Effective Learning Rules as Natural Gradient Descent

Lucas Shoji
lshoji@mit.edu
Department of Physics and Department of Brain and Cognitive Sciences, MIT, Cambridge, MA 02139, USA

Kenta Suzuki
kjsuzuki@princeton.edu
Department of Mathematics, Princeton University, Princeton, NJ 08544, USA

Leo Kozachkov
leokoz8@brown.edu
Department of Brain and Cognitive Sciences, MIT, Cambridge, MA 02139, USA; and Thomas J. Watson Research Center, IBM Research, Yorktown Heights, NY 10598, USA

We establish that a broad class of effective learning rules—those that improve a scalar performance measure over a given time window—can be expressed as natural gradient descent with respect to an appropriately defined metric. Specifically, parameter updates in this class can always be written as the product of a symmetric positive-definite matrix and the negative gradient of a loss function encoding the task. Given the high level of generality, our findings formally support the idea that the gradient is a fundamental object underlying all learning processes. Our results are valid across a wide range of common settings, including continuous-time, discrete-time, stochastic, and higher-order learning rules, as well as loss functions with explicit time dependence. Beyond providing a unified framework for learning, our results also have practical implications for control as well as experimental neuroscience.

1 Introduction

Identifying the brain's learning rules is a major goal in neuroscience, just as developing effective optimizers is in artificial intelligence research (Lim et al., 2015; Nayebi et al., 2020; Richards & Kording, 2023; Francioni et al., 2023; Bredenberg & Savin, 2024). Decades of work have produced a diverse array of learning rules, varying in biological plausibility and efficacy—from local Hebbian updates to exact gradient-based methods like backpropagation (Hopfield, 1982; Grossberg, 1987; Widrow & Lehr, 1990; Abbott &

Neural Computation **38, 71–96** (2026)
https://doi.org/10.1162/NECO.a.1474

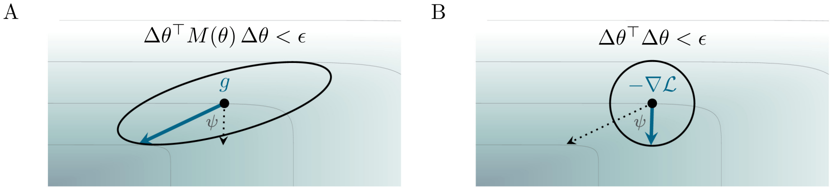
A
$\Delta\theta^\top M(\theta)\,\Delta\theta < \epsilon$
g
ψ
B
$\Delta\theta^\top \Delta\theta < \epsilon$
$-\nabla\mathcal{L}$
ψ

Figure 1: (A) Contour lines of a loss function (darker colors = lower loss). Parameters update in the direction of g. If this update decreases the loss and the step size is small, g is equivalent to steepest descent with a non-Euclidean metric, $M(\theta)$. In this case, the angle ψ between g and the negative gradient is acute. Ellipse: ϵ-ball in this non-Euclidean metric. (B) Steepest descent with the Euclidean metric. Circle: ϵ-ball in the Euclidean metric.

Regehr, 2004; Dayan & Abbott, 2005; Fiete & Seung, 2006; Lillicrap et al., 2016). This letter does not offer any new learning rules. Instead, we show that under mild assumptions, all effective learning rules–those that improve a scalar measure of performance–fit within a simple, unifying framework. Specifically, they can be expressed as the product of a symmetric, positive-definite matrix and the negative gradient of a loss function. This corresponds to performing steepest descent with a non-Euclidean metric (see Figure 1; Amari, 1998).

It is well known that if a learning rule updates parameters by following the negative gradient of a loss function, the loss does not increase along the parameter trajectories (Cauchy, 1847; Nocedal & Wright, 1999). However, many learning rules do not fit this “pure” gradient descent form. Indeed, there are compelling reasons to believe that the brain’s learning rules cannot be expressed as pure gradient descent (Surace et al., 2018; Lillicrap et al., 2016; Bredenberg & Savin, 2024). Fortunately, there are many ways to decrease a loss function beyond traditional gradient descent. One notable class of algorithms, which we focus on in this letter, is natural gradient descent (Amari, 1998).

In natural gradient algorithms, parameter updates are written as the product of a symmetric positive-definite matrix and the negative gradient. If a learning rule can be expressed in this form, it is considered “effective” because it guarantees improvement of a scalar performance measure over time (assuming small step sizes). Given the flexibility of choosing the positive definite matrix, one can ask the converse question: If a learning rule is effective, can it be written as natural gradient descent? We show that for a wide class of effective learning rules, this is indeed the case. For example, our results hold for all effective continuous-time learning rules.

While infinitely many metrics are consistent with a given effective learning rule, we prove that all such metrics share a canonical form. We further identify several metrics that are optimal in terms of allowing us in certain cases to establish tight bounds on the rate of convergence to minima.

1.1 Formal Setting. We consider a set of D real numbers $\theta \in \mathbb{R}^D$ that parameterize the function of a system. In the case of biology, these numbers can represent biophysical variables such as synaptic diffusion constants or receptor densities (Richards & Kording, 2023). In the case of artificial neural networks, these numbers can represent synaptic weights between units. We analyze two common methods for updating θ toward the goal of improving performance on a task (or set of tasks): continuous-time evolution and discrete-time updates. In the former, θ evolves continuously according to a flow,

$$\frac{d\theta}{dt} = g(\theta, t), \tag{1.1}$$

where $g(\theta, t)$ is a potentially nonlinear, time-dependent function. At this stage, we impose no restrictions on this function (e.g., smoothness). In discrete-time updates, changes to θ occur at discrete time intervals,

$$\theta_{t+1} = \theta_t + \eta\, g(\theta_t, t), \tag{1.2}$$

where $\eta > 0$ is a learning rate parameter. This setting is general enough to capture supervised learning, self-supervised learning, as well as in-context learning (where t may be identified with layers in a neural network). Also note that equations 1.1 and 1.2 include techniques that rely on defining higher-order derivatives of θ, such as accelerated gradient methods (Muehlebach & Jordan, 2019). In this case, one can arrive back at the form of equations 1.1 and 1.2 by expanding the state space.[1]

1.2 Effective Learning Rules Do Not Require Monotonic Improvement. We assume that a parameter vector θ can be associated with a system that performs some task. For example, suppose θ contains the weights of a neural network after training. This neural network can then be evaluated based on its performance on some task. We define a learning rule as effective over time interval $m > 0$ with respect to a scalar performance measure if it improves this measure within that interval. We use the loss $\mathcal{L}$ to denote this measure, where improvement means

$$\mathcal{L}(t + m) < \mathcal{L}(t). \tag{1.3}$$

[1] For example, for second-order methods, define the extended state space $[v \;\; \theta]$, where v: $-\dot{\theta}$.

Note that this definition does not require monotonic improvement in the performance measure. In particular, equation 1.3 allows for temporary setbacks, that is, $d\mathcal{L}/dt > 0$, so long as the setbacks do not outweigh the progress on average. This includes, for example, learning rules that take "one step back and two steps forwards." Note also that although the loss does not decrease monotonically along trajectories of equation 1.1, the average loss $\mathcal{L}_{\text{avg}}$ does, because

$$\mathcal{L}_{\text{avg}} := \frac{1}{m}\int_t^{t+m} \mathcal{L}(s)\, ds \qquad \Longrightarrow \qquad \dot{\mathcal{L}}_{\text{avg}} = \frac{\mathcal{L}(t+m) - \mathcal{L}(t)}{m} < 0,$$

where the inequality was obtained by using assumption 1.3. The same argument can be applied to discrete-time updates. In this case, the average loss continually improves, because

$$\mathcal{L}_{\text{avg}}(t) = \frac{1}{m}\sum_{\tau=t}^{t+m-1} \mathcal{L}(\tau) \quad \Longrightarrow \quad \mathcal{L}_{\text{avg}}(t+1) - \mathcal{L}_{\text{avg}}(t) = \frac{\mathcal{L}(t+m) - \mathcal{L}(t)}{m} < 0.$$

Therefore, for the remainder of the letter, we assume without loss of generality that the loss function $\mathcal{L}$ does monotonically decrease. Also note that while the average loss is a particularly convenient measure of asymptotic improvement, we can in fact consider much more general measures that guarantee asymptotic improvement of a performance measure without continual improvement—for example, by considering a sequence of loss (e.g., Lyapunov) functions as done by Ahmadi and Parrilo (2008). Finally, while we only consider differentiable loss functions in this letter, analogous results hold for non-differentiable losses using suitable replacements for the gradient of the loss (Clarke, 1975).

