T, \tag{10}$$

$$\Delta \bar{V}^\ell = \eta_{\bar{V}} (D_u^{\ell+1} h^{\ell+1})(D_d^{\ell+1} \bar{h}^{\ell+1})^T - \gamma \bar{V}^\ell \qquad \propto (D_{\bar{V} post} S_{\bar{V} post})(D_{\bar{V} pre} S_{\bar{V} pre})^T, \tag{11}$$

where η is the time constant of learning (i.e., the learning rate), γ is the strength of local regularization terms that encourage the synapses to be weak when unused. The four sets of rules are almost completely symmetric along all directions – the only asymmetry between $W/\bar{W}$ and $V/\bar{V}$ stems from the fact that W and $\bar{W}$ have the overwriting effect on the

neurons that they connect to, while V and $\bar{V}$ only serve to provide learning signals but do not overwrite the target neurons' activations. This design choice may be explored further in the future. We further discuss the biological evidences for our learning rules in Appendix B.

Both the learning rate and the regularization strength are labeled with a different subscript to emphasize that they could have different time constants. In this work, we always keep the regularization strengths uniform. We empirically search over 64 combinations of the learning rates and use the best one in the experiment.[3] We found it particularly beneficial to fix the ratio between the four learning rates, while tuning the overall factor for different tasks.

A key feature of this architecture is that it allows both the inference network and the learning circuit, in addition to all their connections, to be simultaneously self-assembled. We thus name this algorithm "Self-Assembling Motif" (**SAM**).

B Gradient Learning in Overparametrized Downstream Pathway

The circuits we describe can be viewed as a generalized version of SGD, with a matrix form and learnable learning rate – in this sense, the algorithm is learning the learning algorithm itself. The following theorem is an informal statement of this result. The full formal detail and proof are provided in Appendix A.

Theorem 2. *Consider a ReLU neural network with an arbitrary width and depth and with an overparametrized downstream pathway. If both V^ℓ and $\bar{V}^\ell$ are full-rank for all ℓ, then, for any x such that for all $\ell \in [L]$, $\Delta V^\ell = O(\epsilon)$ and $\Delta \bar{V}^\ell = O(\epsilon)$,*

$$\Delta_{\text{SAM}} W^\ell = H \Delta_{\text{sgd}} W^\ell - \gamma W^\ell + O(\epsilon), \tag{12}$$

for a positive definite matrix $H = \bar{V}^\ell (\bar{V}^\ell)^T$, and $\Delta_{\text{sgd}} W^\ell = -\nabla_W F$ is the SGD update.

Remark. This theorem essentially shows that when V and $\bar{V}$ reach stationarity, the algorithm will run like a SGD algorithm with a matrix learning rate H and weight decay. The fact that matrix H is PSD is crucial from a mathematical perspective, as it guarantees that the loss objective of training will decrease in expectation and that the stationary points of the circuit will be identical to that of SGD. For example, besides SGD itself, Adam or Natural Gradient Descent can also be seen as having a learning rate corresponding to a PSD matrix.

It is worth noticing that this algorithm is different from SGD. H is not only a layer-wise matrix learning rate: it is explicitly dependent on time and is also being learned by the algorithm! This is consistent with observed metaplasticity in biological systems (Abraham & Bear, 1996; Abraham, 2008) – though it extends the idea in a fundamental way since the learning algorithm itself is being learned during the training of the forward network. In Figure 2a, we show examples of the H matrix after training.

[3] See Section E.

