Unsupervised learning by a "softened" correlation game: duality and convergence

Kyle L. Luther*
Dept. of Physics
Princeton University
Princeton, USA
kluther@princeton.edu

Runzhe Yang*
Dept. of Computer Science
Princeton University
Princeton, USA
runzhey@princeton.edu

H. Sebastian Seung
Neuroscience Institute and Dept. of Computer Science
Princeton University
Princeton, USA
sseung@princeton.edu

***Abstract*—Neural networks with Hebbian excitation and anti-Hebbian inhibition form an interesting class of biologically plausible unsupervised learning algorithms. It has recently been shown that such networks can be regarded as online gradient descent-ascent algorithms for solving min-max problems that are dual to unsupervised learning principles formulated with no explicit reference to neural networks. Here we generalize one such formulation, the correlation game, by replacing a hard constraint with a soft penalty function. Our "softened" correlation game contains the nonnegative similarity matching principle as a special case. For solving the primal problem, we derive a projected gradient ascent algorithm that achieves speed through sorting. For solving the dual problem, we derive a projected gradient descent-ascent algorithm, the stochastic online variant of which can be interpreted as a neural network algorithm. We prove strong duality when the inhibitory connection matrix is positive definite, a condition that also prohibits multistability of neural activity dynamics. We show empirically that the neural net algorithm can converge when inhibitory plasticity is faster than excitatory plasticity, and may fail to converge in the opposing case. This is intuitively interpreted using the structure of the min-max problem.**

***Index Terms*—neural networks, Hebbian learning**

I. Introduction

Neural networks with Hebbian feedforward and anti-Hebbian lateral connections are an interesting class of biologically plausible unsupervised learning algorithms [1], [2]. (See also historical references cited in Ref. [3].) Recently it was shown that such networks can be regarded as online gradient descent-ascent (GDA) algorithms for solving min-max problems. This was shown for a linear network[1] by Pehlevan et al. [4], and for a nonlinear network by Seung and Zung [3]. In the latter work, nonnegativity constraints were imposed on the activities and connections, so that the feedforward connections are excitatory and the lateral connections are inhibitory, as in the original model of Földiák [1].

The min-max problems are in turn dual to unsupervised learning principles that are formulated with no explicit reference to neural networks. Pehlevan et al. formulated the similarity matching principle [4], while Seung and Zung [3] formulated the "correlation game," the maximization of input-output correlations subject to an upper bound constraint on output-output correlations.

Here we propose a generalization of the correlation game in which the hard constraint is replaced by a soft penalty function. This contains the nonnegative similarity matching principle [5] as a special case, and the original correlation game [3] as a limiting case. Gradient ascent can be used to find a local optimum of the softened correlation game. We show through numerical experiments that the parameters of the penalty function can be used to control output-output correlations.

Through a duality transformation, the softened correlation game becomes a max-min problem, which can be solved by gradient descent-ascent (GDA). In the neural network interpretation, GDA contains some output variables that represent neural activities. Legendre transform variables represent excitatory and inhibitory neural connections. We show through numerical experiments that GDA may produce approximately the same results as gradient ascent, i.e., that strong duality holds at least approximately in some cases. GDA has both batch and stochastic online variants, the latter of which is more biologically plausible.

We then go on to address two issues: duality and convergence. Pehlevan et al. [4] showed that learning by their linear network satisfies strong duality with the similarity matching principle. Seung and Zung [3] showed only weak duality for their nonlinear network. In this paper we derive sufficient conditions for strong duality of the nonlinear network. It turns out that the inhibitory connection matrix should be positive definite for strong duality.

GDA does not necessarily converge to a steady state. Pehlevan et al. [4] did a linear stability analysis for learning by a linear network, and showed convergence if inhibitory plasticity is sufficiently fast relative to excitatory plasticity. We present example numerical simulations showing that the

*Authors contributed equally.

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC0005. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/IBC, or the U.S. Government.

[1]We refer to their network as "linear," because the dynamics of neural activities is linear. When the modifications to synaptic weights are included, however, the complete dynamics is nonlinear.

nonlinear network converges to a steady state for fast inhibition but exhibits more complex dynamical behaviors for slow inhibition. We give intuitive arguments as to why convergence should be seen with fast inhibition but not slow inhibition. Convergence proofs for fast inhibition should be possible by extending existing methods [6]–[8], but are postponed to future work.

II. Unsupervised learning principle

Given a matrix $U = [\mathbf{u}(1), ..., \mathbf{u}(T)]$ of T input vectors, we define unsupervised learning as the optimization

$$\max_{X \geq 0} F(X) = \max_{X \geq 0} \left\{ \Phi^* \left(\frac{XU^\intercal}{T} \right) - \frac{1}{2} \Psi^* \left(\frac{XX^\intercal}{T} \right) \right\} \quad (1)$$

with respect to the matrix $X = [\mathbf{x}(1), ..., \mathbf{x}(T)]$ of T output vectors. The functions Φ^* and Ψ^* are assumed chosen so that $-F(X)$ is radially unbounded in the nonnegative orthant, a sufficient condition for the existence of the optimum. The output vectors produced by unsupervised learning should ideally be some "useful" representation of the input vectors.

We further assume that Φ^* and Ψ^* are chosen to be nondecreasing functions of the elements of their matrix arguments. Therefore our unsupervised learning principle aims to make the input-output correlation matrix $XU^\intercal/T$ large and make the output-output correlation matrix $XX^\intercal/T$ small. If the input-output correlations are large, that means the output vectors are related to the input vectors. If the output-output correlations are small, that means different elements of the output carry different kinds of information.

