NATURAL-GRADIENT LEARNING FOR SPIKING NEURONS

Elena Kreutzer
Department of Physiology
University of Bern
kreutzer@pyl.unibe.ch

Walter M. Senn[*]
Department of Physiology
University of Bern
walter.senn@unibe.ch

Mihai A. Petrovici[*]
Department of Physiology
University of Bern
Kirchhoff-Institute for Physics
Heidelberg University
mihai.petrovici@unibe.ch

November 26, 2020

ABSTRACT

In many normative theories of synaptic plasticity, weight updates implicitly depend on the chosen parametrization of the weights. This problem relates, for example, to neuronal morphology: synapses which are functionally equivalent in terms of their impact on somatic firing can differ substantially in spine size due to their different positions along the dendritic tree. Classical theories based on Euclidean gradient descent can easily lead to inconsistencies due to such parametrization dependence. The issues are solved in the framework of Riemannian geometry, in which we propose that plasticity instead follows natural gradient descent. Under this hypothesis, we derive a synaptic learning rule for spiking neurons that couples functional efficiency with the explanation of several well-documented biological phenomena such as dendritic democracy, multiplicative scaling and heterosynaptic plasticity. We therefore suggest that in its search for functional synaptic plasticity, evolution might have come up with its own version of natural gradient descent.

1 Introduction

Understanding the fundamental computational principles underlying synaptic plasticity represents a long-standing goal in neuroscience. To this end, a multitude of top-down computational paradigms have been developed, which derive plasticity rules as gradient descent on a particular objective function of the studied neural network (Rosenblatt, 1958; Rumelhart et al., 1986; Pfister et al., 2006; D'Souza et al., 2010; Friedrich et al., 2011).

However, the exact physical quantity to which these synaptic weights correspond often remains unspecified. What is frequently simply referred to as w_{ij} (the synaptic weight from neuron j to neuron i) might relate to different components of synaptic interaction, such as calcium concentration in the presynaptic axon terminal, neurotransmitter concentration in the synaptic cleft, receptor activation in the postsynaptic dendrite or the postsynaptic potential (PSP) amplitude in the spine, the dendritic shaft or at the soma of the postsynaptic cell. All of these biological processes can be linked by transformation rules, but depending on which of them represents the variable with respect to which performance is optimized, the network behavior during training can be markedly different.

As an example we consider the parametrization of the synaptic strength either as PSP amplitude in the soma, w^{s}, or as PSP amplitude in the dendrite, w^{d} (see also Fig. 1 and Section 2.1). Reparametrizing the synaptic strength in this way implies an attenuation factor for each single synapse, but different factors are assigned across the positions on the dendritic tree. As a consequence, the weight vector will follow a different trajectory during learning depending on whether the somatic or dendritic parametrization of the PSP amplitude was chosen.

It certainly could be the case that evolution has favored one particular parametrization over all others during its gradual tuning of synaptic plasticity, but this would necessarily imply sub-optimal convergence for all but a narrow set of neuron morphologies and connectome configurations. An invariant learning rule on the other hand would not only be mathematically unambiguous and therefore more elegant, but could also improve learning, thus increasing fitness.

[*]Joint senior authorship.

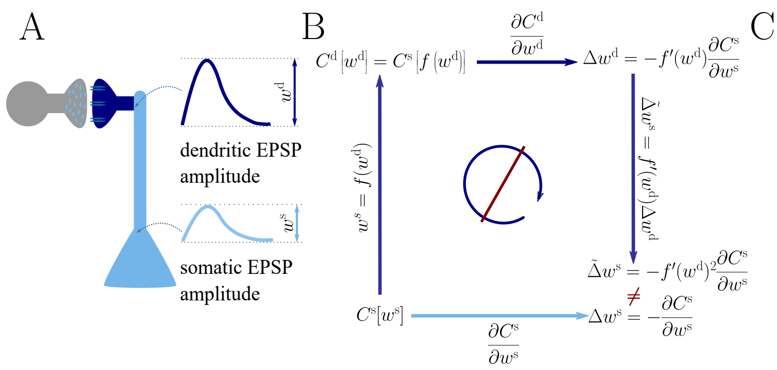
A
dendritic EPSP amplitude
w^{d}
somatic EPSP amplitude
w^{s}
B
$C^{\mathrm{d}}\left[w^{\mathrm{d}}\right] = C^{\mathrm{s}}\left[f\left(w^{\mathrm{d}}\right)\right]$
$\frac{\partial C^{\mathrm{d}}}{\partial w^{\mathrm{d}}}$
$\Delta w^{\mathrm{d}} = -f'(w^{\mathrm{d}})\frac{\partial C^{\mathrm{s}}}{\partial w^{\mathrm{s}}}$
$w^{\mathrm{s}} = f(w^{\mathrm{d}})$
$\tilde{\Delta} w^{\mathrm{s}} = f'(w^{\mathrm{d}})\Delta w^{\mathrm{d}}$
$\tilde{\Delta} w^{\mathrm{s}} = -f'(w^{\mathrm{d}})^2\frac{\partial C^{\mathrm{s}}}{\partial w^{\mathrm{s}}}$
$\neq$
$C^{\mathrm{s}}[w^{\mathrm{s}}]$
$\Delta w^{\mathrm{s}} = -\frac{\partial C^{\mathrm{s}}}{\partial w^{\mathrm{s}}}$
$\frac{\partial C^{\mathrm{s}}}{\partial w^{\mathrm{s}}}$
C
dendritic amplitude
learning
dendritic weight change
reparametrization
w^{d}
Δw^{d}
evokes som. weight change
reparametrization
mismatch
$\tilde{\Delta} w^{\mathrm{s}}$
Δw^{s}
w^{s}
somatic amplitude
learning
somatic weight change

Figure 1: **Classical gradient descent depends on chosen parametrization. (A)** The strength of a synapse can be parametrized in various ways, e.g., as the EPSP amplitude at either the soma w^{s} or the dendrite w^{d}. Biological processes such as attenuation govern the relationship between these variables. Depending on the chosen parametrization, Euclidean gradient descent can yields different results. **(B)** Mathematical derivation. **(C)** Phenomenological correlates. EPSPs before learning are represented as continuous, after learning as dashed curves. The light blue arrow represents gradient descent on the error as a function of the somatic EPSP $C^{\mathrm{s}}\left[w^{\mathrm{s}}\right]$ (also shown in light blue). The resulting weight change leads to an increase Δw^{s} in the somatic EPSP after learning. The dark blue arrows track the calculation of the same gradient, but with respect to the dendritic EPSP (also shown in dark blue): 1) taking the attenuation into account in order to compute the error as a function of w^{d}, 2) calculating the gradient, followed by 3) deriving the associated change in $\tilde{\Delta} w^{\mathrm{s}}$, again considering attenuation. Due to the attenuation $f(w)$ entering the calculation twice, the synaptic weights updates, as well as the associated evolution of a neuron's output statistics over time, will differ under the two parametrizations.

In some aspects, the question of invariant behavior is related to the principle of relativity in physics, which requires the laws of physics – in our case: the improvement of performance during learning – to be the same in all frames of reference. What if neurons would seek to conserve the way they adapt their behavior regardless of, e.g., the specific positioning of synapses along their dendritic tree? Which equations of motion – in our case: synaptic learning rules – are able to fulfill this requirement?

The solution lies in following the path of steepest descent not in relation to a small change in the synaptic weights (Euclidean gradient descent), but rather with respect to a small change in the input-output distribution (natural gradient descent). This requires taking the gradient of the error function with respect to a metric defined directly on the space of possible input-output distributions, with coordinates defined by the synaptic weights. First proposed in Amari (1998), but with earlier roots in information geometry (Amari, 1987; Amari and Nagaoka, 2000), natural gradient methods (Yang and Amari, 1998; Rattray and Saad, 1999; Park et al., 2000; Kakade, 2001) have recently been rediscovered in the context of deep learning (Pascanu and Bengio, 2013; Martens, 2014; Ollivier, 2015; Amari et al., 2019; Bernacchia et al., 2018). Moreover, Pascanu and Bengio (2013) showed that the natural gradient learning rule is closely related to other machine learning algorithms. However, most of the applications focus on rate-based networks which are not inherently linked to a statistical manifold and have to be equipped with Gaussian noise or a probabilistic output layer interpretation in order to allow an application of the natural gradient. Furthermore, a biologically plausible synaptic plasticity rule needs to make all of the required information accessible at the synapse itself, which is usually unnecessary and therefore largely ignored in machine learning.

The stochastic nature of neuronal outputs in-vivo (see, e.g. Softky and Koch, 1993) provides a natural setting for plasticity rules based on information geometry. As a model for biological synapses, natural gradient combines the elegance of invariance with the success of gradient-descent-based learning rules. In this manuscript, we derive a closed-form synaptic learning rule based on natural gradient descent for spiking neurons and explore its implications. Our learning rule equips the synapses with more functionality compared to classical error learning by enabling them to adjust their learning rate to their respective impact on the neuron's output. It naturally takes into account relevant variables such as the statistics of the afferent input or their respective positions on the dendritic tree. This allows a set of predictions which are corroborated by both experimentally observed phenomena such as dendritic democracy and multiplicative weight dynamics and theoretically desirable properties such as Bayesian reasoning (Marceau-Caron and Ollivier, 2017). Furthermore, and unlike classical error-learning rules, plasticity based on the natural gradient is able to incorporate both homo- and heterosynaptic phenomena into a unified framework. While theoretically derived heterosynaptic components of learning rules are notoriously difficult for synapses to implement due to their non-locality, we show that in our learning rule they can be approximated by quantities accessible at the locus of plasticity. In line with results from machine learning, the combination of these features also enables faster convergence during supervised learning.

2 Results

2.1 Naive Euclidean gradient is not parametrization-invariant

We consider a cost function C on the neuronal level that, in the sense of cortical credit assignment (see e.g. Sacramento et al., 2017), can relate to some behavioral cost of the agent that it serves. The output of the neuron depends on the amplitudes of the somatic PSPs elicited by the presynaptic spikes. We denote these "somatic weights" by $\boldsymbol{w}^{\mathrm{s}}$, and may parametrize the neuronal cost as $C = C^{\mathrm{s}}[\boldsymbol{w}^{\mathrm{s}}]$. However, dendritic PSP amplitudes $\boldsymbol{w}^{\mathrm{d}}$ can be argued to offer a more unmitigated representation of synaptic weights, so we might rather wish to express the cost as $C = C^{\mathrm{d}}[\boldsymbol{w}^{\mathrm{d}}]$. These two parametrizations are related by an attenuation factor $\boldsymbol{\alpha}$ (between 0 and 1): $\boldsymbol{w}^{\mathrm{s}} = \boldsymbol{\alpha}\boldsymbol{w}^{\mathrm{d}}$. In general, this attenuation factor depends on the synaptic position and is therefore described by a vector that is multiplied component-wise with the weights.

It may now seem straightforward to switch between the somatic and dendritic representation of the cost by simply substituting variables, for example $C^{\mathrm{d}}[\boldsymbol{w}^{\mathrm{d}}] = C^{\mathrm{s}}[\boldsymbol{w}^{\mathrm{s}}] = C^{\mathrm{s}}[\boldsymbol{\alpha}\boldsymbol{w}^{\mathrm{d}}]$. To derive a plasticity rule for the somatic and dendritic weights we might consider gradient descent on the cost:

$$\Delta\boldsymbol{w}^{\mathrm{d}} = -\frac{\partial C^{\mathrm{d}}}{\partial \boldsymbol{w}^{\mathrm{d}}} = -\frac{\partial \boldsymbol{w}^{\mathrm{s}}}{\partial \boldsymbol{w}^{\mathrm{d}}}\frac{\partial C^{\mathrm{s}}}{\partial \boldsymbol{w}^{\mathrm{s}}} = -\boldsymbol{\alpha}\frac{\partial C^{\mathrm{s}}}{\partial \boldsymbol{w}^{\mathrm{s}}} = \boldsymbol{\alpha}\Delta\boldsymbol{w}^{s} \ . \tag{1}$$

At first glance, this relation seems reasonable: dendritic weight changes affect the cost more weakly then somatic weight changes, so their respective gradient is more shallow by the factor $\boldsymbol{\alpha}$. However, from a functional perspective, the opposite should be true: dendritic weights should experience a larger change than somatic weights in order to elicit the same effect on the cost. This inconsistency can be made explicit by considering that somatic weight changes are, themselves, attenuated dendritic weight changes: $\Delta\boldsymbol{w}^{\mathrm{s}} = \boldsymbol{\alpha}\Delta\boldsymbol{w}^{\mathrm{d}}$. Substituting this into Eqn. 1 leads to a contradiction: $\Delta\boldsymbol{w}^{\mathrm{d}} = \boldsymbol{\alpha}^2\Delta\boldsymbol{w}^{\mathrm{d}}$. This reasoning is visualized in Fig. 1 for the general case where the somatic and dendritic weights are related by some arbitrary function f. To solve the conundrum we need to shift the focus from changing the synaptic input to changing the neuronal output, while at the same time considering a more rigorous treatment of gradient descent (see also Surace et al., 2020).