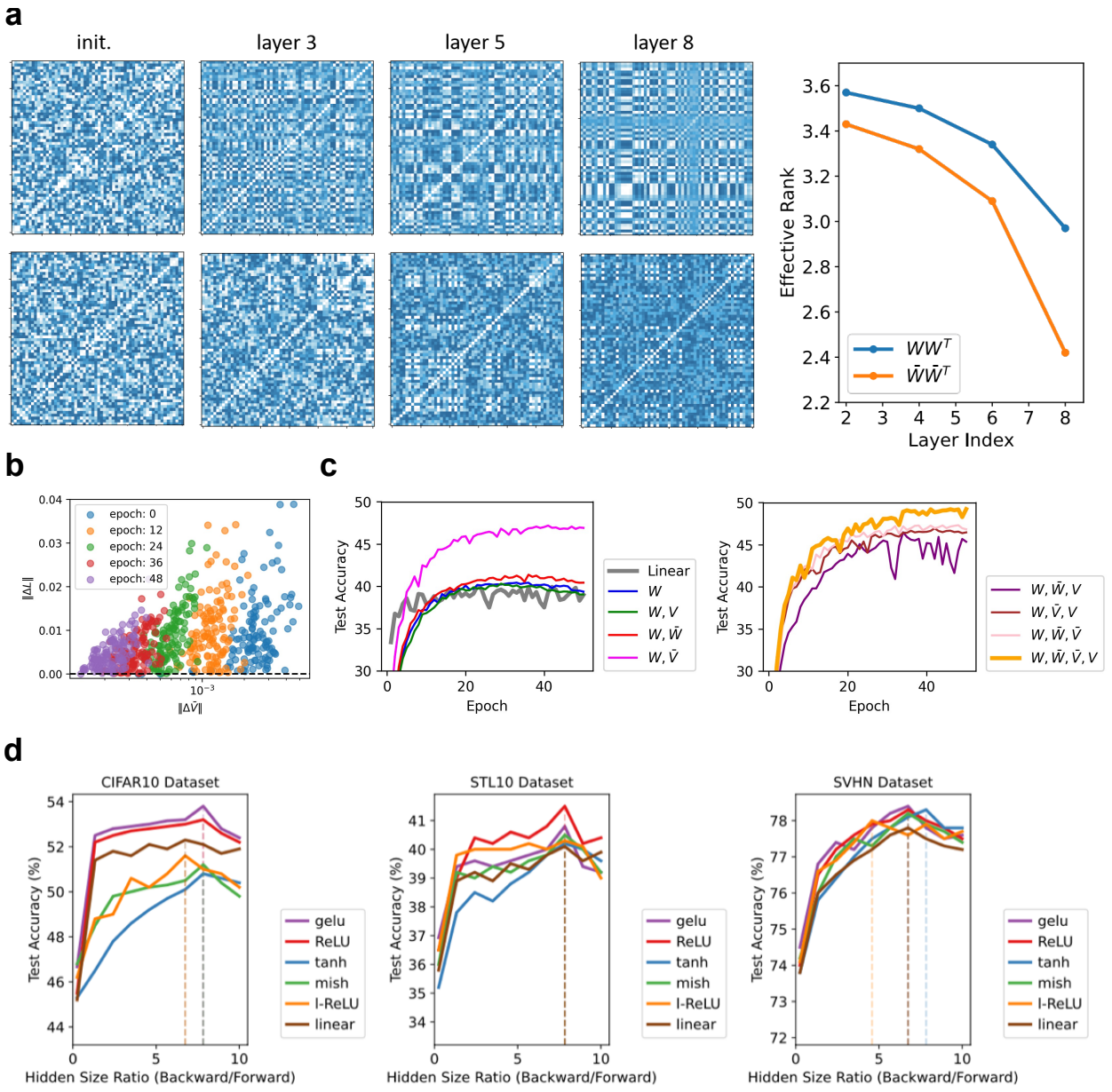

Figure S1: **Detailed analyses, variants and ablation studies of SAM. (a)** The matrices $\bar{V}\bar{V}^T$ (upper) and WW^T (lower) before and after training. Also, recall that $\bar{V}\bar{V}^T = H$ is the matrix learning rate for W (Theorem 1). The neurons within the same layer become correlated after training. The spectra of the weight matrices of the two pathway are interestingly found to be effectively low-rank (right). This visualization was done using the SVHN dataset with a network having 8 hidden layers. **(b)** Both $\bar{V}$ and ΔL becomes smaller and smaller over time, a condition required for the algorithm to simulate gradient descent (Theorem 1). **(c)** Ablation study: see caption in main text. **(d)** The impact of the width of the downstream pathway on performance. We use different activation functions in the forward pathway. The width of the upstream pathway was kept constant at 200, while varying the width of the downstream pathway. We find that having a wider downstream network improves the performance of the feedforward network up to an overparametrization ratio of 7.5.