To guarantee the nondecreasing property, we will write Φ^* and Φ^* as Legendre transforms,

$$\Phi^*(C) = \max_{W \geq 0} \{ \operatorname{Tr} WC^\intercal - \Phi(W) \} \quad (2)$$

$$\Psi^*(C) = \max_{L \geq 0} \{ \operatorname{Tr} LC^\intercal - \Psi(L) \} \quad (3)$$

We will assume that Φ and Ψ are strongly convex, so that the maximum and minimum in the Legendre transforms are attained. The nondecreasing property holds because $\partial \Phi^*(C)/\partial C_{ia} = W_{ia} \geq 0$, where W_{ia} is the solution of Eq. (2) for a given C. We also assume without loss of generality that $\Psi(L) = \Psi(L^\intercal)$, since Ψ^* is a function of a symmetric matrix in Eq. (1).

While our formalism is quite general, we will also be interested in the special case

$$\Psi^*(C) = \frac{1}{2\mu} \left\| [C - D]^+ \right\|^2 \quad (4)$$

where $[z]^+ = \max(z, 0)$ is defined as (half-wave) rectification. The off-diagonal terms of Eq. (4) tend to push output-output correlations to be less than D or encourage them to be small if they are greater than D. Each diagonal term of Eq. (4) tends to push output power $T^{-1} \sum_t X_{it}^2$ to be less than D_{ii} or encourage it to be small if greater than D_{ii}.

In the $\mu \to 0$ limit, Eq. (4) drives the output-output correlation matrix to be less than or equal to D. Then Eq. (1) becomes a penalty function method for solving the constrained optimization

$$\max_{X \geq 0} \Phi^* \left(\frac{XU^\intercal}{T} \right) \text{ subject to } \frac{XX^\intercal}{T} \leq D \quad (5)$$

where the matrix D is an upper bound constraint on the output-output correlations. This was previously called the "correlation game" by Ref. [3]. For finite μ, we obtain a variant of the correlation game with a soft constraint.

If we further set $D = 0$ and $\Phi^*(C) = \operatorname{Tr} C^\intercal C$, we obtain the nonnegative similarity matching principle of Ref. [5],

$$F(X) = -\frac{1}{2} \|X^\intercal X - U^\intercal U\|^2 + \text{ const} \quad (6)$$

(This is easy to see for $\mu = 1$, but also holds up to rescaling for other positive values of μ.)

We will also be interested in the special case

$$\Phi(W) = \frac{\gamma}{2} \sum_{ia} W_{ia}^2 + \frac{\kappa}{2} \sum_i \left(\sum_a W_{ia} \right)^2 \quad (7)$$

The optimum in Eq. (2) satisfies

$$W_{ia} = \gamma^{-1} \left[C_{ia} - \kappa \sum_b W_{ib} \right]^+ \quad (8)$$

The second term of Eq. (7) has effectively induced a kind of "competition" within each row of the matrix C, so that W_{ia} is only positive for those C_{ia} that exceed the threshold $\kappa \sum_b W_{ib}$. As a result, Φ^* ends up depending only on the largest elements of each row of the matrix C. The form of competition is similar to that studied by Seung and Zung [3], who placed $\sum_a W_{ia} - \rho$ inside the parentheses in Eq. (7).

III. Learning by gradient ascent

Rewriting Eq. (1) using the Legendre transforms yields

$$\max_{X \geq 0} F(X) = \max_{X \geq 0} \max_{W \geq 0} \min_{L \geq 0} \mathcal{F}(W, L, X) \quad (9)$$

where

$$\mathcal{F}(W, L, X) = \operatorname{Tr} \left(\frac{WUX^\intercal}{T} \right) - \Phi(W) \quad (10)$$

$$- \frac{1}{2} \left[\operatorname{Tr} \left(\frac{LXX^\intercal}{T} \right) - \Psi(L) \right] \quad (11)$$

The $\max$ in Eq. (3) has changed to $\min$ because of the minus sign before $\Psi^*(L)$ in Eq. (1).

Then the optimization of Eq. (9) can be performed using projected gradient ascent,

$$X \leftarrow \left[X + \eta_X \frac{1}{T} (W^* U - L^* X) \right]^+ \quad (12)$$

where

$$W^* = \arg\max_{W \geq 0} \{ \operatorname{Tr} WUX^\intercal/T - \Phi(W) \}, \quad (13)$$

and

$$L^* = \arg\max_{L \geq 0} \{ \operatorname{Tr} LXX^\intercal/T - \Psi(L) \} \quad (14)$$

These maxima are uniquely defined assuming that Φ and Ψ are strongly convex.

For the special case of the penalty function in Eq. (4), Ψ^* is the Legendre transform of

$$\Psi(L) = \frac{\mu}{2}\sum_{ij} L_{ij}^2 + \sum_{ij} D_{ij}L_{ij} \tag{15}$$

and we can solve for L^* in closed form,

$$L^* = \mu^{-1}\left[\frac{XX^\intercal}{T} - D\right]^+ \tag{16}$$

Note that when $\mu = 0$, as in the case of the hard constraint considered by [3], L^* is undefined so the gradient ascent on X algorithm is not directly applicable to the correlation game with hard constraints.

To find W^* we need to solve Equation (8). The key idea behind the algorithm is that if we know exactly which elements of W^* are nonzero, we can easily compute:

$$\sum_{b:W^*_{ib}\neq 0} W^*_{ib} = \sum_b W^*_{ib} \tag{17}$$

and insert this quantity into Eq. (8) to determine W^*. Summing Eq. (8) over a such that $W^*_{ia} \neq 0$ gives:

$$\sum_{a:W^*_{ia}\neq 0} W^*_{ia} = \frac{1}{\gamma}\sum_{a:W^*_{ia}\neq 0} C_{ia} - k_i^*\frac{\kappa}{\gamma}\sum_{a:W^*_{ia}\neq 0} W^*_{ia} \tag{18}$$

where k_i^* is the number of nonzero elements of W^* in every row i. Bringing the 2nd term on the lefthand side of the equation gives:

$$\sum_{b:W^*_{ib}\neq 0} W^*_{ib} = \frac{1}{\gamma + k_i^*\kappa}\sum_{a:W^*_{ia}\neq 0} C_{ia} \tag{19}$$

Using Eq. (8) we can see that the locations i, a where $W^* > 0$ are at the locations where C_{ia} is one the the top k_i^* elements of the i'th row.

