Statistical whitening of neural populations with gain-modulating interneurons

Lyndon R. Duong [*1] **David Lipshutz** [*2] **David J. Heeger** [1] **Dmitri B. Chklovskii** [2 3] **Eero P. Simoncelli** [1 2]

Abstract

Statistical whitening transformations play a fundamental role in many computational systems, and may also play an important role in biological sensory systems. Individual neurons appear to rapidly and reversibly alter their input-output gains, approximately normalizing the variance of their responses. Populations of neurons appear to regulate their joint responses, reducing correlations between neural activities. It is natural to see whitening as the objective that guides these behaviors, but the mechanism for such joint changes is unknown, and direct adjustment of synaptic interactions would seem to be both too slow, and insufficiently reversible. Motivated by the extensive neuroscience literature on rapid gain modulation, we propose a recurrent network architecture in which joint whitening is achieved through modulation of gains within the circuit. Specifically, we derive an online statistical whitening algorithm that regulates the joint second-order statistics of a multi-dimensional input by adjusting the marginal variances of an *overcomplete* set of interneuron projections. The gains of these interneurons are adjusted individually, using only local signals, and feed back onto the primary neurons. The network converges to a state in which the responses of the primary neurons are whitened. We demonstrate through simulations that the behavior of the network is robust to poor conditioning or noise when the gains are sign-constrained, and can be generalized to achieve a form of local whitening in convolutional populations, such as those found throughout the visual or auditory system.

1. Introduction

Statistical whitening transformations, in which multi-dimensional inputs are decorrelated and normalized to have unit variance, are common in statistical signal processing and machine learning systems. For example, they provide a common step in statistical factorization methods (Hyvärinen & Oja, 2000) and are often used as a preprocessing step for training deep networks (Krizhevsky, 2009). Empirical evidence shows that statistical whitening improves unsupervised feature learning (Coates et al., 2011). More recently, self-supervised learning methods have used statistical whitening or related decorrelation transformations to prevent representational collapse (Ermolov et al., 2021; Zbontar et al., 2021; Bardes et al., 2021; Hua et al., 2021). Whitening in neural networks is often performed in the offline setting. However, online methods are useful, especially when the inputs are from dynamic environments.

In early sensory systems, which receive inputs from dynamic environments, changes in sensory input statistics induce rapid changes in the input-output gains of single neurons, allowing cells to normalize their output variance (Fairhall et al., 2001; Nagel & Doupe, 2006). This is hypothesized to enable maximal information transmission (Barlow, 1961; Laughlin, 1981; Fairhall et al., 2001). At the population level, whitening and related adaptive decorrelation transformations have been reported in sensory areas such as the early visual cortex of cats (Benucci et al., 2013) and the olfactory bulb in zebrafish (Friedrich, 2013; Wanner & Friedrich, 2020) and mice (Giridhar et al., 2011; Gschwend et al., 2015). However, the mechanisms underlying such whitening behaviors are unknown, and would seem to require coordination among all pairs of neurons, as opposed to the single-neuron case which relies only on gain rescaling.

Here, motivated by the large neuroscience literature on rapid gain modulation, we propose a novel recurrent network architecture for statistical whitening that exclusively relies on gain modulation. In particular, we introduce a novel objective for statistical whitening that is expressed solely in terms of the *marginal* variances of an overcomplete representation of the input signal. We derive a recurrent circuit to optimize the objective, and show that it corresponds to a network comprising primary neurons and an auxiliary population of interneurons with scalar *gain modulation*. Importantly, the

[*]Equal contribution [1]Center for Neural Science, New York University, New York, NY [2]Center for Computational Neuroscience, Flatiron Institute, New York, NY [3]Neuroscience Institute, New York University School of Medicine, New York, NY. Correspondence to: Lyndon R. Duong <lyndon.duong@nyu.edu>, David Lipshutz <dlipshutz@flatironinstitute.org>.

Under review.

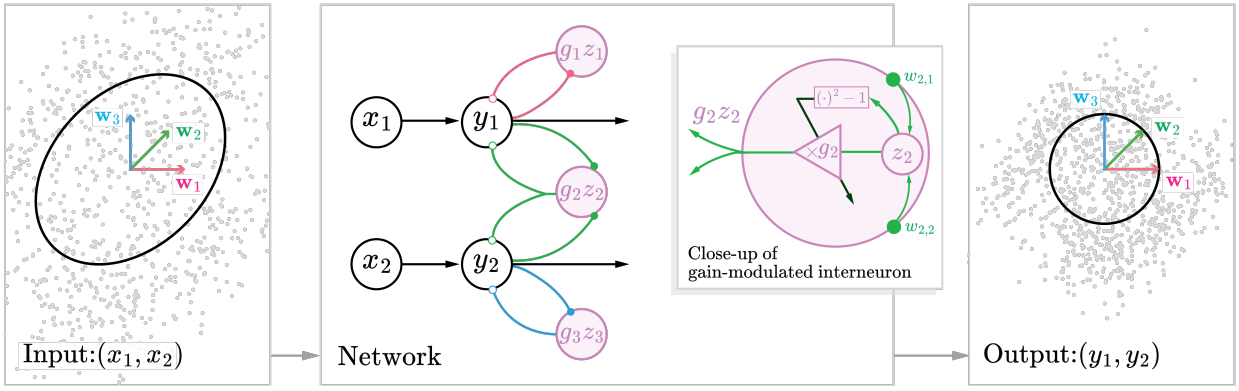

Figure 1. Schematic of a recurrent statistical whitening network with 2 primary neurons and 3 interneurons. **Left**: 2D Scatter plot of the (non-Gaussian) network inputs $\mathbf{x} = (x_1, x_2)$ whose covariance is the ellipse. **Center**: Primary neurons, whose outputs are $\mathbf{y} = (y_1, y_2)$, receive external feedforward inputs, $\mathbf{x}$, and recurrent feedback inputs from an auxiliary population of interneurons, $-\sum_{i=1}^{3} g_i z_i \mathbf{w}_i$. Linear projection vectors $\{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3\} \in \mathbb{R}^2$ encode non-negative feedforward synaptic weights connecting the primary neurons to interneuron $i = 1, 2, 3$ (symmetric weights are used for feedback connections). The weights are shown in the left and right panels with corresponding colors. **Inset:** The i^{th} interneuron (e.g. here $i = 2$) receives input $z_i = \mathbf{w}_i^\top \mathbf{y}$, which is multiplied by its gain g_i to produce output $g_i z_i$. Its gain, g_i, is adjusted s.t. $\Delta g_i \propto z_i^2 - 1$. The dark arrow indicates that the gain update operates on a slower time scale. **Right:** Scatter plots of the whitened network outputs $\mathbf{y}$. Outputs have unit variance along all $\mathbf{w}_i$'s, which is equivalent to having identity covariance matrix, i.e., $\mathbf{C}_{yy} = \mathbf{I}_N$ (black circle).

network operates online, and its responses converge to the classical ZCA whitening solution without supervision or backpropagation. To demonstrate potential applications of this framework, we show that gain modulation serves as an implicit gating mechanism, which facilitates fast context-dependent whitening. Further, we show how non-negative gain modulation provides a novel approach for dealing with ill-conditioned or noisy data. Finally, we relax the overcompleteness constraint in our objective and provide a method for local decorrelation of convolutional populations.

2. A novel objective for ZCA whitening

Consider a neural network with N primary neurons. For each $t = 1, 2, \ldots$, let $\mathbf{x}_t$ and $\mathbf{y}_t$ be N-dimensional vectors whose components respectively denote the inputs and outputs of the primary neurons at time t, Figure 1. Without loss of generality we assume the inputs $\mathbf{x}_t$ are centered.

2.1. Conventional objective

Statistical whitening aims to linearly transform inputs $\mathbf{x}_t$ so that the covariance of the outputs $\mathbf{y}_t$ is identity, i.e.,

$$\mathbf{C}_{yy} = \langle \mathbf{y}_t \mathbf{y}_t^\top \rangle_t = \mathbf{I}_N, \tag{1}$$

where $\langle \cdot \rangle_t$ denotes the expectation operator over t, and $\mathbf{I}_N$ denotes the $N \times N$ identity matrix (see Appendix A for a list of notation used in this work).

It is well known that whitening is not unique: any orthogonal rotation of a random vector with identity covariance matrix also has identity covariance matrix. There are several common choices to resolve this rotational ambiguity, each with their own advantages (Kessy et al., 2018). Here, we focus on the popular whitening transformation called Zero-phase Component Analysis (ZCA) whitening or Mahalanobis whitening, which is the whitening transformation that minimizes the mean-squared error between the inputs and the whitened outputs (alternatively, the one whose transformation matrix is symmetric). Mathematically, the ZCA-whitened outputs are the optimal solution to the minimization problem

$$\min_{\{\mathbf{y}_t\}} \langle \|\mathbf{x}_t - \mathbf{y}_t\|_2^2 \rangle_t \quad \text{s.t.} \quad \langle \mathbf{y}_t \mathbf{y}_t^\top \rangle_t = \mathbf{I}_N, \tag{2}$$

where $\|\cdot\|_2$ denotes the Euclidean norm on $\mathbb{R}^N$. Assuming the covariance of the inputs $\mathbf{C}_{xx} := \langle \mathbf{x}_t \mathbf{x}_t^\top \rangle_t$ is positive definite, the unique solution to the optimization problem in Equation 2 is $\mathbf{y}_t = \mathbf{C}_{xx}^{-1/2} \mathbf{x}_t$ for $t = 1, 2, \ldots$, where $\mathbf{C}_{xx}^{-1/2}$ is the inverse matrix square root of $\mathbf{C}_{xx}$.