2.2 Natural gradient plasticity rule

We consider a neuron with somatic potential V evoked by the spikes x_i of n presynaptic afferents firing at rates r_i. The presynaptic spikes of afferent i cause a train of weighted dendritic potentials $w^{\mathrm{d}}\, x_i^{\epsilon}$ locally at the synaptic site. The x_i^{ϵ} denotes the unweighted synaptic potential (USP) train elicited by the low-pass-filtered spike train x_i. At the soma, each dendritic potential is attenuated by a potentially nonlinear function that depends on the synaptic location: $w_i^{\mathrm{s}} = f_i(w_i^{\mathrm{d}})$. The somatic voltage thus reads as $V = \sum_i^n w_i^{\mathrm{s}}\, x_i^{\epsilon} = \sum_i^n f_i(w_i^{\mathrm{d}})\, x_i^{\epsilon}$.

We further assume that the neuron's firing follows an inhomogeneous Poisson process whose rate $\phi_t(V) := \phi(V_t)$ depends on the current membrane potential through a nonlinear transfer function ϕ. In this case, spiking in a sufficiently short interval $[t, t + \mathrm{d}t]$ is Bernoulli-distributed. The probability of a spike occurring in this interval (denoted as $y_t = 1$) is then given by

$$p_{\boldsymbol{w}}\left(y_t = 1 | \boldsymbol{x}_t^{\epsilon}\right) = \phi_t(V)\ \mathrm{d}t\ , \tag{2}$$

which defines our generalized linear neuron model (Gerstner and Kistler, 2002). Here, we used $\boldsymbol{x}^{\epsilon}$ as a shorthand notation for the USP vector (x_i^{ϵ}). In the following, we drop the time indices for better readability.

Assuming that the neuron strives to reproduce a target firing distribution $p^*\left(y|\boldsymbol{x}^{\epsilon}\right)$, plasticity may follow a supervised-learning paradigm based on gradient descent. In this case, the Kullback-Leibler divergence between the neuron's current and its target firing distribution

$$C\left[p_{\boldsymbol{w}}\right] = D_{\mathrm{KL}}(p^* \| p_{\boldsymbol{w}}) = \mathbb{E}\left[\log\left(\frac{p^*}{p_{\boldsymbol{w}}}\right)\right]_{p^*} \tag{3}$$

represents a natural cost function which measures the error between the current and the desired output distribution in an information-theoretic sense. Minimizing this cost function by naive Euclidean gradient descent with respect to the synaptic weights (denoted by $\nabla_{\boldsymbol{w}}^{e}$) results in the well-known error-correcting rule (Pfister et al., 2006)

$$\dot{\boldsymbol{w}} = -\eta\nabla_{\boldsymbol{w}}^{e}C = -\eta\left[Y^* - \phi(V)\right]\frac{\phi'(V)}{\phi(V)}\boldsymbol{x}^{\epsilon}, \tag{4}$$

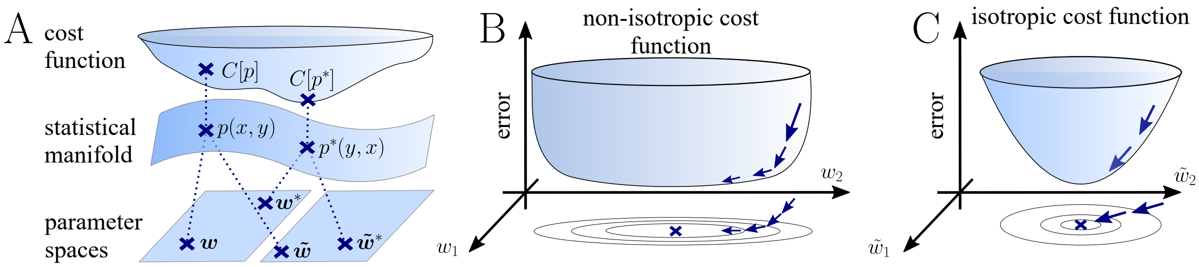

Figure 2: **Natural gradient represents the true gradient direction on the manifold of neuronal input-output distributions.** **(A)** During supervised learning, the error between the current and the target state is measured in terms of a cost function defined on the neuron's output space; in our case, this is the manifold formed by the neuronal output distributions $p(y,x)$. As the output of a neuron is determined by the strength of incoming synapses, the cost C is an implicit function of the afferent weight vector $\boldsymbol{w}$. Since the gradient of a function depends on the distance measure of the underlying space, Euclidean gradient descent, which follows the gradient of the cost as a function of the synaptic weights $\partial C/\partial \boldsymbol{w}$, is not uniquely defined, but depends on how $\boldsymbol{w}$ is parametrized. If, instead, we follow the gradient on the output manifold itself, it becomes independent of the underlying parametrization. Expressed in a specific parametrization, the resulting natural gradient contains a correction term that accounts for the distance distortion between the synaptic parameter space and the output manifold. **(B-C)** Standard gradient descent learning is suited for isotropic (B), rather than for non-isotropic (C) cost functions. For example, the magnitude of the gradient decreases in valley regions where the cost function is flat, resulting in slow convergence to the target. A non-optimal choice of parametrization can introduce such artefacts and therefore harm the performance of learning rules based on Euclidean gradient descent. In contrast, natural gradient learning will locally correct for distortions arising from non-optimal parametrizations.

which is a spike-based version of the classical perceptron learning rule (Rosenblatt, 1958), whose multilayer version forms the basis of the error-backpropagation algorithm (Rumelhart et al., 1986). Here, Y^* denotes a teacher spike train sampled from p^*. On the single-neuron level, a possible biological implementation has been suggested by Urbanczik and Senn (2014), who demonstrated how a neuron may exploit its morphology to store errors, an idea that was recently extended to multilayer networks (Sacramento et al., 2017).

However, as we argued above, learning based on Euclidean gradient descent is not unproblematic. It cannot account for synaptic weight (re)parametrization, as caused, for example, by the diversity of synaptic loci on the dendritic tree. Convergence of learning is therefore harmed by the slow adaptation of distal synapses compared to equally important proximal counterparts. With the multiplicative USP term $\boldsymbol{x}^\epsilon$ in Eqn. 4 being the only manifestation of presynaptic activity, there is no mechanism by which to take into account input variability, which can, in turn, also impede learning. Furthermore, when compared to experimental evidence, this learning rule cannot explain heterosynaptic plasticity, as it is purely presynaptically gated.

In general, Euclidean gradient descent is well-known to exhibit slow convergence in non-isotropic regions of the cost function (Ruder, 2016), with such non-isotropy frequently arising or being aggravated by an inadequate choice of parametrization (see Ollivier, 2015, and Fig. 2). In contrast, natural gradient descent is, by construction, immune to these problems. The key idea of natural gradient as outlined by Amari is to follow the (locally) shortest path in terms of the neuron's firing distribution. Argued from a normative point of view, this is the only "correct" path to consider, since plasticity aims to adapt a neuron's behavior, i.e., its input-output relationship, rather than some internal parameter (Fig. 2).

For the concept of a locally shortest path to make sense in terms of distributions, we require the choice of a distance measure for probability distributions. Since a parametric statistical model, such as the set of our neuron's realizable output distributions, forms a Riemannian manifold (Rao, 1945; Amari and Nagaoka, 2000), a local distance measure can be obtained in form of a Riemannian metric. The Fisher metric (Rao, 1945), an infinitesmial version of the D_{KL}, represents a canonical choice on manifolds of probability distributions, since it is generally the unique metric that remains invariant under sufficient statistics (Cencov, 1972). On a given parameter space, the Fisher metric may be expressed in terms of a bilinear product with the Fisher information matrix

$$G(\boldsymbol{w}) = \mathbb{E}\left[\frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}} \frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}}^T\right]_{p_{\boldsymbol{w}}} . \tag{5}$$

The Fisher metric locally measures distances in the p-manifold as a function of the chosen parametrization. We can then obtain the natural gradient (which intuitively may be thought of as "$\partial C/\partial p$") by correcting the Euclidean gradient $\nabla^e_{\boldsymbol{w}} C := \partial C/\partial \boldsymbol{w}$ with the distance measure above:

$$\nabla^n_{\boldsymbol{w}} C = G\left(\boldsymbol{w}\right)^{-1} \nabla^e_{\boldsymbol{w}} C \, . \tag{6}$$

The natural gradient learning rule is then given as $\dot{\boldsymbol{w}} = -\eta \nabla^{n}_{\boldsymbol{w}} C$. Calculating the right-hand expression for the case of Poisson-spiking neurons (for details, see Supplementary Information, Sections S.1 and S.2), this takes the form

$$\dot{\boldsymbol{w}} = \eta \, \gamma_{\mathrm{s}} \left[Y^{*} - \phi(V) \right] \frac{\phi'(V)}{\phi(V)} \frac{1}{f'(\boldsymbol{w})} \left[c_{\epsilon} \frac{\boldsymbol{x}^{\epsilon}}{\boldsymbol{r}} - \gamma_{\mathrm{u}} + \gamma_{\mathrm{w}} f(\boldsymbol{w}) \right] , \tag{7}$$

where $\boldsymbol{w}$ is an arbitrary weight parametrization that relates to the somatic amplitudes via a component-wise rescaling $w^{\mathrm{s}} = f(\boldsymbol{w}) = \left[f_i(w_i) \right]_{1=1}^{n}$. For easier reading, we use several shorthand notations: multiplications and divisions of vectors, scalar functions and additions of scalars to vectors apply component-wise. Eqn. 7 represents the complete expression of our natural gradient rule, which we discuss throughout the remainder of the manuscript.

Natural gradient learning conserves both the error term $\left[Y^{*} - \phi(V) \right]$ and the USP contribution $\boldsymbol{x}^{\epsilon}$ from classical gradient-descent plasticity. However, by including the relationship between the parametrization of interest $\boldsymbol{w}$ and the somatic PSP amplitudes $f(\boldsymbol{w})$, natural-gradient-based plasticity explicitly accounts for reparametrization distortions, such as those arising from PSP attenuation during propagation along the dendritic tree. Furthermore, natural-gradient learning introduces multiple scaling factors and new plasticity components, whose characteristics will be further explored in dedicated sections below (see also Supplementary Information, Sections S.3.1 and S.3.2 for more details).

First of all, we note the appearance of two scaling factors (more details in Section 2.5). On one hand, the size of the synaptic adjustment is modulated by a global scaling factor γ_{s}, which adjusts synaptic weight updates to the characteristics of the output non-linearity, similarly to the synapse-specific scaling by the inverse of f'. Furthermore γ_{s} also depends on the output statistics of the neuron, harmonizing plasticity across different states in the output distribution (see Supplementary Information, Section S.3.1). On the other hand, a second, synapse-specific learning rate scaling accounts for the statistics of the input at the respective synapse, in the form of a normalization by the afferent input rate $c_{\epsilon}/\boldsymbol{r}$, where c_{ϵ} is a constant that depends on the PSP kernel (see Section 4.1). Unlike the global modulation introduced by γ_{s}, this scaling only affects the USP-dependent plasticity component. Just as for Euclidean-gradient-based learning, the latter is directly evoked by the spike trains arriving at the synapse. Therefore, the resulting plasticity is homosynaptic, affecting only synapses which receive afferent input.

However, in the case of natural-gradient learning, this input-specific adaptation is complemented by two additional forms of heterosynaptic plasticity (Section 2.6). First, the learning rule has a bias term γ_{u} which uniformly adjusts all synapses and may be considered homeostatic, as it usually opposes the USP-dependent plasticity contribution. The amplitude of this bias does not exclusively depend on the afferent input at the respective synapse, but is rather determined by the overall input to the neuron. Thus, unlike the USP-dependent component, this heterosynaptic plasticity component equally affects both active and inactive inactive synaptic connections. Furthermore, natural gradient descent implies the presence of another plasticity component $\gamma_{\mathrm{w}} f(\boldsymbol{w})$ which adapts the synapses depending on their current weight. More specifically, connections that are already strong are subject to larger changes compared to weaker ones. Since the proportionality factor γ_{w} only depends on global variables such as the membrane potential, this component also affects both active and inactive synapses.