We now describe the algorithm used to compute k_i^*. First sort every row of C. Specifically for every row i, find an ordering $\{a_1, a_2, ...a_m\}$ such that $C_{ia_k} \geq C_{ia_{k'}}$ if $k < k'$. Use this ordering to compute the $N \times M$ matrix:

$$S_{ik} = \frac{\kappa}{\gamma + k\kappa}\sum_{j=1}^{k} C_{ia_j} \tag{20}$$

For any $k_i \leq k_i^*$ we know that $W_{ia} > 0$ and therefore $C_{ia_k} - S_{ik} > 0$. Conversely we know that for any $k_i > k_i^*$ we know that $W_{ia} = 0$ and therefore $C_{ia_k} - S_{ik} \leq 0$. We can therefore compute k_i^* by computing the maximal k for every row such that:

$$k_i^* = \max_{k\in\{1,2,...,M\}} k \quad \text{such that } S_{ik} < C_{ia_k} \tag{21}$$

Finally we can analytically compute W_i^* using $S_{ik_i^*}$:

$$W^*_{ia} = \frac{1}{\gamma}\left[C_{ia} - S_{ik_i^*}\right]^+ \tag{22}$$

The time complexity of this algorithm is dominated by the sorting that must be done for every row of C. There are n rows, each of which requires $m\log m$ steps to sort giving us a complexity of $O(nm\log m)$

So long as η_X is sufficiently small, we generally expect that the algorithm at least finds a local maximum of F. We will perform numerical experiments with this algorithm in a later section. A potential drawback of the algorithm is that it is unclear how to extend it to the online setting. In general, we cannot write the gradient as a sum over examples.

IV. Learning by gradient descent-ascent

To derive an online algorithm for approximately solving Eq. (1), we will transform the primal problem (9) into a dual. The maximums over X and W commute, so by the max-min inequality, we can write the upper bound

$$\max_{X\geq 0} F(X) \leq \max_{W\geq 0}\inf_{L\geq 0}\sup_{X\geq 0}\mathcal{F}(W, L, X) \tag{23}$$

$$\leq \max_{W\geq 0}\inf_{L\geq 0} R(W, L) \tag{24}$$

where we have defined

$$R(W, L) = \sup_{X\geq 0}\mathcal{F}(W, L, X) \tag{25}$$

One can attempt to solve this max-min problem by projected gradient descent-ascent (GDA), i.e.

$$\Delta W = \eta_W\frac{\partial R}{\partial W} \tag{26}$$

$$\Delta L = -\eta_L\frac{\partial R}{\partial L} \tag{27}$$

followed by projection of W and L into the nonnegative orthant. Calculating the gradients yields

$$\Delta W \propto \frac{XU^\intercal}{T} - \frac{\partial\Phi}{\partial W} \tag{28}$$

$$\Delta L \propto \frac{XX^\intercal}{T} - \frac{\partial\Psi}{\partial L} \tag{29}$$

where X comes from optimizing

$$\max_{X\geq 0}\mathrm{Tr}\left(WUX^\intercal - \frac{1}{2}LXX^\intercal\right) \tag{30}$$

Here we have replaced the $\sup$ in Eqs. (23) and (25) by $\max$, which is appropriate if all diagonal elements of L are positive. In that case, $-\mathcal{F}$ is radially unbounded for $X \geq 0$, and the maximum over X exists. Note that the L update of Eq. (29) preserves symmetry of L.

A potential complication is that R is not guaranteed to be continuously differentiable everywhere. If $L \succ 0$, then Eq. (30) has a unique global optimum and no other local optima. Then X must change continuously as L varies, so that the derivatives of R in Eqs. (28) and (29) also change continuously. But if $L \not\succ 0$, then X could jump from one optimum to another, and the derivatives of R could have discontinuities.

For regular GDA, a steady state is a stationary point of $R(W, L)$. This should be modified for projected GDA.

Remark 1. *For projected GDA, a steady state is equivalent to a Karush-Kuhn-Tucker (KKT) point.*

The KKT conditions for a max-min problem with nonnegativity constraints are analogous to those for a regular optimization with nonnegativity constraints. For all i and a,

$$\frac{\partial R}{\partial W_{ia}} = 0 \text{ and } W_{ia} > 0 \tag{31}$$

or

$$\frac{\partial R}{\partial W_{ia}} \leq 0 \text{ and } W_{ia} = 0 \tag{32}$$

For all i and j,

$$\frac{\partial R}{\partial L_{ij}} = 0 \text{ and } L_{ij} > 0 \tag{33}$$

or

$$\frac{\partial R}{\partial L_{ij}} \geq 0 \text{ and } L_{ij} = 0 \tag{34}$$

While GDA is still a batch algorithm as written above, it can be turned into an online algorithm. Given W, L, and a randomly chosen input vector $\mathbf{u}$, solve

$$\max_{\mathbf{x} \geq 0} \mathrm{Tr}\left(\mathbf{x}^\intercal W \mathbf{u} - \frac{1}{2}\mathbf{x}^\intercal L \mathbf{x}\right) \tag{35}$$

This optimization can be done by projected diagonally scaled gradient ascent

$$\mathbf{x} \leftarrow \left[\mathbf{x} + \eta_x \mathrm{diag}(L)^{-1}\left(W\mathbf{u} - L\mathbf{x}\right)\right]^+ \tag{36}$$

or by coordinate ascent

$$x_i \leftarrow \frac{1}{L_{ii}}\left[\sum_a W_{ia}u_a - \sum_{j, j\neq i} L_{ij}x_j\right]^+ \tag{37}$$

The matrices W and L (off-diagonal elements), originally introduced as Legendre transform variables, have now become the feedforward and lateral connections of a neural net. The diagonal elements of L can be interpreted as normalizing the input-output function of a single neuron, or normalizing the connections converging onto a single neuron.