Equation 2 provides a starting point for deriving online ZCA whitening algorithms that can be implemented with recurrent neural networks that learn by updating their synaptic weights (Pehlevan & Chklovskii, 2015).

2.2. A novel objective using *marginal* statistics

We formulate a novel objective for learning the ZCA whitening transform via gain modulation. Our innovation exploits the fact that a random vector has identity covariance matrix (i.e., Equation 1 holds) if and only if it has unit marginal

variance along all possible 1D projections (a form of tomography; see Related Work). We can derive a tighter statement, that holds for a finite but *overcomplete* set of at least $K \geq K_N := N(N+1)/2$ distinct axes ('overcomplete' simply means that the number of axes exceeds the dimensionality of the input, i.e., $K > N$). Intuitively, this equivalence holds because an $N \times N$ symmetric matrix has K_N degrees of freedom, so the marginal variances along $K \geq K_N$ distinct axes are sufficient to constrain the $N \times N$ (symmetric) covariance matrix. We formalize this equivalence in the following proposition, whose proof is provided in Appendix B.

Proposition 2.1. *Fix $K \geq K_N$. Suppose $\mathbf{w}_1, \ldots, \mathbf{w}_K \in \mathbb{R}^N$ are unit vectors*[1] *such that*

$$\textit{span}(\{\mathbf{w}_1\mathbf{w}_1^\top, \ldots, \mathbf{w}_K\mathbf{w}_K^\top\}) = \mathbb{S}^N, \tag{3}$$

where $\mathbb{S}^N$ denotes the K_N-dimensional vector space of $N \times N$ symmetric matrices. Then Equation 1 holds if and only if the projection of $\mathbf{y}_t$ onto each unit vector $\mathbf{w}_1, \ldots, \mathbf{w}_K$ has unit variance, i.e.,

$$\langle(\mathbf{w}_i^\top \mathbf{y}_t)^2\rangle_t = 1 \quad \textit{for} \quad i = 1, \ldots, K. \tag{4}$$

Assuming Equation 3 holds, we can interpret the set of vectors $\{\mathbf{w}_1, \ldots, \mathbf{w}_K\}$ as a *frame* (i.e., an overcomplete basis; Casazza et al., 2013) in $\mathbb{R}^N$ such that the covariance of the outputs $\mathbf{C}_{yy}$ can be computed from the variances of the K-dimensional projection onto the set of frame vectors. Thus, we can replace the whitening constraint in Equation 2 with the equivalent *marginal variance* constraint to obtain the following objective:

$$\min_{\{\mathbf{y}_t\}} \langle\|\mathbf{x}_t - \mathbf{y}_t\|_2^2\rangle_t \quad \text{s.t.} \quad \text{Equation 4 holds.} \tag{5}$$

3. A recurrent neural network with gain adaptation for ZCA whitening

In this section, we derive an online algorithm for solving the optimization problem in Equation 5 and map the algorithm onto a recurrent neural network with gain modulation. We first introduce Lagrange multipliers to enforce the constraints, which transforms the minimization problem into a minimax problem. We then solve the minimax problem by taking stochastic gradient steps.

Assume we have an overcomplete frame $\{\mathbf{w}_1, \ldots, \mathbf{w}_K\}$ in $\mathbb{R}^N$ satisfying Equation 3. We concatenate the frame vectors into an $N \times K$ matrix $\mathbf{W} := [\mathbf{w}_1, \ldots, \mathbf{w}_K]$. In our network, primary neurons project onto the layer of K interneurons with the synaptic weights representing matrix $\mathbf{W}$. Then, the post-synaptic currents in interneurons at time t encode the K-dimensional vector $\mathbf{z}_t := \mathbf{W}^\top \mathbf{y}_t$ (Figure 1). We emphasize that the synaptic weight matrix $\mathbf{W}$ will remain *fixed* in our whitening algorithm.

[1]The unit-length assumption is without loss of generality and is imposed here for notational convenience.

3.1. Enforcing the marginal variance constraints with scalar gains

We introduce Lagrange multipliers $g_1, \ldots, g_K \in \mathbb{R}$ to enforce the K constraints in Equation 4. We concatenate the Lagrange multipliers into the K-dimensional vector $\mathbf{g} := [g_1, \ldots, g_K]^\top \in \mathbb{R}^K$, and formulate the problem as a saddle point optimization,

$$\max_{\mathbf{g}} \min_{\{\mathbf{y}_t\}} \langle \ell(\mathbf{x}_t, \mathbf{y}_t, \mathbf{g})\rangle_t, \tag{6}$$

$$\text{where } \ell(\mathbf{x}, \mathbf{y}, \mathbf{g}) := \|\mathbf{x} - \mathbf{y}\|_2^2 + \sum_{i=1}^K g_i \left\{(\mathbf{w}_i^\top \mathbf{y})^2 - 1\right\}.$$

Here, we have interchanged the order of maximization over $\mathbf{g}$ and minimization over $\mathbf{y}_t$, which is justified because $\ell(\mathbf{x}_t, \mathbf{y}_t, \mathbf{g})$ is convex in $\mathbf{y}_t$ and linear in $\mathbf{g}$, see Appendix C.

In our neural network implementation, g_i will correspond to the multiplicative gain associated with the i^{th} interneuron, so that its output at time t is $g_i z_{i,t}$ (Figure 1, Inset). From Equation 6, we see that the gain of the i^{th} interneuron, g_i, enforces the marginal variance of $\mathbf{y}_t$ along the axis spanned by $\mathbf{w}_i$ to be unity. Importantly, the gains are not hyperparameters, but rather they are optimization variables which promote statistical whitening of $\{\mathbf{y}_t\}$, preventing the neural outputs from trivially matching the inputs $\{\mathbf{x}_t\}$.

3.2. Deriving recurrent neural network update rules

To solve Equation 6 in the online setting, we assume there is a time-scale separation between 'fast' neural dynamics and 'slow' gain updates, so that at each time step the neural dynamics equilibrate before the gains are adjusted. This allows us to perform the inner minimization over $\{\mathbf{y}_t\}$ before the outer maximization over the gains. In biological neural networks, this is justifiable because a given neuron's activations (i.e. action potential firing) operate on a much more rapid time-scale than its intrinsic input-output gain, which is driven by slower processes such as changes in calcium ion concentration gradients (Ferguson & Cardin, 2020).

3.2.1. Fast neural activity dynamics

For each time step $t = 1, 2, \ldots$, we minimize the objective $\ell(\mathbf{x}_t, \mathbf{y}_t, \mathbf{g})$ over $\mathbf{y}_t$ by recursively running gradient-descent steps to equilibrium:

$$\begin{aligned}\mathbf{y}_t &\leftarrow \mathbf{y}_t - \frac{\gamma}{2}\nabla_{\mathbf{y}}\ell(\mathbf{x}_t, \mathbf{y}_t(\tau), \mathbf{g}) \\ &= \mathbf{y}_t + \gamma\left\{\mathbf{x}_t - \mathbf{W}(\mathbf{g} \circ \mathbf{z}_t) - \mathbf{y}_t\right\},\end{aligned} \tag{7}$$

where $\gamma > 0$ is a small constant, the circle '$\circ$' denotes the Hadamard (element-wise) product, $\mathbf{g} \circ \mathbf{z}_t$ is a vector of K

gain-modulated interneuron outputs, and we assume the primary cell outputs are initialized at zero.

We see from the right-hand-side of Equation 7 that the 'fast' dynamics of the primary neurons are driven by three terms (inside the curly braces): i) constant feedforward external input $\mathbf{x}_t$; ii) recurrent gain-modulated feedback from interneurons $-\mathbf{W}(\mathbf{g} \circ \mathbf{z}_t)$; and iii) a leak term $-\mathbf{y}_t$. Because the neural activity dynamics are linear, we can analytically solve for their equilibrium (i.e. steady-state), $\bar{\mathbf{y}}_t$, by setting the update in Equation 7 to zero:

$$\begin{aligned}\bar{\mathbf{y}}_t &= \left[\mathbf{I}_N + \mathbf{W}\,\mathrm{diag}\,(\mathbf{g})\,\mathbf{W}^\top\right]^{-1}\mathbf{x}_t \\ &= \left[\mathbf{I}_N + \sum_{i=1}^{K} g_i \mathbf{w}_i \mathbf{w}_i^\top\right]^{-1}\mathbf{x}_t, \end{aligned} \tag{8}$$

where $\mathrm{diag}\,(\mathbf{g})$ denotes the $K \times K$ diagonal matrix whose $(i,i)^{\text{th}}$ entry is g_i, for $i = 1, \ldots, K$. The equilibrium feedforward interneuron inputs are then given by

$$\bar{\mathbf{z}}_t = \mathbf{W}^\top \bar{\mathbf{y}}_t. \tag{9}$$

The gain-modulated outputs of the K interneurons, $\mathbf{g} \circ \mathbf{z}_t$, are then projected back onto the primary cells via symmetric weights, $-\mathbf{W}$ (Figure 1).