1.3 (Natural) Gradient Descent.

Gradient descent is a prototypical algorithm for decreasing a loss function. However, it is by no means the only algorithm that does so. An important generalization of gradient descent is natural gradient descent (Amari, 1998),

$$\dot{\theta} = -M^{-1}(\theta, t)\, \nabla_\theta \mathcal{L}, \tag{1.4}$$

where $M(\theta, t)$ is some symmetric positive-definite matrix.[2] To see that natural gradient descent indeed decreases the loss $\mathcal{L}$ in continuous time, suppose that θ is not at a stationary point, that is, $\|\nabla_\theta \mathcal{L}\| > 0$. Then,

[2] Technically, this is natural gradient flow, called a metric. We will use the term *descent* to refer to both continuous and discrete updates.

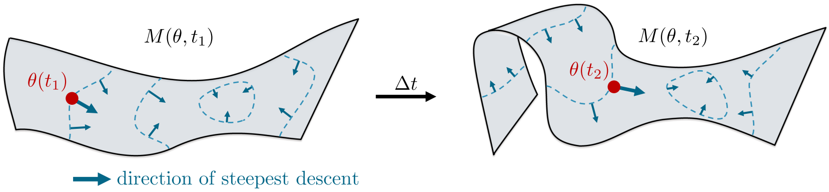

Figure 2: Natural gradient descent minimizes a loss function (dashed contours) by evolving the parameters θ in the direction of steepest descent in a non-Euclidean space. This space, a D-dimensional manifold with metric $M(\theta, t)$, is visualized as a surface embedded in a higher-dimensional Euclidean space. We demonstrate that a wide class of learning rules that decreases the loss function (not necessarily monotonically) fits this framework. In this context, the dynamics of both θ and M are determined by the learning rule and the loss function.

$$\dot{\mathcal{L}} = \nabla_\theta \mathcal{L}^\top \, \dot{\theta} = -\nabla_\theta \mathcal{L}^\top M^{-1}(\theta, t) \, \nabla_\theta \mathcal{L} \leq -\frac{\|\nabla_\theta \mathcal{L}(s)\|^2}{\lambda_{\max}(M)} < 0, \tag{1.5}$$

where $\lambda_{\max}(M) > 0$ denotes the largest eigenvalue of M. The first equality follows from the chain rule, the second equality follows from substituting in equation 1.4, and the inequality is obtained by using the Rayleigh quotient (Horn & Johnson, 2012). The above conclusion also holds in discrete time, for sufficiently small learning rate η.

There are two interesting connections between natural gradient descent and gradient descent. The first is that in the special case when $M = I$, natural gradient descent reduces to gradient descent. The second connection is that both gradient descent and natural gradient descent perform steepest descent: the negative gradient is the direction of steepest descent in Euclidean space, whereas the negative natural gradient denotes the direction of steepest descent in some non-Euclidean space—particular, in a space where unit lengths at point θ satisfy

$$a^\top M(\theta, t) \, a = 1,$$

which is called a metric. Since the metric may change over time, the geometry underlying learning is itself dynamic if the dynamics function g depends explicitly on time, evolving alongside the parameters (see Figure 2). Natural gradients underlie many techniques in machine learning and optimization (Kakade, 2001; Pascanu, 2013; Martens & Grosse, 2015; Martens, 2020; Dangel et al., 2024; Wensing & Slotine, 2020; Ollivier et al., 2017), control theory (Lee et al., 2018; Boffi & Slotine, 2021; Tzen et al., 2023; Wensing & Slotine, 2020), and, more recently, have enjoyed renewed interest in neuroscience (Surace et al., 2018; Pogodin et al., 2023; Bredenberg & Savin, 2024; Cornford et al., 2024).

2 Main Results

2.1 Continuous-Time Learning Rules. To streamline the notation, we define y as the negative gradient of $\mathcal{L}$ and the update vector $g(\theta, t)$, defined in equation 1.1, as g. This allows us to express the monotonic decrease of the loss function more concisely as $y^\top g > 0$. Our goal is to find a symmetric positive-definite matrix M that maps g to y, which ensures that g can be written in the natural gradient form,

$$Mg = y \qquad \Longleftrightarrow \qquad g = -M^{-1}\,\nabla_\theta \mathcal{L}.$$

Toward this goal, consider the matrix

$$M = \frac{1}{y^\top g} y y^\top + \sum_{i=1}^{D-1} u_i u_i^\top. \tag{2.1}$$

Here, the vectors u_i are chosen to span the subspace orthogonal to g, denoted by $g^\perp$: $-\{v \in \mathbb{R}^n : v^\top g = 0\}$. As desired, M maps the update vector g to the negative gradient direction y. By construction, M is symmetric and positive definite. Indeed, for any nonzero vector x, we have

$$x^\top M x = \frac{1}{y^\top g}\left(x^\top y\right)^2 + \sum_{i=1}^{D-1}\left(x^\top u_i\right)^2 > 0.$$

The inequality holds because x cannot be simultaneously orthogonal to both y and all the u_i, as this would contradict the assumption that $y^\top g > 0$. Later on, for a special family of metrics, we derive the full spectrum of M.

2.1.1 Canonical Form of the Metric. We now show in the following proof that any symmetric, positive-definite matrix M such that $Mg = y$, with $g^\top y > 0$, is of the form given in equation 2.1.

Proof. Let M satisfy the requirements given above. Define

$$M' := M - \frac{1}{y^\top g} y y^\top,$$

which is a symmetric matrix. We claim that for any nonzero $u \in g^\perp$,

$$u^\top M' u > 0.$$

If so, since the matrix is symmetric and an "orthogonal" eigendecomposition exists, it follows that M' is of the form $\sum_{i=1}^{D-1} u_i u_i^\top$ for some basis $\{u_i\}$ of $g^\perp$, proving the canonical form. To show this, first note that

$$M' g = M g - \frac{1}{y^\top g} y y^\top g = 0. \tag{2.2}$$

Now take an arbitrary nonzero $u \in g^{\perp}$. Consider the projection of u to $y^{\perp}$ along g,

$$u' = u - \frac{y^{\top} u}{y^{\top} g} g,$$

which is nonzero and orthogonal to y.[3] Together with equation 2.2 we see

$$u^{\top} M' u = (u')^{\top} M' u' = (u')^{\top} M u' > 0,$$

concluding our proof. □

2.1.2 One-Parameter Family of Metrics. Although the matrix M in equation 2.1 is positive-definite, it will be useful later to have an explicit expression for the eigenvalues of M, for example, in terms of the angle between y and g. While this is challenging for a general M, we observe that a one-parameter family of valid metrics M can be written as

$$M = \frac{1}{y^{\top} g} y y^{\top} + \alpha \sum_{i=1}^{D-1} u_i u_i^{\top} = \frac{1}{y^{\top} g} y y^{\top} + \alpha \left(I - \frac{g g^{\top}}{g^{\top} g} \right), \tag{2.3}$$

where $\alpha > 0$ can depend on y and g, and $u_i^{\top} u_j = \delta_{ij}$. These are exactly the matrices M, which acts as the scalar α on the orthogonal complement of the span of g and y. We show in appendix C that the full spectrum of M can be derived for this family of metrics, as a function of α.

