Why do similarity matching objectives lead to Hebbian/anti-Hebbian networks?

Cengiz Pehlevan[1], Anirvan M. Sengupta[1,2], and Dmitri B. Chklovskii[1,3]

[1]Center for Computational Biology, Flatiron Institute, New York, NY

[2]Physics and Astronomy Department, Rutgers University, New Brunswick, NJ

[3]NYU Langone Medical Center, New York, NY

Abstract

Modeling self-organization of neural networks for unsupervised learning using Hebbian and anti-Hebbian plasticity has a long history in neuroscience. Yet, derivations of single-layer networks with such local learning rules from principled optimization objectives became possible only recently, with the introduction of similarity matching objectives. What explains the success of similarity matching objectives in deriving neural networks with local learning rules? Here, using dimensionality reduction as an example, we introduce several variable substitutions that illuminate the success of similarity matching. We show that the full network objective may be optimized separately for each synapse using local learning rules both in the offline and online settings. We formalize the long-standing intuition of the rivalry between Hebbian and anti-Hebbian rules by formulating a min-max optimization problem. We introduce a novel dimensionality reduction objective using fractional matrix exponents. To illustrate the generality of our approach, we apply it to a novel formulation of dimensionality reduction combined with whitening. We confirm numerically that the networks with learning rules derived from principled objectives perform better than those with heuristic learning rules.

1 Introduction

The human brain generates complex behaviors via the dynamics of electrical activity in a network of $\sim 10^{11}$ neurons each making $\sim 10^4$ synaptic connections. As

there is no known centralized authority determining which specific connections a neuron makes or specifying the weights of individual synapses, synaptic connections must be established based on local rules. Therefore, a major challenge in neuroscience is to determine local synaptic learning rules that would ensure that the network acts coherently, i.e. guarantee robust network self-organization.

Much work has been devoted to the self-organization of neural networks for solving unsupervised computational tasks using Hebbian and anti-Hebbian learning rules (Földiak, 1990; Földiak, 1989; Rubner and Tavan, 1989; Rubner and Schulten, 1990; Carlson, 1990; Plumbley, 1993b; Leen, 1991; Plumbley, 1993a; Linsker, 1997). Unsupervised setting is natural in biology because large-scale labeled datasets are typically unavailable. Hebbian and anti-Hebbian learning rules are biologically plausible because they are local: The weight of an (anti-)Hebbian synapse is proportional to the (minus) correlation in activity between the two neurons the synapse connects.

In networks for dimensionality reduction, for example, feedforward connections use Hebbian rules and lateral - anti-Hebbian, Figure 1. Hebbian rules attempt to align each neuronal feature vector, whose components are the weights of synapses impinging onto the neuron, with the input space direction of greatest variance. Anti-Hebbian rules mediate competition among neurons which prevents their feature vectors from aligning in the same direction. A rivalry between the two kinds of rules results in the equilibrium where synaptic weight vectors span the principal subspace of the input covariance matrix, i. e. the subspace spanned by the eigenvectors corresponding to the largest eigenvalues.

However, in most existing single-layer networks, Figure 1, Hebbian and anti-Hebbian learning rules were postulated rather than derived from a principled objective. Having such derivation should yield better performing rules and deeper understanding than has been achived using heuristic rules. But, until recently, all derivations of single-layer networks from principled objectives led to biologically implausible non-local learning rules, where the weight of a synapse depends on the activities of neurons other than the two the synapse connects.

Recently, single-layer networks with local learning rules have been derived from similarity matching objective functions (Pehlevan et al., 2015; Pehlevan and Chklovskii, 2014; Hu et al., 2014). But why do similarity matching objectives lead to neural networks with local, Hebbian and anti-Hebbian learning rules? A clear answer to this question has been lacking.

Here, we answer this question by performing several illuminating variable transformations. Specifically, we reduce the full network optimization problem to a set of trivial optimization problems for each synapse which can be solved locally. Eliminating neural activity variables leads to a min-max objective in terms of feedforward and lateral synaptic weight matrices. This finally formalizes the long-held intuition about the adversarial relationship of Hebbian and anti-Hebbian

learning rules.

In this paper, we make the following contributions. In Section 2, we present a more transparent derivation of the previously proposed online similarity matching algorithm for Principal Subspace Projection (PSP). In Section 3, we propose a novel objective for PSP combined with spherizing, or whitening, the data, which we name Principal Subspace Whitening (PSW), and derive from it a biologically plausible online algorithm. Also, in Sections 2 and 3, we demonstrate that stability in the offline setting guarantees projection onto the principal subspace and give principled learning rate recommendations. In Section 4, by eliminating activity variables from the objectives, we derive min-max formulations of PSP and PSW which yield themselves to game-theoretical interpretations. In Section 5, by expressing the optimization objectives in terms of feedforward synaptic weights only, we arrive at novel formulations of dimensionality reduction in terms of fractional powers of matrices. In Section 6, we demonstrate numerically that the performance of our online algorithms is superior to the heuristic ones.

2 From similarity matching to Hebbian/anti-Hebbian networks for PSP

2.1 Derivation of a mixed PSP from similarity matching

The PSP problem is formulated as follows. Given T centered input data samples, $\mathbf{x}_t \in \mathbb{R}^n$, find T projections, $\mathbf{y}_t \in \mathbb{R}^k$, onto the principal subspace ($k \leq n$), i.e. the subspace spanned by eigenvectors corresponding to the k top eigenvalues of the input covariance matrix:

$$\mathbf{C} \equiv \frac{1}{T}\sum_{t=1}^{T} \mathbf{x}_t \mathbf{x}_t^\top = \frac{1}{T}\mathbf{X}\mathbf{X}^\top, \tag{1}$$

where we resort to a matrix notation by concatenating input column vectors into $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_T]$. Similarly, outputs are $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_T]$.

Our goal is to derive a biologically plausible single-layer neural network implementing PSP by optimizing a principled objective. Biological plausibility requires that the learning rules are local, i.e. synaptic weight update depends on the activity of only the two neurons the synapse connects. The only PSP objective known to yield a single-layer neural network with local learning rules is based on similarity matching (Pehlevan et al., 2015). This objective, borrowed from Multi-Dimensional Scaling (MDS), minimizes the mismatch between the similarity of

inputs and outputs (Mardia et al., 1980; Williams, 2001; Cox and Cox, 2000):

$$\text{PSP}: \qquad \min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} \frac{1}{T^2}\left\|\mathbf{X}^\top\mathbf{X}-\mathbf{Y}^\top\mathbf{Y}\right\|_F^2. \tag{2}$$

Here, similarity is quantified by the inner products between all pairs of inputs (outputs) comprising the Grammians $\mathbf{X}^\top\mathbf{X}$ ($\mathbf{Y}^\top\mathbf{Y}$).

One can understand intuitively that the objective (2) is optimized by the projection onto the principal subspace by considering the following (for a rigorous proof see (Pehlevan and Chklovskii, 2015; Mardia et al., 1980; Cox and Cox, 2000)). First, substitute a Singular Value Decomposition (SVD) for matrices $\mathbf{X}$ and $\mathbf{Y}$ and note that the mismatch is minimized by matching right singular vectors of $\mathbf{Y}$ to that of $\mathbf{X}$. Then, rotating the Grammians to the diagonal basis reduces the minimization problem to minimizing the mismatch between the corresponding singular values squared. Therefore, $\mathbf{Y}$ is given by the top k right singular vectors of $\mathbf{X}$ scaled by corresponding singular values. As the objective (2) is invariant to the left-multiplication of $\mathbf{Y}$ by an orthogonal matrix, it has infinitely many degenerate solutions. One such solution corresponds to the Principal Component Analysis (PCA).

Unlike non-neural-network formulations of PSP or PCA, similarity matching outputs principal components (scores) rather than principal eigenvectors of the input covariance (loadings). Such difference in formulation is motivated by our interest in PSP or PCA neural networks (Diamantaras and Kung, 1996) that output principal components, $\mathbf{y}_t$, rather than principal eigenvectors. Principal eigenvectors are not transmitted downstream of the network but can be recovered computationally from the synaptic weight matrices. Although synaptic weights do not enter the objective (2), in previous work (Pehlevan et al., 2015), they arose naturally in the derivation of the online algorithm (see below) and stored correlations between input and output neural activities.

Next, we derive the min-max PSP objective from Eq. (2), starting with expanding the square of the Frobenius norm:

$$\arg\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} \frac{1}{T^2}\left\|\mathbf{X}^\top\mathbf{X}-\mathbf{Y}^\top\mathbf{Y}\right\|_F^2 = \arg\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} \frac{1}{T^2}\mathrm{Tr}\left(-2\mathbf{X}^\top\mathbf{X}\mathbf{Y}^\top\mathbf{Y}+\mathbf{Y}^\top\mathbf{Y}\mathbf{Y}^\top\mathbf{Y}\right). \tag{3}$$

We can rewrite Eq. (3) by introducing two new dynamical variable matrices in place of covariance matrices $\frac{1}{T}\mathbf{X}\mathbf{Y}^\top$ and $\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top$:

$$\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} \min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \max_{\mathbf{M}\in\mathbb{R}^{k\times k}} L_{PSP}(\mathbf{W},\mathbf{M},\mathbf{Y}), \quad \text{where} \tag{4}$$

$$L_{PSP}(\mathbf{W},\mathbf{M},\mathbf{Y}) \equiv \mathrm{Tr}\left(-\frac{4}{T}\mathbf{X}^\top\mathbf{W}^\top\mathbf{Y}+\frac{2}{T}\mathbf{Y}^\top\mathbf{M}\mathbf{Y}\right)+2\mathrm{Tr}\left(\mathbf{W}^\top\mathbf{W}\right)-\mathrm{Tr}\left(\mathbf{M}^\top\mathbf{M}\right). \tag{5}$$

To see that Eq. (5) is equivalent to Eq. (3) find optimal $\mathbf{W}^* = \frac{1}{T}\mathbf{Y}\mathbf{X}^\top$ and $\mathbf{M}^* = \frac{1}{T}\mathbf{Y}\mathbf{Y}^\top$ by setting the corresponding derivatives of objective (5) to zero. Then, substitute $\mathbf{W}^*$ and $\mathbf{M}^*$ into Eq. (5) to obtain (3).

Finally, we exchange the order of minimization with respect to $\mathbf{Y}$ and $\mathbf{W}$ as well as the order of minimization with respect to $\mathbf{Y}$ and maximization with respect to $\mathbf{M}$ in Eq. (5). The last exchange is justified by the saddle point property (see Proposition 1 in Appendix A). Then, we arrive at the following min-max optimization problem:

$$\min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \max_{\mathbf{M}\in\mathbb{R}^{k\times k}} \min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} L_{PSP}(\mathbf{W},\mathbf{M},\mathbf{Y}), \tag{6}$$

where $L_{PSP}(\mathbf{W},\mathbf{M},\mathbf{Y})$ is defined in Eq. (5). We call this a mixed objective because it includes both output variables, $\mathbf{Y}$, and covariances, $\mathbf{W}$ and $\mathbf{M}$.

2.2 Offline PSP algorithm

In this section, we present an offline optimization algorithm to solve the PSP problem and analyze fixed points of the corresponding dynamics. These results will be used in the next Section for the biologically plausible online algorithm implemented by neural networks.

In the offline setting, we can solve Eq. (6) by the alternating optimization approach used commonly in neural networks literature (Olshausen et al., 1996; Olshausen and Field, 1997; Arora et al., 2015). We, first, minimize with respect to $\mathbf{Y}$ while keeping $\mathbf{W}$ and $\mathbf{M}$ fixed,

$$\mathbf{Y}^* = \arg\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} L_{PSP}(\mathbf{W},\mathbf{M},\mathbf{Y}), \tag{7}$$

and, second, make a gradient descent-ascent step with respect to $\mathbf{W}$ and $\mathbf{M}$ while keeping $\mathbf{Y}$ fixed:

$$\begin{bmatrix} \mathbf{W} & \mathbf{M} \end{bmatrix} \longleftarrow \begin{bmatrix} \mathbf{W} & \mathbf{M} \end{bmatrix} + \begin{bmatrix} -\eta\frac{\partial L_{PSP}(\mathbf{W},\mathbf{M},\mathbf{Y}^*)}{\partial \mathbf{W}} & \frac{\eta}{\tau}\frac{\partial L_{PSP}(\mathbf{W},\mathbf{M},\mathbf{Y}^*)}{\partial \mathbf{M}} \end{bmatrix}, \tag{8}$$

where η is the $\mathbf{W}$ learning rate and $\tau > 0$ is a ratio of learning rates for $\mathbf{W}$ and $\mathbf{M}$. In Appendix C, we analyze how τ affects linear stability of the fixed point dynamics. These two phases are iterated until convergence (Algorithm 1)[1].

[1]This alternating optimization is identical to a gradient descent-ascent (see Proposition 2 in Appendix B) in $\mathbf{W}$ and $\mathbf{M}$ on the objective:

$$l_{PSP}(\mathbf{W},\mathbf{M}) \equiv \min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} L_{PSP}(\mathbf{W},\mathbf{M},\mathbf{Y}).$$

Algorithm 1 Offline min-max PSP

1: Initialize $\mathbf{W}$. Initalize $\mathbf{M}$ as a positive definite matrix.
2: Iterate until convergence:
3: Minimize Eq. (5) with respect to $\mathbf{Y}$, keeping $\mathbf{W}$ and $\mathbf{M}$ fixed:

$$\mathbf{Y} = \mathbf{M}^{-1}\mathbf{W}\mathbf{X}. \tag{9}$$

4: Perform a gradient descent-ascent step with respect to $\mathbf{W}$ and $\mathbf{M}$ for a fixed $\mathbf{Y}$:

$$\begin{aligned} \mathbf{W} &\longleftarrow \mathbf{W} + 2\eta \left(\frac{1}{T}\mathbf{Y}\mathbf{X}^\top - \mathbf{W} \right), \\ \mathbf{M} &\longleftarrow \mathbf{M} + \frac{\eta}{\tau} \left(\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top - \mathbf{M} \right). \end{aligned} \tag{10}$$

where the step size, $0 < \eta < 1$, may depend on the iteration.

Optimal $\mathbf{Y}$ in Eq. (9) exists because $\mathbf{M}$ stays positive definite if initialized as such.

2.3 Linearly stable fixed points of Algorithm 1 correspond to the PSP

Here we demonstrate that convergence of Algorithm 1 to fixed $\mathbf{W}$ and $\mathbf{M}$ implies that $\mathbf{Y}$ is a PSP of $\mathbf{X}$. To this end, we approximate the gradient descent-ascent dynamics in the limit of small learning rate with the system of differential equations:

$$\begin{aligned} \mathbf{Y}(t) &= \mathbf{M}^{-1}(t)\mathbf{W}(t)\mathbf{X}, \\ \frac{d\mathbf{W}(t)}{dt} &= \frac{2}{T}\mathbf{Y}(t)\mathbf{X}^\top - 2\mathbf{W}(t), \\ \tau\frac{d\mathbf{M}(t)}{dt} &= \frac{1}{T}\mathbf{Y}(t)\mathbf{Y}(t)^\top - \mathbf{M}(t), \end{aligned} \tag{11}$$

where t is now the time index for gradient descent-ascent dynamics.

To state our main result in Theorem 1, we define the "filter matrix" $\mathbf{F}(t)$ whose rows are "neural filters"

$$\mathbf{F}(t) := \mathbf{M}^{-1}(t)\mathbf{W}(t), \tag{12}$$

so that, according to Eq. (9),

$$\mathbf{Y}(t) = \mathbf{F}(t)\mathbf{X}. \tag{13}$$

Theorem 1. *Fixed points of the dynamical system* (11) *have the following properties:*

1. *The neural filters,* $\mathbf{F}$*, are orthonormal, i.e.* $\mathbf{F}\mathbf{F}^\top = \mathbf{I}$.

2. *The neural filters span a* k*-dimensional subspace in* $\mathbb{R}^n$ *spanned by some* k *eigenvectors of the input covariance matrix.*

3. *Stability of a fixed point requires that the neural filters span the* **principal** *subspace of* $\mathbf{X}$.