3.2.2. Slow gain dynamics

After the fast neural activities reach steady-state, the interneuron gains are updated by taking a stochastic gradient-ascent step with respect to $\mathbf{g}$:

$$\begin{aligned}\mathbf{g} &\leftarrow \mathbf{g} + \frac{\eta}{2}\nabla_{\mathbf{g}}\ell(\mathbf{x}_t, \bar{\mathbf{y}}_t, \mathbf{g}) \\ &= \mathbf{g} + \eta\left(\bar{\mathbf{z}}_t^{\circ 2} - \mathbf{1}\right), \end{aligned} \tag{10}$$

where $\eta > 0$ is the learning rate, the superscript '$\circ 2$' denotes the element-wise squaring operation (i.e., $\bar{\mathbf{z}}_t^{\circ 2} = [\bar{z}_{t,1}^2, \ldots, \bar{z}_{t,K}^2]^\top$) and $\mathbf{1} = [1, \ldots, 1]^\top$ is the K-dimensional vector of ones[2]. Remarkably, the update to the i^{th} interneuron's gain g_i (Equation 10) depends only on the online estimate of the *variance* of its equilibrium input $\bar{z}_{t,i}^2$, and its distance away from the target variance, 1. Networks such as these which adapt using only local signals to each interneuron are suitable candidates for hardware implementations using low-power neuromorphic chips (Pehlevan & Chklovskii, 2019). Thus, although statistical whitening inherently requires a *joint* transformation in response to joint statistics, our recurrent network solution operates solely using single-neuron gain changes in response to *marginal* statistics.

[2]Appendix D generalizes the gain update to allowing for temporal-weighted averaging of the variance over past samples.

3.2.3. Online unsupervised algorithm

By combining Equations 7 – 10, we arrive at our online recurrent neural network algorithm for statistical whitening via gain modulation (Algorithm 1). We also provide batched and offline versions of the algorithm in Appendix E.

Algorithm 1 Online ZCA whitening via gain modulation

1: **Input:** Centered inputs $\mathbf{x}_1, \mathbf{x}_2, \cdots \in \mathbb{R}^N$
2: **Initialize:** $\mathbf{W} \in \mathbb{R}^{N \times K}$; $\mathbf{g} \in \mathbb{R}^K$; $\eta, \gamma > 0$
3: **for** $t = 1, 2, \ldots$ **do**
4: $\quad \mathbf{y}_t \leftarrow \mathbf{0}$
5: $\quad \{$Run $\mathbf{y}_t$ and $\mathbf{z}_t$ dynamics to equilibrium$\}$
6: $\quad$ **while** not converged **do**
7: $\quad\quad \mathbf{z}_t \leftarrow \mathbf{W}^\top \mathbf{y}_t$
8: $\quad\quad \mathbf{y}_t \leftarrow \mathbf{y}_t + \gamma\left\{\mathbf{x}_t - \mathbf{W}(\mathbf{g} \circ \mathbf{z}_t) - \mathbf{y}_t\right\}$
9: $\quad$ **end while**
10: $\quad \mathbf{g} \leftarrow \mathbf{g} + \eta\left(\mathbf{z}_t^{\circ 2} - \mathbf{1}\right)$ $\{$Update gains$\}$
11: **end for**

There are a few points worth noting about this network:

- The weights $\mathbf{W}$ remain *fixed* in Algorithm 1. Rather, the gains $\mathbf{g}$ adapt to statistically whiten the outputs. This allows the whitening to be easily adjusted and reversed, by simply returning the gains to their default states.
- While the objective is effectively in the form of an auto-encoding loss function involving an ℓ_2 reconstruction term (Eq. 6), the recurrent network never explicitly reconstructs its inputs.
- Since all recurrent dynamics are linear, it is possible to bypass the inner loop representing the fast dynamics of the primary cells (lines 6 – 9 of Algorithm 1), by directly computing the equilibrium responses of $\bar{\mathbf{y}}_t$, and $\bar{\mathbf{z}}$ directly (Eqs. 8, 9).

4. Numerical experiments and applications

We provide different applications of our recurrent ZCA whitening network via gain modulation. In particular, we emphasize that gain adaptation is distinct from, while also complementary to, a synaptic weight learning. We therefore side-step the goal of learning the frame $\mathbf{W}$, and assume it is known. This allows us to decouple and analyze the general properties of our proposed gain modulation framework, independently from the choice of frame.

4.1. Gain modulation: a new solution to ZCA whitening

We first demonstrate that our algorithm succeeds in yielding statistically whitened outputs. We simulated a network with interneuron weights, $\mathbf{W}$, as illustrated in Figure 1 (N=2,

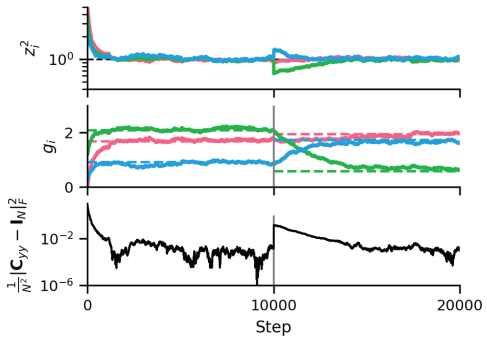

Figure 2. Network from Figure 1 (with corresponding colors; N=2, K=K_N=3, η=2E-3) whitening to two randomly generated statistical contexts online (10K steps each). **Top:** Marginal variances (log scale) measured by interneurons approach 1 over time. **Middle:** Dynamics of interneuron gains, which are applied to z_i before feeding back onto the primary cells. Dashed lines are optimal gains (Appendix F). **Bottom:** Whitening error over time.

K=K_N=3). Figure 2 shows network adaptation to inputs from two contexts with randomly generated underlying input covariances $\mathbf{C}_{xx}$ (10K gain update steps each). As update steps progress, all marginal variances converge to unity, as expected from the objective (top panel). To achieve ZCA whitening at equilibrium, then $\mathbf{I}_N + \sum_{i=1}^{K} g_i \mathbf{w}_i \mathbf{w}_i^\top = \mathbf{C}_{xx}^{1/2}$ (Equation 8). When the number of interneurons satisfies K=K_N, the optimal gains to achieve ZCA whitening can be solved analytically (see Appendix F for details). These are displayed as dashed lines in the (middle panel). We found that the network successfully adapted to the two random statistical contexts, and converged to the optimal set of gains to achieve whitened $\mathbf{y}_t$ (Figure 2). Accordingly, the whitening error, as measured by the Frobenius norm between $\mathbf{C}_{yy}$ and $\mathbf{I}_N$, approached zero (bottom panel). Thus, with each interneuron monitoring their respective *marginal* input variances z_i^2, and re-scaling their input-output gains to modulate feedback onto the primary neurons, the network succeeded in adapting to each context and yielded whitened outputs.

4.2. Rate of convergence depends on frame W

Thus far, we have assumed the frame, $\mathbf{W}$, was fixed and known (e.g., optimized through pre-training or long timescale development). This distinguishes our method from existing ZCA whitening methods, which typically operate by estimating the eigenvectors of the data. By contrast, our network obviates learning the principal axes of the data altogether, and instead uses a statistical sampling approach along a fixed set of measurement axes.

If the number of interneurons K=K_N, their gains will descend the gradient of the objective (Equation 10), and by Proposition Theorem 2.1, the outputs will become whitened. We were interested in how effectively the network whitened randomly sampled inputs with fixed input covariance depending on its initialization. Figure 3 summarizes an empirical convergence test of 100 networks where $N = 2$ with three different kinds of frame $\mathbf{W} \in \mathbb{R}^{N \times K_N}$: i) with i.i.d. Gaussian entries ('Random'); ii) through an optimization procedure that finds a frame whose columns have minimum mutual coherence and cover the ambient space ('Optimized'); and iii) a frame whose first N columns were the eigenvectors of the data and the remaining $K_N - N$ columns were random Gaussian entries ('Spectral'). For clarity, we have removed the effects of sampling stochasticity by running the offline version of our network, which assumes having direct access to the input covariance (Appendix E); the online version was qualitatively similar.

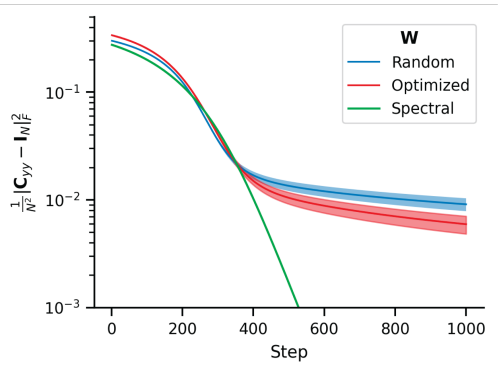

Figure 3. Convergence depends on qualitative structure of $\mathbf{W}$. Networks each had N=2, K=K_N=3, η=5E-3. Shaded error regions are standard errors over the 100 repeats.