Another interesting aspect of the algorithm (see Theorem 1) is that it allows (but does not require) the backward pathway to be overparametrized. That is, the number of downstream connections can be larger than the number of forward connections. This is consistent with observations in biological circuits (Briggs, 2020), but it is inconsistent with previous biologically inspired algorithms of learning, which require the same number of connections for ascending and descending pathways. Also, it turns out that the backward pathway is highly flexible in the choice of the activation; with any choice of D_d, the theorem holds. This prediction of the robustness of the backward pathway is numerically supported by the experiment in Section C.2.

When is it possible to satisfy the condition that V and $\bar{V}$ are stationary? There are many possible ways. Two notable cases are when V and $\bar{V}$ are permutation matrices, and D_d is the Jacobian matrix of D_u. In this case, H is identity, and so our algorithm is completely identical to SGD. In a different case, the feedback pathway is very expressive (e.g., by having a large number of feedback neurons), and so it is capable, from an approximation perspective, of approximating the forward net. Figure 2b shows how $\Delta\bar{V}$ approaches zero during training, and how this accompanies the decrease and $\|\Delta L\|$, consistent with the theory.

C Main Observations

C.1 SAM is an Efficient Learning Algorithm

We evaluate our algorithm on seven widely used image classification datasets: CIFAR10 (Krizhevsky et al., 2009), MNIST (Yann, 1998), Fashion MNIST (Xiao et al., 2017), ChestMNIST (Netzer et al., 2011), PathMNIST (Yang et al., 2021, 2023), SVHN (Netzer et al., 2011), and STL10 (Coates et al., 2011). For details about these datasets, see Appendix C. We compare our algorithm's performance against the standard backpropagation and other biologically plausible algorithms. These include Feedback Alignment (FA) (Lillicrap et al., 2016), which uses random fixed weights for error propagation instead of transposed weights; Weight Mirroring (WM) (Akrout et al., 2019a), a method enforcing symmetry between the forward and backward passes by mirroring weights; and the KP algorithm (Akrout et al., 2019a), which adjusts synaptic weights without requiring weight transport, simplifying network architecture and enhancing learning in large networks. While the primary goal of this paper is not to pursue State-of-the-Art (SOTA) performance but to design a neural network algorithm that can be deployed in the brain, our approach achieves SOTA among biologically plausible algorithms on all datasets (see in Table 1). Notably, our algorithm outperforms models trained with backpropagation on six out of the seven datasets. The only exception is ChestMNIST, where it performs 0.02% lower than backpropagation.

C.2 Overparameterization in the Downstream Pathway is Beneficial

In the nervous system, the number of backprojections can be much larger than that of the forward stream. For example, the feedback connections V1 to LGN is can be roughly 10

times the RGC synapses from the retina (Briggs, 2020). We have observed a potential computational explanation of this phenomenon in our simulations. Our models perform better as the feedback pathway becomes overparametrized and achieves the best performance across multiple datasets when the parameter count of the downstream pathway is approximately 5 to 8 times that of the upstream pathway. Specifically, as shown in Figure 2d for CIFAR10, performance peaks when the parameter count of the downstream pathway is 8 times that of the upstream pathway. We perform experiments on the three most challenging datasets from the seven datasets used in Table 1: CIFAR10 (Krizhevsky et al., 2009), SVHN (Netzer et al., 2011), and STL10 (Coates et al., 2011).

C.3 Learned Representations are Structurally Low-Rank

Now, we show some examples of the learned weight matrices. We train a width-60 network on SVHN dataset for 50 epochs. See Figure 2a. As is common in deep learning, we plot the matrix WW^T and also $\bar{V}\bar{V}^T$. We see that before training, these matrices are primarily diagonal matrices due to i.i.d. Gaussian initialization. After training, correlations between neurons and weights appear. We characterize the rank of a matrix through a continuous metric referred to as the effective rank (Roy & Vetterli, 2007). We find that these weight matrices also become effectively low-rank.