The full expressions for γ_{s}, γ_{u} and γ_{w} are complicated functions of the membrane potential, its mean and variance, as well as, for γ_{u} and γ_{w}, of the total input $\sum_{i=1}^{n} \boldsymbol{x}_i^{\epsilon}$ and the total instantaneous presynaptic rate $\sum_{i=1}^{n} r_i$. However, under reasonable assumptions such as a high number of presynaptic partners and for a large, diverse set of empirically tested scenarios, we have shown that these factors can be reduced to simple functions of variables that are fully accessible at the locus of individual synapses. The above learning rule along with closed-form expressions for these factors (Supplementary Information, Sections S.3.1 and S.3.2) represent the main analytical findings of this paper.

We note that, while having used a standard sigmoidal transfer function throughout the paper, Eqn. 7 holds for every sufficiently smooth ϕ. Moreover, there exists a quadratic transfer function for which our learning rule becomes particularly simple, which we discuss in Section S.4 of the Supplementary Information.

In the following, we demonstrate that the additional terms introduced in natural-gradient-based plasticity confer important advantages compared to Euclidean gradient descent, both in terms of of convergence as well as with respect to biological plausibility. More precisely, we show that our plasticity rule improves convergence in a supervised learning task involving an anisotropic cost function, a situation which is notoriously hard to deal with for Euclidean-gradient-based learning rules (Ruder, 2016). We then proceed to investigate natural-gradient learning from a biological point of view, deriving a number of predictions that can be experimentally tested, with some of them related to in-vivo observations that are otherwise difficult to explain with classical gradient-based learning rules.

2.3 Natural gradient speeds up learning

Non-isotropic cost landscapes can easily be provoked by non-homogeneous input conditions. In nature, these phenomena arise under a wide range of circumstances, for elementary reasons that boil down to morphology (neurons are not

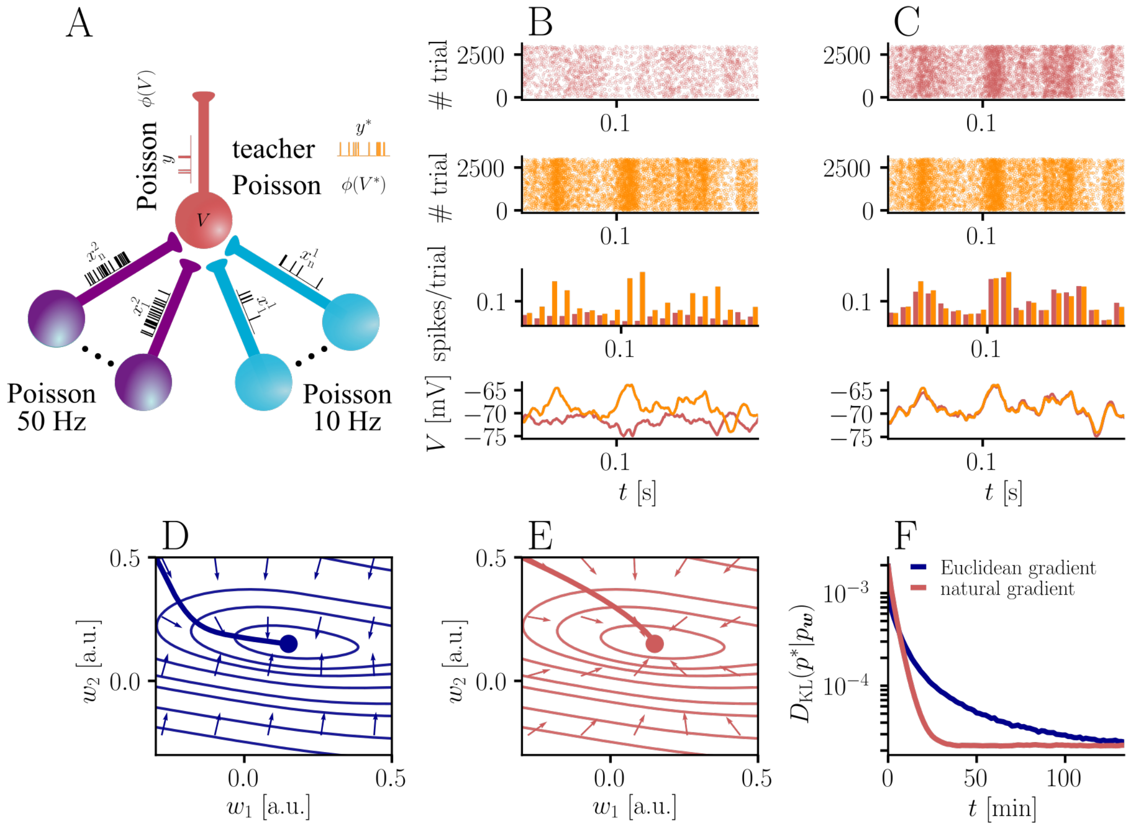

Figure 3: **Natural gradient speeds up learning in a simple regression task. (A)** We tested the performance of the natural gradient rule in a supervised learning scenario, where a single output neuron had to adapt its firing distribution to a target distribution, delivered in form of spikes from a teacher neuron. The input consisted of Poisson spikes from $n = 100$ afferents, half of them firing at 10 Hz and 50 Hz, respectively. **(B-C)** Spike trains, PSTHs and voltage traces for teacher (orange) and student (red) neuron before (B) and after (C) learning with natural-gradient plasticity. During learning, the firing patterns of the student neuron align to those of the teacher neuron. **(D-E)** Exemplary weight evolution during Euclidean-gradient (D) and natural-gradient (E) learning given $n = 2$ afferents with the same two rates as before. Thick solid lines represent contour lines of the cost function C. The respective vector fields depict normalized negative Euclidean and natural gradients of the cost C, averaged over 2000 input samples. The thin solid lines represent the paths traced out by the input weights during learning. **(F)** Learning curves for $n = 100$ afferents using natural-gradient and Euclidean-gradient plasticity. The plot shows averages over 1000 trials with initial and target weights randomly chosen from a uniform distribution $\boldsymbol{\mathcal{U}}(-1/n, 1/n)$. Fixed learning rates were tuned for each algorithm separately to exhibit the fastest possible convergence to a root mean squared error of 0.8 Hz in the student neuron's output rate.

symmetrical geometric objects) and function (neurons receive input from multiple afferents that perform different computations and thus behave differently). To evaluate the convergence behavior of our learning rule and compare it to Euclidean gradient descent, we considered a very generic situation in which a neuron is required to map a diverse set of inputs onto a target output.

In order to induce a simple and intuitive anisotropy of the error landscape, we divided the afferent population into two equally sized groups of neurons with different firing rates (Fig. 3A). This resulted in an asymmetric cost function, as visible from the elongated contour lines (Fig. 3D,E). We further chose a realizable teacher by simulating a different neuron with the same input populations connected via a predefined set of target weights $\boldsymbol{w}^*$. Fig. 3B,C show that our natural-gradient rule enables the student neuron to adapt its weights to reproduce the teacher voltage V^* and thereby its output distribution.

In the following, we compare learning in two student neurons, one endowed with Euclidean-gradient plasticity (Eqn. 4, Fig. 3D) and one with our natural-gradient rule (Eqn. 7, Fig. 3E). To better visualize the difference between the two rules, we used a two-dimensional input weight space, i.e., one neuron per afferent population. While the negative Euclidean gradient vectors stand, by definition, perpendicular to the contour lines of C, the negative natural gradient vectors point directly towards the target weight configuration $\boldsymbol{w}^*$. Due to the anisotropy of C induced by the different

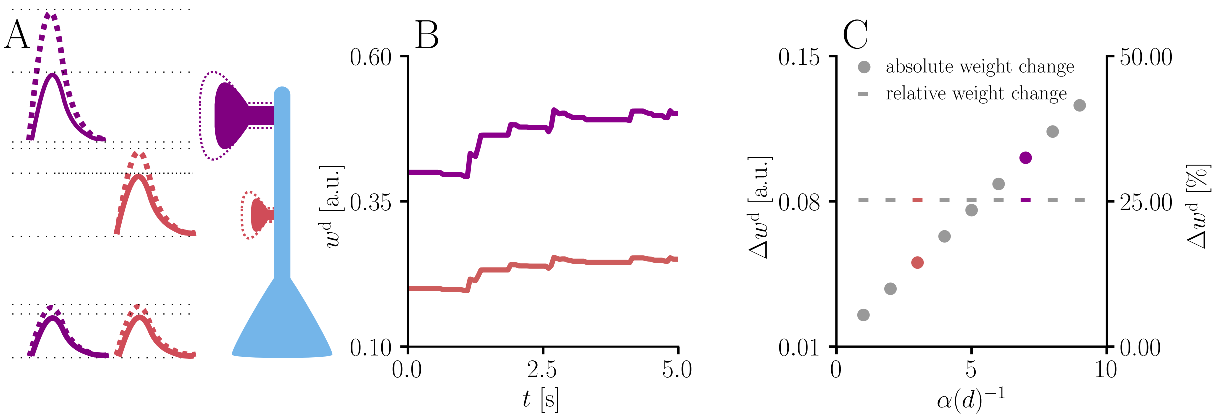

Figure 4: **Natural-gradient learning scales synaptic weight updates depending on their distance from the soma.** We stimulated a single excitatory synapse with Poisson input at 5 Hz, paired with a Poisson teacher spike train at 20 Hz. The distance d from soma was varied between 1 µm and 10 µm and attenuation was assumed to be linear and proportional to the inverse distance from soma. To make weight changes comparable, we scaled dendritic PSP amplitudes inversely with $d + 1$ in order for all of them to produce the same PSP amplitude at the soma. **(A)** Example PSPs before (solid lines) and after (dashed lines) learning for two synapses at 3 µm and 7 µm. Application of our natural-gradient rule results in equal changes for the somatic PSPs. **(B)** Example traces of synaptic weights for the two synapses in (A). **(C)** Absolute and relative dendritic amplitude change after 5 s as a function of a synapse's distance from the soma.

input rates (see also Fig. 2B), Euclidean-gradient learning starts out by mostly adapting the high-rate afferent weight and only gradually begins learning the low-rate afferent. In contrast, natural gradient adapts both synaptic weights homogeneously. This is clearly reflected by paths traced by the synaptic weights during learning.

Overall, this lead to faster convergence of the natural gradient plasticity rule compared to Euclidean gradient descent. In order to enable a meaningful comparison, learning rates were tuned separately for each plasticity rule in order to optimize their respective convergence speed. The faster convergence of natural-gradient plasticity is a robust effect, as evidenced in Fig. 3F by the average learning curves over 1000 trials.

In addition to the functional advantages described above, natural-gradient learning also makes some interesting predictions about biology, which we address below.

2.4 Democratic plasticity

As discussed in the introduction, classical gradient-based learning rules do not usually account for neuron morphology. Since attenuation of PSPs is equivalent to weight reparametrization and our learning rule is, by construction, parametrization-invariant, it naturally compensates for the distance between synapse and soma. In Eqn. 7, this is reflected by a component-wise rescaling of the synaptic changes with the inverse of the attenuation function f', which is induced by the Fisher information metric (see also Fig. 8 and the corresponding section in the Methods). Under the assumption of passive attenuation along the dendritic tree, we have $w_i^{\mathrm{s}} = f_{d_i}(w_i^{\mathrm{d}}) = \alpha(d_i) w_i^{\mathrm{d}}$, where d_i denotes the distance of the ith synapse from the soma. More specifically, $\alpha(\boldsymbol{d}) = e^{-\boldsymbol{d}/\lambda}$, where λ represents the electrotonic length scale. We can write the natural-gradient rule as

$$\dot{\boldsymbol{w}}^{\mathrm{d}} = \gamma_{\mathrm{s}} \left[Y^* - \phi(V) \right] \frac{\phi'(V)}{\phi(V)} \left[\frac{c_\epsilon}{\alpha(\boldsymbol{d})} \frac{\boldsymbol{x}^\epsilon}{\boldsymbol{r}} - \frac{\gamma_{\mathrm{u}}}{\alpha(\boldsymbol{d})} + \gamma_{\mathrm{w}} \boldsymbol{w}^{\mathrm{d}} \right] . \tag{8}$$

For functionally equivalent synapses (i.e., with identical input statistics), synaptic changes in distal dendrites are scaled up compared to proximal synapses. As a result, the effect of synaptic plasticity on the neuron's output is independent of the synapse location, since dendritic attenuation is precisely counterbalanced by weight update amplification.