The Spectral frame defines a bound on achievable performance, converging much faster than the Random and Optimized frames. This is because the interneuron axes were aligned with the input's principal axes, and a simple gain scaling along those directions is the optimal whitening solution. Interestingly, we found that networks with optimized weights systematically converged faster than randomly-initialized frames. These results indicate that the choice of frame *does* in fact play an important role in the effectiveness of our algorithm. Namely, increased coverage of the space by the frame vectors facilitates whitening with our gain re-scaling mechanism. The random sampling approach has little hope of scaling to high dimensional inputs, and the green line in Figure 3 shows that one would benefit from aligning the frame vectors to the principal axes of the inputs.

4.3. Implicit gating via gain modulation

Motivated by the findings in Figure 3, we wished to demonstrate a way in which our adaptive gain modulation network could complement or augment a network in which context-dependent *weights* have already been learned. We performed an experiment involving a network with 'pre-trained' $\mathbf{W}$ (N=6, K=K_N=21) whitening inputs from

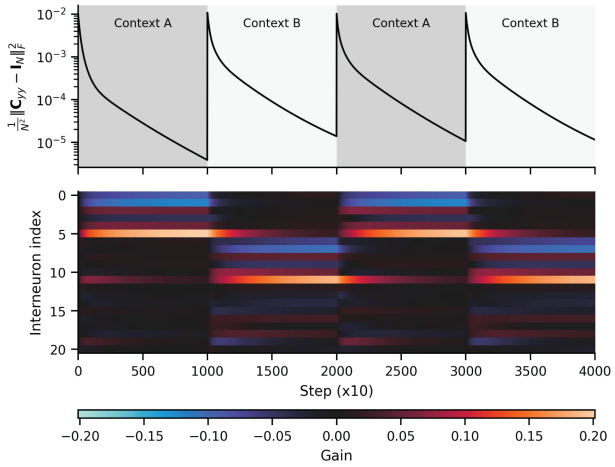

Figure 4. Gains can act as an implicit gating mechanism. **Top**: Whitening error over time with a network (N=6; K_N=21; η=1E-3) adapting to 2 alternating statistical contexts A and B, with different input covariances for 10K steps each. $\mathbf{W}$ was initialized as a Spectral frame, with the first $2N$ columns set to be the eigenvectors of covariances of contexts A and B, respectively. **Bottom**: Gains can be seen to act as switches for context, gating the spectral components to optimally whiten each context.

two alternating statistical contexts, A and B, for 10K steps each. The frame was constructed such that the first and second N columns were the eigenvectors of context A and B's covariance, respectively, and the remaining $K - 2N$ columns' elements were random i.i.d. Gaussian. Figure 4 (top panel) shows that the network adaptively whitens the inputs from each successive context. Surprisingly, upon closer inspection to the K interneurons' gains over time (bottom panel) showed that they approximately served to 'select' the frame vectors corresponding to the eigenvectors of each respective condition (as indicated by the blue/red intensity on the figure). Our gain modulation framework thus serves as an effective means of *gating* context-dependent information without an explicit context signal.

4.4. Normalizing ill-conditioned data

When inputs are low-rank, $\mathbf{C}_{xx}$ is ill-conditioned (Figure 5A), and whitening can amplify directions of small variance that are due to noise. In this section, we show how our gain-modulating network can be simply modified to handle these types of inputs. To prevent amplification of inputs below a certain threshold, we can replace the unit marginal variance equality constraints with upper bound constraints:

$$\langle(\mathbf{w}_i^\top \mathbf{y}_t)^2\rangle_t \leq 1 \quad \text{for} \quad i = 1, \ldots, K. \tag{11}$$

Our modified network objective then becomes

$$\min_{\{\mathbf{y}_t\}} \langle \|\mathbf{x}_t - \mathbf{y}_t\|_2^2 \rangle_t \quad \text{s.t.} \quad \text{Equation 11 holds.} \tag{12}$$

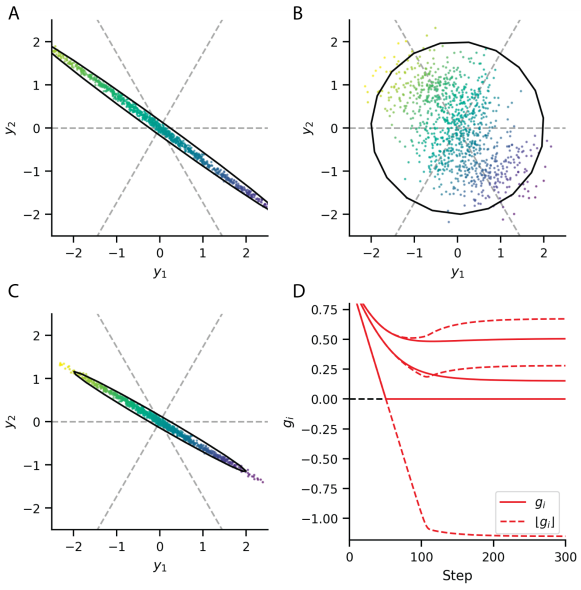

Figure 5. Two networks (N=2, K=3, η=0.02) whitening ill-conditioned inputs. **A:** Outputs without whitening. 2D scatterplot of a non-Gaussian density whose underlying signal lies close to a latent 1D axis. The signal magnitude along that axis is denoted by the colors. The covariance matrix is depicted as a black ellipse. Gray dashed lines are axes spanned by $\mathbf{W}$ (here chosen to be an equi-angular frame). **B:** ZCA whitening boosts small-amplitude noise lying along the uninformative direction. **C:** Modulating gains according to Eq. 14 rescales the data *without* amplifying noise. **D:** Gains updated with Eq. 10 (solid) vs. Eq. 14 (dashed).

Intuitively, if the projected variance along a given direction is already less than or equal to unity, then it will not affect the overall loss. To enforce the upper bound constraints, we introduce gains as Lagrange multipliers as before, but restrict the domain of $\mathbf{g}$ to be the non-negative orthant $\mathbb{R}_+^K$, resulting in non-negative optimal gains:

$$\max_{\mathbf{g}\in\mathbb{R}_+^K} \min_{\{\mathbf{y}_t\}} \langle \ell(\mathbf{x}_t, \mathbf{y}_t, \mathbf{g})\rangle_t, \tag{13}$$

where $\ell(\mathbf{x}, \mathbf{y}, \mathbf{g})$ is defined as in Equation 6. At each time step t, we optimize Equation 13 by first taking gradient-descent steps with respect to $\mathbf{y}_t$, resulting in the same neural dynamics (Equation 7) and equilibrium solution (Equation 8) as before. After the neural activities equilibrate, we take a *projected* gradient-ascent step with respect to $\mathbf{g}$:

$$\mathbf{g} \leftarrow \lfloor \mathbf{g} + \eta(\bar{\mathbf{z}}_t^{\circ 2} - \mathbf{1}) \rfloor \tag{14}$$

where $\lfloor \cdot \rfloor$ denotes the element-wise half-wave rectification operation that projects its inputs onto the positive orthant $\mathbb{R}_+^K$, i.e., $\lfloor \mathbf{v} \rfloor := [\max(v_1, 0), \ldots, \max(v_K, 0)]^\top$.

We simulated a network with gains set to either updates using unconstrained gains (Equation 10), or rectified gains (Equation 14), and observed that these two models converged to two different solutions (Figure 5B, C). When

g_i was not constrained to be non-negative, the network achieved global whitening, as before. By contrast, the gains constrained to be non-negative converged to different values altogether, with one of them converging to zero rather than becoming negative. The whitening error for this network unsurprisingly converged to a non-zero value with the non-negative gain constraint. Thus, with a non-negative constraint, the network failed to fully whiten $\mathbf{y}$, but in doing so, it *did not amplify the noise*. In Appendix G we show additional cases that provide further geometric intuition on differences between ZCA whitening and non-negative gain constrained ZCA whitening with our network.

4.5. Gain modulation enables local spatial decorrelation

The requirement of K_N interneurons to ensure a statistically white output becomes prohibitively costly for high-dimensional inputs due to the number of interneurons scaling as $\mathcal{O}(N^2)$. This led us to ask: how many interneurons are needed in practice? For natural sensory inputs such as images, it is well known that inter-pixel correlation is highly structured, decaying as a function of distance. Using a Gaussian random walk, we simulated gaze fixation and micro-saccadic eye movements, drawing 12×12 patch samples from a natural image (Figure 6A; Hateren & Schaaf, 1998). We did this for different randomly selected regions of the image (colors). The content of each region is quite different, but the inter-pixel correlation within each context fell rapidly with distance (Figure 6B).

We *relaxed* the $\mathcal{O}(N^2)$ marginal variance constraint to instead target whitening of *spatially local neighborhoods* of primary neurons with image patch inputs. That is, the frame $\mathbf{W}$ spanned $K < K_N$ axes in $\mathbb{R}^N$, but was constructed such that *overlapping* neighborhoods of 4×4 primary neurons were decorrelated, each by a population of interneurons that was 'overcomplete' with respect to that neighborhood (see Appendix H for frame construction details). Importantly, taking into account convolutional structure dramatically reduces the interneuron complexity from $\mathcal{O}(N^2) \rightarrow \mathcal{O}(N)$ (Appendix H). This frame is still overcomplete ($K > N$), but because $K<K_N$, we no longer guarantee at equilibrium that $\mathbf{C}_{yy} = \mathbf{I}_N$.