After iterating to convergence, make the updates

$$\Delta W \propto \mathbf{x}\mathbf{u}^\intercal - \partial\Phi/\partial W \tag{38}$$

$$\Delta L \propto \mathbf{x}\mathbf{x}^\intercal - \partial\Psi/\partial L \tag{39}$$

If these are averaged over the random choice of the input vector $\mathbf{u}$, then they are equivalent to the gradient updates. In other words, the above is a stochastic online variant of GDA.

For the special cases of Eqs. (4) and (7), these take the form

$$\Delta W_{ia} \propto x_i u_a - \gamma W_{ia} - \kappa \sum_b W_{ib} \tag{40}$$

$$\Delta L_{ij} \propto x_i x_j - \mu L_{ij} - D_{ij} \tag{41}$$

The W update is Hebbian, because it is driven by the correlation of postsynaptic activity x_i and presynaptic activity u_a. The γ term is a linear weight decay, and the κ term gives rise to competition between connections that converge onto the same neuron i.

The L update is anti-Hebbian for $i \neq j$, because it is driven by the correlation of activities x_i and x_j. Following Földiák [1], we use the term "anti-Hebbian" because strengthening of inhibition by correlated activity makes the interaction between neurons i and j more negative. For the correlation game with hard constraint on output-output correlations ($\mu = 0$), there is only constant weight decay. For nonnegative similarity matching ($D = 0$), there is only linear weight decay. Both kinds of weight decay are included in the general case of our softened correlation game.

V. The Penalty Function and Correlations

We now empirically investigate the impact of modifying μ in Eq. (4) for the same choice of $\Phi(W)$ and D studied in the previous section. For fixed D, we intuitively expect that when μ is small, $XX^\intercal/T$ should not be significantly larger than D, because otherwise the penalty function Ψ^* of Eq. (4) would contribute a large negative value to the total objective which we are trying to maximize. On the other hand we expect that increasing μ will make the objective function more tolerant of $XX^\intercal/T$ being larger than D.

We set the matrix D, a soft constraint on output-output correlations, to have q^2 on the diagonal and p^2 elsewhere,

$$D = \begin{bmatrix} q^2 & p^2 & \dots & p^2 \\ p^2 & q^2 & \dots & p^2 \\ \vdots & \vdots & \ddots & \vdots \\ p^2 & p^2 & \dots & q^2 \end{bmatrix} \tag{42}$$

We use the gradient ascent algorithm to train networks using 64 neurons on the first 1000 examples of the MNIST handwritten digits dataset. We set $q = 1.0, p = 0.3$ and $\gamma = 1.0, \kappa = 0.1$. We explore $\mu \in \{0.5, 1.0, 2.0, 4.0, 8.0, 16.0, 32.0, 64.0\}$. We also train another network with $\mu = 1.0$ and $D = 0$, which is the case of nonnegative similarity matching [5]. We empirically see the distributions of both on- and off-diagonal elements of $XX^\intercal/T$ shift to the right as we increase μ.

Fig. 1 also plots the ratio of the average off-diagonal to the average on-diagonal correlations. As $\mu \to 0$ we see most off-diagonal correlations are tightly clustered around p^2 while most on-diagonal correlations are tightly clustered around q^2 so the ratio is roughly p^2/q^2. In this limit, we approach the original correlation game with hard constraints [3].

On the other hand when μ grows, both the on-diagonal and off-diagonal correlations grow. For these settings, the ratio appears to asymptote to 0.2, which is the ratio achieved by setting $D = 0$ and $\mu = 1.0$ as in nonnegative similarity matching [5]. This makes sense because when μ is large, $XX^\intercal/T$ is large relative to D so $[XX^\intercal/T - D]^+$ is more nearly equal to $[XX^\intercal/T]^+$. Another way to see this is by examining the updates for L

$$\Delta L \propto \frac{XX^\intercal}{T} - \mu L - D \tag{43}$$

When μ is large, D becomes relatively unimportant so $\mu L - D \approx \mu L$

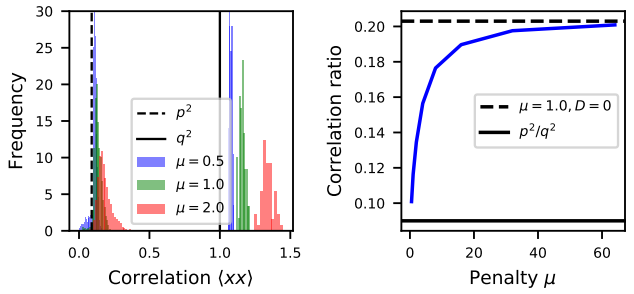

Fig. 1. Output-output correlations (elements of $XX^\intercal/T$) for varying μ. Left: histograms of $XX^\intercal/T$ for $\mu \in \{0.5, 1.0, 2.0\}$. Decreasing μ seems to shift the distributions of both the on- and off-diagonal elements of $XX^\intercal/T$ downward and decrease the variance of these distributions. Right: we plot the ratio of the average off-diagonal correlation to the average on-diagonal correlation as a function of μ. When μ is small, this ratio is nearly equal to p^2/q^2. As μ grows this ratio appears to asymptote the same output-output correlation ratio when $\mu = 1.0, D = 0.0$.