We illustrate this effect with simulations of synaptic weight updates at different locations along a dendritic tree in Fig. 5. Such "democratic plasticity", which enables distal synapses to contribute just as effectively to changes in the output as proximal synapses, is reminiscent of the concept of "dendritic democracy" (Magee and Cook, 2000). These experiments show increased synaptic amplitudes in the distal dendritic tree of multiple cell types, such as rat hippocampal CA1 neurons; dendritic democracy has therefore been presumed to serve the purpose of giving distal inputs a "vote" on the neuronal output. Still, experiments show highly diverse PSP amplitudes in neuronal somata (Williams and Stuart, 2002). Our plasticity rule refines the notion of democracy by asserting that learning itself rather than its end result is rescaled in

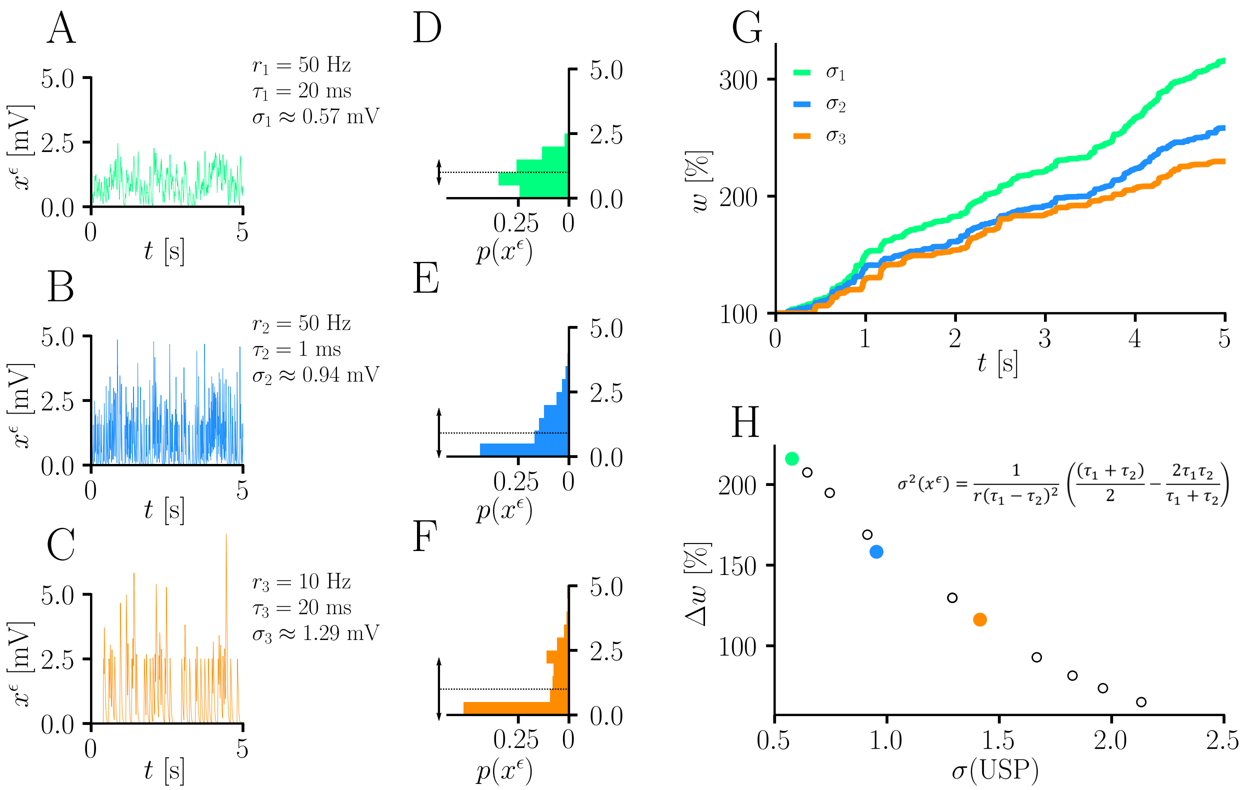

Figure 5: **Natural gradient learning scales with input variance.** **(A-C)** Exemplary USPs x_i^ϵ and **(D-F)** their distributions for three different scenarios between which the USP variance $\sigma^2(x_i^\epsilon)$ is varied. In each scenario, a neuron received a single excitatory input with a given rate r and synaptic time constant τ_s. The soma always received teacher spikes at a rate of 80 Hz. To enable a meaningful comparison, the mean USP was conserved by appropriately rescaling the height ϵ_0 of the USP kernel ϵ (see Section 4.1). **(A,D)** Reference simulation. **(B,E)** Reduced synaptic time constant, resulting in an increased USP variance σ_2^2. **(C,F)** Reduced input rate, resulting in an increased USP variance σ_3^2. **(K)** Synaptic weight changes over 5 s for the three scenarios above. **(L)** Total synaptic weight change after $t_0 = 5$ s as a function of USP variance. Each data point represents a different pair of r and τ_s. The three scenarios above are marked with their respective colors.

accordance with the neuronal morphology. Whether such democratic plasticity ultimately leads to distal and proximal synapses having the same effective vote at the soma depends on their respective importance towards reaching the target output. In particular, if synapses from multiple afferents that encode the same information are randomly distributed along the dendritic tree, then democratic plasticity also predicts dendritic democracy, as the scaling of weight changes implies a similar scaling of the final learned weights. Note, however, that the absence of dendritic democracy does not contradict the presence of democratic plasticity, as afferents from different cortical regions might target specific positions on the dendritic tree (see, e.g., Markram et al., 2004).

2.5 Input and output-specific scaling

In addition to undoing distortions induced by, e.g., attenuation, the natural gradient rule predicts further modulations of the homosynaptic learning rate. The factor γ_s in Eqn. 7 represents an output-dependent global scaling factor (for both homo- and heterosynaptic plasticity):

$$\gamma_\mathrm{s} = \mathbb{E}\left[\frac{\phi'(V)^2}{\phi(V)}\right]^{-1}_{p_\mathrm{usp}} . \tag{9}$$

It increases the learning rate in regions where the sigmoidal transfer function is flat (see also Section S.3.1). This represents an unmediated reflection of the philosophy of natural gradient descent, which finds the steepest path for a small change in output, rather than in the numeric value of some parameter. The desired change in the output requires scaling the corresponding input change by the inverse slope of the transfer function.

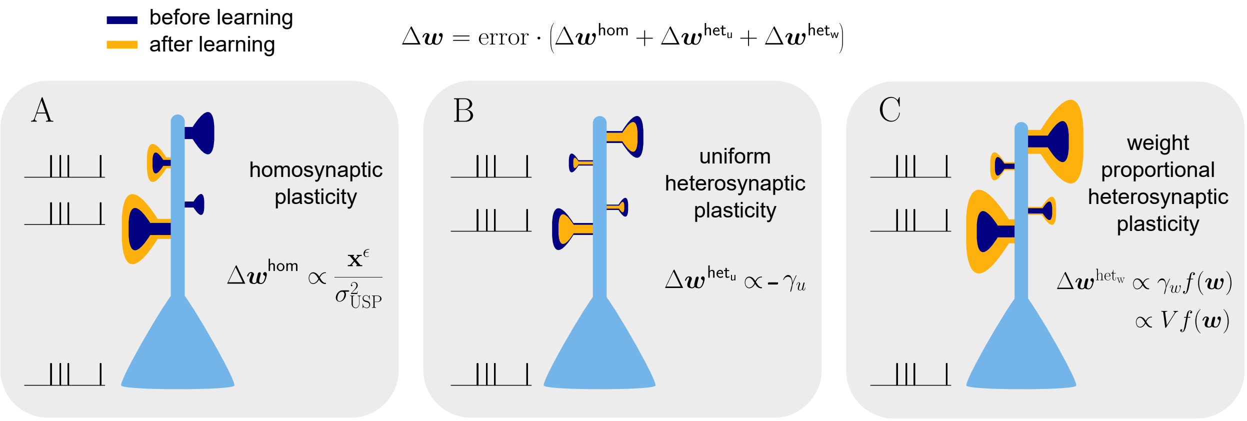
before learning
after learning
$\Delta \boldsymbol{w} = \text{error} \cdot \left(\Delta \boldsymbol{w}^{\text{hom}} + \Delta \boldsymbol{w}^{\text{het}_\text{u}} + \Delta \boldsymbol{w}^{\text{het}_\text{w}}\right)$
A
homosynaptic plasticity
$\Delta \boldsymbol{w}^{\text{hom}} \propto \frac{\mathbf{x}^\epsilon}{\sigma^2_{\text{USP}}}$
B
uniform heterosynaptic plasticity
$\Delta \boldsymbol{w}^{\text{het}_\text{u}} \propto - \gamma_u$
C
weight proportional heterosynaptic plasticity
$\Delta \boldsymbol{w}^{\text{het}_\text{w}} \propto \gamma_w f(\boldsymbol{w})$
$\propto V f(\boldsymbol{w})$

Figure 6: **Natural-gradient learning combines multiple forms of plasticity.** Spike trains to the left of the neuron represent afferent inputs to two of the synapses and teacher input to the soma. The two synapses on the right of the dendritic tree receive no stimulus. The teacher is assumed to induce a positive error. **(A)** The homosynaptic component adapts all stimulated synapses, leaving all unstimulated synapses untouched. **(B)** The uniform heterosynaptic component changes all synapses in the same manner, only depending on global activity levels. **(C)** The proportional heterosynaptic component contributes a weight change that is proportional to the current synaptic strength. The magnitude of this weight change is approximately proportional to a product of the current membrane potential above baseline and the weight vector.

Furthermore, synaptic learning rates are inversely correlated to the USP variance $\sigma^2(\boldsymbol{x}^\epsilon)$ (Fig. 5). In particular, for the homosynaptic component, the scaling is exactly equal to $\sigma^2(\boldsymbol{x}^\epsilon)^{-1} = c_\epsilon / r_i$ (see Eqn. 7 and Section 4.1). In other words, natural gradient learning explicitly scales synaptic updates with the (un)reliability of their input. To demonstrate this effect in isolation, we simulated the effects of changing the USP variance while conserving its mean. Moreover, to demonstrate its robustness, we independently varied two contributors to the input reliability, namely input rates (which enter $\sigma^2(\boldsymbol{x}^\epsilon)$ directly) and synaptic time constants (which affect the PSP-kernel-dependent scaling constant c_ϵ). Fig. 5 shows how unreliable input leads to slower learning, with an inverse dependence of synaptic weight changes on the USP variance. We note that this observation also makes intuitive sense from a Bayesian point of view, under which any information needs to be weighted by the reliability of its source (cf. also Aitchison and Latham, 2014, although our interpretation is different).

2.6 Interplay of homosynaptic and heterosynaptic plasticity

One elementary property of update rules based on Euclidean gradient descent is their presynaptic gating, i.e., all weight updates are scaled with their respective synaptic input $\boldsymbol{x}^\epsilon$. Therefore, they are necessarily restricted to homosynaptic plasticity, as studied in classical LTP and LTD experiments (Bliss and Lømo, 1973; Dudek and Bear, 1992). As discussed above, natural-gradient learning retains a rescaled version of this homosynaptic contribution, but at the same time predicts the presence of two additional plasticity components. Contrary to homosynaptic plasticity, these components also adapt synapses to currently non-active afferents, given a sufficient level of global input. Due to their lack of input specificity, they give rise to heterosynaptic weight changes, a form of plasticity that has been observed in hippocampus (Chen et al., 2013; Lynch et al., 1977), cerebellum (Ito and Kano, 1982) and neocortex (Chistiakova and Volgushev, 2009), mostly in combination with homosynaptic plasticity. A functional interpretation of heterosynaptic plasticity, to which our learning rule also alludes, is as a prospective adaptation mechanism for temporarily inactive synapses such that, upon activation, they are already useful for the neuronal output.