Figure 2a can be compared with similar matrices trained with SGD in Ziyin et al. (2024) and Huh et al. (2021), for example, and they seem to have similar high-level structures, suggesting that our algorithm can lead to a meaningful representation learning. Detailed analysis of the learned representation using our algorithm may be an interesting future problem.

C.4 Interconnection Plasticity is Crucial

A remaining question is whether learning at some of the interconnections is more important than at others, and if so, which is more important? Additionally, are any interconnections not important at all? We perform this ablation study on CIFAR-10, where we initialize all the parameters randomly but only update a subset of all the connections. See Figure 2c. Note that W is always plastic. To gauge how much the network has learned about the nonlinear relations in the data, we also compare with the learning trajectory of a linear model trained with SGD.

We discover the following phenomena:

- Only updating W works but only works as well as the linear model; making other parts plastic is thus important for learning nonlinear functions;
- If only one additional connection is made plastic, $\bar{V}$ is the most important one, improving performance from 40% to roughly 46%;
- Making $\bar{V}$ plastic is similar to making W, V plastic together, suggesting that some loss of learning capabilities due to connection lesions can be compensated by other connections (but not all);

- Making all four connections plastic improves the performance further, to the SGD level (cf. Table 1); this means that all connections need to be plastic to achieve the best performance.

Therefore, none of the pathways we introduced are redundant, and this partially explains the outperformance of our proposed algorithm over the existing two-pathway biologically plausible learning algorithms. Another reflection of the fact that there is an asymmetry between different pathways is that different connections require different learning time constants to work. In particular, the $\bar{V}$ matrices require the largest learning rates, usually four times that of W (see Appendix).

D Theorems and Proofs

The following Lemma shows that the interstream connections evolve towards becoming aligned with each other after training.

Lemma 1. *If $\Delta V^\ell = 0$ and $\Delta \bar{V}^\ell = 0$ for all l, then*

$$\bar{V}^{\ell-1} = (V^\ell)^T. \tag{13}$$

Proof. This is a consequence of the dynamics of $\bar{V}^{\ell-1}$ and $\bar{V}^\ell$ being essentially identical. By definition, we have that

$$\Delta V^\ell = \bar{h}^\ell (h^\ell)^T - \gamma V^\ell, \tag{14}$$
$$\Delta (\bar{V}^{\ell-1})^T = \bar{h}^\ell (h^\ell)^T - \gamma (\bar{V}^{\ell-1})^T. \tag{15}$$

This implies that

$$\Delta (V^\ell - (V^{\ell-1})^T) = -\gamma (V^\ell - (V^{\ell-1})^T), \tag{16}$$

which is an exponential decay to zero. This finishes the proof. □

The following definition states what it means for the downstream pathway to be "overparametrized."

Definition 1. *The downstream pathway is said to be overparametrized if for all ℓ and $V^\ell \in \mathbb{R}^{\bar{d}_\ell \times d_\ell}$,*

$$d_\ell \leq \bar{d}_\ell. \tag{17}$$

We also say that the matrix V^ℓ is overparametrized if this condition holds.

The following Lemma is a technical step in the theorem proof.

Lemma 2. *If $V^\ell D_u^\ell (W^\ell)^T = \bar{W}^\ell D_d^{\ell+1} (\bar{V}^\ell)^T$, then,*

$$P \bar{W}^\ell D_d^{\ell+1} (\bar{V}^\ell)^T = \bar{W}^\ell D_d^{\ell+1} (\bar{V}^\ell)^T, \tag{18}$$

where $P = ((V^\ell)^T)^+ (V^\ell)^T$ is a projection matrix and A^+ denotes the pseudoinverse of the matrix A.