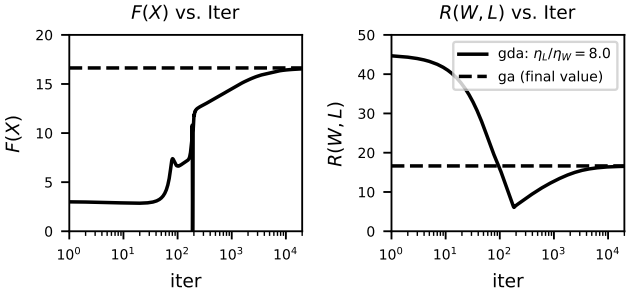

Fig. 2. Comparing $F(X)$ and $R(W, L)$ from gradient descent-ascent algorithm with $\eta_L/\eta_W = 8.0$ (solid curves) to final value of $F(X)$ from gradient ascent algorithm (dashed line). A logarithmic timescale is used for the x-axis. There seems to be 2 distinct phases, the first occurs in the first 100 steps where R decreases rapidly the 2nd occurs after where R gradually rises. Both gradient ascent and GDA yield solutions with similar value of the objective $F(X)$.

VI. Experiments with gradient ascent and GDA

We have introduced two algorithms, gradient ascent and GDA, for solving the unsupervised learning problem of Eq. (1). There are several ways in which the two algorithms might yield different results.

1) The global optimum of the dual problem might not equal the global optimum of the primal problem (1), as Eq. (23) is only an inequality.
2) Gradient ascent might find a local optimum of the primal problem (1) that is not a global optimum.
3) GDA might find a local optimum of the dual problem (right hand side of Eq. 23) that is not a global optimum.
4) GDA might not converge.

To probe how important these issues are in practice, we conducted numerical experiments with the gradient ascent and GDA algorithms.

We used 64 neurons and set $q = 1.0, p = 0.3, \mu = 1.0$ and $\gamma = 1.0, \kappa = 0.1$. We ran both gradient ascent and GDA using full batch updates with the first 1000 examples of the MNIST handwritten digits dataset. We used batch GDA to investigate duality, because the stochasticity of online GDA could complicate comparisons.

For GDA, we randomly initialize W by drawing its elements from a Uniform$(0, 1)$ distribution, and then normalizing the rows to sum to one. We initialize L to the identity matrix. We set $\eta_W = 0.0005$ and $\eta_L = 0.004$. The optimization over X in Eq. (30) uses projected gradient ascent on X to find a local optimum. We ignore the possibility that this could be a local optimum that is not a global optimum.

For gradient ascent (12), we use the same randomly initialized W to initialize $X = UW$, and a learning rate parameter of $\eta_X/T = 0.01$.

We train both algorithms for $20,000$ iterations, long enough for convergence. In Fig. 2, we plot the values of both $F(X)$ and $R(W, L)$ along the GDA training trajectory. We compare these values to the final value of F returned by gradient ascent. We observe that all three values are nearly equal by $20,000$ iterations. In this particular numerical experiment, the issues mentioned at the beginning of this section do not seem problematic.

VII. Strong duality

We observed empirically that gradient ascent and GDA can yield similar results. It would be interesting to find sufficient conditions guaranteeing that the results are the same. To address this question, we study whether the global solutions of Eqs. (1) and (23) are the same, i.e., whether the upper bound in Eq. (23) is an equality. For the time being, we neglect the remaining items in the list, i.e., whether these global solutions are actually found by gradient ascent and GDA.

A global minimax point is not necessarily a local minimax point [9].[2] But for the following theorem, we will assume that a global solution of the max-min problem in Eq. (23) is a local solution. This simplifies the situation because a local solution must be a KKT point. If a solution is not a KKT point, it is of less interest as it cannot be a steady state of GDA anyway.

Theorem 1. *Suppose that a global solution (W^*, L^*, X^*) of right hand side of Eq. (23) exists, and L^* is positive definite. Then the upper bound in Eq. (23) becomes an equality.*

Before proving this theorem, we must take care of some preliminaries.

Lemma 1. *Suppose that $(\mathbf{x}^*, \mathbf{y}^*)$ is a global minimax point of $f(\mathbf{x}, \mathbf{y})$ with nonnegativity constraints, i.e., a global solution of*

$$\min_{\mathbf{x}\geq 0} \max_{\mathbf{y}\geq 0} f(\mathbf{x}, \mathbf{y}) \tag{44}$$

Suppose that f is twice differentiable in a neighborhood of $(\mathbf{x}^*, \mathbf{y}^*)$, and $f(\mathbf{x}, \cdot)$ is strongly concave for all $\mathbf{x}$ in a neighborhood of $\mathbf{x}^*$. Then $(\mathbf{x}^*, \mathbf{y}^*)$ is a local minimax point.

[2]Standard optimization is simpler: a global solution is guaranteed to be a local solution.

Proof. This is a variant of Theorem 23 of Ref. [9], which also gives a precise definition of local minimax point. □

Lemma 2. *A local minimax point with nonnegativity constraints is a KKT point.*

Proof. This is a variant of Proposition 18 of [9]. □

Definition 1.

$$S(L, X) = \mathcal{F}(W^*, L, X) \tag{45}$$

where W^* minimizes Eq. (23)

Lemma 3. *If (L^*, X^*) is a global minimax point of S, and $L^* \succ 0$ then (L^*, X^*) must be a local minimax point, and hence a KKT point.*

Proof. If $L^* \succ 0$, then for all L in a neighborhood of L^*, $L \succ 0$ and $S(L, \cdot)$ is strongly concave. Apply Lemmas 1 and 2. □

Lemma 4. *(From [10]) If (L^*, X^*) is a saddle point of S, i.e. $\forall\, X, L$:*

$$S(L^*, X) \leq S(L^*, X^*) \leq S(L, X^*) \tag{46}$$

then minimax equality holds:

$$\sup_X \inf_L S(L, X) = \inf_L \sup_X S(L, X) \tag{47}$$

Having taken care of the preliminaries, we now return to Theorem 1.