Our natural-gradient learning rule Eqn. 7 can be more summarily rewritten as

$$\Delta \boldsymbol{w} = \Delta \boldsymbol{w}^{\text{hom}} + \Delta \boldsymbol{w}^{\text{het}_\text{u}} + \Delta \boldsymbol{w}^{\text{het}_\text{w}} \,, \tag{10}$$

where the three additive terms represent the variance-normalized homosynaptic plasticity, the uniform heterosynaptic plasticity and the weight-dependent heterosynaptic plasticity:

$$\begin{aligned} \Delta \boldsymbol{w}^{\text{hom}} &\propto c_\epsilon \frac{\boldsymbol{x}^\epsilon}{\boldsymbol{r}} \,, && (11) \\ \Delta \boldsymbol{w}^{\text{het}_\text{u}} &\propto -\gamma_\text{u} \mathbf{1} \,, && (12) \\ \Delta \boldsymbol{w}^{\text{het}_\text{w}} &\propto \gamma_\text{w} f\left(\boldsymbol{w}\right) = c_\text{w} V f(\boldsymbol{w}) \,, && (13) \\ & && (14) \end{aligned}$$

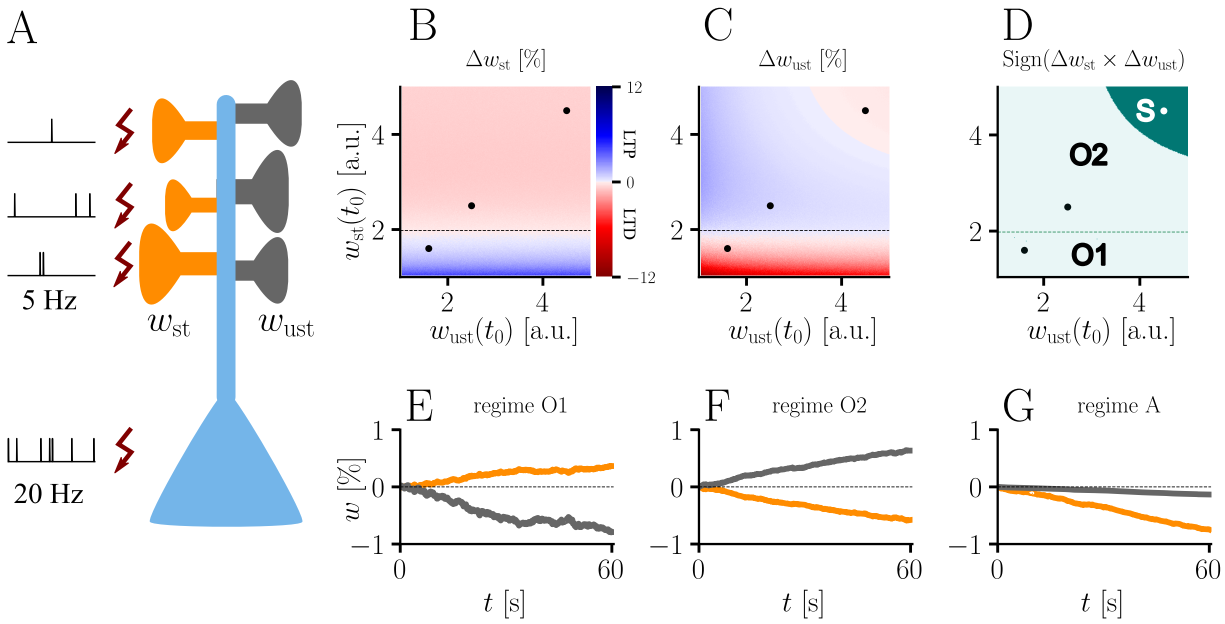

Figure 7: **Interplay of homo- and heterosynaptic plasticity in natural-gradient learning. (A)** Simulation setup. Five out of ten inputs received excitatory Poisson input at 5 Hz. To avoid singularities in the learning rule, the other five received extremely weak input at 0.01 Hz. In addition, we assumed the presence of tonic inhibition as a balancing mechanism for keeping the neuron's output within a reasonable regime. Afferent stimulus was paired with teacher spike trains at 20 Hz and plasticity at both stimulated and unstimulated synapses was evaluated in comparison with their initial weights. For simplicity, initial weights within each group were assumed to be equal. **(B)** Weight change of stimulated weights (homosynaptic). Homosynaptic plasticity is independent of unstimulated weights. Equilibrium (dashed black line) is reached when the neuron's output matches its teacher and the error vanishes. For increasing stimulated weights, potentiation switches to depression at the equilibrium line. **(C)** Weight change of unstimulated weights (heterosynaptic). For very high activity caused by very large synaptic weights, heterosynaptic plasticity always causes synaptic depression. Otherwise, it behaves exactly opposite to homosynaptic plasticity. Increasing the size of initial stimulated weights results in a change from depression to potentiation at the same point where homosynaptic potentiation turns into depression. **(D)** Direct comparison of the signs of homo- and heterosynaptic plasticity. The light green area (O1, O2) represents opposing signs, dark green (S) represents the same sign (more specifically, depression). Their shared equilibrium is marked by the dashed green line and represents the switch from positive to negative error. **(E-G)** Relative weight changes of synaptic weights for stimulated and unstimulated synapses during learning, with initial weights picked from the different regimes indicated by the crosses in (B, C, D).

with the common proportionality factor $\eta\gamma_{\mathrm{s}}\left[Y^{*}-\phi(V)\right]\frac{\phi'(V)}{\phi(V)}f'\left(\boldsymbol{w}\right)^{-1}$ composed of the learning rate, the output-dependent global scaling factor, the postsynaptic error, a sensitivity factor and the inverse attenuation function, in order of their appearance. The effect of these three components is visualized in Fig. 6B. The homosynaptic term $\Delta\boldsymbol{w}^{\mathrm{hom}}$ is experienced only by stimulated synapses, while the two heterosynaptic terms act on all synapses. The first heterosynaptic term $\Delta\boldsymbol{w}^{\mathrm{het_u}}$ introduces a uniform adjustment to all components by the same amount, depending on the global activity level. For a large number of presynaptic inputs, it can be approximated by a constant (see Section S.3.2). Furthermore, it usually opposes the homosynaptic change, which we address in more detail below.

In contrast, the contribution of the second heterosynaptic term $\Delta\boldsymbol{w}^{\mathrm{het_w}}$ is weight-dependent, adapting all synapses in proportion to their current strength. This explains experimental results such as Loewenstein et al. (2011), which found in-vivo weight changes in the neocortex to be proportional to the spine size, which itself is correlated with synaptic strength (Asrican et al., 2007). Our simulations show that $\Delta\boldsymbol{w}^{\mathrm{het_w}}$ is roughly a linear function of the membrane potential (more specifically, its deviation with respect to its baseline). Since the latter can be interpreted as a scalar product between the afferent input vector and the synaptic weight vector, it implies that input transmitted by strong synapses has the largest impact on this heterosynaptic plasticity component. In comparison, input from weak synapses only has a small effect, thus requiring persistent and strong stimulation of these synapses to induce significant changes and "override the status quo" of the neuron. Since, following a period of learning, afferents connected via weak synapses can be considered uninformative for the neuron's target output, this mechanism ensures a form of heterosynaptic robustness towards noise.

The homo- and heterosynaptic terms exhibit an interesting relationship. To illustrate the nature of their interplay, we simulated a simple experiment (Fig. 7A) with varying initial synaptic weights for both active and inactive presynaptic afferents. Stimulated synapses (Fig. 7B) are seen to undergo strong potentiation (LTP) for very small initial weights; the magnitude of weight changes decreases for larger initial amplitudes until the neuron's output matches its teacher, at which point the sign of the postsynaptic error term flips. For even larger initial weights, potentiation at stimulated synapses therefore turns into depression (LTD), which becoms stronger for higher initial values of the stimulated synapses' weights. This is in line with the error learning paradigm, in which changes in synaptic weights seek to reduce the difference between a neuron's target and its output.

For unstimulated synapses (Fig. 7C), we observe a reversed behavior. For small weights, the negative uniform term $\Delta \boldsymbol{w}^{\mathrm{het}_\mathrm{u}}$ dominates and plasticity is depressing. As for the homosynaptic case, the sign of plasticity switches when the weights become large enough for the error to switch sign. Therefore, in the regime where stimulated synapses experienced potentiation, unstimulated synapses are depressed and vice-versa. This reproduces various experimental observations: on one hand, homosynaptic potentiation has often been found to be accompanied by heterosynaptic depression (Lynch et al., 1977), such as in the amygdala (Royer and Paré, 2003) or the visual cortex (Arami et al., 2013); on the other hand, when the postsynaptic error term switches sign, depression at unstimulated synapses transforms into potentiation (Wöhrl et al., 2007; Royer and Paré, 2003).

While plasticity at stimulated synapses is unaffected by the initial state of the unstimulated synapses, plasticity at unstimulated synaptic connections depends on both the stimulated and unstimulated weights. In particular, when either of these grow large enough, the proportional term $\Delta \boldsymbol{w}^{\mathrm{het}_\mathrm{w}}$ overtakes the uniform term $\Delta \boldsymbol{w}^{\mathrm{het}_\mathrm{u}}$ and heterosynaptic plasticity switches sign again. Thus, for very large weights (top right corner of Fig. 7C), heterosynaptic potentiation transforms back into depression, in order to more quickly quench excessive output activity. This behavior is useful for both supervised and unsupervised learning scenarios (Zenke and Gerstner, 2017), where it was shown that pairing Hebbian terms with heterosynaptic and homeostatic plasticity is crucial for stability.

In summary, we can distinguish three plasticity regimes for natural-gradient learning (Fig. 7D-G). In two of these regimes, heterosynaptic and homosynaptic plasticity are opposed (O1, O2), whereas in the third, they are aligned and lead to depression (S). The two opposing regimes are separated by the zero-error equilibrium line, at which plasticity switches sign.

3 Discussion

As a consequence of the fundamentally stochastic nature of evolution, it is no surprise that biology withstands confinement to strict laws. Still, physics-inspired arguments from symmetry and invariance can help uncover abstract principles that evolution may have gradually discovered and implemented into our brains. Here, we have considered parametrization invariance in the context of learning, which, in biological terms, translates to the fundamental ability of neurons to deal with diversity in their morphology and input-output characteristics. This requirement ultimately leads to various forms of scaling and heterosynaptic plasticity that are experimentally well-documented, but can not be accounted for by classical paradigms that regard plasticity as Euclidean gradient descent. In turn, these biological phenomena can now be seen as a means to jointly improve and accelerate error-correcting learning.

Inspired by insights from information geometry, we applied the framework of natural gradient descent to biologically realistic neurons with extended morphology and spiking output. Compared to classical error-correcting learning rules, our plasticity paradigm requires the presence of several additional ingredients. First, a global factor adapts the learning rate to the particular shape of the voltage-to-spike transfer function and to the desired statistics of the output, thus addressing the diversity of neuronal response functions observed in vivo (Markram et al., 2004). Second, the homosynaptic component of plasticity is normalized by the variance of presynaptic inputs, which provides a direct link to Bayesian frameworks of neuronal computation (Aitchison and Latham, 2014; Jordan et al., 2020). Third, our rule contains a uniform heterosynaptic term that opposes homosynaptic changes, downregulating plasticity and thus acting as a homeostatic mechanism (Chen et al., 2013; Chistiakova et al., 2015). Fourth, we find a weight-dependent heterosynaptic term that also accounts for the shape of the neuron's activation function, while increasing its robustness towards noise. Finally, our natural-gradient-based plasticity correctly accounts for the somato-dendritic reparametrization of synaptic strengths.

These features enable faster convergence on non-isotropic error landscapes, in line with results for multilayer perceptrons (Yang and Amari, 1998; Rattray and Saad, 1999) and rate-based deep neural networks (Pascanu and Bengio, 2013; Ollivier, 2015; Bernacchia et al., 2018). Importantly, our learning rule can be formulated as a simple, fully local expression, only requiring information that is available at the locus of plasticity.

We further note an interesting property of our learning rule, which it inherits directly from the Fisher information metric that underlies natural gradient descent, namely invariance under sufficient statistics. This is especially relevant for biological neurons, whose stochastic firing effectively communicates information samples rather than explicit distributions. Thus, downstream computation is likely to require a reliable sample-based, i.e., statistically sufficient, estimation of the afferent distribution's parameters, such as the sample mean and variance. This singles out our natural-gradient approach from other second-order-like methods as a particularly appealing framework for biological learning.

Many of the biological phenomena predicted by our invariant learning rule are reflected in existing experimental results. Our "synaptic democracy" can give rise to dendritic democracy, as observed by Magee and Cook (2000). Our plasticity rule requires heterosynaptic plasticity, which has been observed in neocortex, as well as in deeper brain regions such as amygdala and hippocampus (Lynch et al., 1977; Engert and Bonhoeffer, 1997; White et al., 1990; Royer and Paré, 2003; Wöhrl et al., 2007; Chistiakova and Volgushev, 2009; Arami et al., 2013; Chen et al., 2013; Chistiakova et al., 2015), often in combination with homosynaptic weight changes. Moreover, we find that heterosynaptic plasticity generally opposes homosynaptic plasticity, which qualitatively matches many experimental findings (Lynch et al., 1977; White et al., 1990; Royer and Paré, 2003; Wöhrl et al., 2007) and can be functionally interpreted as an enhancement of competition. For very large weights, heterosynaptic plasticity aligns with homosynaptic changes, pushing the synaptic weights back to a sensible range (Chistiakova et al., 2015), as shown to be necessary for unsupervised learning (Zenke and Gerstner, 2017). In supervised learning it helps speed up convergence by keeping the weights in the operating range.