4. *Suppose the neural filters span the principal subspace. Define*

$$\gamma_{ij} := 2 + \frac{(\sigma_i - \sigma_j)^2}{\sigma_i \sigma_j}, \tag{14}$$

where $i = 1, \ldots, k$*,* $j = 1, \ldots, k$ *and* $\{\sigma_1, \ldots, \sigma_k\}$ *are the top* k *principal eigenvalues of* $\mathbf{C}$*. We assume* $\sigma_k \neq \sigma_{k+1}$*. This fixed point is linearly stable if and only if:*

$$\tau < \frac{1}{2 - 4/\gamma_{ij}} \tag{15}$$

for all (i, j) *pairs. By linearly stable we mean that linear perturbations of* $\mathbf{W}$ *and* $\mathbf{M}$ *converge to a configuration in which the new neural filters are merely rotations within the principal subspace of the original neural filters.*

Proof. See Appendix C. □

Based on Theorem 1 we claim that, provided the dynamics converges to a fixed point, Algorithm 1 has found a PSP of input data. Note that the orthonormality of the neural filters is desired and consistent with PSP since, in this approach, outputs, $\mathbf{Y}$, are interpreted as coordinates with respect to a basis spanning the principal subspace.

Theorem 1 yields a practical recommendation for choosing learning rate parameters in simulations. In a typical situation, one will not know the eigenvalues of the covariance matrix a priori but can rely on the fact, $\gamma_{ij} \geq 2$. Then, Eq. (15) implies that for $\tau \leq 1/2$ the principal subspace is linearly stable leading to numerical convergence and stability.

2.4 Online neural min-max optimization algorithms

Unlike the offline setting considered so far, where all the input data are available from the outset, in the online setting, input data are streamed to the algorithm

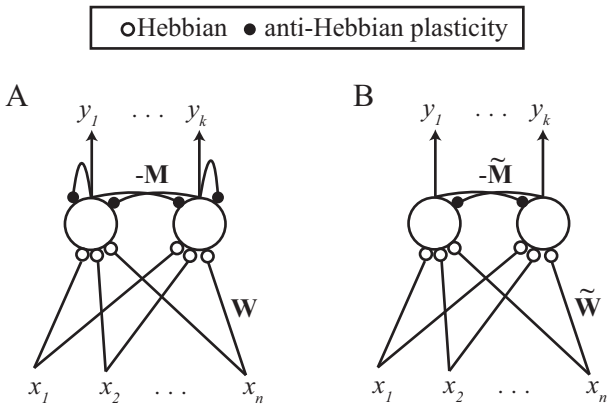

Figure 1: Dimensionality reduction neural networks derived by min-max optimization in the online setting. A. Network with autapses. B. Network without autapses.

sequentially, one at a time. The algorithm must compute the corresponding output before the next input arrives and transmit it downstream. Once transmitted, the output cannot be altered. Moreover, the algorithm cannot store in memory any sizable fraction of past inputs or outputs but only a few, $\mathcal{O}(nk)$, state variables.

Whereas developing algorithms for the online setting is more challenging than that for the offline, it is necessary both for data analysis and for modeling biological neural networks. The size of modern datasets may exceed that of available RAM and/or the output must be computed before the dataset is fully streamed. Biological neural networks operating on the data streamed by the sensory organs are incapable of storing any significant fraction of it and compute the output on the fly.

Pehlevan et al. (2015) gave a derivation of a neural online algorithm for PSP, starting from the original similarity matching cost function (2). Here, instead, we start from the min-max form of similarity matching (6) and end up with a class of algorithms that reduce to the algorithm of Pehlevan et al. (2015) for special choices of learning rates. Our main contribution, however, is that the current derivation is much more intuitive and simpler, with insights to why similarity matching leads to local learning rules.

We start by rewriting the min-max PSP objective (6) as a sum of time-separable terms that can be optimized independently:

$$\min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \max_{\mathbf{M}\in\mathbb{R}^{k\times k}} \frac{1}{T}\sum_{t=1}^{T} l_{PSP,t}(\mathbf{W},\mathbf{M}), \tag{16}$$

where

$$l_{PSP,t}(\mathbf{W},\mathbf{M}) \equiv 2\mathrm{Tr}\left(\mathbf{W}^\top\mathbf{W}\right) - \mathrm{Tr}\left(\mathbf{M}^\top\mathbf{M}\right) + \min_{\mathbf{y}_t\in\mathbb{R}^{k\times 1}} l_t(\mathbf{W},\mathbf{M},\mathbf{y}_t), \tag{17}$$

and

$$l_t(\mathbf{W},\mathbf{M},\mathbf{y}_t) = -4\mathbf{x}_t^\top\mathbf{W}^\top\mathbf{y}_t + 2\mathbf{y}_t^\top\mathbf{M}\mathbf{y}_t. \tag{18}$$

This separation in time is a benefit of the min-max PSP objective (6), and leads to a natural way to derive an online algorithm that was not available for the original similarity matching cost function (2).

To solve the optimization problem, Eq. (16), in the online setting, we optimize sequentially each $l_{PSP,t}$. For each t, first, minimize Eq.(18) with respect to $\mathbf{y}_t$ while keeping $\mathbf{W}_t$ and $\mathbf{M}_t$ fixed. Second, make a gradient descent-ascent step with respect to $\mathbf{W}_t$ and $\mathbf{M}_t$ for fixed $\mathbf{Y}$:

$$\begin{aligned}\mathbf{W}_{t+1} &= \mathbf{W}_t - \eta_t\frac{\partial l_{PSP,t}(\mathbf{W}_t,\mathbf{M}_t)}{\partial\mathbf{W}_t},\\ \mathbf{M}_{t+1} &= \mathbf{M}_t + \frac{\eta_t}{\tau}\frac{\partial l_{PSP,t}(\mathbf{W}_t,\mathbf{M}_t)}{\partial\mathbf{M}_t},\end{aligned} \tag{19}$$

where $0 < \eta_t < 1$ is the $\mathbf{W}$ learning rate and $\tau > 0$ is the ratio of $\mathbf{W}$ and $\mathbf{M}$ learning rates. As before, Proposition 2 (Appendix B) ensures that the online gradient descent-ascent updates, Eq. (19), follow from alternating optimization (Olshausen et al., 1996; Olshausen and Field, 1997; Arora et al., 2015) of $l_{PSP,t}$.

Algorithm 2 Online min-max PSP

1: At $t = 0$, initialize the synaptic weight matrices, $\mathbf{W}_1$ and $\mathbf{M}_1$. $\mathbf{M}_1$ must be symmetric and positive definite.
2: Repeat for each $t = 1, \dots T$
3: Receive input $\mathbf{x}_t$
4: Neural activity: Run until convergence

$$\frac{d\mathbf{y}_t(\gamma)}{d\gamma} = \mathbf{W}_t\mathbf{x}_t - \mathbf{M}_t\mathbf{y}_t. \tag{20}$$

5: Plasticity: Update synaptic weight matrices,

$$\begin{aligned}\mathbf{W}_{t+1} &= \mathbf{W}_t + 2\eta_t\left(\mathbf{y}_t\mathbf{x}_t^\top - \mathbf{W}_t\right),\\ \mathbf{M}_{t+1} &= \mathbf{M}_t + \frac{\eta_t}{\tau}\left(\mathbf{y}_t\mathbf{y}_t^\top - \mathbf{M}_t\right).\end{aligned} \tag{21}$$

Algorithm 2 can be implemented by a biologically plausible neural network. The dynamics (20) corresponds to neural activity in a recurrent circuit, where $\mathbf{W}_t$ is the feedforward synaptic weight matrix and $-\mathbf{M}_t$ is the lateral synaptic weight matrix, Fig. 1A. Since $\mathbf{M}_t$ is always positive definite, Eq. (18) is a Lyapunov function for neural activity. Hence the dynamics is guaranteed to converge to a unique fixed point, $\mathbf{y}_t = \mathbf{M}_t^{-1}\mathbf{W}_t\mathbf{x}_t$, where matrix inversion is computed iteratively in a distributed manner.

Updates of covariance matrices, Eq. (21), can be interpreted as synaptic learning rules: Hebbian for feedforward and anti-Hebbian (due to the "$-$" sign in (20)) for lateral synaptic weights. Importantly, these rules are local - the weight of each synapse depends only on the activity of the pair of neurons that synapse connects - and therefore biologically plausible.

Even requiring full optimization with respect to $\mathbf{y}_t$ vs. a gradient step with respect to $\mathbf{W}_t$ and $\mathbf{M}_t$ may have a biological justification. As neural activity dynamics is typically faster than synaptic plasticity, it may settle before the arrival of the next input.

To see why similarity matching leads to local learning rules let us consider Eqs. (6) and (16). Aside from separating in time, useful for derivation of online learning rules, $L_{PSP}(\mathbf{W}, \mathbf{M}, \mathbf{Y})$ also separates in synaptic weights and their pre- and postsynaptic neural activities,

$$L_{PSP}(\mathbf{W}, \mathbf{M}, \mathbf{Y}) = \sum_t \left[\sum_{ij} \left(2W_{ij}^2 - 4W_{ij} x_{t,j} y_{t,i} \right) - \sum_{ij} \left(M_{ij}^2 + 2M_{ij} y_{t,j} y_{t,i} \right) \right]. \tag{22}$$

Therefore, a derivative with respect to a synaptic weight depends only on the quantities accessible to the synapse.

Finally, we address two potential criticisms of the neural PSP algorithm. First is the existence of autapses, i.e. self-coupling of neurons, in our network manifested in nonzero diagonals of the lateral connectivity matrix, $\mathbf{M}$, Fig 1A. Whereas autapses are encountered in the brain, they are rarely seen in principal neurons (Ikeda and Bekkers, 2006). Second is the symmetry of lateral synaptic weights in our network which is not observed experimentally. We derive an autapse-free network architecture (zeros on the diagonal of the lateral synaptic weight matrix $\mathbf{M}_t$) with asymmetric lateral connectivity, Fig 1B, by using coordinate descent (Pehlevan et al., 2015) in place of gradient descent in the neural dynamics stage (20) (see Appendix F). The resulting algorithm produces the same outputs as the current algorithm and for the special case $\tau = 1/2$ and $\eta_t = \eta/2$, reduces to the algorithm with "forgetting" of Pehlevan et al. (2015).

3 From constrained similarity matching to Hebbian/anti-Hebbian networks for PSW

The variable substitution method we introduced in the previous section can be applied to other computational objectives in order to derive neural networks with local learning rules. To give an example, we derive a neural network for PSW, which can be formulated as a constrained similarity matching problem. This example also illustrates how an optimization constraint can be implemented by biological mechanisms.

3.1 Derivation of PSW from constrained similarity matching

The PSW problem is closely related to PSP: project centered input data samples onto the principal subspace ($k \leq n$), and "spherize" the data in the subspace so that the variances in all directions are 1. To derive a neural PSW algorithm, we use the similarity matching objective with an additional constraint:

$$\text{PSW}: \qquad \min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} \frac{1}{T^2}\left\|\mathbf{X}^\top\mathbf{X}-\mathbf{Y}^\top\mathbf{Y}\right\|_F^2, \qquad \text{s.t.} \quad \frac{1}{T}\mathbf{Y}\mathbf{Y}^\top=\mathbf{I} \tag{23}$$

We rewrite Eq. (23) by expanding the Frobenius norm squared and dropping the $\mathrm{Tr}\left(\mathbf{Y}^\top\mathbf{Y}\mathbf{Y}^\top\mathbf{Y}\right)$ term, which is constant under the constraint, thus reducing (23) to a constrained similarity alignment problem:

$$\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} \left(-\frac{1}{T^2}\mathbf{X}^\top\mathbf{X}\mathbf{Y}^\top\mathbf{Y}\right), \qquad \text{s.t.} \quad \frac{1}{T}\mathbf{Y}\mathbf{Y}^\top=\mathbf{I}. \tag{24}$$

To see that objective (24) is optimized by the PSW, first, substitute a Singular Value Decomposition (SVD) for matrices $\mathbf{X}$ and $\mathbf{Y}$ and note that the alignment is maximized by matching right singular vectors of $\mathbf{Y}$ to $\mathbf{X}$ and rotating to the diagonal basis (for a rigorous proof see (Pehlevan and Chklovskii, 2015)). Since the squared singular values of $\mathbf{Y}$ equal unity, the objective (24) is reduced to a summation of k squared singular values of $\mathbf{X}$ and is optimized by choosing the top k. Then, $\mathbf{Y}$ is given by the top k right singular vectors of $\mathbf{X}$ scaled by $\sqrt{T}$. As before, objective (24) is invariant to the left-multiplication of $\mathbf{Y}$ by an orthogonal matrix and, therefore, has infinitely many degenerate solutions.

Next, we derive a mixed PSW objective from Eq. (24) by introducing two new dynamical variable matrices: the input-output correlation matrix, $\mathbf{W}=\frac{1}{T}\mathbf{X}\mathbf{Y}^\top$, and the Lagrange multiplier matrix, $\mathbf{M}$, for the whitening constraint:

$$\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} \min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \max_{\mathbf{M}\in\mathbb{R}^{k\times k}} L_{PSW}(\mathbf{W},\mathbf{M},\mathbf{Y}), \tag{25}$$

where

$$L_{PSW}(\mathbf{W}, \mathbf{M}, \mathbf{Y}) \equiv -\frac{2}{T}\mathrm{Tr}\left(\mathbf{X}^\top \mathbf{W}^\top \mathbf{Y}\right) + \mathrm{Tr}\left(\mathbf{W}^\top \mathbf{W}\right) + \mathrm{Tr}\left(\mathbf{M}\left(\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top - \mathbf{I}\right)\right). \tag{26}$$

To see that Eq. (26) is equivalent to Eq. (24), find optimal $\mathbf{W}^* = \frac{1}{T}\mathbf{Y}\mathbf{X}^\top$ by setting the corresponding derivatives of the objective (26) to zero. Then, substitute $\mathbf{W}^*$ into Eq. (26) to obtain the Lagrangian of Eq. (24).

Finally, we exchange the order of minimization with respect to $\mathbf{Y}$ and $\mathbf{W}$ as well as the order of minimization with respect to $\mathbf{Y}$ and maximization with respect to $\mathbf{M}$ in Eq. (26) (see Proposition 5 in Appendix D for a proof). Then, we arrive at the following min-max optimization problem with a mixed objective:

$$\min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \max_{\mathbf{M}\in\mathbb{R}^{k\times k}} \min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} L_{PSW}(\mathbf{W}, \mathbf{M}, \mathbf{Y}), \tag{27}$$

where $L_{PSW}(\mathbf{W}, \mathbf{M}, \mathbf{Y})$ is defined in Eq. (26).

3.2 Offline PSW algorithm

Next, we give an offline algorithm for the PSW problem, using the alternating optimization procedure as before. We solve Eq. (27) by, first, optimizing with respect to $\mathbf{Y}$ for fixed $\mathbf{W}$ and $\mathbf{M}$ and, second, making a gradient descent-ascent step with respect to $\mathbf{W}$ and $\mathbf{M}$ while keeping $\mathbf{Y}$ fixed[2]. We arrive at the following algorithm:

[2]This alternating optimization is identical to a gradient descent-ascent (see Proposition 2 in Appendix B) in $\mathbf{W}$ and $\mathbf{M}$ on the objective:

$$l_{PSW}(\mathbf{W}, \mathbf{M}) \equiv \min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} L_{PSW}(\mathbf{W}, \mathbf{M}, \mathbf{Y}).$$

Algorithm 3 Offline min-max PSW

1: Initialize $\mathbf{W}$. Initialize $\mathbf{M}$ as a positive definite matrix.
2: Iterate until convergence:
3: Minimize Eq. (26) with respect to $\mathbf{Y}$, keeping $\mathbf{W}$ and $\mathbf{M}$ fixed:

$$\mathbf{Y} = \mathbf{M}^{-1}\mathbf{W}\mathbf{X}. \tag{28}$$

4: Perform a gradient descent-ascent step with respect to $\mathbf{W}$ and $\mathbf{M}$ for a fixed $\mathbf{Y}$:

$$\begin{aligned} \mathbf{W} &\longleftarrow \mathbf{W} + 2\eta \left(\frac{1}{T}\mathbf{Y}\mathbf{X}^\top - \mathbf{W} \right), \\ \mathbf{M} &\longleftarrow \mathbf{M} + \frac{\eta}{\tau} \left(\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top - \mathbf{I} \right). \end{aligned} \tag{29}$$

where the step size, $0 < \eta < 1$, may depend on the iteration.

Convergence of Algorithm 3 requires the input covariance matrix, $\mathbf{C}$, to have at least k non-zero eigenvalues. Otherwise, a consistent solution cannot be found because update (29) forces $\mathbf{Y}$ to be full-rank while Eq. (28) lowers its rank.