Proof. We have

$$P\bar{W}^\ell D_d^{\ell+1}(\bar{V}^\ell)^T = PV^\ell D_u^\ell (W^\ell)^T = V^\ell D_u^\ell (W^\ell)^T = \bar{W}^\ell D_d^{\ell+1}(\bar{V}^\ell)^T. \tag{19}$$

This finishes the proof. □

Now, we explain briefly that the sample-wise gradient of a ReLU network can be written in a specific form. As implied in the main text, the output of a ReLU network can be written in the following form

$$f(x) = W^L D_u^{L-1} .... D^1 W^1 x \tag{20}$$

where D_u^ℓ is the Jacobian of the ReLU activation at the i-th layer.

Let $F(f(x), y)$ be the loss function for the data point x and label y. The above discussion implies that the gradient of F with respect to W can be written as

$$\nabla_{W^\ell} F = \underbrace{(\nabla_f^T F W^L D_u^{L-1} ... D^\ell)^T}_{\text{error signal}} \underbrace{((D_u^{\ell-1}) W^{\ell-1} ... W_1 x)^T}_{\text{forward signal}}. \tag{21}$$

We will show that the downstream pathway will become self-assembled in a way that it computes the error signal.

Now, we are ready to the prove the main theorem. For the ease of reference, we state the theorem again here.

Theorem 3. *Consider a ReLU neural network with an arbitrary width and depth and with an overparametrized downstream pathway. If both V^ℓ and $\bar{V}^\ell$ are full-rank for all ℓ, then, for any x such that for all $\ell \in [L]$, $\Delta V^\ell = O(\epsilon)$ and $\Delta \bar{V}^\ell = O(\epsilon)$,*

$$\Delta_{\text{SAM}} W^\ell = H \Delta_{\text{sgd}} W^\ell - \gamma W^\ell + O(\epsilon), \tag{22}$$

for a positive definite matrix $H = \bar{V}^\ell (\bar{V}^\ell)^T$, and $\Delta_{\text{sgd}} W^\ell = -\nabla_W F$ is the SGD update.

Proof. By definition of the algorithm,

$$\Delta V^\ell = \bar{h}^\ell (h^\ell)^T - \gamma V^\ell = \bar{W}^\ell D_d^{\ell+1} \bar{h}^{\ell+1} (h^\ell)^T - \gamma V^\ell, \tag{23}$$

$$\Delta (\bar{V}^\ell)^T = \bar{h}^{\ell+1} (h^{\ell+1})^T - \gamma (\bar{V}^\ell)^T = \bar{h}^{\ell+1} (h^\ell)^T D_u^\ell (W^\ell)^T - \gamma (\bar{V}^\ell)^T. \tag{24}$$

If $\Delta V^\ell = O(\epsilon)$ and $\Delta \bar{V}^\ell = O(\epsilon)$, we have

$$(\Delta V^\ell) D_u^\ell (W^\ell)^T = O(\epsilon), \tag{25}$$

$$\bar{W}^\ell D_d^{\ell+1} \Delta (\bar{V}^\ell)^T = O(\epsilon). \tag{26}$$

Thus, we have

$$V^\ell D_u^\ell (W^\ell)^T = \bar{W}^\ell D_d^{\ell+1} (\bar{V}^\ell)^T + O(\epsilon). \tag{27}$$

Because V^ℓ is full-rank and overparametrized, $G_\ell := (V^\ell)^T V^\ell$ is an invertible matrix. We can thus multiply $G^{-1}(V^\ell)^T$ on the left to obtain that

$$D_u^\ell (W^\ell)^T = G_\ell^{-1} (V^\ell)^T \bar{W}^\ell D_d^{\ell+1} (\bar{V}^\ell)^T + O(\epsilon). \tag{28}$$

Now, consider the matrix product between different layers[4]

$$(W^\ell D_u^\ell W^{\ell-1} D_u^{\ell-1})^T = (V^{\ell-1})^+ \bar{W}^{\ell-1} D_d^\ell (\bar{V}^{\ell-1})^T G_\ell^{-1} (V^\ell)^T \bar{W}^\ell D_d^{\ell+1} (\bar{V}^\ell)^T + O(\epsilon). \tag{29}$$