Proof of Theorem 1. By the max-min inequality, we know that for any fixed W

$$\max_X \min_L \mathcal{F}(W, L, X) \leq \min_L \sup_X \mathcal{F}(W, L, X) \tag{48}$$

Let W^* be a maximum of the right hand side. If we can show that equality holds for $W = W^*$, then we are done.

Because we have assumed L^* is positive definite, we can apply Lemma 3 to show that (L^*, X^*) is a KKT point of $S(L, X)$.

Because we have assumed L^* is positive definite, we know that $S(L^*, X)$ is concave in X. From concavity in X and the fact that X^* is a KKT point, we know that $S(L^*, X^*) \geq S(L^*, X)$ for all X.

Because we have assumed $\Psi(L)$ is convex, we have that $S(L, X^*)$ is convex in L. From convexity in L and the fact that L^* is a KKT point we also know that $S(L^*, X^*) \leq S(L, X^*)$ for all L.

These two inequalities imply that (L^*, X^*) is a saddle point of S. Lemma 4 tells us that the existence of a saddle point of S implies minmax equality holds. Therefore when $W = W^*$ equality holds in Eq. (48) □

Note that this proof does not require positive definiteness of L for all W, just for $W = W^*$. One might wonder whether the $L^* \succ 0$ condition is ever satisfied. For the special case of nonnegative similarity matching, $L^* = X^* X^{*\intercal}/T$, which is always positive semidefinite and typically will be positive definite.

For the simulation shown in the previous section, L^* is not positive definite. However the magnitudes of the negative eigenvalues are quite small in comparison to the positive eigenvalues (Fig. 3). Therefore the potential problem of local optima in Eq. (30) might not actually be a problem in this example.

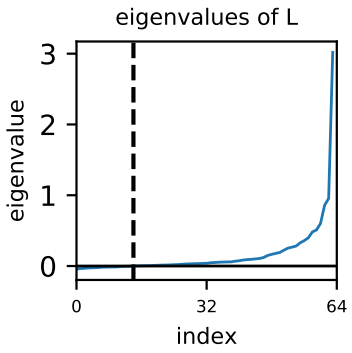

Fig. 3. Eigenvalues of L^* after training with GDA. L^* is not positive definite, but the magnitudes of the negative eigenvalues are quite small. The ratio of the most positive to the most negative eigenvalue is approximately -68. The dashed line indicates the location of the first positive eigenvalue.

VIII. Convergence in Fast Inhibition Regime

The preceding section provided a sufficient condition for strong duality, i.e., for equality to hold in Eq. (23). But even when strong duality holds, GDA might not find a solution of $\max_{W \geq 0} \min_{L \geq 0} R(W, L)$. In fact, GDA might not even converge at all.

In this section, we empirically investigate convergence of GDA using the same Φ, Ψ defined in Eqs. (7, 15) that were studied in Section V. This time we vary the ratio of time scales for inhibitory and excitatory plasticity, η_L/η_W.

For all training runs, we used $\eta_W = 0.0005$. We explored five different learning rates for η_L, giving five different ratios: $\eta_L/\eta_W \in \{0.50, 1.0, 2.0, 4.0, 8.0\}$. To ensure that the results were not dependent on the finite step sizes used, we also generated learning curves using the same five ratios, but with both η_W and η_L halved. We ensured that for each ratio, the curve generated with both learning rates halved was similar to the original curve, except rescaled horizontally by a factor of 2. We present the convergence results in Fig. 4 and Fig. 5, and show the learned output-output correlations and features in Fig. 6 and Fig. 7, respectively.

We provide a summary of our main observations:

1) When $\eta_L/\eta_W \geq 4.0$:
   a) GDA appeared to converge.
   b) GDA learned features that were qualitatively similar to features learned from gradient ascent on F.
   c) GDA found X such that $F(X)$ was nearly equal to the value from direct gradient ascent on F.

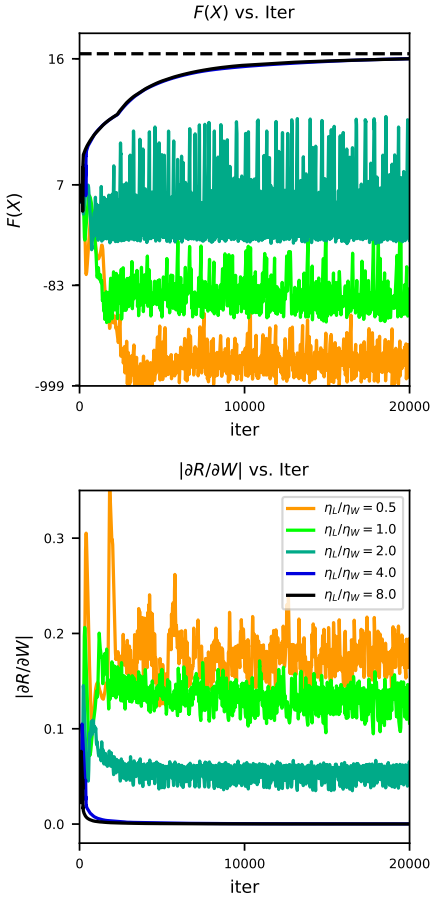

Fig. 4. Simulating the GDA-based neural network learning algorithm with $q = 1.0, p = 0.3, \mu = 1.0$ for various inhibitory-excitatory plasticity ratios η_L/η_W. When inhibitory plasticity is sufficiently slow ($\eta_L/\eta_W \leq 2.0$) learning does not appear to converge. Conversely, when inhibitory plasticity is fast ($\eta_L/\eta_W \geq 4.0$), learning does appear to converge. Note that the updates are non-stochastic and both η_L and η_W are small. What appears to be noise in these small η_L/η_W curves actually appears to be oscillations when zoomed in (see Fig. 5 for an example)