3.3 Linearly stable fixed points of Algorithm 3 correspond to PSW

Here we claim that convergence of Algorithm 3 to fixed $\mathbf{W}$ and $\mathbf{M}$ implies PSW of $\mathbf{X}$. In the limit of small learning rate, the gradient descent-ascent dynamics can be approximated with the system of differential equations:

$$\begin{aligned} \mathbf{Y}(t) &= \mathbf{M}^{-1}(t)\mathbf{W}(t)\mathbf{X}, \\ \frac{d\mathbf{W}(t)}{dt} &= \frac{2}{T}\mathbf{Y}(t)\mathbf{X}^\top - 2\mathbf{W}(t), \\ \tau\frac{d\mathbf{M}(t)}{dt} &= \frac{1}{T}\mathbf{Y}(t)\mathbf{Y}(t)^\top - \mathbf{I}(t), \end{aligned} \tag{30}$$

where t is now the time index for gradient descent-ascent dynamics. We again define the neural filter matrix $\mathbf{F} = \mathbf{M}^{-1}\mathbf{W}$.

Theorem 2. *Fixed points of the dynamical system* (30) *have the following properties:*

1. *The outputs are whitened, i.e.* $\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top = \mathbf{I}$.
2. *The neural filters span a k-dimensional subspace in $\mathbb{R}^n$ which is spanned by some k eigenvectors of the input covariance matrix.*

3. *Stability of the fixed point requires that the neural filters span the* **principal** *subspace of* $\mathbf{X}$.

4. *Suppose the neural filters span the principal subspace. This fixed point is linearly stable if and only if*

$$\tau < \frac{\sigma_i + \sigma_j}{2\left(\sigma_i - \sigma_j\right)^2} \tag{31}$$

for all (i, j) *pairs,* $i \neq j$. *By linear stability we mean that linear perturbations of* $\mathbf{W}$ *and* $\mathbf{M}$ *converge to a rotation of the original neural filters within the principal subspace.*

Proof. See Appendix E. □

Based on Theorem 2 we claim that, provided Algorithm 3 converges, this fixed point corresponds to a PSW of input data. Unlike the PSP case, the neural filters are not orthonormal.

3.4 Online algorithm for PSW

As before, we start by rewriting the min-max PSW objective (27) as a sum of time-separable terms that can be optimized independently:

$$\min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \max_{\mathbf{M}\in\mathbb{R}^{k\times k}} \frac{1}{T}\sum_{t=1}^{T} l_{PSW,t}(\mathbf{W}, \mathbf{M}). \tag{32}$$

where

$$l_{PSW,t}(\mathbf{W}, \mathbf{M}) \equiv \operatorname{Tr}\left(\mathbf{W}^\top \mathbf{W}\right) - \operatorname{Tr}\left(\mathbf{M}\right) + \frac{1}{2}\min_{\mathbf{y}_t\in\mathbb{R}^{k\times 1}} l_t(\mathbf{W}, \mathbf{M}, \mathbf{y}_t). \tag{33}$$

and $l_t(\mathbf{W}, \mathbf{M}, \mathbf{y}_t)$ is defined in Eq. (18). In the online setting, Eq. (32) can be optimized by sequentially minimizing each $l_{PSW,t}$. For each t, first, minimize (18) with respect to $\mathbf{y}_t$ for fixed $\mathbf{W}_t$ and $\mathbf{M}_t$, second, update $\mathbf{W}_t$ and $\mathbf{M}_t$ according to a gradient descent-ascent step for fixed $\mathbf{y}_t$:

$$\begin{aligned}
\mathbf{W}_{t+1} &= \mathbf{W}_t - \eta_t \frac{\partial l_{PSW,t}(\mathbf{W}_t, \mathbf{M}_t)}{\mathbf{W}_t}, \\
\mathbf{M}_{t+1} &= \mathbf{M}_t + \frac{\eta_t}{\tau}\frac{\partial l_{PSW,t}(\mathbf{W}_t, \mathbf{M}_t)}{\mathbf{M}_t},
\end{aligned} \tag{34}$$

where $0 < \eta_t < 1$ is the $\mathbf{W}$ learning rate and $\tau > 0$ is the ratio of $\mathbf{W}$ and $\mathbf{M}$ learning rates.

As before, Proposition 2 ensures that the online gradient descent-ascent updates, Eq. (34), follow from alternating optimization (Olshausen et al., 1996; Olshausen and Field, 1997; Arora et al., 2015) of $l_{PSW,t}$.

Algorithm 4 Online min-max PSW

1: At $t = 0$, initialize the synaptic weight matrices, $\mathbf{W}_1$ and $\mathbf{M}_1$. $\mathbf{M}_1$ must be symmetric and positive definite.
2: Repeat for each $t = 1, \dots, T$
3: Receive input $\mathbf{x}_t$
4: Neural activity: Run until convergence

$$\frac{d\mathbf{y}_t(\gamma)}{d\gamma} = \mathbf{W}_t\mathbf{x}_t - \mathbf{M}_t\mathbf{y}_t. \tag{35}$$

5: Plasticity: Update synaptic weight matrices,

$$\begin{aligned}\mathbf{W}_{t+1} &= \mathbf{W}_t + 2\eta_{W,t}\left(\mathbf{y}_t\mathbf{x}_t^\top - \mathbf{W}_t\right),\\ \mathbf{M}_{t+1} &= \mathbf{M}_t + \eta_{M,t}\left(\mathbf{y}_t\mathbf{y}_t^\top - \mathbf{I}_t\right).\end{aligned} \tag{36}$$

Algorithm 4 can be implemented by a biologically plausible single-layer neural network with lateral connections as in Algorithm 2, Fig. 1A. Updates to synaptic weights, Eq. (36), are local, Hebbian/anti-Hebbian plasticity rules. An autapse-free network architecture, Fig 1B, may be obtained using coordinate descent (Pehlevan et al., 2015) in place of gradient descent in the neural dynamics stage (35) (see Appendix G).

The lateral connections here are the Lagrange multipliers introduced in the offline problem, Eq. (26). In the PSP network, they resulted from a variable transformation of the output covariance matrix. This difference caries over to the learning rules, where in Algorithm 4, the lateral learning rule is enforcing the whitening of the output, but in Algorithm 2, the lateral learning rule sets the lateral weight matrix to the output covariance matrix.

4 Game theoretical interpretation of Hebbian/anti-Hebbian learning

In the original similarity matching objective, Eq. (2), the only variables are neuronal activities which, at the optimum, represent principal components. In Section 2, we rewrote these objectives by introducing matrices $\mathbf{W}$ and $\mathbf{M}$ corresponding

to synaptic connection weights, Eq. (5). Here, we eliminate neural activity variables altogether and arrive at a min-max formulation in terms of feedforward, $\mathbf{W}$, and lateral, $\mathbf{M}$, connection weight matrices only. This formulation lends itself to a game-theoretical interpretation.

Since in the offline PSP setting, optimal $\mathbf{M}^*$ in Eq. (6) is an invertible matrix (because $\mathbf{M}^* = \frac{1}{T}\mathbf{Y}^*\mathbf{Y}^{*\top}$, see also Appendix A), we can restrict our optimization to invertible matrices, $\mathbf{M}$, only. Then, we can optimize objective (5) with respect to $\mathbf{Y}$ and substitute its optimal value $\mathbf{Y}^* = \mathbf{M}^{-1}\mathbf{W}\mathbf{X}$ into (5) and (6) to obtain:

$$\min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \max_{\mathbf{M}\in\mathbb{R}^{k\times k}} -\frac{2}{T}\mathrm{Tr}\left(\mathbf{X}^\top\mathbf{W}^\top\mathbf{M}^{-1}\mathbf{W}\mathbf{X}\right) + 2\mathrm{Tr}\left(\mathbf{W}^\top\mathbf{W}\right) - \mathrm{Tr}\left(\mathbf{M}^\top\mathbf{M}\right), \\ \text{s.t. } \mathbf{M} \text{ is invertible.} \tag{37}$$

This min-max objective admits a game-theoretical interpretation where feedforward, $\mathbf{W}$, and lateral, $\mathbf{M}$, synaptic weight matrices oppose each other. To reduce the objective, feedforward synaptic weight vectors of each output neuron attempt to align with the direction of maximum variance of input data. However, if this was the only driving force then all output neurons would learn the same synaptic weight vectors and represent the same top principal component. At the same time, linear dependency between different feedforward synaptic weight vectors can be exploited by the lateral synaptic weights to increase the objective by cancelling the contributions of different components. To avoid this, the feedforward synaptic weight vectors become linearly independent and span the principal subspace.

A similar interpretation can be given for PSW, where feedforward, $\mathbf{W}$, and lateral, $\mathbf{M}$, synaptic weight matrices oppose each other adversarially.

5 Novel formulations of dimensionality reduction using fractional exponents

In this section, we point to a new class of dimensionality reduction objective functions that naturally follow from the min-max objectives (5) and (6). Eliminating both the neural activity variables, $\mathbf{Y}$, and the lateral connection weight matrix, $\mathbf{M}$, we arrive at optimization problems in terms of the feedforward weight matrix, $\mathbf{W}$, only. The rows of optimal $\mathbf{W}$ form a non-orthogonal basis of the principal subspace. Such formulations of principal subspace problems involve fractional exponents of matrices and, to the best of our knowledge, have not been proposed previously.

By replacing $\max_{\mathbf{M}} \min_{\mathbf{Y}}$ optimization in the min-max PSP objective, Eq. (6), by its saddle point value (see Proposition 1 in Appendix A) we find the following

objective expressed solely in terms of $\mathbf{W}$:

$$\min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \operatorname{Tr}\left(-\frac{3}{T^{2/3}}\left(\mathbf{W}\mathbf{X}\mathbf{X}^\top\mathbf{W}^\top\right)^{2/3}+2\mathbf{W}\mathbf{W}^\top\right), \tag{38}$$

The rows of the optimal $\mathbf{W}$ are not principal eigenvectors, rather the rowspace of $\mathbf{W}$ spans the principal subspace.

By replacing $\max_{\mathbf{M}}\min_{\mathbf{Y}}$ optimization in the min-max PSW objective, Eq. (27), by its optimal value (see Proposition 5 in Appendix D):

$$\min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \operatorname{Tr}\left(-\frac{2}{T^{1/2}}\left(\mathbf{W}\mathbf{X}\mathbf{X}^\top\mathbf{W}^\top\right)^{1/2}+\mathbf{W}\mathbf{W}^\top\right). \tag{39}$$

As before, the rows of the optimal $\mathbf{W}$ are not principal eigenvectors, rather the rowspace of $\mathbf{W}$ spans the principal eigenspace.

We observe that the only material difference between Eqs. (38) and (39) is in the value of the fractional exponent. Based on this, we conjecture that any objective function of such form with a fractional exponent from a continuous range is optimized by $\mathbf{W}$ spanning the principal subspace. Such solutions would differ in the eigenvalues associated with the corresponding components.

A supporting argument for our conjecture comes from the work of Miao and Hua (1998), which studied the cost

$$\min_{\mathbf{W}\in\mathbb{R}^{k\times n}} \operatorname{Tr}\left(-\log\left(\mathbf{W}\mathbf{X}\mathbf{X}^\top\mathbf{W}^\top\right)+\mathbf{W}\mathbf{W}^\top\right). \tag{40}$$

Eq. 40 can be seen as a limiting case of our conjecture, where the fractional exponent goes to zero. Indeed, Miao and Hua (1998) proved that the rows of optimal $\mathbf{W}$ are an orthonormal basis for the principal eigenspace.

6 Numerical experiments

Next, we test our findings using a simple artificial dataset. We generated an $n = 10$ dimensional dataset and we simulated our offline and online algorithms to reduce this dataset to $k = 3$ dimensions, using different values of the parameter τ. The results are plotted in Figs. 2, 3, 4 and 5 along with details of the simulations in the figures' caption.

Consistent with Theorems 1 and 2, small perturbations to PSP and PSW fixed points decayed (solid lines) or grew (dashed lines) depending on the value of τ, Fig. 2A. Offline simulations that start from random initial conditions converged to the PSP (or the PSW) solution if the fixed point was linearly stable, Fig. 2B. Interestingly, the online algorithms' performance were very close to that of the offline, Fig. 2C.

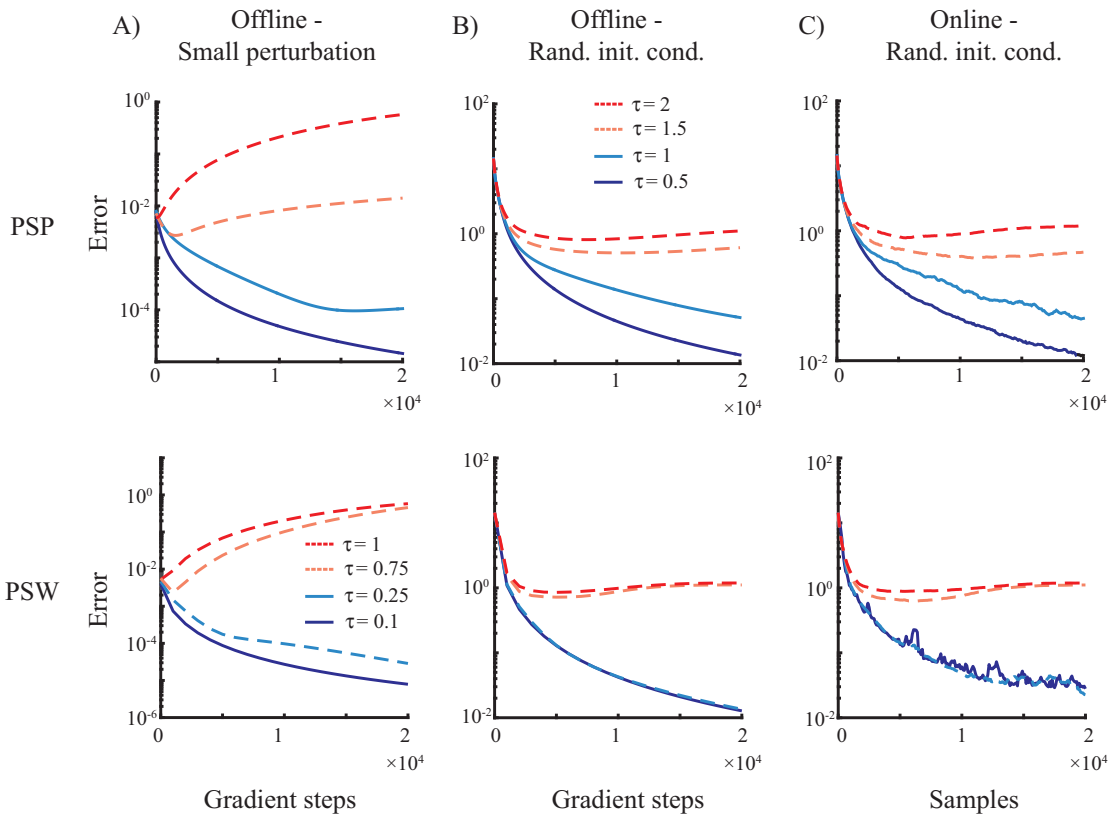

Figure 2: Demonstration of the stability of the PSP (top row) and PSW (bottom row) algorithms. We constructed an $n = 10$ by $T = 2000$ data matrix $\mathbf{X}$ from its SVD, where the left and right singular vectors are chosen randomly, the top three singular values are set to $\{\sqrt{3T}, \sqrt{2T}, \sqrt{T}\}$ and the rest of the singular values are chosen uniformly in $[0, 0.1\sqrt{T}]$. Learning rates were $\eta_t = 1/(10^3 + t)$. Errors were defined using deviation of the neural filters from their optimal values (Pehlevan et al., 2015). Let $\mathbf{U}$ be the 10×3 matrix whose columns are the top 3 left singular vectors of $\mathbf{X}$. PSP error: $\left\|\mathbf{F}(t)^\top \mathbf{F}(t) - \mathbf{U}\mathbf{U}^\top\right\|_F$, PSW error: $\left\|\mathbf{F}(t)^\top \mathbf{F}(t) - \mathbf{U}\mathbf{S}\mathbf{U}^\top\right\|_F$, with $\mathbf{S} = \mathrm{diag}\left([1/3, 1/2, 1]\right)$ in MATLAB notation. Solid (dashed) lines indicate linearly stable (unstable) choices of τ. A) Small perturbations to the fixed point. $\mathbf{W}$ and $\mathbf{M}$ matrices were initialized by adding a random Gaussian variable, $\mathcal{N}(0, 10^{-6})$, elementwise to their fixed point values. B) Offline algorithm, initialized with random $\mathbf{W}$ and $\mathbf{M}$ matrices. C) Online algorithm, initialized with the same initial condition as in B). A random column of $\mathbf{X}$ is processed at each time.