2.1.3 Optimal Metrics. We further show in appendix C that the one-parameter family (see equation 2.3) contains several globally "optimal" metrics. In particular, we prove that among all possible metrics, not just within this one-parameter family, the metric M_{opt}, which achieves the smallest condition number, is given by setting $\alpha = \frac{y^{\top} y}{g^{\top} y}$ in equation 2.3. The condition number of the metric is a crucial quantity that in general can be used to estimate the time until convergence to a minimum (see appendix A for an example involving a strongly convex loss function). The spectrum of M_{opt} can be written in terms of the angle between y and g, which we call $\psi \in (-\frac{\pi}{2}, \frac{\pi}{2})$, as follows:

$$\lambda_{\max/\min}(M_{\mathrm{opt}}) = \frac{||y||}{||g||} \left[\frac{1}{\cos(\psi)} \pm |\tan(\psi)| \right]$$

$$\lambda_d(M_{\mathrm{opt}}) = \frac{||y||}{||g||} \frac{1}{\cos(\psi)}, \tag{2.4}$$

[3] This projection is well defined by the assumption of effective learning, that $y^{\top} g > 0$.

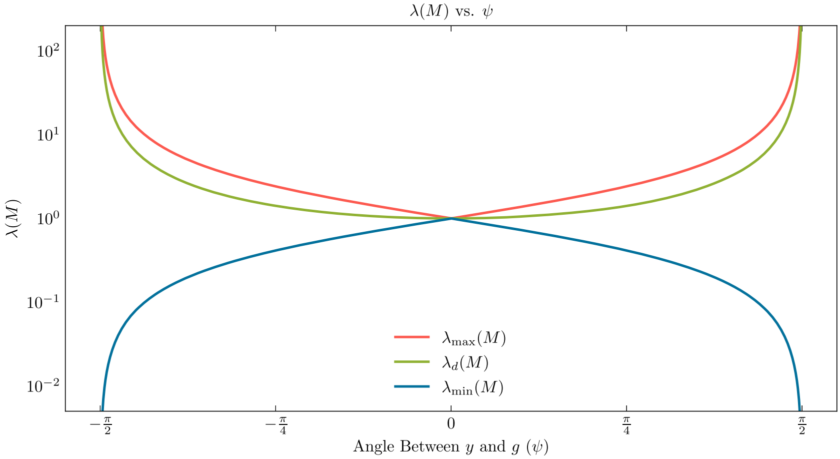

Figure 3: Eigenvalues of the optimal metric M_{opt} as a function of the angle ψ between vectors y and g, with the norm ratio $\|y\|/\|g\|$ fixed at unity. Refer to equation 2.4 in the main text.

where $1 < d < D$. See Figure 3 for a plot of these curves. The condition number κ of the optimal metric M_{opt} has a particularly simple form as a function of ψ:

$$\kappa(M_{\text{opt}}) = \frac{\lambda_{\max}(M_{\text{opt}})}{\lambda_{\min}(M_{\text{opt}})} = \frac{1 + |\sin(\psi)|}{1 - |\sin(\psi)|}. \tag{2.5}$$

Note that $1/\kappa(M_{\text{opt}}) \in (0, 1]$ and can be naturally viewed as a measure of similarity between g and y. We also show in appendix C that among all possible metrics, the one with the minimum possible $\lambda_{\max}(M)$ is asymptotically approached as $\alpha \to 0$. It can be shown that this minimum is given by

$$\lambda_{\max}(M) > \frac{\|y\|}{\|g\|} \frac{1}{\cos(\psi)}.$$

Similarly, the metric with the maximum possible $\lambda_{\min}(M)$ is approached asymptotically when $\alpha \to \infty$. This maximum is given by

$$\lambda_{\min}(M) < \frac{\|y\|}{\|g\|} \cos(\psi).$$

These results will be particularly useful later, particularly when analyzing discrete-time learning rules in section 2.2.

2.1.4 Metric Asymptotics. It is clear from equation 2.1 that the metric M will "blowup" if the negative gradient y becomes orthogonal to the parameter update g. This is expected because in this case, learning does not occur ($d\mathcal{L}/dt = 0$). Furthermore, in this case, we would have that

$$y^\top g = g^\top M g = 0,$$

which contradicts the positive-definiteness of M. This can be confirmed by inspecting the eigenvalues of the metric M given in equation 2.4 and Figure 3. One sees that as the angle ψ between y and g approaches $\pi/2$ or $-\pi/2$, the smallest eigenvalue of the metric goes to zero, causing M to lose its positive definiteness, while the remaining eigenvalues tend to infinity.

2.1.5 Time-Varying Loss. So far, we have only considered loss functions $\mathcal{L}(\theta)$, which do not depend explicitly on the time t. However, there are many cases of interest where the loss can be thought of as changing in time, for example, in online convex optimization (Hazan, 2016). In this case, we can show that effective learning of a time-varying loss $\mathcal{L}(\theta, t)$ implies that the learning dynamics of an extended parameter vector may be written as natural gradient descent of this loss. We define this new extended vector as v and its time derivative as $\dot{v}$,

$$v := \begin{bmatrix} \theta & t \end{bmatrix}^\top \qquad \Longrightarrow \qquad \dot{v} = \begin{bmatrix} \dot{\theta} & 1 \end{bmatrix}^\top .$$

Then the total derivative of the time-varying loss as θ evolves in time is given by,

$$\dot{\mathcal{L}} = \frac{\partial \mathcal{L}}{\partial \theta}^\top \dot{\theta} + \frac{\partial \mathcal{L}}{\partial t} = \frac{\partial \mathcal{L}}{\partial v}^\top \dot{v} \ < 0.$$

Thus, we may conclude that updates to the extended variable v perform natural gradient descent on the time-varying loss $\mathcal{L}$,

$$\dot{v} = -M^{-1} \frac{\partial \mathcal{L}}{\partial v},$$

where M is constructed as before, with $y = -\frac{\partial \mathcal{L}}{\partial v}$ and $g = \dot{v}$.

2.2 Discrete-Time Learning Rules. Consider a discrete-time learning rule that decreases a loss function $\mathcal{L}$ at every step,

$$\theta_{t+1} = \theta_t + \eta\, g(\theta_t, t) \qquad \text{and} \qquad \mathcal{L}(\theta_{t+1}) - \mathcal{L}(\theta_t) < 0. \tag{2.6}$$

Note that the additive form of the update in equation 2.6 is purely for convenience, and is equivalent to a nonadditive form (e.g., multiplicative, as in Cornford et al., 2024) via a simple redefinition of terms:

$$\theta_{t+1} = \tilde{g}(\theta_t, t)\colon -\theta_t + \eta\, g(\theta_t, t).$$

In this section, we show that equation 2.6 implies that the updates g can always be written in the form of a positive-definite matrix multiplied by the discrete gradient, which we define below. We will also show that for smooth loss functions $\mathcal{L}$ and sufficiently small η, it is possible to construct at every time t a symmetric positive-definite matrix M (in general, different from the M considered above) such that

$$g(\theta_t) = -M^{-1}\, \nabla_\theta \mathcal{L}(\theta_t).$$

To prove this, we recall Taylor's theorem (Rudin, 1964; Nocedal & Wright, 1999) and the definition of a discrete gradient (Gonzalez, 1996; McLachlan et al., 1999).

Theorem 1 (Taylor's Theorem). *Suppose that $\mathcal{L} : \mathbb{R}^D \to \mathbb{R}$ is a twice continuously differentiable function, and that $p \in \mathbb{R}^D$. Then there exists some $\lambda \in (0, 1)$ such that*

$$\mathcal{L}(x + p) = \mathcal{L}(x) + p^\top \nabla \mathcal{L}(x) + \frac{1}{2} p^\top\, \nabla^2 \mathcal{L}\,(x + \lambda p)\; p. \tag{2.7}$$

It is important to note that equation 2.7 is an equality, and not an approximation (although it can certainly be used to generate an excellent approximation of the difference between $\mathcal{L}(x + p)$ and $\mathcal{L}(x)$ when the norm of p is small and $\mathcal{L}$ is smooth).