A further prediction that follows from our plasticity rule is the normalization of weight changes by the presynaptic variance. We would thus anticipate that increasing the jitter in presynaptic spike trains should reduce LTP in standard plasticity induction protocols. Also, we expect to observe a significant dependence of synaptic plasticity on neuronal response functions and output statistics. For example, flatter response functions should correlate with faster learning, in contrast to the inverse correlation predicted by classical learning rules derived from Euclidean gradient descent. These propositions remain to be tested experimentally.

By following gradients with respect to the neuronal output rather than the synaptic weights themselves, we were able to derive a parametrization-invariant error-correcting plasticity rule on the single-neuron level. Error-correcting learning rules are an important ingredient in understanding biological forms of error backpropagation Sacramento et al. (2017). In principle, our learning rule can be directly incorporated as a building block into spike-based frameworks of error backpropagation such as (Sporea and Grüning, 2013; Schiess et al., 2016). Based on these models, top-down feedback can provide a target for the somatic spiking of individual neurons, towards which our learning rule could be used to speed up convergence. Explicitly and exactly applying natural gradient at the network level does not appear biologically feasible due to the existence of cross-unit terms in the Fisher information matrix G. However, methods such as the unit-wise natural-gradient approach (Ollivier, 2015) could be employed to approximate the natural gradient using a block-diagonal form of G. For spiking networks, this would reduce global natural-gradient descent to our local rule for single neurons.

4 Methods

4.1 Neuron model

We chose a Poisson neuron model whose firing rate depends on the somatic membrane potential V above the resting potential $V_{\text{rest}} = -70\,\text{mV}$. They relate via a sigmoidal activation function

$$\phi(V) = \frac{\phi_{\max}}{1 + \exp\left[-\beta(V - \theta)\right]}, \tag{15}$$

with $\beta = 0.3$, $\theta = 10\,\text{mV}$ and a maximal firing rate $\phi_{max} = 100\,\text{Hz}$. This means the activation function is centered at $-60\,\text{mV}$, saturating around $-50\,\text{mV}$. Note that the derivation of our learning rule does not depend on the explicit choice of activation function, but holds for arbitrary monotonically increasing, positive functions that are sufficiently smooth. For the sake of simplicity, refractoriness was neglected. For the same reason, we assumed synaptic input to be current-based, such that incoming spikes elicit a somatic membrane potential above baseline given by

$$V = \sum_i w_i x_i^\epsilon\,, \tag{16}$$

where

$$x_i^\epsilon(t) = \left[x_i * \epsilon\right](t) \tag{17}$$

denotes the unweighted synaptic potential (USP) evoked by a spike train

$$x_i = \sum_{t_i^{\mathrm{f}}} \delta\left(t - t_i^{\mathrm{f}}\right) \tag{18}$$

of afferent i, and w_i is the corresponding synaptic weight. Here, t_i^{f} denote the firing times of afferent i, and the synaptic response kernel ϵ is modeled as

$$\epsilon(t) = \epsilon_0 \frac{\Theta(t)}{(\tau_{\mathrm{m}} - \tau_{\mathrm{s}})} \left[\exp\left(-\frac{t}{\tau_{\mathrm{m}}}\right) - \exp\left(-\frac{t}{\tau_{\mathrm{s}}}\right)\right], \tag{19}$$

where ϵ_0 is a scaling factor with units $\mathrm{mV\,ms}$, and unless specified otherwise, we chose $\epsilon_0 = 1\,\mathrm{mV\,ms}$. For slowly changing input rates, mean and variance of the stationary unweighted synaptic potential are then given as (Petrovici, 2016)

$$\mathbb{E}\left[x_i^{\epsilon}\right] \approx \epsilon_0 r_i \tag{20}$$

and

$$\mathrm{Var}\left(x_i^{\epsilon}\right) \approx c_{\epsilon}^{-1} r_i\ , \tag{21}$$

with $c_{\epsilon} = \left(\int_0^{\infty} \epsilon^2 dt\right)^{-1}$. Unless indicated otherwise, simulations were performed with a membrane time constant $\tau_{\mathrm{m}} = 10\,\mathrm{ms}$ and a synaptic time constant $\tau_{\mathrm{s}} = 3\,\mathrm{ms}$. Hence, USPs had an amplitude of $60\,\mathrm{mV}$ and were normalized with respect to area under the curve and multiplied by the synaptic weights. Initial and target weights were chosen such that the resulting average membrane potential was within operating range of the activation function. As an example, in Fig. 3F, the average initial excitatory weight was 0.005, corresponding to an EPSP amplitude of $300\,\mu\mathrm{V}$.

In Fig. 5, the scaling factor ϵ_0 was additionally normalized proportionally to the input rate r_i at the synapse in order to keep the mean USP constant and allow a comparison based solely on the variance.

4.2 Derivation of the somatic natural gradient learning rule

The choice of a Poisson neuron model implies that spiking in a small interval $[t, t + \mathrm{d}t]$ is governed by a Poisson distribution. For sufficiently small interval lengths $\mathrm{d}t$, the probability of having a single spike in $[t, t + \mathrm{d}t]$ becomes Bernoulli with parameter $\phi\left(V_t\right)\mathrm{d}t$. The aim of supervised learning is to bring this distribution closer to a given target distribution with density p^*, delivered in form of a teacher spike train. We measure the error between the desired and the current input-output spike distribution in terms of the Kullback-Leibler divergence given in Eqn. 3, which attains its single minimum when the two distributions are equal. Note that while the D_{KL} is a standard measure to characterize "how far" two distributions are apart, its behavior can sometimes be slightly unintuitive since it is not a metric. In particular, it is not symmetric and does not satisfy the triangle inequality.

Classical error learning follows the Euclidean gradient of this cost function, given as the vector of partial derivatives with respect to the synaptic weights. A short calculation (Supplementary Information, Section S.1) shows that the resulting Euclidean gradient descent learning rule is given by Eqn. 4. By correcting the vector of partial derivatives for the distance distortion between the manifold of input-output distributions and the synaptic weight space, given in terms of the Fisher information matrix $G(\boldsymbol{w})$, we obtain the natural gradient (Eqn. 6). We then followed an approach by Amari (Amari, 1998) to derive an explicit formula for the product on the right hand side of Eqn. 6.

In Section S.1 of the Supplementary Information, we show that given independent input spike trains, the Fisher information matrix defined in Eqn. 5 can be decomposed with respect to the vector of input rates $\boldsymbol{r}$ and the current weight vector $\boldsymbol{w}$ as

$$G^{\mathrm{s}}\left(\boldsymbol{w}^{\mathrm{s}}\right) = c_1\left(\epsilon_0^2 \boldsymbol{r}\boldsymbol{r}^T + \Sigma_{\mathrm{usp}}\right) + c_2\epsilon_0\left(\Sigma_{\mathrm{usp}}\boldsymbol{w}^{\mathrm{s}}\boldsymbol{r}^T + \boldsymbol{r}\Sigma_{\mathrm{usp}}\boldsymbol{w}^{\mathrm{s}T}\right) + c_3\left(\Sigma_{\mathrm{usp}}\boldsymbol{w}^{\mathrm{s}}\right)\left(\Sigma_{\mathrm{usp}}\boldsymbol{w}^{\mathrm{s}}\right)^T . \tag{22}$$

Here, $\Sigma_{\mathrm{usp}} = \mathrm{diag}\{\mathrm{c}_{\epsilon}^{-1}\mathrm{r_i}\}_{\mathrm{i=1}}^{\mathrm{n}}$ is the covariance matrix of the unweighted synaptic potentials, and c_1, c_2, c_3 are coefficients (Eqn. S32 for their definition) depending on the mean μ_V and variance σ_V^2 of the membrane potential, and on the total rate $q = c_{\epsilon}\epsilon_0^2 \sum_i r_i$. Through repeated application of the Sherman-Morrison-Formula, the inverse of $G(\boldsymbol{w})$ can be obtained as

$$G^{\mathrm{s}}\left(\boldsymbol{w}^{\mathrm{s}}\right)^{-1} = \gamma_{\mathrm{s}}\left(\Sigma_{\mathrm{usp}}^{-1} + \left(c_{\epsilon}\epsilon_0 g_1 \mathbf{1} + g_2\boldsymbol{w}^{\mathrm{s}}\right)c_{\epsilon}\epsilon_0\mathbf{1}^T + \left[c_{\epsilon}\epsilon_0 g_3\mathbf{1} + g_4\boldsymbol{w}^{\mathrm{s}}\right)\boldsymbol{w}^{\mathrm{s}T}\right] . \tag{23}$$

Here the coefficients $\gamma_{\mathrm{s}}, g_1, g_2, g_3, g_4$, which are defined in Eqn. S52, are again functions of mean and variance of the membrane potential, and of the total rate. Consequently, the natural gradient rule in terms of somatic amplitudes is given by

$$\dot{\boldsymbol{w}}^{\mathrm{s}} = \gamma_{\mathrm{s}}\left[Y^* - \phi(V)\right]\frac{\phi'(V)}{\phi(V)}\left(\frac{c_{\epsilon}\boldsymbol{x}^{\epsilon}}{\boldsymbol{r}} - \gamma_{\mathrm{u}}\mathbf{1} + \gamma_{\mathrm{w}}\boldsymbol{w}^{\mathrm{s}}\right) . \tag{24}$$

Note that the formulas for

$$\gamma_{\mathrm{u}} = -c_\epsilon \epsilon_0 \left(g_1 c_\epsilon \epsilon_0 \sum_{i=1}^{n} x_i^\epsilon + g_3 V \right) \text{ and } \gamma_{\mathrm{w}} = \left(g_2 c_\epsilon \epsilon_0 \sum_{i=1}^{n} x_i^\epsilon + g_4 V \right) . \tag{25}$$

arise from the product of the inverse Fisher information matrix and the Euclidean gradient, using $\mathbf{1}^T \boldsymbol{x}^\epsilon = \sum_{i=1}^n x_i^\epsilon$ and ${\boldsymbol{w}^{\mathrm{s}}}^T \boldsymbol{x}^\epsilon = V$. Due to the complicated expressions for $g_1, \ldots, g_4$ (Eqn. S32), Eqn. 25 only provides limited information about the behavior of γ_{u} and γ_{w}. Therefore, we performed an empirical analysis based on simulation data (Supplementary Information, Section S.3.2, Fig. S2). In a stepwise manner we first evaluated $g_1, \ldots, g_4$ under various conditions, which revealed that the products with g_2 and g_3 in Eqn. 25 are neglible in most cases compared to the other terms, hence $\gamma_{\mathrm{u}} \approx c_\epsilon^2 \epsilon_0^2 \sum_{i=1}^n x_i^\epsilon$ and $\gamma_{\mathrm{w}} \approx g_4 V$. Furthermore, for a sufficient number of input afferents, we can approximate $g_1 \approx -q^{-1}$. Since $q = c_\epsilon \epsilon_0 \mathbb{E}[\sum_{i=1}^n x_i^\epsilon]$, by the central limit theorem we have $\gamma_u \approx c_{\mathrm{u}} c_\epsilon$ for large n and $c_{\mathrm{u}} = \epsilon_0 = 1$. Moreover, while the variance of g_4 across weight samples increases with the number and firing rate of input afferents, its mean stays approximately constant across conditions. This lead to the approximation $\gamma_{\mathrm{w}} \approx c_{\mathrm{w}} V$, where c_{w} is constants across weights, input rates and the number of input afferents.

To evaluate the quality of our approximations, we tested the performance of learning in the setting of Fig. 3 when γ_{u} and γ_{w} were replaced by their approximations (Eqn. S60). The test was performed for several input patterns (Section 4.4.6, Section S.3.3, Fig. S3). It turned out that a convergence behavior very similar to natural gradient descent could be achieved with $c_{\mathrm{u}} = 0.95$, which worked much better in practice than $c_{\mathrm{u}} = 1.$. For c_{w} a choice of 0.05 which was close to the mean of c_{w} worked well.

For these input rate configurations and choices of constants, we additionally sampled the negative gradient vectors for random initial weights and USPs (Section 4.4.6, Supplementary Information, Section S.3.3, Fig. S3) and compared the angles and length difference between natural gradient vectors and the approximation to the ones between natural and Euclidean gradient.