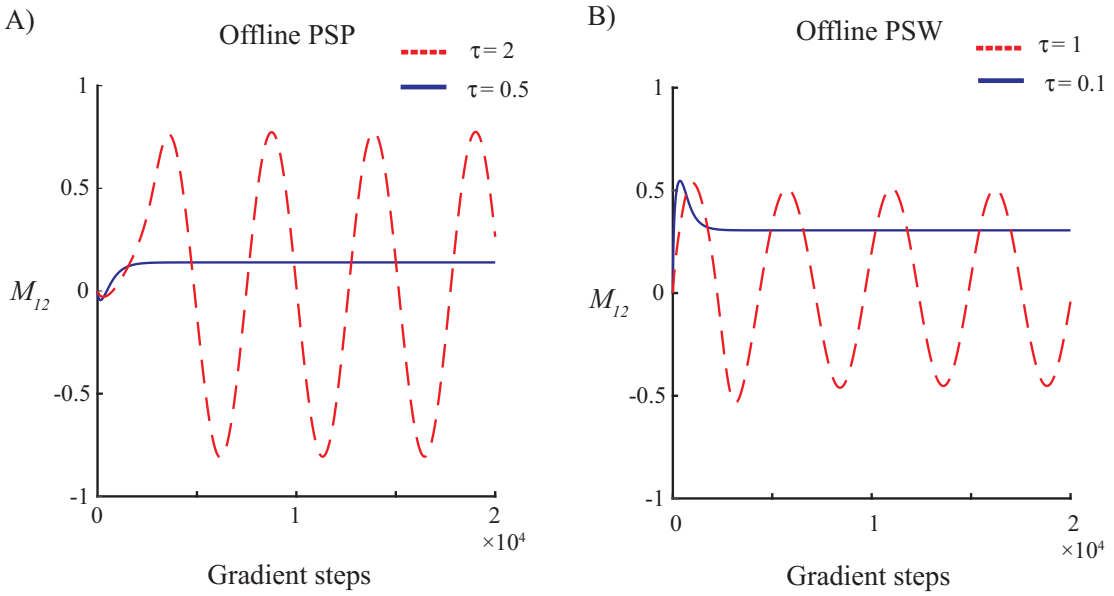

Figure 3: Evolution of a synaptic weight. Same dataset was used as in Fig. 2. $\eta = 10^{-3}$.

The error for linearly unstable simulations in Fig. 2 saturates rather than blowing up. This may seem at odds with Theorems 1 and 2, which stated that if there is a stable fixed point of the dynamics, it should be the PSP/PSW solution. A closer look resolves this dilemma. In Fig. 3, we plot the evolution of an element of the $\mathbf{M}$ matrix in the offline algorithms for stable and unstable choices of τ. When the principal subspace is linearly unstable, the synaptic weights exhibit undamped oscillations. The dynamics seems to be confined to a manifold with a fixed distance (in terms of the error metric) from the principal subspace. That the error does not grow to infinity is a result of the stabilizing effect of min-max antagonism of the synaptic weights. Online algorithms behave similarly.

Next, we studied in detail the effect of τ parameter on the convergence. In the offline algorithm, we plot the error after a fixed number of gradient steps, as a function of τ. For PSP, there is an optimal τ. Decreasing τ beyond the optimal value doesn't lead to a degradation in performance, however increasing it leads to a rapid increase in the error. For PSW, there is a plateau of low error for low values of τ but a rapid increase as one approaches the linear instability threshold. Online algorithms behave similarly.

Finally, we compared the performance of our online PSP algorithm to neural PSP algorithms with heuristic learning rules such as the Subspace Network (Oja, 1989) and the Generalized Hebbian Algorithm (GHA) (Sanger, 1989), on the same dataset. We found that our algorithm converges much faster (Fig. 5). Previously,

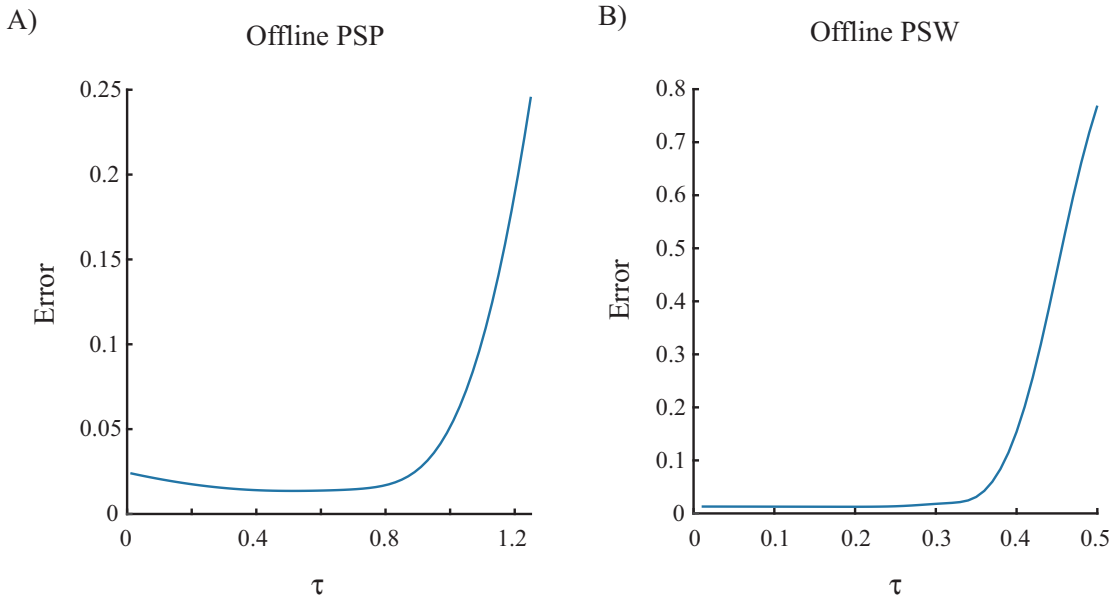

Figure 4: Effect of τ of performance. Error after 2×10^4 gradient steps are plotted as a function of different choices of τ. Same dataset was used as in Fig. 2 with same network initalization and learning rates. Both curves start from $\tau = 0.01$ and go to the maximum value allowed for linear stability.

the original similarity matching network (Pehlevan et al., 2015), which is a special case of the online PSP algorithm of this paper, was shown to converge faster than the APEX (Kung et al., 1994) and Földiak's (Földiak, 1989) networks.

7 Discussion

In this paper, through transparent variable substitutions, we demonstrated why biologically plausible neural networks can be derived from similarity matching objectives, mathematically formalized the adversarial relationship between Hebbian feedforward and anti-Hebbian lateral connections using min-max optimization lending itself to a game-theoretical interpretation, and formulated dimensionality reduction tasks as optimizations of fractional powers of matrices. The formalism we developed should generalize to unsupervised tasks other than dimensionality reduction and could provide a theoretical foundation for both natural and artificial neural networks.

In comparing our networks with biological ones, most importantly, our networks rely only on local learning rules that can be implemented by synaptic plasticity. While Hebbian learning is famously observed in neural circuits (Bliss and Lømo,

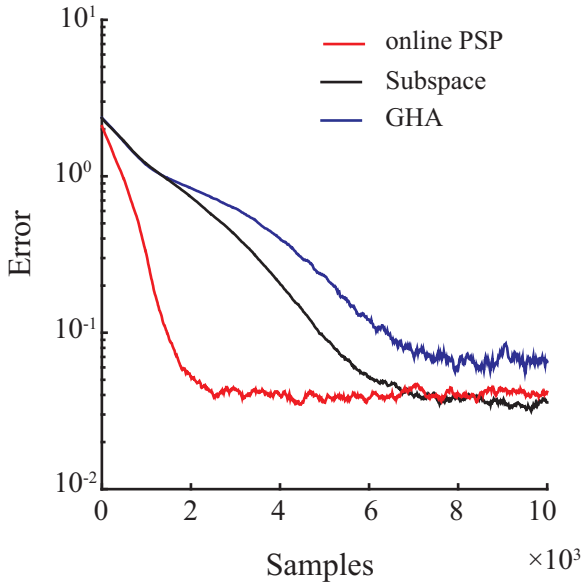

Figure 5: Comparison of the online PSP algorithm with the Subspace Network (Oja, 1989) and the GHA (Sanger, 1989). The dataset and the error metric is as in Fig. 2. For fairness of comparison, the learning rates in all networks were set to $\eta = 10^{-3}$. $\tau = 1/2$ for the online PSP algorithm . Feedforward connectivity matrices were initialized randomly. For the online PSP algorithm, lateral connectivity matrix was initialized to the identity matrix. Curves show averages over 10 trials.

1973; Bliss and Gardner-Medwin, 1973), our networks also require anti-Hebbian learning, which can be interpreted as the long-term potentiation of inhibitory postsynaptic potentials. Experimentally, such long-term potentiation can arise from pairing action potentials in inhibitory neurons with subthreshold depolarization of postsynaptic pyramidal neurons (Komatsu, 1994; Maffei et al., 2006). However, plasticity in inhibitory synapses does not have to be Hebbian, i.e. depend on the correlation between pre- and postsynaptic activity (Kullmann et al., 2012).

To make progress, we had to make several simplifications sacrificing biological realism. In particular, we assumed that neuronal activity is a continuous variable which would correspond to membrane depolarization (in graded potential neurons) or firing rate (in spiking neurons). We ignored the nonlinearity of the neuronal input-output function. Such linear regime could be implemented via a resting state bias (in graded potential neurons) or resting firing rate (in spiking neurons).

The applicability of our networks as models of biological networks can be judged by experimentally testing the following predictions. First, we predict a relationship between the feedforward and lateral synaptic weight matrices which could be tested using modern connectomics datasets. Second, we suggest that similarity of output activity matches that of the input which could be tested by neuronal population activity measurements using calcium imaging.

Often the choice of a learning rate is crucial to the learning performance of neural networks. Here, we encountered a nuanced case where the ratio of feedforward and lateral weights, τ, affects the learning performance significantly. First, there is a maximum value of such ratio, beyond which the principal subspace solution is linearly unstable. The maximum value depends on the principal eigenvalues, but for PSP, $\tau \leq 1/2$ is always linearly stable. For PSW there isn't an always safe choice. Having the same learning rates for feedforward and lateral weights, $\tau = 1$, may actually be unstable. Second, linear stability is not the only thing that affects performance. In simulations, for PSP, we observed that there is an optimal value of τ. For PSW, decreasing τ seems to increase performance until a plateau is reached. This difference between PSP and PSW may be attributed to the difference of origins of lateral connectivity. In PSW algorithms, lateral weights originate from Lagrange multipliers enforcing an optimization constraint. Low τ, meaning higher lateral learning rates, force the network to satisfy the constraint during the whole evolution of the algorithm.

Based on these observation, we can make practical suggestions for the τ parameter. For PSP, $\tau = 1/2$ seems to be a good choice, which is also preferred from another derivation of an online similarity matching algorithm (Pehlevan et al., 2015). For PSW, the smaller the τ, the better it is, although one should make sure that the lateral weight learning rate η/τ is still sufficiently small.

Acknowledgments

We thank Alex Genkin, Sebastian Seung, Mariano Tepper and Jonathan Zung for discussions.

A Proof of strong min-max property for PSP objective

Here we show that minimization with respect to $\mathbf{Y}$ and maximization with respect to $\mathbf{M}$ can be exchanged in Eq. (5). We will make use of the following min-max theorem (Boyd and Vandenberghe, 2004), for which we give a proof for completeness:

Theorem 3. *Let $f : \mathbb{R}^n \times \mathbb{R}^m \longrightarrow \mathbb{R}$. Suppose the saddle-point property holds, i.e. $\exists \mathbf{a}^* \in \mathbb{R}^n$, $\mathbf{b}^* \in \mathbb{R}^m$ such that $\forall \mathbf{a} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$*

$$f(\mathbf{a}^*, \mathbf{b}) \leq f(\mathbf{a}^*, \mathbf{b}^*) \leq f(\mathbf{a}, \mathbf{b}^*). \tag{41}$$

Then,

$$\max_{\mathbf{b}} \min_{\mathbf{a}} f(\mathbf{a}, \mathbf{b}) = \min_{\mathbf{a}} \max_{\mathbf{b}} f(\mathbf{a}, \mathbf{b}) = f(\mathbf{a}^*, \mathbf{b}^*). \tag{42}$$

Proof. $\forall \mathbf{c} \in \mathbb{R}^n$, $\min_{\mathbf{a}} \max_{\mathbf{b}} f(\mathbf{a}, \mathbf{b}) \leq \max_{\mathbf{b}} f(\mathbf{c}, \mathbf{b})$, which implies

$$\min_{\mathbf{a}} \max_{\mathbf{b}} f(\mathbf{a}, \mathbf{b}) \leq \max_{\mathbf{b}} f(\mathbf{a}^*, \mathbf{b}) = f(\mathbf{a}^*, \mathbf{b}^*) = \min_{\mathbf{a}} f(\mathbf{a}, \mathbf{b}^*) \leq \max_{\mathbf{b}} \min_{\mathbf{a}} f(\mathbf{a}, \mathbf{b}). \tag{43}$$

Since $\max_{\mathbf{b}} \min_{\mathbf{a}} f(\mathbf{a}, \mathbf{b}) \leq \min_{\mathbf{a}} \max_{\mathbf{b}} f(\mathbf{a}, \mathbf{b})$ is always true, we get an equality. □

Now, we present the main result of this section.

Proposition 1. *Define*

$$f(\mathbf{Y}, \mathbf{M}, \mathbf{A}) := \mathrm{Tr}\left(-\frac{4}{T}\mathbf{A}^\top \mathbf{Y} + \frac{2}{T}\mathbf{Y}^\top \mathbf{M}\mathbf{Y}\right) - \mathrm{Tr}\left(\mathbf{M}^\top \mathbf{M}\right), \tag{44}$$

where $\mathbf{Y}$, $\mathbf{M}$ and $\mathbf{A}$ are arbitrary sized, real-valued matrices. f obeys a strong min-max property:

$$\min_{\mathbf{Y}} \max_{\mathbf{M}} f(\mathbf{Y}, \mathbf{M}, \mathbf{A}) = \max_{\mathbf{M}} \min_{\mathbf{Y}} f(\mathbf{Y}, \mathbf{M}, \mathbf{A}) = -\frac{3}{T^{2/3}} \mathrm{Tr}\left(\left(\mathbf{A}\mathbf{A}^\top\right)^{2/3}\right). \tag{45}$$

Proof. We will show that the saddle-point property holds for Eq. (44). Then the result follows from Theorem 1.