Definition 1 (Discrete Gradient). *Suppose that $\mathcal{L} : \mathbb{R}^D \to \mathbb{R}$ is a differentiable function and that $p \in \mathbb{R}^D$. Then $\bar{\nabla}\mathcal{L} : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}^D$ is a discrete gradient of $\mathcal{L}$ if it is continuous and*

$$\begin{cases} p^\top \bar{\nabla}\mathcal{L}(x, x + p) = \mathcal{L}(x + p) - \mathcal{L}(x) \\ \qquad\quad \bar{\nabla}\mathcal{L}(x, x) = \nabla \mathcal{L}(x). \end{cases} \tag{2.8}$$

2.2.1 Discrete-Time Metric. As in the analysis of continuous-time learning rules above, we define the negative discrete gradient as

$$\bar{y} := -\bar{\nabla}\mathcal{L}(\theta_t, \theta_{t+1}). \tag{2.9}$$

Note that equations 2.6 and 2.8 together imply that updates of the parameter vector θ will correlate with $\bar{y}$:

$$\eta\, g^\top \bar{y} = -\left[\mathcal{L}(\theta_{t+1}) - \mathcal{L}(\theta_t)\right] \ > \ 0.$$

Using this observation, we define the discrete analog of the metric 2.1 as

$$\bar{M} := \frac{\bar{y}\bar{y}^\top}{\bar{y}^\top g} + \sum_{i=1}^{D-1} u_i u_i^\top, \tag{2.10}$$

where, as before, the vectors u_i are chosen to span the subspace orthogonal to g, denoted by $g^\perp := \{v \in \mathbb{R}^n : v^\top g = 0\}$. We can see from this definition of $\bar{M}$ that

$$\bar{M}g = \bar{y} \qquad \text{and} \qquad \bar{M} = \bar{M}^\top \succ 0.$$

This implies, via equation 2.6, that the parameter updates can be written in the form of a positive-definite matrix multiplied by the discrete gradient, as claimed:

$$\theta_{t+1} = \theta_t - \eta \bar{M}^{-1}\, \bar{\nabla}\mathcal{L}\left(\theta_t,\ \theta_{t+1}\right). \tag{2.11}$$

Although equation 2.11 bears a resemblance to the natural gradient descent rule, they are not identical. This is because the discrete gradient does not always correspond to the gradient of a specific loss function. In the following section, we explore the conditions under which equation 2.11 can be considered a "true" natural gradient descent. In order to do this, we introduce a new discrete gradient, derived from the Hessian of the loss function.

2.2.2 Small Learning Rate Regime. Motivated by Taylor's theorem, we now introduce the following particular discrete gradient,

$$\bar{\nabla}\mathcal{L}(x, x+p) := \nabla\mathcal{L}(x) + \frac{1}{2}\nabla^2\mathcal{L}(x+\lambda p)\, p, \tag{2.12}$$

where $\lambda \in (0, 1)$ is derived from equation 2.7. It can be easily verified that the discrete gradient conditions (see equation 2.8) hold. Taking $p = \eta\, g(\theta_t)$, we see that for $\bar{y}$ as in equation 2.9,

$$\bar{y} = -\nabla\mathcal{L}(\theta_t) - \frac{\eta}{2}\nabla^2\mathcal{L}(\theta_t + \lambda\eta g)\, g = -\nabla\mathcal{L}(\theta_t) - \eta H g,$$

where

$$H := \frac{1}{2}\nabla^2\mathcal{L}(\theta_t + \lambda\eta g).$$

Note that for this particular choice of discrete gradient, we also have that

$$\bar{y} \to y \quad \text{as} \quad \eta \to 0.$$

Since equation 2.11 can be rewritten as $\theta_{t+1} - \theta_t = \eta \bar{M}^{-1} \bar{y}$, from equation 2.12, we obtain

$$\bar{M}\left(\frac{\theta_{t+1} - \theta_t}{\eta}\right) = \bar{y} = -\nabla \mathcal{L}(\theta_t) - H\big(\theta_{t+1} - \theta_t\big).$$

Adding the Hessian term to both sides, we have

$$\big[\bar{M} + \eta H\big]\left(\frac{\theta_{t+1} - \theta_t}{\eta}\right) = -\nabla \mathcal{L}(\theta_t). \tag{2.13}$$

Equation 2.13 is almost in the desired natural gradient form. In order to put it in exactly natural gradient form, we would like the matrix $\bar{M} + H$ to be positive-definite. We will now show that this can be done by choosing η sufficiently small. In the case where the loss function $\mathcal{L}$ is convex, $\bar{M} + H$ is always positive-definite. We therefore only deal with the case when the loss $\mathcal{L}$ is nonconvex, so that H has a negative minimum eigenvalue. That is, we assume

$$\exists\, h > 0 \quad \text{such that} \quad \lambda_{\min}(H) = -h.$$

Using the results of section 2.1 on picking a metric with an easily calculable minimum eigenvalue, and the fact (Horn & Johnson, 2012) that

$$\lambda_{\min}(\bar{M} + \eta H) \geq \lambda_{\min}(\bar{M}) + \eta \lambda_{\min}(H),$$

we can ensure that $\lambda_{\min}(\bar{M} + \eta H) > 0$ by choosing η to be sufficiently small:

$$\eta < \frac{1}{h} \frac{\|\bar{y}\|}{\|g\|} \cos(\bar{\psi}), \tag{2.14}$$

where $\bar{\psi}$ is the angle between the negative discrete-gradient $\bar{y}$ and the update vector g and is always between $-\pi/2$ and $\pi/2$. If η satisfies this inequality, then we can invert the $\bar{M} + \eta H$ term, yielding

$$\frac{\theta_{t+1} - \theta_t}{\eta} = -\big[\bar{M} + \eta H\big]^{-1} \nabla \mathcal{L}(\theta_t), \tag{2.15}$$

which is precisely a discrete-time natural gradient update rule.

2.2.3 Limit as Learning Rate Goes to Zero ($\eta \to 0$). Using the fact that

$$\bar{M} \to M \quad \text{and} \quad \eta H \to 0 \quad \text{as} \quad \eta \to 0,$$

we obtain that the limit of equation 2.15 as $\eta \to 0$ recovers the natural gradient descent, equation 1.4,

$$\dot{\theta} = -M^{-1}\, \nabla\mathcal{L}(\theta).$$

2.2.4 Stochastic Learning Rules. When the discrete learning rule is stochastic, there is a probability distribution over θ_{t+1} given a known θ_t. In this case, the average update will be given as

$$\eta \left\langle g(\theta_t) \right\rangle = \langle \theta_{t+1} - \theta_t \rangle = \langle \theta_{t+1} \rangle - \theta_t.$$

Effective learning on average for a given loss $\mathcal{L}$, up to the generality of integrating this in time, can be defined as any learning rule that yields

$$\left\langle \mathcal{L}(\theta_{t+1}) - \mathcal{L}(\theta_t) \right\rangle < 0.$$

Similar to the deterministic case, we can define

$$\bar{M} = \frac{\langle \bar{y} \rangle \langle \bar{y} \rangle^\top}{\langle g \rangle^\top \langle \bar{y} \rangle} + \sum_{i=1}^{D-1} u_i u_i^\top,$$

where $\bar{y}$ is the negative discrete gradient defined previously and vectors u_i span the subspace orthogonal to $\langle g \rangle$. This yields

$$\bar{M}\langle g \rangle = -\nabla L(\theta_t) - \eta \langle H g \rangle.$$

In the case where $\langle Hg \rangle^\top \langle g \rangle = 0$, $\bar{M}$ already works as a metric. Otherwise, we want the matrix

$$M\colon -\bar{M} + \eta \frac{\langle Hg \rangle \langle Hg \rangle^\top}{\langle Hg \rangle^\top \langle g \rangle} \tag{2.16}$$

to be positive-definite to allow the average learning rule to be expressed as natural gradient descent,

$$\langle g \rangle = M^{-1} \nabla L(\theta_t). \tag{2.17}$$

Similar to the deterministic case, this will always hold for small enough η. This is because the second term in equation 2.16 can be made arbitrarily small as $\eta \to 0$.