After running this local whitening network on the inputs drawn from the red context, we found that (Figure 6C): i) inter-pixel correlations drop within the region specified by the local neighborhood; and ii) surprisingly, correlations at longer-range are dramatically reduced. Accordingly, the covariance eigenspectrum of the locally whitened outputs was significantly flatter compared to the inputs (Figure 6D left vs. right columns). We also provide a 1D example in Appendix H. We remark that this empirical result is not at all obvious – that whitening individual *overlapping* neighborhoods of neurons should produce a more globally whitened output covariance. Indeed, studying whether and when a globally whitened solution is possible from whitening of spatial overlapping neighborhoods is an interesting problem that is worth pursuing.

5. Related work

5.1. Biologically plausible whitening networks

Biological circuits operate in the online setting and, due to physical constraints, learn exclusively using local signals. Therefore, to plausibly model neural computation, a neural network model must operate in the online setting (i.e., streaming data) and use local learning rules. There are a few existing normative models of statistical whitening and related transformations; however, these models use synaptic plasticity mechanisms (i.e., changing $\mathbf{W}$) to adapt to changing input statistics (Pehlevan & Chklovskii, 2015; Pehlevan et al., 2017; Chapochnikov et al., 2021; Lipshutz et al., 2022). Adaptation of neural population responses to changes in sensory inputs statistics occurs rapidly, on the order of seconds (Benucci et al., 2013; Wanner & Friedrich, 2020), so it could potentially be accounted for by short-term synaptic plasticity, which operates on the timescale of tens of milliseconds to minutes (Zucker et al., 2002), but not by long-term synaptic plasticity, which operates on the timescale of minutes or longer (Martin et al., 2000). Here, we explore the alternative hypothesis that modulation of neural gains, which operates on the order of tens of milliseconds to minutes (Fairhall et al., 2001), facilitates rapid adaptation of neural populations to changing input statistics.

5.2. Tomography and "sliced" density measurements

Our leveraging of 1D projections to compute the ZCA whitening transform is reminiscent of approaches taken in the field of tomography. Geometrically, our method represents an ellipsoid (i.e., the N dimensional covariance matrix) using noisy 1D projections of the ellipsoid onto axes spanned by frame vectors (i.e., estimates of the marginal variances). This is a special case of reconstruction problems that have been studied in geometric tomography (Karl et al., 1994; Gardner, 1995). An important distinction between tomographic reconstruction and our solution to ZCA whitening is that we are not using the 1D projections to reconstruct the multi-dimensional inputs; instead, we are utilizing the univariate measurements to transform the ellipsoid into a new shape (a hyper-sphere, in the case of whitening).

In optimal transport, "sliced" methods offer a way to measure otherwise intractable p-Wasserstein distances in high dimensions (Bonneel et al., 2015), thereby enabling its use in optimization loss functions. Sliced methods compute Wasserstein distance by repeatedly taking series of 1D projections of two densities, then computing the expectation

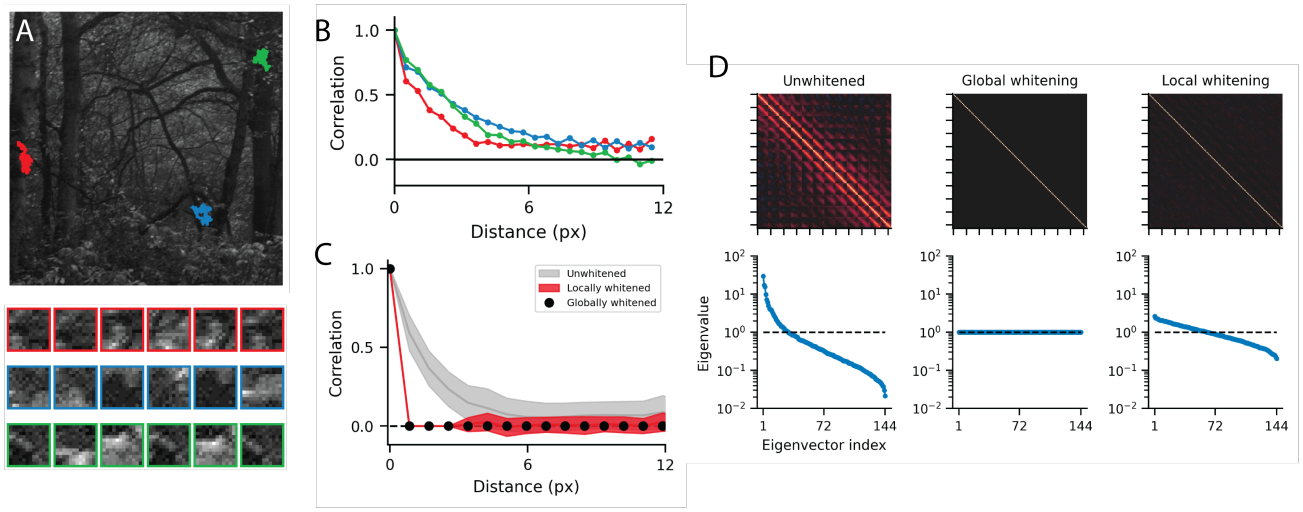

Figure 6. Local spatial whitening. **A)** Large grayscale image from which 12×12 image patch samples are drawn. Colors represent random-walk sampling from regions of the image corresponding to contexts with different underlying statistics. Six samples from each context are shown below. **B)** Without whitening, mean pairwise output pixel correlations decay rapidly with spatial distance in each context, suggesting that local whitening may be effective. **C)** Pairwise output pixel correlation of patches from the red context before (gray) and after global (black dots) vs. convolutional whitening with overlapping 4×4 neighborhoods (red). Shaded regions represent standard deviations. **D)** Top: Expected correlation matrices of all flattened patches of the red context before whitening, and after global/local ZCA whitening. Correlation and not covariance matrices are displayed here to facilitate comparison; all panels use the same color scale. Bottom: Corresponding covariance eigenspectra.

over all 1D Wasserstein distances, for which there exists an analytic solution. Notably, the 2-Wasserstein distance between a 1D zero-mean Gaussian with variance σ^2 and a standard normal (i.e. white) density is

$$W_2\left(\mathcal{N}\left(0,\sigma^2\right);\mathcal{N}\left(0,1\right)\right) = \|\sigma - 1\|\,.$$

Comparing this with the rule by which we update each interneuron gain, $g_i \leftarrow g_i + \eta((\mathbf{w}_i^\top \bar{\mathbf{y}}_t)^2 - 1)$ (Equation 10), reveals striking similarity between our recurrent neural network and methods optimizing using sliced Wasserstein distances. However, distinguishing characteristics of our approach include: 1) minimizing distance between univariate *variances* rather than standard deviations; 2) the directions along which we compute slices (columns of $\mathbf{W}$) are fixed, whereas sliced methods typically compute a new set of random projections at each optimization step; 3) most importantly, our network operates online, and minimizes sliced variance distances *without* backpropagation.

6. Discussion

We have derived a novel family of recurrent models for whitening, which use *gain modulation* to transform joint second-order statistics of their inputs based on *marginal* variance measurements. We showed that, given sufficiently many marginal measurements along unique axes, the network will produce ZCA whitened outputs. In particular, our objective (Equation 5) provides an elegant way to think about the classical problem of statistical whitening, and draws connections to old concepts in tomography and transport theory. The framework developed here is flexible, with several generalizations or extensions that we omitted due to space limitations. For example, by replacing the unity marginal variance constraint by a set of target variances differing from 1, the network can be used to transform (i.e. transport) its input density to one matching the corresponding (non-white) covariance.

Modulating feature gains has proven effective in adapting pre-trained neural networks to novel inputs with out-of-training distribution statistics (Ballé et al., 2020; Duong et al., 2022; Mohan et al., 2021). In fact, adaptive gain modulation is an old concept in neuroscience which we believe would be of importance to the broader machine learning community. In real neural networks, there exist several computational processes operating concurrently at different time-scales. Examples include synaptic weights encoding long-term information, while faster processes like gain modulation facilitate rapid adaptation to different contexts. Indeed, the demonstrations in this study were largely agnostic to the exact structure of the weights $\mathbf{W}$, and instead focused on the computational role of adaptive gain modulation itself. We showed how gains can adaptively decorrelate a network's outputs *without* modifying its pre-trained weights in an online setting. Specifically, we showed that gain modulation: 1) enables fast switching between pre-learned context-dependent weight regimes; 2) can be used in conjunction with properly-aligned interneuron projection weights to handle ill-conditioned inputs; and 3) reduce

long-range dependencies by modifying *local* signals.

Feature whitening and decorrelation has become an important objective constraint in self-supervised contrastive learning methods to help prevent representational collapse (Bardes et al., 2021; Zbontar et al., 2021; Ermolov et al., 2021). We believe that the networks developed in this study, motivated by extensive neuroscience research on rapid gain modulation, provide an effective whitening solution for these methods – particularly in regimes which prioritize streaming data, and networks designed for low-power consumption hardware.