If the saddle point exists, it is when $\nabla f = 0$,

$$\begin{aligned} \mathbf{M}^* &= \frac{1}{T}\mathbf{Y}^* {\mathbf{Y}^*}^\top, \\ \mathbf{M}^* \mathbf{Y}^* &= \mathbf{A}. \end{aligned} \tag{46}$$

Note that $\mathbf{M}^*$ is symmetric and positive semidefinite. Multiplying the first equation by $\mathbf{M}^*$ on the left and the right, and using the the second equation, we arrive at

$$\mathbf{M}^{*3} = \frac{1}{T}\mathbf{A}\mathbf{A}^\top. \tag{47}$$

Solutions to Eq. (46) are not unique, because $\mathbf{M}^*$ may not be invertible depending on $\mathbf{A}$. However, all solutions give the same value of f:

$$\begin{aligned} f\left(\mathbf{Y}^*, \mathbf{M}^*, \mathbf{A}\right) &= \mathrm{Tr}\left(-\frac{4}{T}\mathbf{A}^\top\mathbf{Y}^* + \frac{2}{T}\mathbf{Y}^{*\top}\mathbf{M}^*\mathbf{Y}^*\right) - \mathrm{Tr}\left(\mathbf{M}^{*2}\right) \\ &= \mathrm{Tr}\left(-\frac{4}{T}\mathbf{Y}^{*\top}\mathbf{M}^*\mathbf{Y}^* + \frac{2}{T}\mathbf{Y}^{*\top}\mathbf{M}^*\mathbf{Y}^*\right) - \mathrm{Tr}\left(\mathbf{M}^{*2}\right) \\ &= -3\mathrm{Tr}\left(\mathbf{M}^{*2}\right) = -\frac{3}{T^{2/3}}\mathrm{Tr}\left(\left(\mathbf{A}\mathbf{A}^\top\right)^{2/3}\right). \end{aligned} \tag{48}$$

Now, we check if the saddle-point property, Eq. (41), holds. The first inequality is satisfied:

$$\begin{aligned} f\left(\mathbf{Y}^*, \mathbf{M}^*, \mathbf{A}\right) - f\left(\mathbf{Y}^*, \mathbf{M}, \mathbf{A}\right) &= \mathrm{Tr}\left(\frac{2}{\mathrm{T}}\mathbf{Y}^{*\top}\left(\mathbf{M}^* - \mathbf{M}\right)\mathbf{Y}^*\right) - \mathrm{Tr}\left(\mathbf{M}^{*2}\right) + \mathrm{Tr}\left(\mathbf{M}^\top\mathbf{M}\right) \\ &= -2\,\mathrm{Tr}\left(\mathbf{M}^*\mathbf{M}\right) + \mathrm{Tr}\left(\mathbf{M}^{*2}\right) + \mathrm{Tr}\left(\mathbf{M}^\top\mathbf{M}\right) \\ &= \left\|\mathbf{M}^* - \mathbf{M}\right\|_F^2 \geq 0. \end{aligned} \tag{49}$$

The second inequality is also satisfied:

$$\begin{aligned} f\left(\mathbf{Y}, \mathbf{M}^*, \mathbf{A}\right) - f\left(\mathbf{Y}^*, \mathbf{M}^*, \mathbf{A}\right) &= \mathrm{Tr}\left(-\frac{4}{T}\mathbf{A}^\top\left(\mathbf{Y} - \mathbf{Y}^*\right) + \frac{2}{T}\mathbf{Y}^\top\mathbf{M}^*\mathbf{Y} - \frac{2}{T}\mathbf{Y}^{*\top}\mathbf{M}^*\mathbf{Y}^*\right) \\ &= \mathrm{Tr}\left(-\frac{4}{T}\mathbf{Y}^{*\top}\mathbf{M}^*\mathbf{Y} + \frac{2}{T}\mathbf{Y}^\top\mathbf{M}^*\mathbf{Y} + \frac{2}{T}\mathbf{Y}^{*\top}\mathbf{M}^*\mathbf{Y}^*\right) \\ &= \frac{2}{T}\,\mathrm{Tr}\left(\left(\mathbf{Y} - \mathbf{Y}^*\right)^\top\mathbf{M}^*\left(\mathbf{Y} - \mathbf{Y}^*\right)\right) \geq 0, \end{aligned} \tag{50}$$

where the last line follows form $\mathbf{M}^*$ being positive semidefinite.

Eq.s (49) and (50) show that the saddle-point property (41) holds, and therefore max and min can be exchanged and the value of f at the saddle-point is $f\left(\mathbf{Y}^*, \mathbf{M}^*, \mathbf{A}\right) = -\frac{3}{T^{2/3}}\mathrm{Tr}\left(\left(\mathbf{A}\mathbf{A}^\top\right)^{2/3}\right)$. □

B Taking a derivative using a chain rule

Proposition 2. *Suppose a differentiable, scalar function* $H(\mathbf{a}_1, \ldots, \mathbf{a}_m)$*, where* $\mathbf{a}_i \in \mathbb{R}^{d_i}$ *with arbitrary* d_i*. Assume a finite minimum with respect to* $\mathbf{a}_m$ *exists for a given set of* $\{\mathbf{a}_1, \ldots, \mathbf{a}_{m-1}\}$*:*

$$h(\mathbf{a}_1, \ldots, \mathbf{a}_{m-1}) = \min_{\mathbf{a}_m} H(\mathbf{a}_1, \ldots, \mathbf{a}_m), \tag{51}$$

and the optimal $\mathbf{a}_m^* = \arg\min_{\mathbf{a}_m} H(\mathbf{a}_1, \ldots, \mathbf{a}_m)$ *is a stationary point*

$$\left.\frac{\partial H}{\partial \mathbf{a}_m}\right|_{\{\mathbf{a}_1, \ldots, \mathbf{a}_{m-1}, \mathbf{a}_m^*\}} = 0. \tag{52}$$

Then, for $i = 1, \ldots, m-1$

$$\left.\frac{\partial h}{\partial \mathbf{a}_i}\right|_{\{\mathbf{a}_1, \ldots, \mathbf{a}_{m-1}\}} = \left.\frac{\partial H}{\partial \mathbf{a}_i}\right|_{\{\mathbf{a}_1, \ldots, \mathbf{a}_{m-1}, \mathbf{a}_m^*\}} \tag{53}$$

Proof. The result follows from application of the chain rule and the stationarity of the minimum:

$$\left.\frac{\partial h}{\partial \mathbf{a}_i}\right|_{\{\mathbf{a}_1, \ldots, \mathbf{a}_{m-1}\}} = \left.\frac{\partial H}{\partial \mathbf{a}_i}\right|_{\{\mathbf{a}_1, \ldots, \mathbf{a}_{m-1}, \mathbf{a}_m^*\}} + \left(\left.\frac{\partial H}{\partial \mathbf{a}_m}\right|_{\{\mathbf{a}_1, \ldots, \mathbf{a}_{m-1}, \mathbf{a}_m^*\}}\right)^\top \left.\frac{\partial \mathbf{a}_m}{\partial \mathbf{a}_i}\right|_{\{\mathbf{a}_1, \ldots, \mathbf{a}_{m-1}\}} \tag{54}$$

where the second term is zero due to Eq. (52). □

C Proof of Theorem 1

Here we prove Theorem 1 using methodology from (Pehlevan et al., 2015).

The fixed points of Eq. (11) are (using $^-$for fixed point):

$$\bar{\mathbf{W}} = \bar{\mathbf{F}}\mathbf{C}, \qquad \bar{\mathbf{M}} = \bar{\mathbf{F}}\mathbf{C}\bar{\mathbf{F}}^\top, \tag{55}$$

where $\mathbf{C}$ is the input covariance matrix defined as in Eq. (1).

C.1 Proof of item 1

The result follows from Eq.s (12) and (55):

$$\mathbf{I} = \bar{\mathbf{M}}^{-1}\bar{\mathbf{M}} = \bar{\mathbf{M}}^{-1}\bar{\mathbf{F}}\mathbf{C}\bar{\mathbf{F}}^\top = \bar{\mathbf{M}}^{-1}\bar{\mathbf{W}}\bar{\mathbf{F}}^\top = \bar{\mathbf{F}}\bar{\mathbf{F}}^\top \tag{56}$$

C.2 Proof of item 2

First note that at fixed points, $\bar{\mathbf{F}}^\top\bar{\mathbf{F}}$ and $\mathbf{C}$ commute:

$$\bar{\mathbf{F}}^\top\bar{\mathbf{F}}\mathbf{C} = \mathbf{C}\bar{\mathbf{F}}^\top\bar{\mathbf{F}}. \tag{57}$$

Proof. The result follows from Eq.s (12) and (55):

$$\bar{\mathbf{F}}^\top\bar{\mathbf{F}}\mathbf{C} = \bar{\mathbf{F}}^\top\bar{\mathbf{W}} = \bar{\mathbf{F}}^\top\bar{\mathbf{M}}\bar{\mathbf{F}} = \bar{\mathbf{W}}^\top\bar{\mathbf{F}} = \mathbf{C}\bar{\mathbf{F}}^\top\bar{\mathbf{F}}. \tag{58}$$

□

$\bar{\mathbf{F}}^\top\bar{\mathbf{F}}$ and $\mathbf{C}$ share the same eigenvectors, because they commute. Orthonormality of neural filters, Eq. (56), implies that the k rows of $\bar{\mathbf{F}}$ are degenerate eigenvectors of $\bar{\mathbf{F}}^\top\bar{\mathbf{F}}$ with unit eigenvalue. To see this: $\left(\bar{\mathbf{F}}^\top\bar{\mathbf{F}}\right)\bar{\mathbf{F}}^\top = \bar{\mathbf{F}}^\top$. Because the filters are degenerate, the corresponding k shared eigenvectors of $\mathbf{C}$ may not be the filters themselves but linear combinations of them. Nevertheless, the shared eigenvectors composed of filters span the same space as the filters.

Since we are interested in PSP, it is desirable that it is the top k eigenvectors of $\mathbf{C}$ that spans the filter space. A linear stability analysis around the fixed point reveals that any other combination is unstable, and that the PS is stable if τ is chosen appropriately.

C.3 Proof of item 3

Preliminaries

In order to perform a linear stability analysis, we linearize the system of equations (11) around the fixed point. Even though Eq. (11) depends on $\mathbf{W}$ and $\mathbf{M}$, we will find it convenient to change variables and work with $\mathbf{F}$ and $\mathbf{M}$ instead.

Using the relation $\mathbf{F} = \mathbf{M}^{-1}\mathbf{W}$, one can express linear perturbations of $\mathbf{F}$ around its fixed point, $\delta\mathbf{F}$, in terms of perturbations of $\mathbf{W}$ and $\mathbf{M}$:

$$\delta\mathbf{F} = \delta\left(\mathbf{M}^{-1}\right)\bar{\mathbf{W}} + \bar{\mathbf{M}}^{-1}\delta\mathbf{W} = -\bar{\mathbf{M}}^{-1}\delta\mathbf{M}\bar{\mathbf{F}} + \bar{\mathbf{M}}^{-1}\delta\mathbf{W} \tag{59}$$

Linearization of Eq. (11) gives:

$$\frac{d\delta\mathbf{W}}{dt} = 2\delta\mathbf{F}\mathbf{C} - 2\delta\mathbf{W}, \tag{60}$$

and

$$\tau\frac{d\delta\mathbf{M}}{dt} = \delta\mathbf{F}\mathbf{C}\bar{\mathbf{F}}^{\top} + \bar{\mathbf{F}}\mathbf{C}\delta\bar{\mathbf{F}}^{\top} - \delta\mathbf{M}. \tag{61}$$

Using these, we arrive at:

$$\frac{d\delta\mathbf{F}}{dt} = -\frac{1}{\tau}\bar{\mathbf{M}}^{-1}\left(\delta\mathbf{F}\mathbf{C}\bar{\mathbf{F}}^{\top} + \bar{\mathbf{F}}\mathbf{C}\delta\bar{\mathbf{F}}^{\top} + (2\tau-1)\delta\mathbf{M}\right)\bar{\mathbf{F}} + 2\bar{\mathbf{M}}^{-1}\delta\mathbf{F}\mathbf{C} - 2\delta\mathbf{F}. \tag{62}$$

Eq.s (61) and (62) define a closed system of equations.

It will be useful to decompose $\delta\mathbf{F}$ into components[3]:

$$\delta\mathbf{F} = \delta\mathbf{A}\bar{\mathbf{F}} + \delta\mathbf{S}\bar{\mathbf{F}} + \delta\mathbf{B}\bar{\mathbf{G}} \tag{63}$$

where $\delta\mathbf{A}$ is an $k \times k$ anti-symmetric matrix, $\delta\mathbf{S}$ is an $k \times k$ symmetric matrix and $\delta\mathbf{B}$ is an $k \times (n-k)$ matrix. $\bar{\mathbf{G}}$ is an $(n-k) \times n$ matrix with orthonormal rows, which are orthogonal to the rows of $\bar{\mathbf{F}}$. $\delta\mathbf{A}$ and $\delta\mathbf{S}$ are perturbations that keep the neural filters within the filter space. Anti-symmetric $\delta\mathbf{A}$ corresponds to rotations of filters within the filter space, preserving orthonormality. Symmetric $\delta\mathbf{S}$ destroys orthonormality. $\delta\mathbf{B}$ is a perturbation that takes the neural filters outside of the filter space.

Let $\mathbf{v}_{1,\ldots,n}$ be the eigenvectors $\mathbf{C}$ and $\sigma_{1,\ldots,n}$ be the corresponding eigenvalues. We label them such that $\bar{\mathbf{F}}$ spans the same space as the space spanned by the first k eigenvectors. We choose rows of $\bar{\mathbf{G}}$ to be the remaining eigenvectors, i.e. $\bar{\mathbf{G}}^{\top} := [\mathbf{v}_{k+1}, \ldots, \mathbf{v}_n]$. Note that, with this choice,

$$\sum_k C_{ik}\bar{G}^{\top}_{kj} = \sigma_{j+m}\bar{G}^{\top}_{ij}. \tag{64}$$

[3]see Lemma 3 in (Pehlevan et al., 2015) for a proof of why such a decomposition always exists.

Proof

The proof of item 3 in Theorem 1 follows from studying the stability of $\delta\mathbf{B}$ component.

Multiplying Eq. (62) on the right by $\bar{\mathbf{G}}^\top$, one arrives at a decoupled equation for $\delta\mathbf{B}$:

$$\frac{d\delta B_i^j}{dt} = \sum_m P_{im}^j \delta B_m^j, \qquad P_{im}^j := 2\left(\bar{M}_{im}^{-1}\sigma_{j+k} - \delta_{im}\right), \tag{65}$$

where for convenience we changed our notation to $\delta B_{kj} = \delta B_k^j$. For each j, the dynamics is linearly stable if all eigenvalues of all $\mathbf{P^j}$ are negative. In turn, this implies that for stability, eigenvalues of $\bar{\mathbf{M}}$ should be greater than $\sigma_{k+1,\ldots,n}$.

Eigenvalues of $\bar{\mathbf{M}}$ are:

$$\mathrm{eig}(\bar{\mathbf{M}}) = \{\sigma_1, \ldots, \sigma_k\}. \tag{66}$$

Proof. The eigenvalue equation

$$\bar{\mathbf{F}}\mathbf{C}\bar{\mathbf{F}}^\top\boldsymbol{\lambda} = \lambda\boldsymbol{\lambda} \tag{67}$$

implies that

$$\mathbf{C}\left(\bar{\mathbf{F}}^\top\boldsymbol{\lambda}\right) = \lambda\left(\bar{\mathbf{F}}^\top\boldsymbol{\lambda}\right), \tag{68}$$

which can be seen by multiplying Eq. (67) on the left by $\bar{\mathbf{F}}^\top$, using the commutation of $\bar{\mathbf{F}}^\top\bar{\mathbf{F}}$ and $\mathbf{C}$, and the orthonormality of neural filters. Further, orthonormality of neural filters implies:

$$\bar{\mathbf{F}}^\top\bar{\mathbf{F}}\left(\bar{\mathbf{F}}^\top\boldsymbol{\lambda}\right) = \left(\bar{\mathbf{F}}^\top\boldsymbol{\lambda}\right). \tag{69}$$

Then, $\left(\bar{\mathbf{F}}^\top\boldsymbol{\lambda}\right)$ is a shared eigenvector[4] between $\mathbf{C}$ and $\bar{\mathbf{F}}^\top\bar{\mathbf{F}}$. Shared eigenvectors of $\mathbf{C}$ with unit eigenvalue in $\bar{\mathbf{F}}^\top\bar{\mathbf{F}}$ are $\mathbf{v}_1, \ldots, \mathbf{v}_k$. Since the eigenvalue of $\left(\bar{\mathbf{F}}^\top\boldsymbol{\lambda}\right)$ with respect to $\bar{\mathbf{F}}^\top\bar{\mathbf{F}}$ is 1, $\bar{\mathbf{F}}^\top\boldsymbol{\lambda}$ must be one of $\mathbf{v}_1, \ldots, \mathbf{v}_k$. Then Eq. (68) implies that $\lambda = \{\sigma_1, \ldots, \sigma_k\}$ and

$$\mathrm{eig}(\bar{\mathbf{M}}) = \{\sigma_1, \ldots, \sigma_k\}. \tag{70}$$

□

Then, it follows that linear stability requires

$$\{\sigma_1, \ldots, \sigma_k\} > \{\sigma_{k+1}, \ldots, \sigma_n\}. \tag{71}$$

This proves our claim that if at the fixed point, the neural filters span a subspace other than the principal subspace, the fixed point is linearly unstable.

[4]One might worry that $\left(\bar{\mathbf{F}}^\top\boldsymbol{\lambda}\right) = \mathbf{0}$, but this would require $\bar{\mathbf{F}}\left(\bar{\mathbf{F}}^\top\boldsymbol{\lambda}\right) = \boldsymbol{\lambda} = \mathbf{0}$, which is a contradiction.

C.4 Proof of item 4

We now assume that the fixed point is the principal subspace. From item 3, we know that the $\delta\mathbf{B}$ perturbations are stable. The proof of item 4 in Theorem 1, follows from the linear stabilities of $\delta\mathbf{A}$ and $\delta\mathbf{S}$.