2.2.5 Larger Learning Rates. When η is outside the bounds of equation 2.14, strict natural gradient-descent dynamics may not hold. For a counterexample, consider the one-dimensional case with loss $L(\theta) = -\theta + 2\sin\theta$, and learning rule $g(\theta_t) = \theta_{t+1} - \theta_t = 2\pi$ with $\eta = 1$ and initial condition $\theta_0 = 0$. The updates are guaranteed to reduce the loss, as

$$L(\theta_{t+1}) - L(\theta_t) = -2\pi,$$

and so the learning rule is effective for L. The gradient will be simply the derivative

$$\partial_\theta L(\theta_t) = -1 + 2\cos(2\pi t) = 1 > 0,$$

where the second equality holds since t is discrete. This will give a metric,

$$M = -\frac{\langle g(\theta)\rangle}{\langle \partial_x L\rangle} = -1 < 0,$$

violating the condition for the metric to be positive-definite. Notice that this is outside the bounds of equation 2.14, which would give $H = -\frac{1}{\pi}$, $h = \frac{1}{\pi}$, $\bar{y} = -1$, and $\eta_{\max} = \frac{1}{2}$. This learning rule is still descending in the direction of the discrete gradient with equation 2.11.

3 Numerical Experiments

We provide two numerical experiments supporting the theory just developed. In the first, we show that a stable linear time-invariant (LTI) dynamical system, which in general cannot be written as the gradient of a scalar function, can be written in the natural gradient form. In the second, we show that a popular biologically plausible alternative to propagation, feedback alignment, can also be written as a natural gradient descent.

3.1 Linear Time-Invariant Dynamics. We consider the stable LTI system

$$\dot{\theta} = g(\theta, t) = A\theta, \tag{3.1}$$

where A is an asymmetric matrix with eigenvalues strictly on the left-hand side of the complex plane. Because A is asymmetric, the dynamics, equation 3.1, cannot be written as the gradient of a scalar function (because this

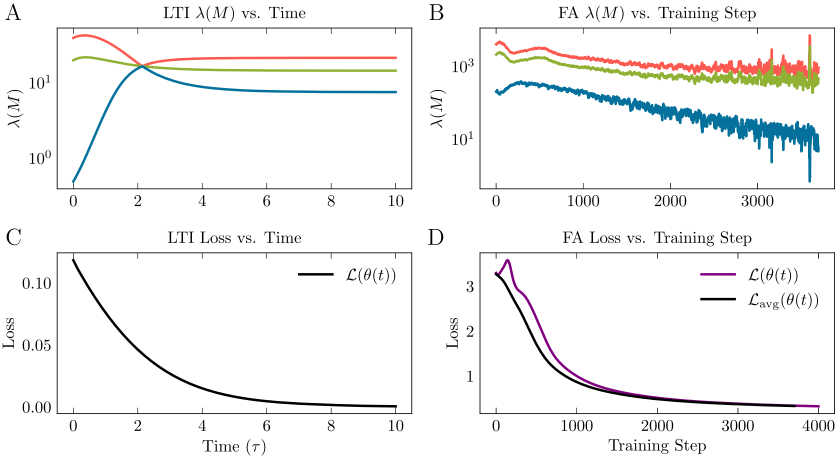

Figure 4: (A) Eigenvalues of M_{opt} for stable linear time-invariant (LTI) dynamics over time. The oscillations arise from the complex eigenvalues of the LTI matrix. (B) Lyapunov function (loss) corresponding to the dynamics in panel A, demonstrating a monotonic decrease. (C) Eigenvalues of M_{opt} for a small multilayer network trained with a biologically plausible learning rule (feedback alignment) to classify MNIST digits. (D) Training loss of feedback alignment as a function of training steps, showing that while the instantaneous loss is not strictly monotonic, the average loss decreases over time.

would imply the Hessian, A, is symmetric). Of course, it is well known that the trajectories of equation 3.1 *do* decrease the Lyapunov function,

$$\mathcal{L}(\theta) = \theta^\top P\theta \qquad \text{where} \qquad PA + A^\top P = -Q,$$

with $Q = Q^\top$, $P = P^\top \succ 0$ (Brockett, 2015). In simulations, we set $Q = I$ and solved for P by using the SciPy (Virtanen et al., 2020) function,

`scipy.linalg.solve_continuous_lyapunov`. In this case, we have that

$$y = -\nabla_\theta \mathcal{L} = -2P\theta,$$

and the metric M can be calculated according to our results, putting the dynamics, equation 3.1, in the natural gradient form,

$$\dot{\theta} = -M^{-1}(\theta)\, \nabla_\theta \mathcal{L}.$$

Figure 4 shows the results.

3.2 Biologically Plausible Learning (Feedback Alignment). Feedback alignment (FA) is a biologically plausible alternative to backpropagation (BP) (Lillicrap et al., 2016) with strong performance on benchmarks and favorable scaling for large networks (Nøkland, 2016; Launay et al., 2020). FA uses a random, fixed backward connectivity structure instead of BP's symmetric weights. We use FA to train a simple two-layer linear network to classify digits from the MNIST data set and, as expected, the network improves performance. Then we computed the eigenvalues of the NGD metric with equation 2.4. The results are shown in Figure 4. Notably, M does not provide a straightforward mapping from BP's individual updates to those of FA when using stochastic gradient descent, as is our case. This discrepancy arises because each update relies on a different data subset to compute its loss and gradient, meaning the updates are not solely functions of the weights, complicating comparisons between steps. For a consistent evaluation of efficient learning, we used the test loss averaged over 300 steps, $\mathcal{L}_{\text{avg}}(\theta)$ in Figure 4D. The metric maps the weight updates to the negative gradient of this loss. Details of our code can be found in GitHub.[4] We note that the condition number of the metric can be used to provide a principled measure of the quality of the feedback matrix, in the sense that a large condition number implies slow convergence of feedback alignment, whereas a small condition number predicts fast convergence (see appendix A).

4 Discussion

We have demonstrated that a broad class of effective learning rules—those that improve a scalar performance measure—can be unified under the framework of natural gradient descent. This result offers a conceptual bridge between diverse learning paradigms, including biologically plausible mechanisms, by showing that they can be understood as natural gradient algorithms (Nayebi et al., 2020; Bredenberg & Savin, 2024; Richards & Kording, 2023). Our findings hold in both continuous and discrete time, providing formal support for the idea that gradient-based updates are fundamental to all learning processes.

4.1 Connections to Related Work. Previous work has shown that if a continuous-time dynamical system admits a strict Lyapunov function, its evolution can be described by a symmetric positive-definite matrix multiplied by the negative gradient of that function (McLachlan et al., 1999; Bárta et al., 2012). In our case, the loss function serves as the Lyapunov function governing the learning dynamics.

[4] https://github.com/kozleo/all_learning_natural_gradient/tree/main.

Our work extends the results of McLachlan et al. (1999) by generalizing their findings to nonmonotonic and time-varying loss functions, as well as discrete-time and stochastic settings. We establish that the class of metrics considered in McLachlan et al. (1999) and Bárta et al. (2012) is canonical—meaning it is the only class of valid metrics. Furthermore, we derive a one-parameter family of metrics whose spectral properties can be computed exactly. Within this family, we identify an optimal metric, in the sense of minimizing the condition number, which proves especially useful in providing convergence results and control, and analyzing discrete-time learning rules.

4.2 Implications for Deep Learning. Recent work has shown that many widely used deep learning optimizers—such as Adam, Shampoo, and Prodigy—correspond to steepest descent under a fixed norm (Bernstein & Newhouse, 2024). Our results generalize this perspective by showing that any effective optimizer can be viewed as steepest descent with respect to a (state-dependent, time-varying) weighted Euclidean norm. This correspondence suggests the intriguing possibility of combining different metrics and norms, potentially revealing that a broader class of optimizers used "in the wild" can also be understood as steepest descent. We will explore this direction in future work.

4.3 Future Directions. One natural conjecture emerging from our work is that any sequence of parameter updates leading to an overall improvement in a loss function—even if not strictly monotonic—can be reformulated as steepest descent under some norm. Many update rules that at first glance seem disconnected from an optimization process can in fact be reformulated as minimizing a loss function. For example, the Hebbian learning rule in Hopfield networks can be viewed as minimizing cosine similarity loss (Tolmachev & Manton, 2020). Another related line of work shows that any asymptotically stable dynamical system can be mapped, via a suitable coordinate transformation, to an exponentially stable system (Grüne et al., 1999). In such cases, the metric may naturally arise from the Jacobian of this transformation—a correspondence that we leave for future investigation.