However, by Lemma 1, at stationarity, we must have

$$\bar{V}^{\ell-1} = (V^\ell)^T, \tag{30}$$

and so

$$(\bar{V}^{\ell-1})^T G_\ell^{-1} (V^\ell)^T = V^\ell G_\ell^{-1} (V^\ell)^T = P^2 = P, \tag{31}$$

where $P = ((V^\ell)^T)^+ (V^\ell)^T$ is a projection matrix, and the $+$ superscript denotes the pseudoinverse. Thus,

$$(W^\ell D_u^\ell W^{\ell-1} D_u^{\ell-1})^T = (V^{\ell-1})^+ \bar{W}^{\ell-1} D_d^\ell P \bar{W}^\ell D_d^{\ell+1} (\bar{V}^\ell)^T + O(\epsilon). \tag{32}$$

Using Lemma 2, we have that

$$P D_d^\ell \bar{W}^\ell (\bar{V}^\ell)^T = D_d^\ell \bar{W}^\ell (\bar{V}^\ell)^T, \tag{33}$$

which implies that

$$(W^\ell D_u^\ell W^{\ell-1} D_u^{\ell-1})^T = (V^{\ell-1})^+ \bar{W}^{\ell-1} D_d^\ell \bar{W}^\ell D_d^{\ell+1} (\bar{V}^\ell)^T + O(2\epsilon). \tag{34}$$

Namely, the intermediate projections P have no effect. This argument can be applied repetitively to show that

$$(W^\ell D_u^\ell ... W^{\ell-n} D_u^{\ell-n})^T = (V^{\ell-n})^+ \bar{W}^{\ell-n} D_d^{\ell-n+1} ... \bar{W}^\ell D_d^{\ell+1} (\bar{V}^\ell)^T + O((\ell-n)\epsilon) \tag{35}$$

for any l and $n < l$. This means that the backward net has essentially become the transpose of the forward net.

Now, by the definition of our algorithm, we have

$$\begin{aligned}
\bar{V}^\ell \bar{h}^{\ell+1} &= \bar{V}^\ell \bar{W}^{\ell+1} D_d^{\ell+2} ... \bar{W}^{L-1} D_d^L \bar{h}^L && (36)\\
&= \bar{V}^\ell V^{\ell+1} (W^{L-1} D_u^{L-1} ... W^\ell D_u^\ell)^T ((\bar{V}^\ell)^T)^{-1} \bar{h}^L && (37)\\
&= \bar{V}^\ell (\bar{V}^\ell)^T (W^{L-1} D_u^{L-1} ... W^\ell D_u^\ell)^T ((\bar{V}^\ell)^T)^{-1} \bar{h}^L && (38)\\
&= Z (W^{L-1} D_u^{L-1} ... W^\ell D_u^\ell)^T \bar{h}^L + O(\ell\epsilon), && (39)
\end{aligned}$$

where $Z = \bar{V}^\ell (\bar{V}^\ell)^T$ is positive definite, we have used the definition $\bar{V}^L = I$. By definition, $\bar{h}^L$ is the label gradient.

Therefore,

$$\begin{aligned}
D_u^\ell \bar{V}^\ell \bar{h}^{\ell+1} (h^\ell)^T D_u^\ell &= \underbrace{D_u^\ell Z D_u^\ell}_{\text{effective learning rate: } H} D_u^\ell (W^\ell)^T D_u^{L-1} (W^{L-1})^T \bar{h}^L (h^\ell)^T D_u^\ell + O(\ell\epsilon), && (40)\\
&= H \nabla_{h^{\ell+1}} F && (41)
\end{aligned}$$

The proof is complete.

□

[4]Note that $(V^\ell)^+ = G_\ell^{-1} (V^\ell)^T$.

Note that this update rule in Eq. (40) is zero if and only if the SGD update is zero. It thus has the same stationary points as SGD. To see this, note that D_u^ℓ is a projection matrix, and Z is full-rank and symmetric. Thus, the null space of H is the same as the null space of D_u^ℓ. Therefore, if $\nabla_{h^{\ell+1}} F \neq 0$, it must be the case that $H \nabla_{h^{\ell+1}} F \neq 0$.