Multiplying Eq. (62) on the right by $\bar{\mathbf{F}}^\top$,

$$\frac{d\delta\mathbf{A}}{dt} + \frac{d\delta\mathbf{S}}{dt} = \left(2 - \frac{1}{\tau}\right)\left(\bar{\mathbf{M}}^{-1}\left(\delta\mathbf{A} + \delta\mathbf{S}\right)\bar{\mathbf{M}} - \bar{\mathbf{M}}^{-1}\delta\mathbf{M} - \delta\mathbf{A}\right) - \left(2 + \frac{1}{\tau}\right)\delta\mathbf{S}. \tag{72}$$

Unlike the case of $\delta\mathbf{B}$, this equation is coupled to $\delta\mathbf{M}$, whose dynamics, Eq. (61), reduces to:

$$\tau\frac{d\delta\mathbf{M}}{dt} = \left(\delta\mathbf{A} + \delta\mathbf{S}\right)\bar{\mathbf{M}} + \bar{\mathbf{M}}\left(-\delta\mathbf{A} + \delta\mathbf{S}\right) - \delta\mathbf{M}. \tag{73}$$

We will only consider symmetric $\delta\mathbf{M}$ perturbations, although if antisymmetric perturbations were allowed, they would stably decay to zero, because the only antisymmetric term on the RHS of (73) would come from $\delta\mathbf{M}$.

From Eq.s (72) and (73), it follows that

$$\frac{d}{dt}\left(\delta\mathbf{A} + \delta\mathbf{S} - \left(2\tau - 1\right)\bar{\mathbf{M}}^{-1}\delta\mathbf{M}\right) = -4\delta\mathbf{S}. \tag{74}$$

The RHS is symmetric. Therefore, the antisymmetric part of the LHS must equal zero. This gives us an integral of the dynamics

$$\mathbf{\Omega} := \delta\mathbf{A}(t) - \left(\tau - \frac{1}{2}\right)\left(\bar{\mathbf{M}}^{-1}\delta\mathbf{M}(t) - \delta\mathbf{M}(t)\bar{\mathbf{M}}^{-1}\right), \tag{75}$$

where $\mathbf{\Omega}$ is a constant, skew symmetric matrix. This reveals an interesting point, after the perturbation $\delta\mathbf{A}$ and $\delta\mathbf{M}$ will not decay to $\mathbf{0}$, even if the fixed point is stable. In hindsight, this is expected because due to the symmetry of the problem: there is a manifold of stable fixed points (bases in principal subspace), and perturbations within this manifold should not decay. A similar situation was observed in (Pehlevan et al., 2015).

The symmetric part of Eq. (74) gives,

$$\frac{d}{dt}\left(\delta\mathbf{S} - \left(\tau - \frac{1}{2}\right)\left(\bar{\mathbf{M}}^{-1}\delta\mathbf{M} + \delta\mathbf{M}\bar{\mathbf{M}}^{-1}\right)\right) = -4\delta\mathbf{S}, \tag{76}$$

which, using (73), implies

$$\begin{aligned}\frac{d\delta\mathbf{S}}{dt} &= \left(1-\frac{1}{2\tau}\right)\left[\bar{\mathbf{M}}^{-1}\delta\mathbf{A}\bar{\mathbf{M}} - \bar{\mathbf{M}}\delta\mathbf{A}\bar{\mathbf{M}}^{-1}\right] \\ &+ \left(1-\frac{1}{2\tau}\right)\left[\bar{\mathbf{M}}^{-1}\delta\mathbf{S}\bar{\mathbf{M}} + \bar{\mathbf{M}}\delta\mathbf{S}\bar{\mathbf{M}}^{-1} + 2\delta\mathbf{S}\right] - 4\delta\mathbf{S} \\ &- \left(1-\frac{1}{2\tau}\right)\left(\bar{\mathbf{M}}^{-1}\delta\mathbf{M} + \delta\mathbf{M}\bar{\mathbf{M}}^{-1}\right). \end{aligned} \tag{77}$$

To summarize, we analyze the linear stability of the system of equations, defined by Eq.s (73), (75), (77).

Next, we change to a basis where $\bar{\mathbf{M}}$ is diagonal. $\bar{\mathbf{M}}$ is symmetric, its eigenvalues are the principal eigenvectors $\{\sigma_1, \ldots, \sigma_k\}$ as shown in Appendix C.3 and it has an orthonormal set of eigenvectors. Let $\mathbf{U}$ be the matrix that contains the eigenvectors of $\bar{\mathbf{M}}$ in its columns. Define

$$\begin{aligned}\delta\mathbf{A}^U &:= \mathbf{U}^\top\delta\mathbf{A}\mathbf{U}, \\ \delta\mathbf{S}^U &:= \mathbf{U}^\top\delta\mathbf{S}\mathbf{U}, \\ \delta\mathbf{M}^U &:= \mathbf{U}^\top\delta\mathbf{M}\mathbf{U}, \\ \mathbf{\Omega}^U &:= \mathbf{U}^\top\mathbf{\Omega}\mathbf{U} \end{aligned} \tag{78}$$

Expressing Eq.s (73), (75), (77) in this new basis, in component form, and eliminating δA^U_{ij}:

$$\frac{d}{dt}\begin{bmatrix}\delta M^U_{ij} \\ \delta S^U_{ij}\end{bmatrix} = \mathbf{H}^{ij}\begin{bmatrix}\delta M^U_{ij} \\ \delta S^U_{ij}\end{bmatrix} + \begin{bmatrix}\frac{1}{\tau}\left(\sigma_j - \sigma_i\right) \\ \left(1-\frac{1}{2\tau}\right)\left(\frac{\sigma_j}{\sigma_i} - \frac{\sigma_i}{\sigma_j}\right)\end{bmatrix}\Omega^U_{ij} \tag{79}$$

where

$$\mathbf{H}^{ij} := \begin{bmatrix} \left(1-\frac{1}{2\tau}\right)\left(\sigma_j-\sigma_i\right)\left(\frac{1}{\sigma_i}-\frac{1}{\sigma_j}\right) - \frac{1}{\tau} & \frac{1}{\tau}\left(\sigma_j+\sigma_i\right) \\ \left(1-\frac{1}{2\tau}\right)\left[\left(\frac{\sigma_j}{\sigma_i}-\frac{\sigma_i}{\sigma_j}\right)\left(\tau-\frac{1}{2}\right)\left(\frac{1}{\sigma_i}-\frac{1}{\sigma_j}\right) - \left(\frac{1}{\sigma_i}+\frac{1}{\sigma_j}\right)\right] & \left(1-\frac{1}{2\tau}\right)\left(\frac{\sigma_j}{\sigma_i}+\frac{\sigma_i}{\sigma_j}+2\right) - 4 \end{bmatrix} \tag{80}$$

This is a closed system of equations for each (i,j) pair! The fixed point of this system of equations is at

$$\begin{aligned}\delta S^U_{ij} &= 0, \\ \delta M^U_{ij} &= \frac{\Omega^U_{ij}}{\frac{1}{\sigma_j-\sigma_i} - \left(\tau-\frac{1}{2}\right)\left(\frac{1}{\sigma_i}-\frac{1}{\sigma_j}\right)}. \end{aligned} \tag{81}$$

Hence, if the linear perturbations are stable, the perturbations that destroy the orthonormality of neural filters will decay to zero, and orthonormality will be restored.

The stability of the fixed point is governed by the trace and the determinant of the matrix $\mathbf{H}^{ij}$. The trace is

$$\mathrm{Tr}(\mathbf{H}^{ij}) = -4 + \left(2 - \frac{1}{\tau}\right)\left(\frac{\sigma_i}{\sigma_j} + \frac{\sigma_j}{\sigma_i},\right) - \frac{1}{\tau} \tag{82}$$

and the determinant is

$$\det(\mathbf{H}^{ij}) = 8 + \left(\frac{2}{\tau} - 4\right)\left(\frac{\sigma_i}{\sigma_j} + \frac{\sigma_j}{\sigma_i}\right). \tag{83}$$

The system (79) is linearly stable if both the trace is negative and the determinant is positive.

Defining the following function of covariance eigenvalues:

$$\gamma_{ij} := \left(\frac{\sigma_i}{\sigma_j} + \frac{\sigma_j}{\sigma_i}\right) = 2 + \frac{(\sigma_i - \sigma_j)^2}{\sigma_i \sigma_j}, \tag{84}$$

the trace is negative if and only if

$$\tau < \frac{1 + 1/\gamma_{ij}}{2 - 4/\gamma_{ij}} \tag{85}$$

The determinant is positive if and only if

$$\tau < \frac{1}{2 - 4/\gamma_{ij}} \tag{86}$$

Since $\gamma_{ij} > 0$, Eq. (86) implies Eq. (85). For stability, Eq. (86) has to be satisfied for all (i, j) pairs. When $i = j$, $\gamma_{ii} = 2$, Eq. (86) is satisfied because RHS is infinity. When $i \neq j$, Eq. (86) is nontrivial, and depends on relations between covariance eigenvalues. Since $\gamma_{ij} \geq 2$, $\tau \leq 1/2$ is always stable.

Collectively, our results prove item 4 of Theorem 1.

D Proof of strong min-max property for PSW objective

Here we show that minimization with respect to $\mathbf{Y}$ and maximization with respect to $\mathbf{M}$ can be exchanged in Eq. (26). We do this by explicitly calculating the value of

$$-\frac{2}{T}\mathrm{Tr}\left(\mathbf{X}^\top\mathbf{W}^\top\mathbf{Y}\right)+\mathrm{Tr}\left(\mathbf{M}\left(\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top-\mathbf{I}\right)\right) \tag{87}$$

with respect to min-max and max-min optimizations, and showing that the value does not change.

Proposition 3. *Let $\mathbf{A}\in\mathbb{R}^{k\times T}$ with $k\leq T$. Then*

$$\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}}\max_{\mathbf{M}\in\mathbb{R}^{k\times k}}-\frac{2}{T}\mathrm{Tr}\left(\mathbf{A}^\top\mathbf{Y}\right)+\mathrm{Tr}\left(\mathbf{M}\left(\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top-\mathbf{I}\right)\right)=-\frac{2}{T^{1/2}}\mathrm{Tr}\left(\left(\mathbf{A}\mathbf{A}^\top\right)^{1/2}\right). \tag{88}$$

Proof. Left side of Eq. (88) is a constrained optimization problem:

$$\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}}-\frac{2}{T}\mathrm{Tr}\left(\mathbf{A}^\top\mathbf{Y}\right)\qquad\text{s.t.}\quad\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top=\mathbf{I}. \tag{89}$$

Suppose an SVD of $\mathbf{A}=\sum_{i=1}^k\sigma_{A,i}\mathbf{u}_{A,i}\mathbf{v}_{A,i}^\top$ and an SVD of $\mathbf{Y}=\sum_{i=1}^k\sigma_{Y,i}\mathbf{u}_{Y,i}\mathbf{v}_{Y,i}^\top$. The constraint sets $\sigma_{Y,i}=\sqrt{T}$. Then the optimization problem reduces to:

$$\min_{\mathbf{u}_{Y,1},\ldots,\mathbf{u}_{Y,k},\mathbf{v}_{Y,1},\ldots,\mathbf{v}_{Y,k}}-\frac{2}{\sqrt{T}}\sum_{i=1}^k\sigma_{A,i}\sum_{j=1}^k\mathbf{u}_{A,i}^\top\mathbf{u}_{Y,j}\mathbf{v}_{A,i}^\top\mathbf{v}_{Y,j},\qquad\text{s.t.}\quad\mathbf{u}_{Y,i}^\top\mathbf{u}_{Y,j}=\delta_{ij},\quad\mathbf{v}_{Y,i}^\top\mathbf{v}_{Y,j}=\delta_{ij}. \tag{90}$$

Note that $\sum_{j=1}^k\mathbf{u}_{A,i}^\top\mathbf{u}_{Y,j}\mathbf{v}_{A,i}^\top\mathbf{v}_{Y,j}\leq 1$[5] and therefore the cost is lower bounded by $-\frac{2}{\sqrt{T}}\sum_{i=1}^k\sigma_{A,i}$. The lower bound is achieved when $\mathbf{u}_{A,i}=\mathbf{u}_{Y,i}$ and $\mathbf{v}_{A,i}=\mathbf{v}_{Y,i}$, with the optimal value of the objective $-\frac{2}{\sqrt{T}}\sum_{i=1}^k\sigma_{A,i}=-\frac{2}{\sqrt{T}}\mathrm{Tr}\left(\left(\mathbf{A}\mathbf{A}^\top\right)^{1/2}\right)$. □

Proposition 4. *Let $\mathbf{A}\in\mathbb{R}^{k\times T}$ with $k\leq T$. Then*

$$\max_{\mathbf{M}\in\mathbb{R}^{k\times k}}\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}}-\frac{2}{T}\mathrm{Tr}\left(\mathbf{A}^\top\mathbf{Y}\right)+\mathrm{Tr}\left(\mathbf{M}\left(\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top-\mathbf{I}\right)\right)=-\frac{2}{T^{1/2}}\mathrm{Tr}\left(\left(\mathbf{A}\mathbf{A}^\top\right)^{1/2}\right). \tag{91}$$

[5]Define $\alpha_j:=\mathbf{u}_{A,i}^\top\mathbf{u}_{Y,j}$ and $\beta_j:=\mathbf{v}_{A,i}^\top\mathbf{v}_{Y,j}$. Because $\mathbf{u}_{Y,i}^\top\mathbf{u}_{Y,j}=\mathbf{v}_{Y,i}^\top\mathbf{v}_{Y,j}=\delta_{ij}$, it follows that $\sum_{i=1}^k\alpha_i^2=1$ and $\sum_{i=1}^k\beta_i^2\leq 1$. The sum in question is $\sum_{i=1}^k\alpha_i\beta_i$, which is an inner product of a unit vector and a vector with magnitude less than or equal to 1. Hence, the maximal inner product can be 1.

Proof. Note that we only need to consider the symmetric part of $\mathbf{M}$, because its antisymmetric component does not contribute to the cost. Below, we use $\mathbf{M}$ to mean its symmetric part. We will evaluate the value of the objective

$$-\frac{2}{T}\mathrm{Tr}\left(\mathbf{A}^\top\mathbf{Y}\right)+\mathrm{Tr}\left(\mathbf{M}\left(\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top-\mathbf{I}\right)\right) \tag{92}$$

considering the following cases:

1. $\mathbf{A}=\mathbf{0}$. In this case the first term in Eq. (92) drops. Minimization of the second term with respect to $\mathbf{Y}$ gives $-\infty$ if $\mathbf{M}$ has a negative eigenvalue, or a 0 if $\mathbf{M}$ is positive semidefinite. Hence, the max-min objective is zero, and the proposition holds.

2. $\mathbf{A}\neq\mathbf{0}$ and $\mathbf{A}$ is full-rank.

   (a) $\mathbf{M}$ has at least one negative eigenvalue. Then, minimization of Eq. (92) with respect to $\mathbf{Y}$ gives $-\infty$.

   (b) $\mathbf{M}$ is positive semidefinite and has at least one zero eigenvalue. Then, minimization of Eq. (92) with respect to $\mathbf{Y}$ gives $-\infty$. To achieve this solution, one chooses all columns of $\mathbf{Y}$ to be one of the zero eigenvectors. The sign of the eigenvector is chosen such that $\mathrm{Tr}\left(\mathbf{A}^\top\mathbf{Y}\right)$ is positive. Multiplying $\mathbf{Y}$ by a positive scalar, one can reduce the objective indefinitely.

   (c) $\mathbf{M}$ is positive definite. Then, $\mathbf{Y}^*=\mathbf{M}^{-1}\mathbf{A}$ minimizes Eq. (92) with respect to $\mathbf{Y}$. Plugging this back to (92), we get the objective

   $$-\frac{1}{T}\mathrm{Tr}\left(\mathbf{A}^\top\mathbf{M}^{-1}\mathbf{A}\right)-\mathrm{Tr}\left(\mathbf{M}\right). \tag{93}$$

   The positive definite $\mathbf{M}$ that maximizes Eq. (93) can be found by setting its derivative to zero

   $$\mathbf{M}^{*2}=\frac{1}{T}\mathbf{A}\mathbf{A}^\top. \tag{94}$$

   Plugging this back in Eq. (93), one gets the objective

   $$-\frac{2}{\sqrt{T}}\mathrm{Tr}\left(\left(\mathbf{A}\mathbf{A}^\top\right)^{1/2}\right), \tag{95}$$

   which is maximal with respect to all possible $\mathbf{M}$. Therefore the proposition holds.

3. $\mathbf{A}\neq\mathbf{0}$ and $\mathbf{A}$ has rank $r<k$.

(a) $\mathbf{M}$ has at least one negative eigenvalue. Then, minimization of Eq. (92) with respect to $\mathbf{Y}$ gives $-\infty$, as before.