Additionally, there is a growing body of research on non-Euclidean synaptic plasticity, particularly in the context of mirror descent, which has been linked to synaptic weight distributions (Pogodin et al., 2023; Cornford et al., 2024). Mirror descent generalizes gradient descent by performing updates in a dual space (Nemirovsky & Yudin, 1983) and can be viewed as a special case of natural gradient descent. However, not all natural gradient updates can be expressed in the mirror descent framework (Gunasekar et al., 2021). In appendix B we provide conditions under which natural gradient descent may be viewed as mirror descent. Our results therefore encompass a broader class of learning rules, which could be further constrained in future work to better align with biological and algorithmic constraints.

In summary, our findings provide a unified theoretical foundation for understanding learning rules through the lens of natural gradient descent, with implications spanning neuroscience, control, and machine learning.

Appendix A: Convergence Rate and Metric Conditioning

In this appendix, we give an argument for why choosing the metric with the smallest possible condition number is optimal. Consider a particularly simple (yet nonlinear) flow,

$$\dot{\theta} = -M^{-1}\nabla\mathcal{L},$$

where the loss $\mathcal{L}$ is assumed to be strongly convex and the metric M is assumed for simplicity to be slowly time-varying (effectively constant) along the system trajectories. This situation corresponds to preconditioned gradient descent and is a special case of natural gradient descent.

We are interested in analyzing the convergence of this system to the global equilibrium, obtained when flow stops: $\dot{\theta} = 0$. To do so, we consider the following Lyapunov function, which may be regarded as a generalized kinetic energy for the system

$$V\colon -\frac{1}{2}\dot{\theta}^\top M\dot{\theta}.$$

Since M is positive-definite, convergence of V to zero implies convergence of the update $\dot{\theta}$ to zero, which in turn implies convergence of the optimizer to the global equilibrium. The time derivative of V is

$$\dot{V} = \dot{\theta}^\top M\frac{\partial\dot{\theta}}{\partial\theta}\dot{\theta} = -\dot{\theta}^\top H\dot{\theta},$$

where H denotes the Hessian of the loss $\mathcal{L}$. Since the Hessian is positive-definite, there exists some $\beta > 0$ such that

$$H \succeq \beta M.$$

Plugging in this inequality, we find that the kinetic energy converges exponentially quickly to zero because

$$\dot{V} \leq -2\beta V \qquad \Longrightarrow \qquad V(t) \leq V(0)e^{-2\beta t}.$$

Plugging in the definition of $V(t)$, this can be rewritten as

$$\dot{\theta}(t)^\top M \dot{\theta}(t) \leq \dot{\theta}(0)^\top M \dot{\theta}(0) e^{-2\beta t}.$$

Taking the lower bound on the left side and the upper bound on the right side, it follows from this equation that

$$\lambda_{\min}(M) \left\|\dot{\theta}(t)\right\|^2 \leq \lambda_{\max}(M) \left\|\dot{\theta}(0)\right\|^2 e^{-2\beta t}.$$

Finally, we can rewrite this in terms of the condition number of the metric M,

$$\left\|\dot{\theta}(t)\right\| \leq \sqrt{\frac{\lambda_{\max}}{\lambda_{\min}}} \left\|\dot{\theta}(0)\right\| e^{-\beta t} = \sqrt{\kappa(M)} \left\|\dot{\theta}(0)\right\| e^{-\beta t}.$$

Thus, the "optimal" metric in this setting is the one with the smallest condition number, as it provides the tightest convergence rate for the optimizer.

Appendix B: Relation to Mirror Descent

Another line of work involving non-Euclidean synaptic plasticity has been to connect mirror descent (MD) to synaptic weight distributions (Pogodin et al., 2023; Cornford et al., 2024). MD is a generalization of gradient descent where an invertible vector link function $\nabla\psi$ maps from original weight space to a dual space, where the gradient steps happen. These updates are given by

$$\nabla\psi(\theta_{t+1}) - \nabla\psi(\theta_t) = -\nabla_\theta \mathcal{L}(\theta_t).$$

As Gunasekar et al. (2021) pointed out, an equivalent description of MD is as a partial discretization of the natural gradient descent dynamics,

$$\dot{\theta} = (\nabla^2\psi(\theta))^{-1} \nabla_\theta \mathcal{L},$$

with metric $M(\theta) = \nabla^2\psi(\theta)$. However, not all metrics have a corresponding link function $\nabla\psi$, which requires M to be a Jacobian and satisfy the Poincaré condition,

$$\frac{\partial}{\partial\theta_k} M_{ij} = \frac{\partial}{\partial\theta_j} M_{ik}.$$

To summarize, mirror descent is a subset of natural gradient descent: all MD steps can be written as NGD, but not all NGD can be described by MD. An important reminder is that the steps in MD do not correspond to the steps of discrete natural gradient descent, but of the partial discretization of the continuous flow.

Appendix C: Optimal Metric

C.1 Eigenvalues of One-Parameter Family. For convenience, we begin by setting $\alpha = \gamma \frac{y^\top y}{y^\top g}$, for some $\gamma > 0$.

Lemma 1. *The eigenvalues of*

$$M = \frac{yy^\top}{y^\top g} + \gamma \frac{y^\top y}{y^\top g}\left(I - \frac{gg^\top}{g^\top g}\right) \tag{C.1}$$

are

$$\lambda_{\max/\min}(M) = \frac{\|y\|}{2\|g\| \cos(\psi)}\left((1+\gamma) \pm \sqrt{(1+\gamma)^2 - 4\gamma \cos^2(\psi)}\right),$$

with multiplicity one each, and

$$\frac{\|y\|}{\|g\| \cos(\psi)}\gamma$$

with multiplicity $D - 2$.

Proof. Since

$$M = \frac{\|y\|}{\|g\| \cos(\psi)}\left(\hat{y}\hat{y}^\top + \gamma(I - \hat{g}\hat{g}^\top)\right),$$

it suffices to compute the eigenvalues of

$$A_0 := \hat{y}\hat{y}^\top - \gamma \hat{g}\hat{g}^\top.$$

A_0 acts on a vector $v = \hat{y} + \zeta \hat{g}$ as

$$\begin{aligned} A_0 v &= \big(\hat{y} - \cos(\psi)\gamma \hat{g}\big) + \zeta\big(\cos(\psi)\hat{y} - \gamma \hat{g}\big) \\ &= \big(1 + \zeta \cos(\psi)\big)\hat{y} - \big(\cos(\psi)\gamma + \zeta\gamma\big)\hat{g}, \end{aligned}$$

which is a multiple of v exactly when

$$\zeta\big(1 + \zeta \cos(\psi)\big) = -\big(\cos(\psi)\gamma + \zeta\gamma\big).$$

This is equivalent to $\cos(\psi)\zeta^2 + (1+\gamma)\zeta + \gamma \cos(\psi) = 0$, that is,

$$\zeta = \frac{1}{2\cos(\psi)}\left(- (1+\gamma) \pm \sqrt{(1+\gamma)^2 - 4\gamma\cos^2(\psi)}\right).$$

Thus, the corresponding eigenvalues of A_0 are

$$\lambda_{\max/\min}(A_0) = 1 + \zeta\cos(\psi) = \frac{1}{2}\left(1 - \gamma \pm \sqrt{(1+\gamma)^2 - 4\gamma\cos^2(\psi)}\right).$$

They correspond to the eigenvalues

$$\lambda_{\max/\min}(M) = \frac{\|y\|}{2\|g\|\cos(\psi)}\left(1 + \gamma \pm \sqrt{(1+\gamma)^2 - 4\gamma\cos^2(\psi)}\right).$$