References

Ballé, J., Chou, P. A., Minnen, D., Singh, S., Johnston, N., Agustsson, E., Hwang, S. J., and Toderici, G. Nonlinear Transform Coding. *arXiv:2007.03034 [cs, eess, math]*, 2020.

Bardes, A., Ponce, J., and LeCun, Y. VICReg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

Barlow, H. B. Possible Principles Underlying the Transformations of Sensory Messages. In *Sensory Communication*, pp. 216–234. The MIT Press, 1961.

Benucci, A., Saleem, A. B., and Carandini, M. Adaptation maintains population homeostasis in primary visual cortex. *Nature neuroscience*, 16(6):724–729, 2013.

Bonneel, N., Rabin, J., Peyré, G., and Pfister, H. Sliced and Radon Wasserstein Barycenters of Measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45, January 2015. ISSN 0924-9907, 1573-7683.

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

Casazza, P. G., Kutyniok, G., and Philipp, F. Introduction to Finite Frame Theory. In Casazza, P. G. and Kutyniok, G. (eds.), *Finite Frames*, pp. 1–53. Birkhäuser Boston, Boston, 2013. ISBN 978-0-8176-8372-6 978-0-8176-8373-3.

Chapochnikov, N. M., Pehlevan, C., and Chklovskii, D. B. Normative and mechanistic model of an adaptive circuit for efficient encoding and feature extraction. *bioRxiv*, 2021.

Coates, A., Ng, A., and Lee, H. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011.

Duong, L. R., Li, B., Chen, C., and Han, J. Multi-rate adaptive transform coding for video compression. *arXiv:2210.14308 [eess.IV]*, October 2022.

Ermolov, A., Siarohin, A., Sangineto, E., and Sebe, N. Whitening for self-supervised representation learning. In *International Conference on Machine Learning*, pp. 3015–3024. PMLR, 2021.

Fairhall, A. L., Lewen, G. D., and Bialek, W. Efficiency and ambiguity in an adaptive neural code. *Nature*, 412: 787–792, 2001.

Ferguson, K. A. and Cardin, J. A. Mechanisms underlying gain modulation in the cortex. *Nature Reviews Neuroscience*, 21(2):80–92, 2020. ISSN 1471-0048.

Friedrich, R. W. Neuronal computations in the olfactory system of zebrafish. *Annual review of neuroscience*, 36: 383–402, 2013.

Gardner, R. J. *Geometric tomography*, volume 58. Cambridge University Press Cambridge, 1995.

Giridhar, S., Doiron, B., and Urban, N. N. Timescale-dependent shaping of correlation by olfactory bulb lateral inhibition. *Proceedings of the National Academy of Sciences*, 108(14):5843–5848, 2011.

Gschwend, O., Abraham, N. M., Lagier, S., Begnaud, F., Rodriguez, I., and Carleton, A. Neuronal pattern separation in the olfactory bulb improves odor discrimination learning. *Nature Neuroscience*, 18(10):1474–1482, 2015.

Hateren, J. H. v. and Schaaf, A. v. d. Independent component filters of natural images compared with simple cells in primary visual cortex. *Proceedings: Biological Sciences*, 265(1394):359–366, Mar 1998.

Hua, T., Wang, W., Xue, Z., Ren, S., Wang, Y., and Zhao, H. On feature decorrelation in self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9598–9608, 2021.

Hyvärinen, A. and Oja, E. Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5): 411–430, 2000.

Karl, W. C., Verghese, G. C., and Willsky, A. S. Reconstructing Ellipsoids from Projections. *CVGIP: Graphical Models and Image Processing*, 56(2):124–139, 1994. ISSN 1049-9652.

Kessy, A., Lewin, A., and Strimmer, K. Optimal whitening and decorrelation. *The American Statistician*, 72(4):309–314, 2018.

Krizhevsky, A. Learning multiple layers of features from tiny images. Master's thesis, University of Toronto, 2009.

Laughlin, S. A Simple Coding Procedure Enhances a Neuron's Information Capacity. *Zeitschrift fur Naturforschung. C, Journal of biosciences*, pp. 910–2, 1981.

Lipshutz, D., Pehlevan, C., and Chklovskii, D. B. Interneurons accelerate learning dynamics in recurrent neural networks for statistical adaptation. *arxiv preprint arxiv:2209.10634*, 2022.

Martin, S., Grimwood, P. D., and Morris, R. G. Synaptic plasticity and memory: an evaluation of the hypothesis. *Annual Review of Neuroscience*, 23(1):649–711, 2000.

Mohan, S., Vincent, J. L., Manzorro, R., Crozier, P. A., Simoncelli, E. P., and Fernandez-Granda, C. Adaptive Denoising via GainTuning. *arXiv:2107.12815 [cs.CV]*, July 2021.

Nagel, K. I. and Doupe, A. J. Temporal Processing and Adaptation in the Songbird Auditory Forebrain. *Neuron*, 51(6):845–859, September 2006.

Pehlevan, C. and Chklovskii, D. B. A normative theory of adaptive dimensionality reduction in neural networks. *Advances in Neural Information Processing Systems*, 28, 2015.

Pehlevan, C. and Chklovskii, D. B. Neuroscience-Inspired Online Unsupervised Learning Algorithms: Artificial Neural Networks. *IEEE Signal Processing Magazine*, 36(6):88–96, November 2019. ISSN 1053-5888, 1558-0792.

Pehlevan, C., Sengupta, A. M., and Chklovskii, D. B. Why do similarity matching objectives lead to hebbian/anti-hebbian networks? *Neural Computation*, 30(1):84–124, 2017.

Wanner, A. A. and Friedrich, R. W. Whitening of odor representations by the wiring diagram of the olfactory bulb. *Nature neuroscience*, 23(3):433–442, 2020.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pp. 12310–12320. PMLR, 2021.

Zucker, R. S., Regehr, W. G., et al. Short-term synaptic plasticity. *Annual Review of Physiology*, 64(1):355–405, 2002.

A. Notation

For $N \geq 2$, let $K_N := N(N+1)/2$. Let $\mathbb{R}^N$ denote N-dimensional Euclidean space equipped with the Euclidean norm, denoted $\|\cdot\|_2$. Let $\mathbb{R}^N_+$ denote the non-negative orthant in $\mathbb{R}^N$. Given $K \geq 2$, let $\mathbb{R}^{N\times K}$ denote the set of $N \times K$ real-valued matrices and $\mathbb{S}^N$ denote the set of $N \times N$ symmetric matrices.

Matrices are denoted using bold uppercase letters (e.g., $\mathbf{M}$) and vectors are denoted using bold lowercase letters (e.g., $\mathbf{v}$). Given a matrix $\mathbf{M}$, M_{ij} denotes the entry of $\mathbf{M}$ located at the i^{th} row and j^{th} column. Let $\mathbf{1} = [1, \ldots, 1]^\top$ denote the N-dimensional vector of ones. Let $\mathbf{I}_N$ denote the $N \times N$ identity matrix.

Given vectors $\mathbf{v}, \mathbf{w} \in \mathbb{R}^N$, define their Hadamard product by $\mathbf{v} \circ \mathbf{w} := (v_1 w_1, \ldots, v_N w_N) \in \mathbb{R}^N$. Define $\mathbf{v}^{\circ 2} := (v_1^2, \ldots, v_N^2) \in \mathbb{R}^N$. Define $\text{diag}(\mathbf{v})$ to be the $N \times N$ diagonal matrix whose $(i,i)^{\text{th}}$ entry is equal to v_i, for $i = 1, \ldots, N$.

Let $\langle \cdot \rangle_t$ denote expectation over $t = 1, 2, \ldots$.

The $\text{diag}\,(\cdot)$ operator, similar to `numpy.diag()` or MATLAB's `diag()`, can either: 1) map a vector in $\mathbb{R}^K$ to the diagonal of a $K \times K$ zeros matrix; or 2) map the diagonal entries of a $K \times K$ matrix to a vector in $\mathbb{R}^K$. The specific operation being used should be clear by context.

B. Proof of Proposition 2.1

Proof of Proposition 2.1. Suppose Equation 1 holds. Then, for $i = 1, \ldots, K$,

$$\langle (\mathbf{w}_i^\top \mathbf{y}_t)^2 \rangle_t = \langle \mathbf{w}_i^\top \mathbf{y}_t \mathbf{y}_t^\top \mathbf{w}_i \rangle_t = \mathbf{w}_i^\top \mathbf{w}_i = 1.$$

Therefore, Equation 4 holds.