(b) $\mathbf{M}$ is positive semidefinite and has at least one zero eigenvalue.

i. If at least one of the zero-eigenvectors of $\mathbf{M}$ is not a left zero-singular vector of $\mathbf{A}$, then, minimization of Eq. (92) with respect to $\mathbf{Y}$ gives $-\infty$. To achieve this solution, one chooses all columns of $\mathbf{Y}$ to be the zero-eigenvector of $\mathbf{M}$ that is not a left zero-singular vector of $\mathbf{A}$. The sign of the eigenvector is chosen such that $\mathrm{Tr}\left(\mathbf{A}^\top \mathbf{Y}\right)$ is positive. Multiplying $\mathbf{Y}$ by a positive scalar, one can reduce the objective indefinitely.

ii. If all of the zero-eigenvectors of $\mathbf{M}$ are also left zero-singular vectors of $\mathbf{A}$, then Eq. (92) can be reformulated in the subspace spanned by top r eigenvectors of $\mathbf{M}$. Suppose a SVD for $\mathbf{A} = \sum_{i=1}^{r} \sigma_{A,i} \mathbf{u}_{A,i} \mathbf{v}_{M,i}^\top$ with $\sigma_{A,1} \geq \sigma_{A,2} \geq \ldots \geq \sigma_{A,r}$. One can decompose $\mathbf{Y} = \mathbf{Y}^A + \mathbf{Y}^\perp$, where columns of $\mathbf{Y}^\perp$ are perdendicular to the space spanned by $\{\mathbf{u}_{A,1}, \ldots, \mathbf{u}_{A,r}\}$. Then value of the objective Eq. (92) only depends on $\mathbf{Y}^A$. Defining new matrices $\tilde{\mathbf{A}}_{i,:} = \mathbf{u}_{A,i}^\top \mathbf{A}$, $\tilde{\mathbf{Y}}_{i,:} = \mathbf{u}_{A,i}^\top \mathbf{Y}^A$, $\tilde{\mathbf{M}}_{ij} = \mathbf{u}_{A,i}^\top \mathbf{M} \mathbf{u}_{A,j}$, where $i, j = 1, \ldots, r$, we can rewrite Eq. (92) as

$$-\frac{2}{T}\mathrm{Tr}\left(\tilde{\mathbf{A}}^\top \tilde{\mathbf{Y}}\right) + \mathrm{Tr}\left(\tilde{\mathbf{M}}\left(\frac{1}{T}\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^\top - \mathbf{I}\right)\right). \tag{96}$$

Now $\tilde{\mathbf{A}}$ is full-rank and $\tilde{\mathbf{M}}$ is positive definite. As in 2.(c), the objective which is maximal with respect to positive definite $\tilde{\mathbf{M}}$ matrices is

$$-\frac{2}{\sqrt{T}}\mathrm{Tr}\left(\left(\tilde{\mathbf{A}}\tilde{\mathbf{A}}^\top\right)^{1/2}\right) = -\frac{2}{\sqrt{T}}\mathrm{Tr}\left(\left(\mathbf{A}\mathbf{A}^\top\right)^{1/2}\right). \tag{97}$$

(c) $\mathbf{M}$ is positive definite. As in 2.(c), the objective which is maximal with respect to positive definite $\mathbf{M}$ matrices is

$$-\frac{2}{\sqrt{T}}\mathrm{Tr}\left(\left(\mathbf{A}\mathbf{A}^\top\right)^{1/2}\right). \tag{98}$$

This is also maximal with respect to all possible $\mathbf{M}$. Therefore the proposition holds.

Collectively, these arguments prove Eq. (92). □

Propositions (3) and (4) imply the strong min-max property for the PSW cost.

Proposition 5. *The strong min-max property for the PSW cost:*

$$\begin{aligned}
&\min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} \max_{\mathbf{M}\in\mathbb{R}^{k\times k}} -\frac{2}{T}\mathrm{Tr}\left(\mathbf{X}^\top\mathbf{W}^\top\mathbf{Y}\right) + \mathrm{Tr}\left(\mathbf{M}\left(\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top - \mathbf{I}\right)\right) \\
&\quad = \max_{\mathbf{M}\in\mathbb{R}^{k\times k}} \min_{\mathbf{Y}\in\mathbb{R}^{k\times T}} -\frac{2}{T}\mathrm{Tr}\left(\mathbf{X}^\top\mathbf{W}^\top\mathbf{Y}\right) + \mathrm{Tr}\left(\mathbf{M}\left(\frac{1}{T}\mathbf{Y}\mathbf{Y}^\top - \mathbf{I}\right)\right) \\
&\quad = -\frac{2}{T^{1/2}}\mathrm{Tr}\left(\left(\mathbf{W}\mathbf{X}\mathbf{X}^\top\mathbf{W}^\top\right)^{1/2}\right). \qquad (99)
\end{aligned}$$

E Proof of Theorem 2

Here we prove Theorem 2.

E.1 Proof of item 1

Item 1 directly follows from the fixed point equations of the dynamical system (30), ($^{\bar{}}$for fixed point).

$$\begin{aligned}\bar{\mathbf{W}} &= \bar{\mathbf{Y}}\mathbf{X}^\top = \bar{\mathbf{F}}\mathbf{C} \\ \mathbf{I} &= \bar{\mathbf{Y}}\bar{\mathbf{Y}}^\top = \bar{\mathbf{F}}\mathbf{C}\bar{\mathbf{F}}^\top.\end{aligned} \tag{100}$$

E.2 Proof of item 2

We will prove item 2, making use of the normalized neural filters:

$$\mathbf{R} := \mathbf{F}\mathbf{C}^{1/2}, \tag{101}$$

where the input covariance matrix $\mathbf{C}$ is defined as in Eq. (1). At the fixed point, the normalized neural filters are orthonormal:

$$\bar{\mathbf{R}}\bar{\mathbf{R}}^\top = \bar{\mathbf{F}}\mathbf{C}\bar{\mathbf{F}}^\top = \bar{\mathbf{Y}}\bar{\mathbf{Y}}^\top = \mathbf{I}. \tag{102}$$

Normalized filters commute with the covariance matrix:

$$\bar{\mathbf{R}}^\top\bar{\mathbf{R}}\mathbf{C} = \mathbf{C}\bar{\mathbf{R}}^\top\bar{\mathbf{R}}. \tag{103}$$

Proof.

$$\begin{aligned}\bar{\mathbf{R}}^\top\bar{\mathbf{R}}\mathbf{C} &= \mathbf{C}^{1/2}\bar{\mathbf{F}}^\top\bar{\mathbf{F}}\mathbf{C}^{3/2} = \mathbf{C}^{1/2}\bar{\mathbf{F}}^\top\bar{\mathbf{W}}\mathbf{C}^{1/2} = \mathbf{C}^{1/2}\bar{\mathbf{F}}^\top\bar{\mathbf{M}}\bar{\mathbf{F}}\mathbf{C}^{1/2} \\ &= \mathbf{C}^{1/2}\bar{\mathbf{W}}^\top\bar{\mathbf{F}}\mathbf{C}^{1/2} = \mathbf{C}\mathbf{C}^{1/2}\bar{\mathbf{F}}^\top\bar{\mathbf{F}}\mathbf{C}^{1/2} = \mathbf{C}\bar{\mathbf{R}}^\top\bar{\mathbf{R}}.\end{aligned} \tag{104}$$

□

Therefore, as argued in Appendix C.2, rows of $\mathbf{R}$ span a subspace spanned by some k eigenvectors of $\mathbf{C}$. If $\mathbf{C}$ is invertible, rowspace of $\mathbf{F}$ is the same as $\mathbf{R}$ (follows from Eq. (101)) and item 2 follows.

E.3 Proof of item 3

Preliminaries

In order to perform a linear stability analysis, we linearize the system of equations (30) around the fixed point. The evolution of $\mathbf{W}$ and $\mathbf{M}$ perturbations follow from

linearization of (30):

$$\tau \frac{d\delta \mathbf{M}}{dt} = \delta \mathbf{R} \bar{\mathbf{R}}^\top + \bar{\mathbf{R}} \delta \mathbf{R}^\top,$$
$$\frac{d\delta \mathbf{W}}{dt} = 2\delta \mathbf{R} \mathbf{C}^{1/2} - 2\delta \mathbf{W}. \tag{105}$$

Even though Eq. (30) depends on $\mathbf{W}$ and $\mathbf{M}$, we will find it convenient to change variables and work with $\mathbf{R}$, as defined in Eq. (101), and $\mathbf{M}$ instead. Since $\mathbf{R}$, $\mathbf{W}$ and $\mathbf{M}$ are interdependent, we express the perturbations of $\mathbf{R}$ in terms of $\mathbf{W}$ and $\mathbf{M}$ perturbations:

$$\delta \mathbf{R} = \delta \mathbf{M}^{-1} \bar{\mathbf{W}} \mathbf{C}^{1/2} + \bar{\mathbf{M}}^{-1} \delta \mathbf{W} \mathbf{C}^{1/2} = -\bar{\mathbf{M}}^{-1} \delta \mathbf{M} \bar{\mathbf{R}} + \bar{\mathbf{M}}^{-1} \delta \mathbf{W} \mathbf{C}^{1/2}, \tag{106}$$

which implies that

$$\frac{d\delta \mathbf{R}}{dt} = -\bar{\mathbf{M}}^{-1} \frac{d\,\delta \mathbf{M}}{dt} \bar{\mathbf{R}} + \bar{\mathbf{M}}^{-1} \frac{d\,\delta \mathbf{W}}{dt} \mathbf{C}^{1/2}. \tag{107}$$

Plugging these in and eliminating $\delta \mathbf{W}$, we arrive at a linearized equation for $\delta \mathbf{R}$:

$$\frac{d\delta \mathbf{R}}{dt} = -\frac{1}{\tau} \bar{\mathbf{M}}^{-1} \left(\delta \mathbf{R} \bar{\mathbf{R}}^\top + \bar{\mathbf{R}} \delta \mathbf{R}^\top + 2\tau \delta \mathbf{M} \right) \bar{\mathbf{R}} + 2\bar{\mathbf{M}}^{-1} \delta \mathbf{R} \mathbf{C} - 2\delta \mathbf{R}. \tag{108}$$

To asses the stability of $\delta \mathbf{R}$, we expand it as in Appendix C.3:

$$\delta \mathbf{R} = \delta \mathbf{A} \bar{\mathbf{R}} + \delta \mathbf{S} \bar{\mathbf{R}} + \delta \mathbf{B} \bar{\mathbf{G}} \tag{109}$$

where $\delta \mathbf{A}$ is an $k \times k$ skew-symmetric matrix, $\delta \mathbf{S}$ is an $k \times k$ symmetric matrix and $\delta \mathbf{B}$ is an $k \times (n-k)$ matrix. $\bar{\mathbf{G}}$ is an $(n-k) \times n$ matrix with orthonormal rows. These rows are chosen to be orthogonal to the rows of $\bar{\mathbf{R}}$. As before, skew-symmetric $\delta \mathbf{A}$ corresponds to rotations of filters within the normalized filter space, symmetric $\delta \mathbf{S}$ keeps the normalized filter space invariant but destroys orthonormality and $\delta \mathbf{B}$ is a perturbation that takes the normalized neural filters outside of the filter space.

Let $\mathbf{v}_{1,\dots,n}$ be the eigenvectors $\mathbf{C}$ and $\sigma_{1,\dots,n}$ be the corresponding eigenvalues. We label them such that $\bar{\mathbf{R}}$ spans the same space as the space spanned by the first k eigenvectors. We choose rows of $\bar{\mathbf{G}}$ to be the remaining eigenvectors, i.e. $\bar{\mathbf{G}}^\top := [\mathbf{v}_{k+1}, \dots, \mathbf{v}_n]$.

Proof

Proof of item 3 of Theorem 2 follows from studying the stability of $\delta \mathbf{B}$ component. Multiplying Eq. (108) on the right by $\bar{\mathbf{G}}^\top$, we arrive at a decoupled evolution

equation:

$$\frac{d\delta B_i^j}{dt} = \sum_m P_{im}^j \delta B_m^j, \qquad P_{im}^j := 2\left(\bar{M}_{im}^{-1}\sigma_{j+k} - \delta_{im}\right), \tag{110}$$

where for convenience we change our notation to $\delta B_{kj} = \delta B_k^j$.

Eq.s (100) and (102) imply $\bar{\mathbf{M}}^2 = \bar{\mathbf{W}}\mathbf{C}\bar{\mathbf{W}}^\top = \bar{\mathbf{R}}\mathbf{C}^2\bar{\mathbf{R}}^\top$ and hence:

$$\bar{\mathbf{M}} = \bar{\mathbf{R}}\mathbf{C}\bar{\mathbf{R}}^\top. \tag{111}$$

Taking into account Eq.s (102) and (103), the case at hand reduces to the proof presented in Appendix C.3: stable solutions are those for which

$$\{\sigma_1, \ldots, \sigma_k\} > \{\sigma_{k+1}, \ldots, \sigma_n\}. \tag{112}$$

This proves that if at the fixed point, normalized neural filters span a subspace other than the principal subspace, the fixed point is linearly unstable. Since the span of normalized neural filters is that of the neural filters, item 3 follows.

E.4 Proof of item 4

Proof of item 4 follows from the linear stabilities of $\delta\mathbf{A}$ and $\delta\mathbf{S}$. Multiplying Eq. (108) on the right by $\bar{\mathbf{R}}^\top$, and separating the resulting equation in to into its symmetric and anti-symmetric parts, we arrive at:

$$\begin{aligned}
\frac{d\delta\mathbf{A}}{dt} &= -\frac{1}{\tau}\left(\bar{\mathbf{M}}^{-1}\delta\mathbf{S} - \delta\mathbf{S}\bar{\mathbf{M}}^{-1}\right) - \bar{\mathbf{M}}^{-1}\delta\mathbf{M} + \delta\mathbf{M}\bar{\mathbf{M}}^{-1} - 2\delta\mathbf{A} \\
&\quad + \bar{\mathbf{M}}^{-1}\delta\mathbf{A}\bar{\mathbf{M}} + \bar{\mathbf{M}}\delta\mathbf{A}\bar{\mathbf{M}}^{-1} + \bar{\mathbf{M}}^{-1}\delta\mathbf{S}\bar{\mathbf{M}} - \bar{\mathbf{M}}\delta\mathbf{S}\bar{\mathbf{M}}^{-1}, \\
\frac{d\delta\mathbf{S}}{dt} &= -\frac{1}{\tau}\left(\bar{\mathbf{M}}^{-1}\delta\mathbf{S} + \delta\mathbf{S}\bar{\mathbf{M}}^{-1}\right) - \bar{\mathbf{M}}^{-1}\delta\mathbf{M} - \delta\mathbf{M}\bar{\mathbf{M}}^{-1} - 2\delta\mathbf{S} \\
&\quad + \bar{\mathbf{M}}^{-1}\delta\mathbf{A}\bar{\mathbf{M}} - \bar{\mathbf{M}}\delta\mathbf{A}\bar{\mathbf{M}}^{-1} + \bar{\mathbf{M}}^{-1}\delta\mathbf{S}\bar{\mathbf{M}} + \bar{\mathbf{M}}\delta\mathbf{S}\bar{\mathbf{M}}^{-1}
\end{aligned} \tag{113}$$

To obtain a closed set of equations, we complement these equations with $\delta\mathbf{M}$ evolution, which we obtain by plugging the expansion (109) into Eq. (105):

$$\tau\frac{d\delta\mathbf{M}}{dt} = 2\delta\mathbf{S} \tag{114}$$

We only consider symmetric $\delta\mathbf{M}$ below, since our algorithm preserves the symmetry of $\mathbf{M}$ in runtime.