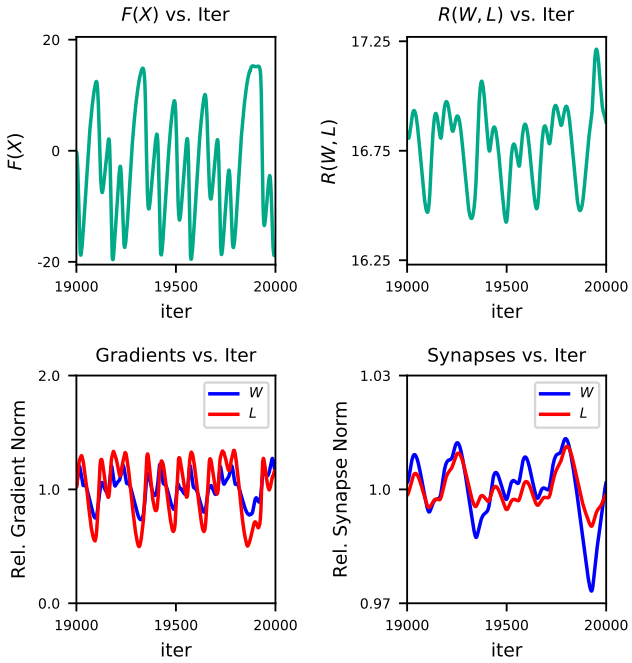

Fig. 5. Non-transient oscillations when $\eta_L/\eta_W = 2.0$. Top left: $F(X)$ vs. iter (this is just a zoomed in version of the $\eta_L/\eta_W = 2.0$ curve in Fig. 4) Top right: $R(W, L)$ vs. iter. Bottom left: normalized gradient norms, specifically the $\ell 1$ norm of $\partial R/\partial W$ divided by the average $\ell 1$ norm of $\partial R/\partial W$ from 19,000 to 20,000 iterations and similarly for L. Bottom right: normalized synapse norms, specifically the $\ell 1$ norm of W divided by the average $\ell 1$ norm of WW from 19,000 to 20,000 iterations and similarly for L

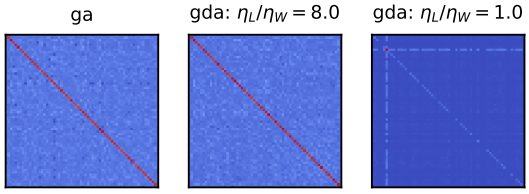

Fig. 6. Comparing output-output correlation matrices after 20k steps of a) gradient ascent algorithm b) gradient ascent-descent algorithm with $\eta_L/\eta_W = 8.0$ c) gradient ascent-descent algorithm with $\eta_L/\eta_W = 1.0$. When $\eta_L/\eta_W = 8.0$, learning appears to converge and output-output correlations appear qualitatively similar to output-output correlations from the gradient ascent algorithm ($\langle x_i^2 \rangle \approx q^2$ and $\langle x_i x_j \rangle \approx p^2$ for $i \neq j$). When $\eta_L/\eta_W = 1.0$, learning does not appear to converge and after 20k steps we observe that one of the neurons has $\langle x_i^2 \rangle \gg q^2$.

d) GDA learning curves did not seem to change much by increasing η_L/η_W beyond 4.0.

2) When $\eta_L/\eta_W \leq 2.0$
   a) GDA did not appear to converge.
   b) GDA dynamics exhibited oscillations whose amplitude increased with decreasing η_L/η_W.
   c) Decreasing η_L/η_W lowered the averaged value of $F(X)$ along GDA trajectories.

We provide as follows one potential explanation for the observation that when η_L/η_W is sufficiently large, GDA appears to converge and it yields X with similar value of $F(X)$ to the gradient ascent algorithm.

Setting the learning rate for inhibitory synapses large suggests that L quickly moves to a location such that $\partial R/\partial L \approx 0$. Because we know that $R(W, L)$ is strictly convex in L, this suggests that L in fact finds the unique minimum $L \approx \arg\min_{L' \geq 0} R(W, L')$. Further, strict convexity in L suggests that small changes in W do not drastically change L, in other words, $\partial \left[\arg\min_{L' \geq 0} R(W, L')\right] / \partial W$ should be finite.

Because W is updating slowly, W effectively "sees" the function $\min_{L' \geq 0} R(W, L')$ which is continuous in W and the updates for W are essentially just gradient ascent updates on a continuous function which are generally expected to converge.

To understand why the converse does not hold, i.e. why setting η_W large does not necessarily yield convergence, we examine the behavior of $R(W, L)$ for fixed L. In particular, for some values of L, $\max_{W \geq 0} R(W, L)$ does not even exist,

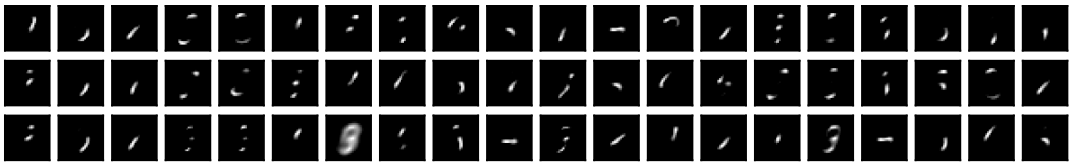

Fig. 7. Learned features after 20k steps of (top row) gradient ascent, (middle row) GDA with $\eta_L/\eta_W = 8.0$ (bottom row) GDA with $\eta_L/\eta_W = 1.0$. Each image shows one of the first 20 rows of W^*, reshaped into the original 28x28 image space. We observe that features learned by gradient ascent and GDA with $\eta_L/\eta_W = 8.0$ appear qualitatively similar. Many of the features learned by GDA with $\eta_L/\eta_W = 1.0$ also appear qualitatively similar to those of gradient ascent. However there are outlier features, especially the one in column 7, which corresponds to the one neuron in Fig. 6 with a high value of $\langle x_i^2 \rangle$.

at least for the form of $\Phi(W)$ we consider.