Proposition 1. *Let Φ denote the empirical loss. If H is positive semi-definite, and if we update the parameters by*

$$\dot{\theta} = -H \nabla_\theta \Phi, \tag{42}$$

then, $\Phi(\theta(t))$ is a monotonic decreasing function of t for any initialization $\theta(0)$.

Proof. For the first part, consider the time evolution of

$$\dot{\Phi} = (\nabla \Phi)^T \dot{\theta} = -(\nabla \Phi)^T H \nabla \Phi \leq 0. \tag{43}$$

□

E Experimental Setup

All experiments are conducted with PyTorch on one NVIDIA A100 80GB GPU. The batch size is set to 256.

Datasets. We use six datasets to evaluate our method. *CIFAR-10* (Krizhevsky et al., 2009) consists of 60,000 color images in 10 different classes, with each image having a resolution of 32x32 pixels. The dataset is divided into 50,000 training images and 10,000 testing images. The classes are mutually exclusive and include common objects such as airplanes, cars, cats, and dogs.

The *MNIST* (Yann, 1998) contains 70,000 grayscale images of handwritten digits (0-9), each with a resolution of 28x28 pixels. The dataset is divided into 60,000 training images and 10,000 testing images. MNIST is known for its simplicity and has been a standard dataset for testing machine learning algorithms.

Fashion MNIST (Xiao et al., 2017) contains 70,000 grayscale images of clothing items like T-shirts, trousers, shoes, and bags, with each image having a resolution of 28x28 pixels. The dataset is also split into 60,000 training and 10,000 testing images. Fashion MNIST is considered more challenging than MNIST due to the variability in the visual features of clothing items.

The *SVHN* (Netzer et al., 2011) contains over 600,000 color images of digits (0-9), each with a resolution of 32x32 pixels. The dataset is divided into a training set of 73,257 images, a testing set of 26,032 images, and an additional 531,131 images for extra training. SVHN is challenging due to the varying digit sizes, orientations, and complex backgrounds.

The ChestMNIST (Yang et al., 2021, 2023) comprises chest X-ray images for multi-label classification tasks. Derived from the NIH ChestX-ray14 dataset, it includes images annotated with 14 different pathological labels, such as pneumonia and emphysema. It is widely used in medical imaging studies to develop and evaluate machine learning models capable of diagnosing multiple conditions from X-ray images.