We now change to a basis where $\bar{\mathbf{M}}$ is diagonal. $\bar{\mathbf{M}}$ is symmetric and has an orthonormal set of eigenvectors. Its eigenvalues are the principal eigenvalues $\{\sigma_1, \ldots, \sigma_k\}$ (from Appendix C.3). Let $\mathbf{U}$ be the matrix that contains the

eigenvectors of $\bar{\mathbf{M}}$ in its columns. Define

$$\begin{aligned}\delta\mathbf{A}^U &:= \mathbf{U}^\top \delta\mathbf{A}\mathbf{U},\\ \delta\mathbf{S}^U &:= \mathbf{U}^\top \delta\mathbf{S}\mathbf{U},\\ \delta\mathbf{M}^U &:= \mathbf{U}^\top \delta\mathbf{M}\mathbf{U}.\end{aligned} \tag{115}$$

In this new basis, the linearized equations, in component form, become:

$$\frac{d}{dt}\begin{bmatrix}\delta M_{ij}^U\\ \delta A_{ij}^U\\ \delta S_{ij}^U\end{bmatrix} = \mathbf{H}^{ij}\begin{bmatrix}\delta M_{ij}^U\\ \delta A_{ij}^U\\ \delta S_{ij}^U\end{bmatrix}, \tag{116}$$

where

$$\mathbf{H}^{ij} := \begin{bmatrix} 0 & 0 & \frac{2}{\tau}\\ \frac{1}{\sigma_j}-\frac{1}{\sigma_i} & -2+\frac{\sigma_j}{\sigma_i}+\frac{\sigma_i}{\sigma_j} & -\frac{1}{\tau}\left(\frac{1}{\sigma_i}-\frac{1}{\sigma_j}\right)+\frac{\sigma_j}{\sigma_i}-\frac{\sigma_i}{\sigma_j}\\ -\frac{1}{\sigma_j}-\frac{1}{\sigma_i} & \frac{\sigma_j}{\sigma_i}-\frac{\sigma_i}{\sigma_j} & -\frac{1}{\tau}\left(\frac{1}{\sigma_i}+\frac{1}{\sigma_j}\right)+\frac{\sigma_j}{\sigma_i}+\frac{\sigma_i}{\sigma_j}-2\end{bmatrix} \tag{117}$$

Linear stability is governed by the three eigenvalues of $\mathbf{H}^{ij}$. One of the eigenvalues is 0, due to the existence of the rotational symmetry in the problem. The corresponding eigenvector is $[\sigma_j - \sigma_i, 1, 0]$. Note that the third element of the eigenvector is zero, showing that the orthogonality of the normalized neural filters are not spoiled even in this mode.

For stability of the principal subspace, the other two eigenvalues must be negative, which means their sum should be negative, and their multiplication should be positive. It is easy to show that both the negativity of the summation and the positivity of the multiplication holds if and only if for all (i, j) pairs with $i \neq j$:

$$\tau < \frac{\sigma_i + \sigma_j}{2\left(\sigma_i - \sigma_j\right)^2}. \tag{118}$$

Hence we have showed that linear perturbations of fixed point weights decay to a configuration in which normalized neural filters are rotations of the original normalized neural filters within the subspace. It follows from Eq. (101), that the same holds for neural filters.

F Autapse-free similarity matching network with asymmetric lateral connectivity

Here, we derive an alternative neural network algorithm for PSP, which does not feature autaptic connections and has asymmetric lateral connections. To this end, we replace the gradient descent neural dynamics defined by Eq. (20) by a coordinate descent dynamics.

In the coordinate descent approach, at every step, one finds the optimal value of one component of $\mathbf{y}_t$, while keeping the rest fixed. By taking the derivative of the cost $-4\mathbf{x}_t^\top \mathbf{W}\mathbf{y}_t + 2\mathbf{y}_t^\top \mathbf{M}\mathbf{y}_t$ with respect to $y_{t,i}$ and setting it to zero we find:

$$y_{t,i} = \sum_{j=1} \frac{W_{t,ij}}{M_{t,ii}} x_{t,j} - \sum_{j\neq i} \frac{M_{t,ij}}{M_{t,ii}} y_{t,j}. \tag{119}$$

The components can be cycled through in any order until the iteration converges to a fixed point. The iteration is guaranteed to converge under very mild assumptions: diagonals of $\mathbf{M}$ have to be positive (Luo and Tseng, 1991), which is satisfied if $\mathbf{M}$ is initialized that way, see Eq. (21). Finally, Eq. (119) can be interpreted as a Gauss-Seidel iteration and generalizations to other iterative schemes are possible, see (Pehlevan et al., 2015).

The coordinate descent iteration, Eq. (119), can be interpreted as the dynamics of an asynchronous autapse-free neural network, Fig. 1B, where synaptic weights are:

$$\tilde{W}_{t,ij} = \frac{W_{t,ij}}{M_{t,ii}}, \qquad \tilde{M}_{t,ij} = \frac{M_{t,,j}}{M_{t,ii}}, \qquad \tilde{M}_{t,ii} = 0. \tag{120}$$

With this definition, the lateral weights are now asymmetric because $M_{t,ii} \neq M_{t,jj}$ if $i \neq j$.

We can derive updates for these synaptic weights from the updates for $\mathbf{W}_t$ and $\mathbf{M}_t$, Eq. (21). By defining another scalar state variable for each ith neuron $\tilde{D}_{t,i} := \tau M_{t,ii}/\eta_{t-1}$, we arrive at[6]:

[6]These update rules can be derived as follows. Start by the definition of the synaptic weights, Eq. (120): $M_{t+1,ii}\tilde{M}_{t+1,ij} = M_{t+1,ij}$. By the gradient-descent update Eq. (21), $M_{t+1,ij} = \left(1-\frac{\eta_t}{\tau}\right) M_{t,ij} + \frac{\eta_t}{\tau} y_{t,i}y_{t,j} = \left(1-\frac{\eta_t}{\tau}\right) \tilde{M}_{t,ij} M_{t,ii} + \frac{\eta_t}{\tau} y_{t,i}y_{t,j}$, where in the second equality we again used Eq. (120). But note that $(1-\frac{\eta_t}{\tau})M_{t,ii} = M_{t+1,ii} - \frac{\eta_t}{\tau} y_{t,i}^2$, from Eq. (21). Combining all of these, $\tilde{M}_{t+1,ij} = \tilde{M}_{t,ij} + \frac{\eta_t}{\tau M_{t+1,ii}} \left(y_{t,i}x_{t,j} - y_{t,i}^2 \tilde{M}_{t,ij} \right)$. Similar derivation can be given for feedforward updates.

$$
\begin{aligned}
\tilde{D}_{t+1,i} &= \frac{\eta_{t-1}}{\eta_t}\left(1-\frac{\eta_t}{\tau}\right)\tilde{D}_{t,i} + y_{t,i}^2, \\
\tilde{W}_{t+1,ij} &= \left(\frac{1-2\eta_t}{1-\eta_t/\tau}\right)\tilde{W}_{t,ij} + \frac{1}{\tilde{D}_{t+1,i}}\left(2\tau y_{t,i}x_{t,j} - \left(\frac{1-2\eta_t}{1-\eta_t/\tau}\right)y_{t,i}^2\tilde{W}_{t,ij}\right), \\
\tilde{M}_{t+1,i,j\neq i} &= \tilde{M}_{t,ij} + \frac{1}{\tilde{D}_{t+1,i}}\left(y_{t,i}y_{t,j} - y_{t,i}^2\tilde{M}_{t,ij}\right). \\
\tilde{M}_{t+1,ii} &= 0,
\end{aligned} \tag{121}
$$

Here, in addition to synaptic weights, the neurons need to keep track of a post-synaptic activity depended variable $\tilde{D}_{t,i}$ and the gradient descent-ascent learning rate parameters η_t, η_{t-1} and τ. The updates are local.

For the special case of $\tau = 1/2$ and $\eta_t = \eta/2$, these plasticity rules simplify to,

$$
\begin{aligned}
\tilde{D}_{t+1,i} &= (1-\eta)\tilde{D}_{t,i} + y_{t,i}^2, \\
\tilde{W}_{t+1,ij} &= \tilde{W}_{t,ij} + \frac{1}{\tilde{D}_{t+1,i}}\left(y_{t,i}x_{t,j} - y_{t,i}^2\tilde{W}_{t,ij}\right) \\
\tilde{M}_{t+1,i,j\neq i} &= \tilde{M}_{t,ij} + \frac{1}{\tilde{D}_{t+1,i}}\left(y_{t,i}y_{t,j} - y_{t,i}^2\tilde{M}_{t,ij}\right). \\
\tilde{M}_{t+1,ii} &= 0,
\end{aligned} \tag{122}
$$

which is precisely the neural online similarity matching algorithm we previously gave in (Pehlevan et al., 2015). Both feedforward and lateral updates have the same form as a single-neuron Oja's rule (Oja, 1982).

Note that the algorithm derived above is essentially the same as the one in the main text: given the same initial conditions and the same inputs, $\mathbf{x}_t$, they will produce the same outputs, $\mathbf{y}_t$. The only difference is a rearrangement of synaptic weights in the neural network implementation.

G Autapse-free constrained similarity matching network with asymmetric lateral connectivity

Following similar steps to Appendix F, we derive an autapse-free PSW neural algorithm with asymmetric lateral connections. We replace the gradient descent neural dynamics defined by Eq. (35) by a coordinate descent dynamics, where at every step, one finds the optimal value of one component of $\mathbf{y}_t$, while keeping the rest fixed:

$$y_{t,i} = \sum_{j=1} \frac{W_{t,ij}}{M_{t,ii}} x_{t,j} - \sum_{j \neq i} \frac{M_{t,ij}}{M_{t,ii}} y_{t,j}. \tag{123}$$

The components can be cycled through in any order until the iteration converges to a fixed point.

The coordinate descent iteration, Eq. (123), can be interpreted as the dynamics of an asynchronous autapse-free neural network, Fig. 1B, with synaptic weights:

$$\tilde{W}_{t,ij} = \frac{W_{t,ij}}{M_{t,ii}}, \qquad \tilde{M}_{t,ij} = \frac{M_{t,,j}}{M_{t,ii}}, \qquad \tilde{M}_{t,ii} = 0. \tag{124}$$

As in Appendix F, the new lateral weights are asymmetric.

Updates for these synaptic weights can be derived from the updates for $\mathbf{W}_t$ and $\mathbf{M}_t$, Eq. (36). Defining another scalar state variable for each ith neuron $\tilde{D}_{t,i} := \tau M_{t,ii}/\eta_{t-1}$, we arrive at

$$\begin{aligned}
\tilde{D}_{t+1,i} &= \frac{\eta_{t-1}}{\eta_t}\left(1 - \frac{\eta_t}{\tau}\right)\tilde{D}_{t,i} + y_{t,i}^2 - 1, \\
\tilde{W}_{t+1,ij} &= (1-2\eta_t)\,\tilde{W}_{t,ij} + \frac{1}{\tilde{D}_{t+1,i}}\left(2\tau y_{t,i} x_{t,j} - (1-2\eta_t)\left(y_{t,i}^2 - 1\right)\tilde{W}_{t,ij}\right), \\
\tilde{M}_{t+1,i,j\neq i} &= \tilde{M}_{t,ij} + \frac{1}{\tilde{D}_{t+1,i}}\left(y_{t,i} y_{t,j} - \left(y_{t,i}^2 - 1\right)\tilde{M}_{t,ij}\right), \\
\tilde{M}_{t+1,ii} &= 0.
\end{aligned} \tag{125}$$

As in Appendix F, in addition to synaptic weights, the neurons need to keep track of a postsynaptic activity depended variable $\tilde{D}_{t,i}$ and gradient descent-ascent learning rate parameters $\eta_{W,t}$, $\eta_{M,t}$ and $\eta_{M,t-1}$.

For the special case of $\eta_t = \eta/2$ and $\tau = 1/2$, these plasticity rules simplify to,

$$\begin{aligned}
\tilde{D}_{t+1,i} &= (1-\eta)\tilde{D}_{t,i} + y_{t,i}^2, \\
\tilde{W}_{t+1,ij} &= (1-\eta)\tilde{W}_{t,ij} + \frac{1}{\tilde{D}_{t+1,i}}\left(y_{t,i}x_{t,j} - (1-\eta)\left(y_{t,i}^2 - 1\right)\tilde{W}_{t,ij}\right) \\
\tilde{M}_{t+1,i,j\neq i} &= \tilde{M}_{t,ij} + \frac{1}{\tilde{D}_{t+1,i}}\left(y_{t,i}y_{t,j} - \left(y_{t,i}^2 - 1\right)\tilde{M}_{t,ij}\right), \\
\tilde{M}_{t+1,ii} &= 0.
\end{aligned} \tag{126}$$

References

Arora, S., Ge, R., Ma, T., and Moitra, A. (2015). Simple, efficient, and neural algorithms for sparse coding. *arXiv preprint arXiv:1503.00778*.

Bliss, T. V. and Gardner-Medwin, A. (1973). Long-lasting potentiation of synaptic transmission in the dentate area of the unanaesthetized rabbit following stimulation of the perforant path. *The Journal of physiology*, 232(2):357.

Bliss, T. V. and Lømo, T. (1973). Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path. *The Journal of physiology*, 232(2):331–356.

Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

Carlson, A. (1990). Anti-hebbian learning in a non-linear neural network. *Biological cybernetics*, 64(2):171–176.

Cox, T. F. and Cox, M. A. (2000). *Multidimensional scaling*. CRC Press.

Diamantaras, K. I. and Kung, S. Y. (1996). *Principal component neural networks: theory and applications*. John Wiley & Sons, Inc.

Földiak, P. (1989). Adaptive network for optimal linear feature extraction. In *Int. Joint Conf. on Neural Networks*, pages 401–405. IEEE.

Földiak, P. (1990). Forming sparse representations by local anti-hebbian learning. *Biological cybernetics*, 64(2):165–170.

Hu, T., Pehlevan, C., and Chklovskii, D. B. (2014). A hebbian/anti-hebbian network for online sparse dictionary learning derived from symmetric matrix factorization. In *Asilomar Conference on Signals, Systems and Computers*, pages 613–619. IEEE.

Ikeda, K. and Bekkers, J. M. (2006). Autapses. *Current Biology*, 16(9):R308.

Komatsu, Y. (1994). Age-dependent long-term potentiation of inhibitory synaptic transmission in rat visual cortex. *Journal of Neuroscience*, 14(11):6488–6499.

Kullmann, D. M., Moreau, A. W., Bakiri, Y., and Nicholson, E. (2012). Plasticity of inhibition. *Neuron*, 75(6):951–962.

Kung, S.-Y., Diamantaras, K., and Taur, J.-S. (1994). Adaptive principal component extraction (apex) and applications. *IEEE Transactions on Signal Processing*, 42(5):1202–1217.

Leen, T. K. (1991). Dynamics of learning in linear feature-discovery networks. *Network*, 2(1):85–105.

Linsker, R. (1997). A local learning rule that enables information maximization for arbitrary input distributions. *Neural Computation*, 9(8):1661–1665.

Luo, Z.-Q. and Tseng, P. (1991). On the convergence of a matrix splitting algorithm for the symmetric monotone linear complementarity problem. *SIAM Journal on Control and Optimization*, 29(5):1037–1060.

Maffei, A., Nataraj, K., Nelson, S. B., and Turrigiano, G. G. (2006). Potentiation of cortical inhibition by visual deprivation. *Nature*, 443(7107):81–84.

Mardia, K. V., Kent, J. T., and Bibby, J. M. (1980). Multivariate analysis (probability and mathematical statistics).

Miao, Y. and Hua, Y. (1998). Fast subspace tracking and neural network learning by a novel information criterion. *IEEE Transactions on Signal Processing*, 46(7):1967–1979.

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

Oja, E. (1989). Neural networks, principal components, and subspaces. *International journal of neural systems*, 1(01):61–68.

Olshausen, B. A. et al. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609.

Olshausen, B. A. and Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325.

Pehlevan, C. and Chklovskii, D. (2015). A normative theory of adaptive dimensionality reduction in neural networks. In *Advances in Neural Information Processing Systems*, pages 2260–2268.

Pehlevan, C. and Chklovskii, D. B. (2014). A hebbian/anti-hebbian network derived from online non-negative matrix factorization can cluster and discover sparse features. In *Asilomar Conference on Signals, Systems and Computers*, pages 769–775. IEEE.

Pehlevan, C., Hu, T., and Chklovskii, D. B. (2015). A hebbian/anti-hebbian neural network for linear subspace learning: A derivation from multidimensional scaling of streaming data. *Neural Computation*, 27:1461–1495.

Plumbley, M. D. (1993a). Efficient information transfer and anti-hebbian neural networks. *Neural Networks*, 6(6):823–833.

Plumbley, M. D. (1993b). A hebbian/anti-hebbian network which optimizes information capacity by orthonormalizing the principal subspace. In *Proc. 3rd Int. Conf. on Artificial Neural Networks*, pages 86–90.

Rubner, J. and Schulten, K. (1990). Development of feature detectors by self-organization. *Biol Cybern*, 62(3):193–199.

Rubner, J. and Tavan, P. (1989). A self-organizing network for principal-component analysis. *EPL*, 10:693.

Sanger, T. (1989). Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural networks*, 2(6):459–473.

Williams, C. K. (2001). On a connection between kernel pca and metric multidimensional scaling. In *NIPS*, pages 675–681. MIT Press.