This can be seen even in the simplest case when U, W, L, X are all just non-negative scalars. In this situation, we can directly $X = WU/L$ and therefore analytically write R:

$$R(W, L) = \frac{1}{2}W^2U^2/L - \frac{1}{2}(\gamma + \kappa)W^2 + \frac{1}{2}\Psi(L) \qquad (49)$$

For any $L < \frac{U^2}{\gamma+\kappa}$ we have that $R \to \infty$ and $W \to \infty$. When η_W is large, we might see oscillations as W quickly grows in magnitude before L has enough time to stabilize the growth.

IX. Conclusion

We have examined two different algorithms for performing the optimization of Eq. (1). The first algorithm directly maximizes $F(X)$ via projected gradient ascent on X. The second constructs an upper bound $\max_{W\geq 0} \min_{L\geq 0} R(W, L)$ whose $\max\min$ points are yield an upper bound on F. Projected GDA is then used to find $\max\min$ points.

Interestingly, the case where strong duality is not guaranteed to hold is exactly the case where the inner optimization $\max_{X\geq 0} \mathcal{F}(W, L, X)$ has the potential for multiple local optima. This is the situation where L is not positive definite and therefore $\mathcal{F}(W, L, \cdot)$ is not strongly concave.

We have also empirically investigated convergence properties of GDA. Unlike gradient ascent, which is guaranteed to at least find a local maximum of F, GDA is not guaranteed to converge to a steady state, and even if it does, the relation of the steady state to the $\max\min$ points of the objective R is unclear. When inhibitory plasticity was slow, we observed that learning did not appear to converge. When inhibitory plasticity was fast, we observed that learning converged to a point W^*, L^*, X^* such that $R(W^*, L^*)$ that was very nearly equal to $F(X^*)$.

We gave an informal explanation of convergence in the fast inhibitory plasticity regime as a consequence of the strict concavity assumption on L. We used this to argue that W effectively did gradient ascent on $\min_{L'\geq 0} R(W, L')$. Lack of convergence in the slow inhibitory plasticity regime was explained via the observation that $\max_{W\geq 0} R(W, L)$ was not guaranteed to exist for arbitrary L.

An even more informal explanation is that for these learning rules, inhibitory plasticity is a key source of negative feedback regulating feedforward excitation (there is also negative feedback in the update rules for W resulting from the term $-\Phi'(W)$). When this negative feedback is too slow, feedforward excitation is allowed to run away before the inhibition has time to catch up and stabilize growth. This may explain the oscillations of W and L seen in Fig. (5).

Convergence of GDA would be easy to prove if R were concave-convex. In our case, R is guaranteed to be convex in L but is generally nonconcave in W. Lyapunov function proofs [6], [7] guarantee convergence of nonconcave-convex GDA in the limit of fast L dynamics, but should be extended to cover projected GDA. The approach of Ref. [8] should also be applicable, but also should be extended to projected GDA.

It would be interesting to explore other ways to stabilize learning dynamics besides modifying η_L/η_W. We also wonder whether learning instabilities analogous to the ones we have simulated might be observed in real brains.

References

[1] P. Földiak, "Forming sparse representations by local anti-hebbian learning," *Biological cybernetics*, vol. 64, no. 2, pp. 165–170, 1990.

[2] C. Pehlevan, T. Hu, and D. B. Chklovskii, "A hebbian/anti-hebbian neural network for linear subspace learning: A derivation from multidimensional scaling of streaming data," *Neural computation*, vol. 27, no. 7, pp. 1461–1495, 2015.

[3] H. S. Seung and J. Zung, "A correlation game for unsupervised learning yields computational interpretations of hebbian excitation, anti-hebbian inhibition, and synapse elimination," *arXiv preprint arXiv:1704.00646*, 2017.

[4] C. Pehlevan, A. M. Sengupta, and D. B. Chklovskii, "Why do similarity matching objectives lead to hebbian/anti-hebbian networks?" *Neural computation*, vol. 30, no. 1, pp. 84–124, 2018.

[5] C. Pehlevan and D. B. Chklovskii, "A hebbian/anti-hebbian network derived from online non-negative matrix factorization can cluster and discover sparse features," in *48th Asilomar Conference on Signals, Systems and Computers, ACSSC 2014, Pacific Grove, CA, USA, November 2-5, 2014*, 2014, pp. 769–775. [Online]. Available: https://doi.org/10.1109/ACSSC.2014.7094553

[6] H. S. Seung, T. J. Richardson, J. C. Lagarias, and J. J. Hopfield, "Minimax and hamiltonian dynamics of excitatory-inhibitory networks," in *Advances in neural information processing systems*, 1998, pp. 329–335.

[7] H. S. Seung, "Convergence of gradient descent-ascent analyzed as a newtonian dynamical system with dissipation," *arXiv preprint arXiv:1903.02536*, 2019.

[8] T. Lin, C. Jin, and M. I. Jordan, "On gradient descent ascent for nonconvex-concave minimax problems," *CoRR*, vol. abs/1906.00331, 2019. [Online]. Available: http://arxiv.org/abs/1906.00331

[9] C. Jin, P. Netrapalli, and M. I. Jordan, "What is local optimality in nonconvex-nonconcave minimax optimization?" *arXiv preprint arXiv:1902.00618*, 2019.

[10] D. Bertsekas, "6.253 lecture notes on convex analysis and optimization," Spring 2004.