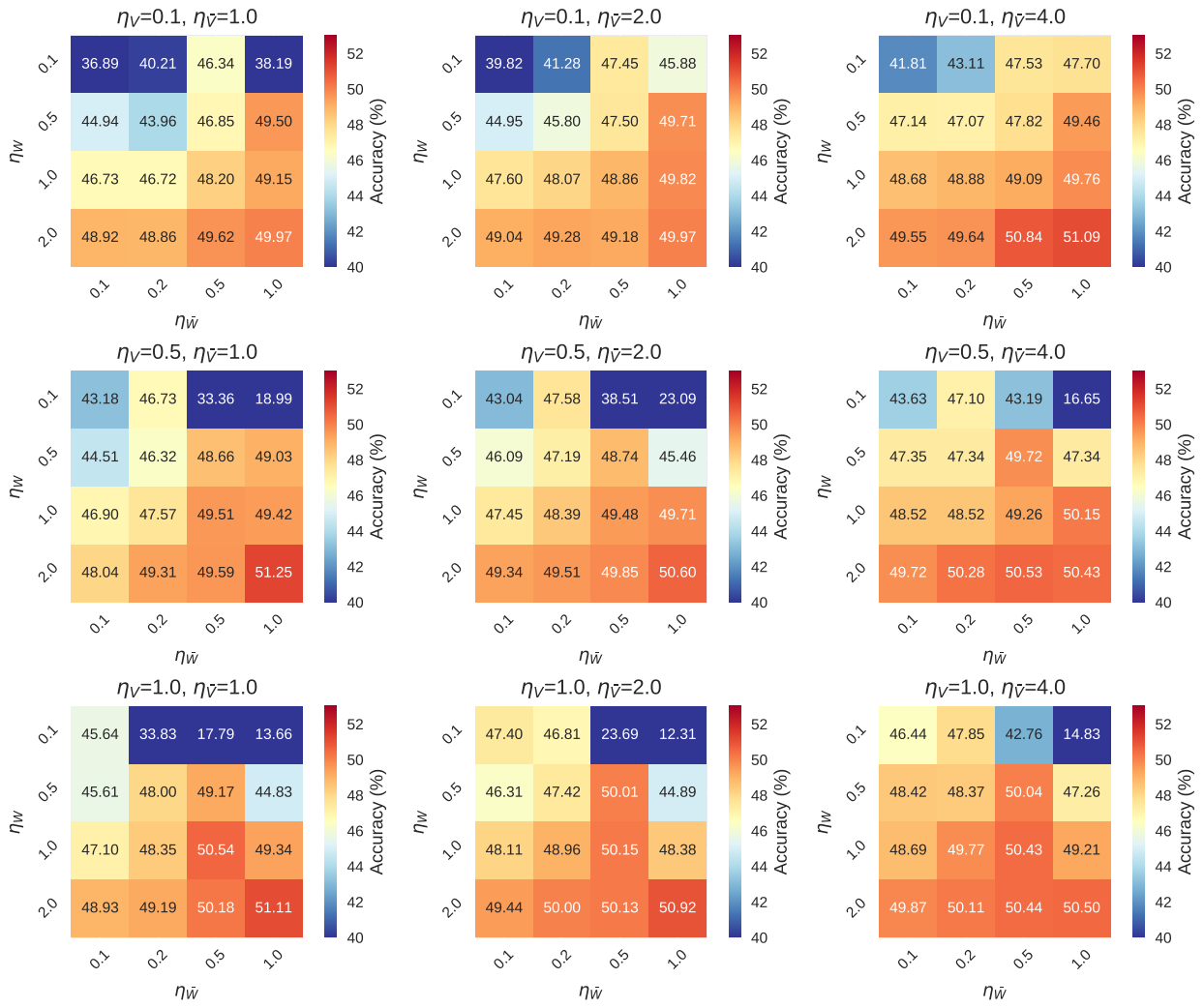

Figure S2: Performance of a four-layer MLP for different choices of learning rates. Interestingly, the best performance is achieved when η_V is the smallest.

PathMNIST (Yang et al., 2021, 2023) is a pathology image dataset designed for multi-class classification tasks. It originates from the NCT-CRC-HE-100K dataset and includes histological images of nine different types of tissue from colorectal cancer samples, such as adenocarcinoma and lymph node tissue. This dataset is used primarily for training models to differentiate between various cancerous tissue types based on their histopathological features.

STL-10 (Coates et al., 2011) consists of 13,000 labeled images across 10 different classes and 100,000 unlabeled images. Each image has a resolution of 96x96 pixels, which is significantly larger than CIFAR-10 images, making the dataset more challenging. The dataset has a training set of 5,000 labeled images and a test set of 8,000 labeled images.

Baselines. We compare our method with four baselines, including SGD and three biologically-motivated algorithms. In the four baseline we assume that there is one feedback pathway for each feedforward pathhway – a strong assumption on developmental mechanisms that SAM

does not require.

Stochastic Gradient Descent (SGD) is the standard optimization method that updates parameters iteratively using a subset of data to minimize the objective function.

Weight Mirroring (Akrout et al., 2019a) is a technique generally used in neural networks to enforce symmetry between forward and backward passes. This can be implemented by mirroring the weights used in the forward pass for use in the backward pass, which often helps with learning efficiency and stability.

Feedback Alignment (Lillicrap et al., 2016) offers an alternative to traditional backpropagation by using random, fixed weights instead of transposed forward-pass weights to propagate error signals.

Kolen-Pollack (KP) algorithm (Kolen & Pollack, 1994) enables synaptic weight adjustments in neural networks without weight transport, simplifying the architecture and enhancing learning in large networks.