□

C.2 Optimal Condition Number for One-Parameter Family. The condition number for M as in equation C.1 is

$$\begin{aligned}\kappa(M) &= \frac{1+\gamma+\sqrt{(1+\gamma)^2 - 4\gamma\cos^2(\psi)}}{1+\gamma-\sqrt{(1+\gamma)^2 - 4\gamma\cos^2(\psi)}}\\ &= \frac{\left(1+\gamma+\sqrt{(1+\gamma)^2 - 4\gamma\cos^2(\psi)}\right)^2}{4\gamma\cos^2(\psi)},\end{aligned}$$

so

$$\begin{aligned}\sqrt{\kappa(M)} &= \frac{1+\gamma+\sqrt{(1+\gamma)^2 - 4\gamma\cos^2(\psi)}}{2\sqrt{\gamma}\cos(\psi)}\\ &= \frac{\gamma^{1/2}+\gamma^{-1/2}+\sqrt{(\gamma^{1/2}+\gamma^{-1/2})^2 - 4\cos^2(\psi)}}{2\cos(\psi)}.\end{aligned}$$

Now let $\cos(\psi)z = \gamma^{1/2} + \gamma^{-1/2}$, which is some variable $z \geq 2/\cos(\psi)$. We are trying to minimize

$$\sqrt{\kappa(M)} = \frac{z + \sqrt{z^2 - 4}}{2}.$$

This is a monotonically increasing function of z so is minimized at $z = 2/\cos(\psi)$. This corresponds to $\gamma = 1$.

C.3 Optimal Condition Number for All Metrics. Let us prove a lemma:

Lemma 2. *Suppose v_0, v_1 are vectors. The following are equivalent:*

1. *for any B a symmetric matrix, $v_0^\top B v_0 = v_1^\top B v_1$; and*
2. *$v_0 = \pm v_1$.*

Proof. One direction is obvious. For the nonobvious direction, let $B = (b_{ij})_{i,j=1}^{D}$ where $b_{ij} = b_{ji}$. Let $v_0 = (x_1, \ldots, x_D)^\top$ and $v_1 = (y_1, \ldots, y_D)^\top$. Then we have the equation

$$\sum_{i,j=1}^{D} b_{ij} x_i x_j = \sum_{i,j=1}^{D} b_{ij} y_i y_j.$$

Thus, we conclude that $x_i x_j = y_i y_j$ for any two indices i and j. When $i = j$, this implies $x_i = \pm y_i$, but these signs must all be the same using all the other equations. □

As a consequence, we can prove the following variant:

Lemma 3. *Suppose v_0 and v_1 are vectors, and g is a nonzero vector. The following are equivalent:*

1. *for any symmetric matrix B such that $Bg = 0$, the equality $v_0^\top B v_0 = v_1^\top B v_1$ holds; and*
2. *there exists an $\alpha \in \mathbb{R}$ such that $v_0 = \pm v_1 + \alpha g$.*

Proof. Again, one direction is obvious. For the nonobvious direction, we consider the projection of v_0 and v_1 to $g^\perp$ along g:

$$v_0' = v_0 - \frac{g^\top v_0}{g^\top g} g, \qquad v_1' = v_1 - \frac{g^\top v_1}{g^\top g} g.$$

Then we see that $(v_0')^\top B v_0' = (v_1')^\top B v_1'$ for all symmetric matrices B on $g^\perp$. Now using lemma 2, we see that $v_0' = \pm v_1'$. □

Proposition 1. *Let M be a positive-definite matrix such that $Mg = y$ where $y^\top g > 0$. Let ψ be the angle between g and y. Then the minimum value for $\kappa(M)$, achieved by equation C.1 when $\gamma = 1$, is*

$$\frac{1 + |\sin(\psi)|}{1 - |\sin(\psi)|}.$$

Proof. We know that $M = \frac{1}{y^\top g} y y^\top + M'$ for some symmetric positive-definite matrix M' on $g^\perp$. Let M' be a matrix attaining the minimum $\kappa(M)$. Consider the perturbation of M' by some symmetric matrix B such that $Bg = 0$. Then perturbation theory tells us

$$\lambda_{\max/\min}(M + \epsilon B) = \lambda_{\max/\min}(M) + v_{\max/\min}^\top B v_{\max/\min} \epsilon + O(\epsilon^2),$$

where $v_{\max/\min}$ are eigenvectors of M with eigenvalue $\lambda_{\max/\min}$ normalized to have norm 1. Thus

$$\frac{\lambda_{\max}(M+\epsilon B)}{\lambda_{\min}(M+\epsilon B)} = \frac{\lambda_{\max}(M) + v_{\max}^{\top} B v_{\max}\epsilon}{\lambda_{\min}(M) + v_{\min}^{\top} B v_{\min}\epsilon} + O(\epsilon^2).$$

Since M' is in particular a local minimum,

$$\lambda_{\max}(M) \cdot v_{\min}^{\top} B v_{\min} = \lambda_{\min}(M) \cdot v_{\max}^{\top} B v_{\max}.$$

By lemma 3, this implies that $v_{\max}$, $v_{\min}$, and g are linearly dependent. Hence by applying M, we see that so are $v_{\max}$, $v_{\min}$, and y. So we can restrict ourself to the two-dimensional subspace spanned by $v_{\max}$ and $v_{\min}$. But then the same argument as before shows optimality. □

C.4 Optimal Maximum and Minimum Metric Eigenvalues. We show that a lower bound (resp., upper bound) for $\lambda_{\min}(M)$ (resp., $\lambda_{\max}(M)$) for matrices M satisfying the conditions of proposition 1 and show that they are never achieved but are asymptotically achieved.

Proposition 2. *Let M be a symmetric positive-definite matrix such that $Mg = y$ where $y^{\top} g > 0$. Let ψ be the angle between g and y. Then $\lambda_{\min}(M) < \frac{\|y\|}{\|g\|}\cos(\psi)$. Moreover, the supremum is asymptotically approached by equation C.1, as $\gamma \to \infty$.*

Proof. Recall that

$$\lambda_{\min}(M) = \min_{v \neq 0} \frac{v^{\top} M v}{v^{\top} v}, \tag{C.2}$$

and, moreover, the minimum is reached by eigenvectors with eigenvalue $\lambda_{\min}$. Thus,

$$\lambda_{\min}(M) \leq \frac{g^{\top} M g}{g^{\top} g} = \frac{g^{\top} y}{g^{\top} g} = \frac{\|y\|}{\|g\|}\cos(\psi).$$

Moreover, equality is not reached since g is not an eigenvector of M. Finally, the limit of the minimum eigenvalue of equation C.1 as $\gamma \to \infty$ is, by lemma 1,

$$\lim_{\gamma\to\infty} \frac{\|y\|}{2\|g\|\cos(\psi)}\left((1+\gamma) - \sqrt{(1+\gamma)^2 - 4\gamma\cos^2(\psi)}\right) = \frac{\|y\|}{\|g\|}\cos(\psi).$$

□

Proposition 3. *Let M be a symmetric positive-definite matrix such that $Mg = y$ where $y^{\top} g > 0$. Let ψ be the angle between g and y. Then $\lambda_{\max}(M) > \frac{\|y\|}{\|g\|\cos(\psi)}$. Moreover, the infimum is asymptotically approached by equation C.1, as $\gamma \to 0$.*

Proof. Recall that (e.g., by using equation C.2 and observing that $\lambda_{\max}(M) = \lambda_{\min}(M^{-1})^{-1}$),

$$\lambda_{\max}(M) = \max_{v \neq 0} \frac{v^\top v}{v^\top M^{-1} v},$$

and, moreover, the minimum is reached by the eigenvectors with eigenvalue $\lambda_{\max}$. Thus,

$$\lambda_{\max}(M) \leq \frac{y^\top y}{y^\top M^{-1} y} = \frac{y^\top y}{y^\top g} = \frac{\|y\|}{\|g\| \cos(\psi)}.$$

Moreover, equality is not reached since y is not an eigenvector of M. Finally, the limit of the maximum eigenvalue of equation C.1, as $\gamma \to 0$ is, by lemma 1,

$$\lim_{\gamma \to 0} \frac{\|y\|}{2\|g\| \cos(\psi)} \left((1+\gamma) + \sqrt{(1+\gamma)^2 - 4\gamma \cos^2(\psi)} \right) = \frac{\|y\|}{\|g\| \cos(\psi)}.$$

$\square$

Acknowledgments

L.K. acknowledges Jean-Jacques Slotine (MIT) for helpful conversations.