Now suppose Equation 4 holds. Let $\mathbf{v} \in \mathbb{R}^N$ be an arbitrary unit vector. Then $\mathbf{v}\mathbf{v}^\top \in \mathbb{S}^N$ and by Equation 3, there exist $g_1, \ldots, g_K \in \mathbb{R}$ such that

$$\mathbf{v}\mathbf{v}^\top = g_1 \mathbf{w}_1 \mathbf{w}_1^\top + \cdots + g_K \mathbf{w}_K \mathbf{w}_K^\top. \tag{15}$$

We have

$$\mathbf{v}^\top \langle \mathbf{y}_t \mathbf{y}_t^\top \rangle_t \mathbf{v} = \text{Tr}(\mathbf{v}\mathbf{v}^\top \langle \mathbf{y}_t \mathbf{y}_t^\top \rangle_t) = \sum_{i=1}^K g_i \,\text{Tr}(\mathbf{w}_i \mathbf{w}_i^\top \langle \mathbf{y}_t \mathbf{y}_t^\top \rangle_t) = \sum_{i=1}^K g_i \,\text{Tr}(\mathbf{w}_i \mathbf{w}_i^\top) = \text{Tr}(\mathbf{v}\mathbf{v}^\top) = 1. \tag{16}$$

The first equality is a property of the trace operator. The second and fourth equalities follows from Equation 15 and the linearity of the trace operator. The third equality follows from Equation 3. The final equality holds because $\mathbf{v}$ is a unit vector. Since Equation 16 holds for every unit vector $\mathbf{v} \in \mathbb{R}^N$, Equation 1 holds. □

C. Saddle point property

We recall the following minmax property for a function that satisfies the saddle point property (Boyd & Vandenberghe, 2004, section 5.4).

Theorem C.1. *Let $V \subseteq \mathbb{R}^n$, $W \subseteq \mathbb{R}^m$ and $f : V \times W \to \mathbb{R}$. Suppose f satisfies the saddle point property; that is, there exists $(\mathbf{a}^*, \mathbf{b}^*) \in V \times W$ such that*

$$f(\mathbf{a}^*, \mathbf{b}) \leq f(\mathbf{a}^*, \mathbf{b}^*) \leq f(\mathbf{a}, \mathbf{b}^*), \qquad \textit{for all } (\mathbf{a}, \mathbf{b}) \in V \times W.$$

Then

$$\min_{\mathbf{a}\in V} \max_{\mathbf{b}\in W} f(\mathbf{a}, \mathbf{b}) = \max_{\mathbf{b}\in W} \min_{\mathbf{a}\in V} f(\mathbf{a}, \mathbf{b}) = f(\mathbf{a}^*, \mathbf{b}^*).$$

D. Weighted average update rule for $\mathbf{g}_i$

The update for $\mathbf{g}$ in Equation 10 can be generalized to allow for a weighted average over past samples. In particular, the general update is given by

$$\mathbf{g} \leftarrow \mathbf{g} + \eta \left(\frac{1}{Z} \sum_{s=1}^t \gamma^{t-s} \mathbf{z}_s^{\circ 2} - \mathbf{1} \right),$$

where $\gamma \in [0, 1]$ determines the decay rate and $Z := 1 + \gamma + \cdots + \gamma^{t-1}$ is a normalizing factor.

E. Batched and offline algorithms for whitening with RNNs via gain modulation

In addition to the fully-online algorithm provided in the main text (Algorithm 1), we also provide two variants below. In many applications, streaming inputs arrive in batches rather than one at a time (e.g. video streaming frames). Similarly for conventional offline stochastic gradient descent training, data is sampled in batches. Algorithm 2 would be one way to accomplish this in our framework, where the main difference between the fully online version is taking the mean across samples in the batch to yield average gain update $\Delta\mathbf{g}$ term. Furthermore, in the fully offline setting when the covariance of the inputs, $\mathbf{C}_{xx}$ is known, Algorithm 3 presents a way to whiten the covariance directly.

Algorithm 2 Batched ZCA whitening

Input: Data matrix $\mathbf{X} \in \mathbb{R}^{N\times T}$ (assumed centered)
Initialize: $\mathbf{W} \in \mathbb{R}^{N\times K}$; $\mathbf{g} \in \mathbb{R}^K$; η; batch size B
while not converged **do**
  $\mathbf{X}_B \leftarrow \texttt{sample\_batch}(\mathbf{X}, B)\{N \times B\}$
  $\mathbf{Y}_b \leftarrow [\mathbf{I}_N + \mathbf{W}\,\mathrm{diag}\,(\mathbf{g})\,\mathbf{W}^\top]^{-1}\mathbf{X}_B$
  $\mathbf{Z}_b \leftarrow \mathbf{W}^\top\mathbf{Y}_b$
  $\Delta\mathbf{g} \leftarrow \mathbf{Z}_B^{\circ 2} - \mathbf{1}$ {Subtract 1 from all entries}
  $\mathbf{g} \leftarrow \mathbf{g} + \eta\ \texttt{mean}(\Delta\mathbf{g}, \texttt{axis=1})$
end while

Algorithm 3 Offline ZCA whitening

Input: Input covariance $\mathbf{C}_{xx}$
Initialize: $\mathbf{W} \in \mathbb{R}^{N\times K}$; $\mathbf{g} \in \mathbb{R}^K$; η
while not converged **do**
  $\mathbf{M} \leftarrow [\mathbf{I}_N + \mathbf{W}\,\mathrm{diag}\,(\mathbf{g})\,\mathbf{W}^\top]^{-1}$
  $\mathbf{C}_{yy} \leftarrow \mathbf{M}\mathbf{C}_{xx}\mathbf{M}^\top$
  $\Delta\mathbf{g} \leftarrow \mathrm{diag}\left(\mathbf{W}^\top\mathbf{C}_{yy}\mathbf{W}\right) - \mathbf{1}$
  $\mathbf{g} \leftarrow \mathbf{g} + \eta\Delta\mathbf{g}$
end while

F. Frame factorizations of symmetric matrices

F.1. Analytic solution for the optimal gains

Recall that the optimal solution of the ZCA objective in Equation 5 is given by $\mathbf{y}_t = \mathbf{C}_{xx}^{-1/2}\mathbf{x}_t$ for $t = 1, 2, \ldots$. In our neural circuit with interneurons and gain control, the outputs of the primary neurons at equilibrium is (given in Equation 8, but repeated here for clarity)

$$\bar{\mathbf{y}}_t = \left[\mathbf{I}_N + \mathbf{W}\,\mathrm{diag}\,(\mathbf{g})\,\mathbf{W}^\top\right]^{-1}\mathbf{x}_t.$$

Therefore, the circuit performs ZCA whitening when the gains $\mathbf{g}$ satisfy the relation

$$\mathbf{I}_N + \mathbf{W}\,\mathrm{diag}\,(\mathbf{g})\,\mathbf{W}^\top = \mathbf{C}_{xx}^{1/2}. \tag{17}$$

When K is exactly $N(N+1)/2$, we can explicitly solve for the optimal gains $\bar{\mathbf{g}}$ (derived in the next subsection):

$$\bar{\mathbf{g}} = \left[\left(\mathbf{W}^\top\mathbf{W}\right)^{\circ 2}\right]^{-1}\left[\mathbf{w}_1^\top\mathbf{C}_{xx}^{1/2}\mathbf{w}_1 - 1, \ldots, \mathbf{w}_N^\top\mathbf{C}_{xx}^{1/2}\mathbf{w}_N - 1\right]^\top. \tag{18}$$

F.2. Deriving optimal gains

We find it useful to first demonstrate that *any* matrix $\mathbf{C} \in \mathbb{S}^N$, where $\mathbb{S}^N$ is the space of symmetric $N \times N$ matrices, can be factorized as

$$\mathbf{W}\,\mathrm{diag}\,(\mathbf{g})\,\mathbf{W}^\top = \mathbf{C} \tag{19}$$

where $\mathbf{W} \in \mathbb{R}^{N\times K}$ is some fixed, arbitrary, frame with $K \geq \frac{N(N+1)}{2}$ (i.e. a representation that is $\mathcal{O}(N^2)$ overcomplete), and $\mathbf{g} \in \mathbb{R}^K$ is a variable vector encoding information about $\mathbf{C}$. We multiply both sides of Equation 19 from the left and right by $\mathbf{W}^\top$ and $\mathbf{W}$, respectively, then take the diagonal[3] of the resultant matrices,

$$\mathrm{diag}\left(\mathbf{W}^\top\mathbf{W}\,\mathrm{diag}\,(\mathbf{g})\,\mathbf{W}^\top\mathbf{W}\right) = \mathrm{diag}\left(\mathbf{W}^\top\mathbf{C}\mathbf{W}\right). \tag{20}$$

[3]Similar to commonly-used matrix libraries, the $\mathrm{diag}\,(\cdot)$ operator here is overloaded and can map a vector to a matrix or vice versa. See Appendix A for details.

Finally, employing a simple matrix identity involving the $\mathrm{diag}\,(\cdot)$ operator yields

$$(\mathbf{W}^\top\mathbf{W})^{\circ 2}\mathbf{g} = \mathrm{diag}\left(\mathbf{W}^\top\mathbf{C}\mathbf{W}\right), \tag{21}$$

$$\implies \mathbf{g} = \left[(\mathbf{W}^\top\mathbf{W})^{\circ 2}\right]^{-1}\mathrm{diag}\left(\mathbf{W}^T\mathbf{C}\mathbf{W}\right), \tag{22}$$

where $(\cdot)^{\circ 2}$ denotes element-wise squaring. Thus, *any* $N \times N$ symmetric matrix, can be encoded as a vector, $\mathbf{g}$, with respect to an arbitrary fixed frame, $\mathbf{W}$, by solving a standard linear system of K equations of the form $\mathbf{A}\mathbf{g} = \mathbf{b}$. Importantly, when $K = N(N+1)/2$, and the columns of $\mathbf{W}$ are not collinear, then the matrix on the LHS, $(\mathbf{W}^\top\mathbf{W})^{\circ 2} \in \mathbb{S}^K_{++}$, is invertible, and the vector $\mathbf{g}$ is unique (Appendix B).