4.3 Reparametrization and general natural gradient rule

To arrive at a more general form of the natural gradient learning rule, we consider a parametrization $\boldsymbol{w}$ of the synaptic weights which is connected to the somatic amplitudes via a smooth component-wise coordinate change $f = (f_1, \ldots, f_n)$, such that $w_i^{\mathrm{s}} = f_i(w_i)$. A Taylor expansion shows that small weight changes then relate via the derivative of f

$$\Delta w_i^{\mathrm{s}} = f_i'(w_i)\, \Delta w_i \, . \tag{26}$$

On the other hand, we can also express the cost function in terms of $\boldsymbol{w}$, with $C[\boldsymbol{w}] = C^{\mathrm{s}}[f(\boldsymbol{w}^{\mathrm{s}})]$, and directly calculate the Euclidean gradient of C in terms of $\boldsymbol{w}$. By the chain rule, we then have

$$\Delta w = -\nabla_{\boldsymbol{w}}^{e} C = -\frac{\partial C}{\partial \boldsymbol{w}} = -\frac{\partial C^{\mathrm{s}}}{\partial \boldsymbol{w}^{\mathrm{s}}} \frac{\partial \boldsymbol{w}^{\mathrm{s}}}{\partial \boldsymbol{w}} = -\mathrm{diag}\{f_i'(w_i)\} \nabla_{\boldsymbol{w}}^{e} C^{\mathrm{s}} = \mathrm{diag}\{f_i'(w_i)\} \Delta w^{\mathrm{s}}. \tag{27}$$

Plugging this into Eqn. 26, we obtain the contradiction $\Delta w_i^{\mathrm{s}} = f_i'(w_i)^2\, \Delta w_i^{\mathrm{s}}$. Hence the predictions of Euclidean gradient learning depend on our choice of synaptic weight parametrization (Fig. 1).

In order to obtain the natural gradient learning rule in terms of $\boldsymbol{w}$, we first express the Fisher Information Matrix in the new parametrization, starting with Eqn. S11

$$G(\boldsymbol{w}) = \mathbb{E}\left[\frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}} \frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}}^T \right]_{p_{\boldsymbol{w}}} \tag{28}$$

$$= \mathbb{E}\left[\left(\frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}^{\mathrm{s}}} \frac{\partial \boldsymbol{w}^{\mathrm{s}}}{\partial \boldsymbol{w}} \right) \left(\frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}^{\mathrm{s}}} \frac{\partial \boldsymbol{w}^{s}}{\partial \boldsymbol{w}} \right)^T \right]_{p_{\boldsymbol{w}}} \tag{29}$$

$$= \mathrm{diag}\{f_i'(w_i)\} \mathbb{E}\left[\frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}^{\mathrm{s}}} \frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}^{\mathrm{s}}}^T \right]_{p_{\boldsymbol{w}}} \mathrm{diag}\{f_i'(w_i)\} \tag{30}$$

$$= \mathrm{diag}\{f_i'(w_i)\} G^{\mathrm{s}}(\boldsymbol{w}^{\mathrm{s}}) \{f_i'(w_i)\}. \tag{31}$$

Inserting both Eqn. 27 and Eqn. 30 into Eqn. 6, we obtain the natural gradient rule in terms of $\boldsymbol{w}$ (Eqn. 7). As illustrated in (Fig. 8), unlike for Euclidean gradient descent, the result is consistent with Eqn. 26.

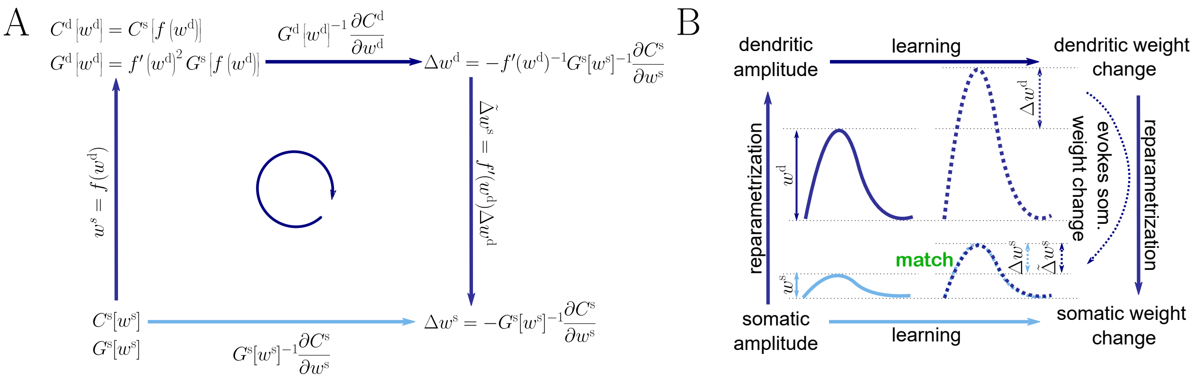
A
$C^{\mathrm{d}}\left[w^{\mathrm{d}}\right] = C^{\mathrm{s}}\left[f\left(w^{\mathrm{d}}\right)\right]$
$G^{\mathrm{d}}\left[w^{\mathrm{d}}\right] = f'\left(w^{\mathrm{d}}\right)^2 G^{\mathrm{s}}\left[f\left(w^{\mathrm{d}}\right)\right]$
$G^{\mathrm{d}}\left[w^{\mathrm{d}}\right]^{-1}\frac{\partial C^{\mathrm{d}}}{\partial w^{\mathrm{d}}}$
$\Delta w^{\mathrm{d}} = -f'(w^{\mathrm{d}})^{-1}G^{\mathrm{s}}[w^{\mathrm{s}}]^{-1}\frac{\partial C^{\mathrm{s}}}{\partial w^{\mathrm{s}}}$
$w^{s} = f(w^{\mathrm{d}})$
$\tilde{\Delta} w^{\mathrm{s}} = f'(w^{\mathrm{d}})\Delta w^{\mathrm{d}}$
$C^{\mathrm{s}}[w^{\mathrm{s}}]$
$G^{\mathrm{s}}[w^{\mathrm{s}}]$
$\Delta w^{\mathrm{s}} = -G^{\mathrm{s}}[w^{\mathrm{s}}]^{-1}\frac{\partial C^{\mathrm{s}}}{\partial w^{\mathrm{s}}}$
$G^{\mathrm{s}}[w^{\mathrm{s}}]^{-1}\frac{\partial C^{\mathrm{s}}}{\partial w^{\mathrm{s}}}$
B
dendritic amplitude
learning
dendritic weight change
reparametrization
w^{d}
Δw^{d}
evokes som. weight change
reparametrization
match
w^{s}
Δw^{s}
$\tilde{\Delta} w^{\mathrm{s}}$
somatic amplitude
learning
somatic weight change

Figure 8: **Natural gradient descent does not depend on chosen parametrization.** **(A)** Mathematical derivation. **(B)** Phenomenological correlates. EPSPs before learning are represented as continuous, after learning as dashed curves. The light blue arrow represents natural gradient descent on the error as a function of the somatic EPSP $C^{\mathrm{s}}\left[w^{\mathrm{s}}\right]$ (also shown in light blue). The resulting weight change leads to an increase Δw^{s} in the somatic EPSP after learning. The dark blue arrows track the calculation of the same gradient, but with respect to the dendritic EPSP (also shown in dark blue): 1) taking the attenuation into account in order to compute the error as a function of w^{d}, 2) calculating the gradient, followed by 3) deriving the associated change in $\tilde{\Delta} w^{\mathrm{s}}$, again considering attenuation. Unlike for Euclidean Gradient descent (Fig. 1), the factor $f'(w)^2$ is compensated, since its inverse enters via the Fisher information. This leads to the synaptic weights updates, as well as the associated evolution of a neuron's output statistics over time, being equal under the two parametrizations.

4.4 Simulation Details

All simulations were performed in python and used the numpy and scipy packages. Differential equations were integrated using a forward Euler method with a time step of $0.5\,\mathrm{ms}$.

4.4.1 Supervised Learning Task

A single output neuron was trained to spike according to a given target distribution in response to incoming spike trains from n independently firing afferents. To create an asymmetry in the input, we chose one half of the afferents' firing rates as $10\,\mathrm{Hz}$, while the remaining afferents fired at $50\,\mathrm{Hz}$. The supervision signal consisted of spike trains from a teacher that received the same input spikes. To allow an easy interpretation of the results, we chose a realizable teacher, firing with rate $\phi\left(V^*\right)$, where $V^* = \boldsymbol{w}^{*T}\boldsymbol{x}^{\epsilon}$ for some optimal set of weights $\boldsymbol{w}^*$. However, our theory itself does not include assumptions about the origin and exact form of the teacher spike train.

For the learning curve in Fig. 3F, initial and target weight components were chosen randomly from a uniform distribution on $\mathcal{U}\left(-1/n, 1/n\right)$, corresponding to maximal PSP amplitudes between $-600\,\mu\mathrm{V}$ and $600\,\mu\mathrm{V}$ for the simulation with $n = 100$ input neurons. Learning curves were averaged over 1000 initial and target weight configurations. We did not enforce Dales' law, thus about half of the synaptic input was inhibitory at the beginning but sign changes were permitted. This means that the mean membrane potential above rest covered a maximal range of $\left[-30\,\mathrm{mV}, 30\,\mathrm{mV}\right]$. Learning rates were optimized as $\eta_{\mathrm{n}} = 6 * 10^{-4}$ for the natural gradient descent algorithm and $\eta_{\mathrm{e}} = 4.5 * 10^{-7}$ for Euclidean gradient descent, providing the fastest possible convergence to a residual root mean squared error in output rates of $0.8\,\mathrm{Hz}$. Per trial, the expectation over USPs in the cost function was evaluated on a randomly sampled test set of 50 USPs that resulted from input spike trains of $250\,\mathrm{ms}$. The expectation over output spikes was calculated analytically.

For the weight path simulation with two neurons (Fig. 3D-E), we chose a fixed initial weight $\boldsymbol{w}_0 = (\frac{-0.3}{n}, \frac{0.5}{n})^T$, a fixed target weight $\boldsymbol{w}^* = (\frac{0.15}{n}, \frac{0.15}{n})^T$, and learning rates $\eta_{\mathrm{n}} = 2.5 * 10^{-4}$ and $\eta_{\mathrm{e}} = 7. * 10^{-7}$. Weight paths were averaged over 500 trials of $6000\,\mathrm{s}$ duration each.

The vector plots in Fig. 3D-E display the average negative normalized natural and Euclidean gradient vectors across 2000 USP samples per synapse ($n = 2$) on a grid of weight positions on $[\frac{-0.4}{n}, \frac{0.6}{n}]^2$, with the first coordinate of the gridpoints in $\{\frac{-0.28}{n}, \frac{-0.1}{n}, \frac{0.08}{n}, \frac{0.26}{n}, \frac{0.44}{n}\}$ and the second in $\{\frac{-0.22}{n}, \frac{-0.02}{n}, \frac{0.26}{n}, \frac{0,5}{n}\}$. Each USP sample was the result of a $1\,\mathrm{s}$ spike train at rate $r_1 = 10\,\mathrm{Hz}$ and $r_2 = 50\,\mathrm{Hz}$ respectively. The contour lines were obtained from 2000 samples of the D_{KL} along a grid on $[\frac{-0.4}{n}, \frac{0.6}{n}]^2$ (distance between two grid points in one dimension: $\frac{0.006}{n}$) and displayed at the levels $0.001, 0.003, 0.005, 0.009, 0.015, 0.02, 0.03, 0.04$.

For the plots in Fig. 3B-C, we used initial, final, and target weights from a sample of the learning curve simulation. We then randomly sampled input spike trains of $250\,\mathrm{ms}$ length and calculated the resulting USPs and voltages according to

Eqn. 17 and Eqn. 16. The output spikes shown in the raster plot were then sampled from a discretized Poisson process with $\mathrm{d}t = 5. * 10^{-4}$. We then calculated the PSTH with a bin size of 12.5 ms.

4.4.2 Distance dependence of amplitude changes

A single excitatory synapse received Poisson spikes at 5 Hz, paired with Poisson teacher spikes at 20 Hz. The distance from the soma was varied between 1 µm and 10 µm. Learning was switched on for 5 s with an initial weight corresponding to 0.05 at the soma, corresponding to a PSP amplitude of 3 mV. Initial dendritic weights were scaled up with the proportionality factor $\alpha(d)^{-1}$ depending on the distance from the soma, in order for input spikes to result in the same somatic amplitude independent of the synaptic position. Example traces are shown for $\alpha(d)^{-1} = 3$ and $\alpha(d)^{-1} = 7$.

4.4.3 Variance dependence of amplitude changes

We stimulated a single excitatory synapse with Poisson spikes, while at the same time providing Poisson teacher spike trains at 80 Hz. To change USP variance independently from mean, unlike in the other exercises, the input kernel in Eqn. 19 was additionally normalized by the input rate. USP variance was varied by either keeping the input rate at 10 Hz while varying the synaptic time constant τ_m between 1 ms and 20 ms, or fixing τ_s at 20 ms and varying the input rate between 10 Hz and 50 Hz.