References

Abbott, L. F., & Regehr, W. G. (2004). Synaptic computation. *Nature*, *431*(7010), 796–803. 10.1038/nature03010

Ahmadi, A. A, & Parrilo, P. A. (2008). Non-monotonic Lyapunov functions for stability of discrete time nonlinear and switched systems. In *Proceedings of the 47th IEEE Conference on Decision and Control* (pp. 614–621).

Amari, S.-I. (1998). Natural gradient works efficiently in learning. *Neural Computation*, *10*(2), 251–276. 10.1162/089976698300017746

Bárta, T., Chill, R., & Fašangová, E. (2012). Every ordinary differential equation with a strict Lyapunov function is a gradient system. *Monatshefte für Mathematik*, *166*(1), 57–72.

Bernstein, J., & Newhouse, L. (2024). *Old optimizer, new norm: An anthology*. arXiv:2409.20325.

Boffi, N. M., & Slotine, J.-J. E. (2021). Implicit regularization and momentum algorithms in nonlinearly parameterized adaptive control and prediction. *Neural Computation*, *33*(3), 590–673. 10.1162/neco_a_01360

Bredenberg, C., & Savin, C. (2024). Desiderata for normative models of synaptic plasticity. *Neural Computation*, *36*(7), 1245–1285. 10.1162/neco_a_01671

Brockett, R. W. (2015). *Finite dimensional linear systems*. SIAM.

Cauchy, A. (1847). Méthode générale pour la résolution des systemes d'équations simultanées. *Comptes Rendus Sciences Paris*, *25*(1847), 536–538.

Clarke, F. H. (1975). Generalized gradients and applications. *Transactions of the American Mathematical Society*, *205*, 247–262. 10.1090/S0002-9947-1975-0367131-6

Cornford, J., Pogodin, R., Ghosh, A., Sheng, K., . . . Richards, B. A. (2024). *Brain-like learning with exponentiated gradients*. bioRxiv. 10.1101/2024.10.25.620272

Dangel, F., Müller, J., & Zeinhofer, M. (2024). *Kronecker-factored approximate curvature for physics-informed neural networks*. arXiv:2405.15603.

Dayan, P., & Abbott, L. F. (2005). *Theoretical neuroscience: Computational and mathematical modeling of neural systems*. MIT Press.

Fiete, I. R., & Seung, H. S. (2006). Gradient learning in spiking neural networks by dynamic perturbation of conductances. *Physical Review Letters*, *97*(4), 048104. 10.1103/PhysRevLett.97.048104

Francioni, V., Tang, V. D., Toloza, E. H., Brown, N. J., & Harnett, M. T. (2023). *Vectorized instructive signals in cortical dendrites during a brain-computer interface task*. bioRxiv, 2023–11.

Gonzalez, O. (1996). Time integration and discrete Hamiltonian systems. *Journal of Nonlinear Science*, *6*, 449–467. 10.1007/BF02440162

Grossberg, S. (1987). Competitive learning: From interactive activation to adaptive resonance. *Cognitive Science*, *11*(1), 23–63. 10.1111/j.1551-6708.1987.tb00862.x

Grüne, L., Sontag, E. D., & Wirth, F. R. (1999). Asymptotic stability equals exponential stability, and ISS equals finite energy gain—if you twist your eyes. *Systems and Control Letters*, *38*(2), 127–134.

Gunasekar, S., Woodworth, B., & Srebro, N. (2021). Mirrorless mirror descent: A natural derivation of mirror descent. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, *130* (pp. 2305–2313).

Hazan, E. (2016). Introduction to online convex optimization. *Foundations and Trends in Optimization*, *2*(3–4), 157–325. 10.1561/2400000013

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. In *Proceedings of the National Academy of Sciences of the United States of America*, *79*(8), 2554–2558. 10.1073/pnas.79.8.2554

Horn, R. A., & Johnson, C. R. (2012). *Matrix analysis*. Cambridge University Press.

Kakade, S. M. (2001). A natural policy gradient. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems*, *14*. MIT Press.

Launay, J., Poli, I., Boniface, F., & Krzakala, F. (2020). Direct feedback alignment scales to modern deep learning tasks and architectures. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems*, *33* (9346). Curran.

Lee, T., Kwon, J., & Park, F. C. (2018). A natural adaptive control law for robot manipulators. In *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 1–9).

Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016). Random synaptic feedback weights support error backpropagation for deep learning. *Nature Communications*, *7*(1), 13276. 10.1038/ncomms13276

Lim, S., McKee, J. L., Woloszyn, L., Amit, Y., Freedman, D. J., Sheinberg, D. L., & Brunel, N. (2015). Inferring learning rules from distributions of firing rates in cortical neurons. *Nature Neuroscience*, *18*(12), 1804–1810. 10.1038/nn.4158

Martens, J. (2020). New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, *21*(146), 1–76.

Martens, J., & Grosse, R. (2015). Optimizing neural networks with a Kronecker-factored approximate curvature. In *Proceedings of the International Conference on Machine Learning* (pp. 2408–2417).

McLachlan, R. I., Quispel, G. R. W., & Robidoux, N. (1999). Geometric integration using discrete gradients. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, *357*(1754), 1021–1045. 10.1098/rsta.1999.0363

Muehlebach, M., & Jordan, M. (2019). A dynamical systems perspective on Nesterov acceleration. In *Proceedings of the International Conference on Machine Learning* (pp. 4656–4662).

Nayebi, A., Srivastava, S., Ganguli, S., & Yamins, D. L. (2020). Identifying learning rules from neural network observables. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems*, *33* (pp. 2639–2650).

Nemirovsky, A. S., & Yudin, D. B. (1983). *Problem complexity and method efficiency in optimization*. Wiley-Interscience.

Nocedal, J., & Wright, S. J. (1999). *Numerical optimization*. Springer.

Nøkland, A. (2016). Direct feedback alignment provides learning in deep neural networks. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, *29*. Curran.

Ollivier, Y., Arnold, L., Auger, A., & Hansen, N. (2017). Information-geometric optimization algorithms: A unifying picture via invariance principles. *Journal of Machine Learning Research*, *18*(18), 1–65.

Pascanu, R. (2013). *Revisiting natural gradient for deep networks*. arXiv:1301.3584.

Pogodin, R., Cornford, J., Ghosh, A., Gidel, G., Lajoie, G., & Richards, B. (2023). *Synaptic weight distributions depend on the geometry of plasticity*. arXiv:2305.19394.

Richards, B. A., & Kording, K. P. (2023). The study of plasticity has always been about gradients. *Journal of Physiology*, *601*(15), 3141–3149. 10.1113/JP282747

Rudin, W. (1964). *Principles of mathematical analysis*, vol. 3. McGraw-Hill.

Surace, S. C., Pfister, J.-P., Gerstner, W., & Brea, J. (2018). On the choice of metric in gradient-based theories of brain function. *PLOS*, *16*(4), e1007640.

Tolmachev, P., & Manton, J. H. (2020). *New insights on learning rules for Hopfield networks: Memory and objective function minimisation*. arXiv:2010.01472.

Tzen, B., Raj, A., Raginsky, M., & Bach, F. (2023). Variational principles for mirror descent and mirror Langevin dynamics. *IEEE Control Systems Letters*, *PP*(99).

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., . . . SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, *17*, 261–272. 10.1038/s41592-019-0686-2

Wensing, P. M., & Slotine, J.-J. E. (2020). *Beyond convexity: Contraction and global convergence of gradient descent*. arXiv:1806.06655.

Widrow, B., & Lehr, M. A. (1990). 30 years of adaptive neural networks: Perceptron, Madaline, and backpropagation. In *Proceedings of the IEEE*, *78*(9), 1415–1442. 10.1109/5.58323

Received May 22, 2025; accepted September 13, 2025.