Without loss of generality, assume that the columns of $\mathbf{W}$ are unit-norm (otherwise, we can always normalize them by absorbing their lengths into the elements of $\mathbf{g}$). Furthermore, assume without loss of generality that $\mathbf{C} \in \mathbb{S}^N_{++}$, the set of all symmetric positive definite matrices (e.g. covariance, precision, PSD square roots, etc.). When $\mathbf{C}$ is a covariance matrix, then $\mathrm{diag}\left(\mathbf{W}^\top\mathbf{C}\mathbf{W}\right)$ can be interpreted as a vector of projected variances of $\mathbf{C}$ along each axis spanned by $\mathbf{W}$. Therefore, Equation 21 states that the vector $\mathbf{g}$ is linearly related to the vector of projected variances via the element-wise squared frame Gramian, $(\mathbf{W}^\top\mathbf{W})^{\circ 2}$.

G. Adaptation with inequality constraint

In general, the modified objective with rectified gains (Equation 14) does not statistically whiten the inputs $\mathbf{x}_1, \mathbf{x}_2, \ldots$, but rather adapts the non-negative gains $g_1, \ldots, g_K$ to ensure that the variances of the outputs $\mathbf{y}_1, \mathbf{y}_2, \ldots$ in the directions spanned by the frame vectors $\{\mathbf{w}_1, \ldots, \mathbf{w}_K\}$ are bounded above by unity (Figure 7). This one-sided normalization carries interesting implications for how and when the circuit statistically whitens its outputs, which can be compared with experimental observations. For instance, the circuit performs ZCA whitening if and only if there are non-negative gains such that Equation 17 holds (see, e.g., the top right example in Figure 7), which corresponds to cases such that the matrix $\mathbf{C}_{xx}^{1/2}$ is an element of the following cone (with its vertex translated by $\mathbf{I}_N$):

$$\left\{\mathbf{I}_N + \sum_{i=1}^{K} g_i \mathbf{w}_i \mathbf{w}_i^\top : \mathbf{g} \in \mathbb{R}^K_+\right\}.$$

On the other hand, if the variance of an input projection is less than unity — i.e., $\mathbf{w}_i^\top \mathbf{C}_{xx} \mathbf{w}_i \leq 1$ for some i — then the corresponding gain g_i remains zero. When this is true for all $i = 1, \ldots, K$, the gains all remain zero and the circuit output is equal to its input (see, e.g., the bottom middle example of Figure 7).

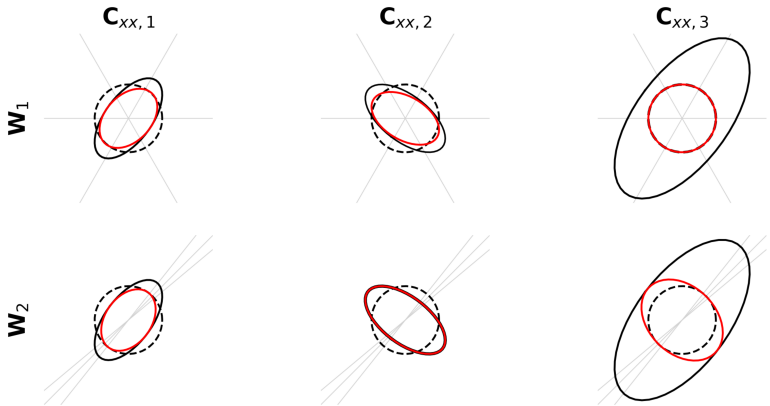

Figure 7. Geometric intuition of whitening with/without inequality constraint. Whitening efficacy using non-negative gains depends on $\mathbf{W}$ and $\mathbf{C}_{xx}$. For $N = 2$ and $K = 3$, examples of covariance matrices $\mathbf{C}_{yy}$ (red ellipses) corresponding to optimal solutions $\mathbf{y}$ of objective 12, for varying input covariance matrices $\mathbf{C}_{xx}$ (black ellipses) and frames $\mathbf{W}$ (spanning axes denoted by gray lines). Unit circles, which correspond to the identity matrix target covariance, are shown with dashed lines. Each row corresponds to a different frame $\mathbf{W}$ and each column corresponds to a different input covariance $\mathbf{C}_{xx}$.

H. Whitening spatially local neighborhoods

H.1. Spatially local whitening in 1D

For an N-dimensional input, we consider a network that whitens spatially local neighborhoods of size $M < N$. To this end, we can construct N filters of the form

$$\mathbf{w}_i = \mathbf{e}_i, \qquad i = 1, \ldots, N$$

and $M(N - \frac{M+1}{2})$ filters of the form

$$\mathbf{w} = \frac{\mathbf{e}_i + \mathbf{e}_j}{\sqrt{2}}, \qquad i, j = 1, \ldots, N, \qquad 1 \leq |i - j| \leq M.$$

The total number of filters is $(M + 1)(N - \frac{M}{2})$, so for fixed M the number of filters scales linearly in N rather than quadratically.

We simulated a network comprising $N = 10$ primary neurons, and a convolutional weight matrix connecting each interneuron to spatial neighborhoods of three primary neurons. Given input data with covariance $\mathbf{C}_{xx}$ illustrated in Figure 8A (left panel), this modified network succeeded to statistically whiten local neighborhoods of size of primary 3 neurons (right panel). Notably, the eigenspectrum (Figure 8B) after local whitening is much closer to being equalized. Furthermore, while the global whitening solution produced a flat spectrum as expected, the local whitening network did not amplify the axis with very low-magnitude eigenvalues (Figure 8B right panel).

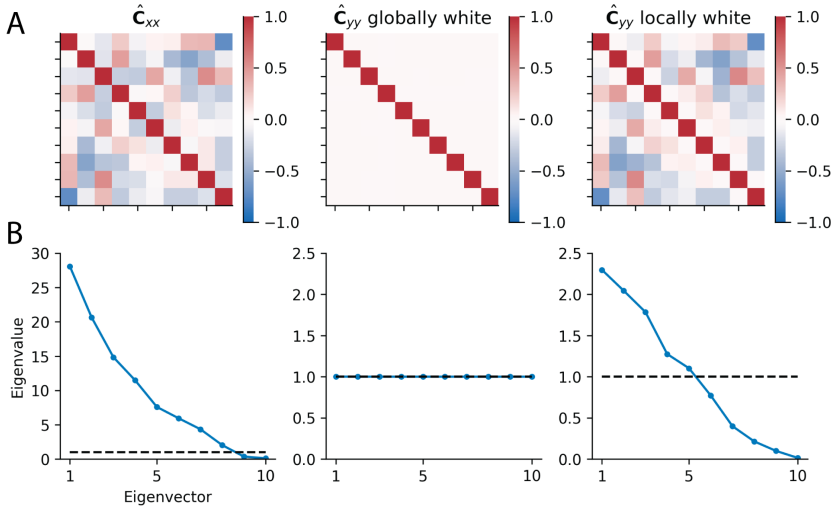

Figure 8. Statistically adapting local neighborhoods of neurons. **A)** $\hat{\mathbf{C}}_{xx}$ denotes correlation matrix, which are shown here for display purposes only, to facilitate comparisons. Network with 10-dimensional input correlation (left) 10-dimensional output correlation matrix after global whitening (middle); and output correlation matrix after statistically whitening local neighborhoods of size 3. The output correlation matrix of the locally adapted circuit has block-identity structure along the diagonal. **B)** Corresponding eigenspectra of *covariance* matrices of unwhitened (left), global whitened (middle), and locally whitened (right) network outputs. The black dashed line denotes unity.

H.2. Filter bank construction in 2D

Here, we describe one way of constructing a set of convolutional weights for overlapping spatial neighborhoods (e.g. image patches) of neurons. Given an $n \times m$ input and overlapping neighborhoods of size $h \times w$ to be statistically whitened, the samples are therefore matrices $X \in \mathbb{R}^{n \times m}$. In this case, filters $\mathbf{w} \in \mathbb{R}^{1 \times n \times m}$ can be indexed by pairs of pixels that are in the same patch:

$$((i, j), (k, \ell)), \qquad 1 \leq i \leq n, \qquad 1 \leq j \leq m, \qquad 0 \leq |i - k| \leq h, \qquad 0 \leq |j - \ell| \leq w$$

We can then construct the filters as,

$$\mathbf{w}_{(i,j),(k,\ell)}(X) = \begin{cases} x_{i,j} & \text{if } (i,j) = (k,\ell), \\ \frac{x_{i,j}+x_{k,\ell}}{\sqrt{2}} & \text{if } (i,j) \neq (k,\ell). \end{cases}$$

In this case there are

$$nm + wh\left[(n-w)(m-h) + (n-w)\frac{(h+1)}{2} + (m-h)\frac{(w+1)}{2} + (h+1)\frac{(w+1)}{2}\right]$$

such filters, so the number of filters required scales linearly with nm rather than quadratically.