4.4.4 Comparison of homo- and heterosynaptic plasticity

Out of $n = 10$ excitatory synapses of a neuron, we stimulated 5 by Poisson spike trains at 5 Hz, together with teacher spikes at 20 Hz, and measured weight changes after 60 s of learning. Initial weights for both unstimulated and stimulated synapses were varied between $\frac{1}{n}$ and $\frac{5}{n}$. For reasons of simplicity, all stimulated weights were assumed to be equal, and tonic inhibition was assumed by a constant shift in baseline membrane potential of −5 mV. Example weight traces are shown for initial weights of $\frac{1.6}{n}$, $\frac{2.5}{n}$, and $\frac{4.5}{n}$ for both stimulated and unstimulated weights. The learning rate was chosen as $\eta = 0.01$.

4.4.5 Approximation of learning rule coefficents

We sampled the values for $g_1, \ldots, g_4$ from Eqn. S32 for different afferent input rates. The input rate r was varied between 5 Hz and 55 Hz for $n = 100$ neurons. The coefficients were evaluated for randomly sampled input weights (20 weight samples of dimension n, each component sampled from a uniform distribution $\mathcal{U}\,(-5/n, 5/n)$).

In a second simulation, we varied the number n of afferents between 10 and 200 for a fixed input rate of 20 Hz, again for randomly sampled input weights (20 weight samples of dimension n, each component sampled from a uniform distribution $\mathcal{U}\,(-5/n, 5/n)$).

In a next step, we compared the sampled values of g_1 as a function of the total input rate $n * r$ to the values of the approximation given by $g_1 \approx -q^{-1}$ (r between 5 Hz and 55 Hz, n between 10 and 200 neurons, 20 weight samples of dimension n, each component sampled from a uniform distribution $\mathcal{U}\,(-5/n, 5/n)$).

Afterwards, we plotted the sampled values of γ_u as a function of the approximation s (Eqn. S59, r between 5 Hz and 55 Hz, $n = 100$, 20 weight samples of dimension n, each component sampled from a uniform distribution $\mathcal{U}\,(-5/n, 5/n)$, 20 USP-samples of dimension n for each rate/weight-combination).

Next, we investigated the behavior of γ_w as a function of $g_4 V$. (r between 5 Hz and 55 Hz, $n = 100$, 20 weight samples of dimension n, each component sampled from a uniform distribution $\mathcal{U}\,(-5/n, 5/n)$, 20 USP-samples of dimension n for each rate/weight-combination), and in last step, as a function of $c_w V$ with a constant $c_w = 0.05$.

4.4.6 Evaluation of approximated natural gradient rule

We evaluated the performance of the approximated natural-gradient rule in Eqn. S62 (with $c_\mathrm{u} = 0.95$ and $c_\mathrm{w} = 0.05$ compared to Euclidean gradient descent and the full rule in Eqn. 7 in the learning task of Fig. 3 under different input conditions (n=100, Group 1: 10 Hz/ Group 2: 30 Hz, Group 1: 10 Hz/ Group 2: 50 Hz, Group 1: 20 Hz/ Group 2: 20 Hz, Group 1: 20 Hz/ Group 2: 40 Hz). The learning curves were averaged over 1000 trials with input and target weight components randomly chosen from a uniform distribution on $\mathcal{U}\,(-1/n, 1/n)$. Learning rate parameters were tuned individually for each learning rule and scenario according to Table 1. All other parameters were the same as for Fig. 3F.

For the angle histograms in Fig. S3A-B, we simulated the natural, Euclidean and approximated natural weight updates for several input and initial weight conditions. Similar to the setup in Fig. 3 we separated the $n = 100$ input afferents in two

Table 1: Learning rates

r_1	r_2	η_n	η_a	η_e
$10\,\mathrm{Hz}$	$30\,\mathrm{Hz}$	0.000655	0.00055	0.00000110
$10\,\mathrm{Hz}$	$50\,\mathrm{Hz}$	0.000600	0.00045	0.00000045
$20\,\mathrm{Hz}$	$20\,\mathrm{Hz}$	0.000650	0.00053	0.00000118
$20\,\mathrm{Hz}$	$40\,\mathrm{Hz}$	0.000580	0.00045	0.00000055

groups firing at different rates (Group1/Group2 : $10\,\mathrm{Hz}/10\,\mathrm{Hz}, 10\,\mathrm{Hz}/30\,\mathrm{Hz}, 10\,\mathrm{Hz}/50\,\mathrm{Hz}, 20\,\mathrm{Hz}/20\,\mathrm{Hz}, 20\,\mathrm{Hz}/40\,\mathrm{Hz}$). For each input pattern, 100 Initial weight components were sampled randomly from a uniform distribution $\mathcal{U}\left(-5/n, 5/n\right)$, while the target weight was fixed at $\boldsymbol{w}^* = \left(\frac{0.15}{n}, \frac{0.15}{n}\right)^T$. For each initial weight, 100 $1\,\mathrm{s}$-long input spike trains were sampled and the average angle between the natural gradient weight update and the approximated natural gradient weight update at $t = 1\,\mathrm{s}$ was calculated. The same was done for the average angle between the natural and the Euclidean weight update.

5 Acknowledgements

This work has received funding from the European Union 7th Framework Programme under grant agreement 604102 (HBP), the Horizon 2020 Framework Programme under grant agreements 720270, 785907 and 945539 (HBP) and the SNF under the personal grant 310030L_156863 (WS) and the Sinergia grant CRSII5_180316. Calculations were performed on UBELIX (http://www.id.unibe.ch/hpc), the HPC cluster at the University of Bern. We thank Oliver Breitwieser for the figure template, and Simone Surace and João Sacramento, as well as the other members of the Senn and Petrovici groups for many insightful discussions. We are also deeply grateful to the late Robert Urbanczik for many enlightening conversations on gradient descent and its mathematical foundations. Furthermore, we wish to thank the Manfred Stärk Foundation for their ongoing support.

References

Aitchison, L. and Latham, P. E. (2014). Bayesian synaptic plasticity makes predictions about plasticity experiments in vivo. *arXiv preprint arXiv:1410.1029*.

Amari, S.-i. (1987). Differential geometrical theory of Statistics. In *Differential geometry in statistical inference*, chapter 2, pages 19–94. Institute of Mathematical Statistics, Hayward.

Amari, S.-i. (1998). Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276.

Amari, S.-i., Karakida, R., and Oizumi, M. (2019). Fisher information and natural gradient learning in random deep networks. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 694–702.

Amari, S.-i. and Nagaoka, H. (2000). *Methods of information geometry*. Translations of Mathematical Monographs 191, American Mathematical Society.

Arami, M. K., Sohya, K., Sarihi, A., Jiang, B., Yanagawa, Y., and Tsumoto, T. (2013). Reciprocal homosynaptic and heterosynaptic long-term plasticity of corticogeniculate projection neurons in layer VI of the mouse visual cortex. *Journal of Neuroscience*, 33(18):7787–7798.

Asrican, B., Lisman, J., and Otmakhov, N. (2007). Synaptic strength of individual spines correlates with bound Ca2+-calmodulin-dependent kinase II. *Journal of Neuroscience*, 27(51):14007–14011.

Bernacchia, A., Lengyel, M., and Hennequin, G. (2018). Exact natural gradient in deep linear networks and its application to the nonlinear case. In *Advances in Neural Information Processing Systems*, pages 5941–5950.

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

Cencov, N. N. (1972). Optimal decision rules and optimal inference. *American Mathematical Society, Rhode Island. Translation from Russian*.

Chen, J.-Y., Lonjers, P., Lee, C., Chistiakova, M., Volgushev, M., and Bazhenov, M. (2013). Heterosynaptic plasticity prevents runaway synaptic dynamics. *Journal of Neuroscience*, 33(40):15915–15929.

Chistiakova, M., Bannon, N. M., Chen, J.-Y., Bazhenov, M., and Volgushev, M. (2015). Homeostatic role of heterosynaptic plasticity: Models and experiments. *Frontiers in Computational Neuroscience*, 9(JULY):89.

Chistiakova, M. and Volgushev, M. (2009). Heterosynaptic plasticity in the neocortex. *Experimental Brain Research*, 199(3-4):377–390.

D'Souza, P., Liu, S. C., and Hahnloser, R. H. (2010). Perceptron learning rule derived from spike-frequency adaptation and spike-time-dependent plasticity. *Proceedings of the National Academy of Sciences of the United States of America*, 107(10):4722–4727.

Dudek, S. M. and Bear, M. F. (1992). Homosynaptic long-term depression in area CA1 of hippocampus and effects of N-methyl-D-aspartate receptor blockade. *Proceedings of the National Academy of Sciences of the United States of America*, 89(10):4363–4367.

Engert, F. and Bonhoeffer, T. (1997). Synapse specificity of long-term potentiation breaks down at short distances. *Nature*, 388(6639):279–284.

Friedrich, J., Urbanczik, R., and Senn, W. (2011). Spatio-temporal credit assignment in neuronal population learning. *PLoS Computational Biology*, 7(6):e1002092.

Gerstner, W. and Kistler, W. M. (2002). *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge University Press.

Ito, M. and Kano, M. (1982). Long-lasting depression of parallel fiber-Purkinje cell transmission induced by conjunctive stimulation of parallel fibers and climbing fibers in the cerebellar cortex. *Neuroscience Letters*, 33(3):253–258.

Jordan, J., Petrovici, M. A., Senn, W., and Sacramento, J. (2020). Conductance-based dendrites perform reliability-weighted opinion pooling. In *ACM International Conference Proceeding Series*, pages 1–3.

Kakade, S. M. (2001). A natural policy gradient. In *Advances in Neural Information Processing Systems 14*, pages 1531–1538.

Loewenstein, Y., Kuras, A., and Rumpel, S. (2011). Multiplicative dynamics underlie the emergence of the log-normal distribution of spine sizes in the neocortex in vivo. *Journal of Neuroscience*, 31(26):9481–9488.

Lynch, G. S., Dunwiddie, T., and Gribkoff, V. (1977). Heterosynaptic depression: A postsynaptic correlate of long-term potentiation. *Nature*, 266(5604):737–739.

Magee, J. C. and Cook, E. P. (2000). Somatic EPSP amplitude is independent of synapse location in hippocampal pyramidal neurons. *Nature Neuroscience*, 3(9):895–903.

Marceau-Caron, G. and Ollivier, Y. (2017). Natural Langevin dynamics for neural networks. In *International Conference on Geometric Science of Information*, pages 451–459, Cham. Springer Verlag.

Markram, H., Toledo-Rodriguez, M., Wang, Y., Gupta, A., Silberberg, G., and Wu, C. (2004). Interneurons of the neocortical inhibitory system. *Nature Reviews Neuroscience*, 5(10):793–807.

Martens, J. (2014). New insights and perspectives on the natural gradient method. *arXiv:1412.1193[Preprint]*.

Ollivier, Y. (2015). Riemannian metrics for neural networks I: Feedforward networks. *Information and Inference: A Journal of the IMA*, 4(2):108–153.

Park, H., Amari, S.-i., and Fukumizu, K. (2000). Adaptive natural gradient learning algorithms for various stochastic models. *Neural Networks*, 13(7):755–764.

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

Petrovici, M. A. (2016). *Form versus function : Theory and models for neuronal substrates*. Springer.

Pfister, J.-P., Toyoizumi, T., Barber, D., and Gerstner, W. (2006). Optimal spike-timing-dependent plasticity for precise action potential firing in supervised learning. *Neural Computation*, 18(6):1318–1348.

Rao, C. R. (1945). Information and the accuracy attainable in the estimation of statistical parameters. *Bull. Calcutta Math. Soc.*, 37:81–91.

Rattray, M. and Saad, D. (1999). Analysis of natural gradient descent for multilayer neural networks. *Physical Review E*, 59(4):4523–4532.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386.

Royer, S. and Paré, D. (2003). Conservation of total synaptic weight through balanced synaptic depression and potentiation. *Nature*, 422(6931):518–522.

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

Sacramento, J., Costa, R. P., Bengio, Y., and Senn, W. (2017). Dendritic error backpropagation in deep cortical microcircuits. *arXiv:1801.00062*.

Schiess, M., Urbanczik, R., and Senn, W. (2016). Somato-dendritic synaptic plasticity and error-backpropagation in active dendrites. *PLoS Comput Biol*, 12(2):e1004638.

Softky, W. R. and Koch, C. (1993). The highly irregular firing of cortical cells is inconsistent with temporal integration of random EPSPs. *The Journal of Neuroscience*, 13(1):334–350.

Sporea, I. and Grüning, A. (2013). Supervised learning in multilayer spiking neural networks. *Neural Computation*, 25(2):473–509.

Stuart, G., Spruston, N., Sakmann, B., and Häusser, M. (1997). Action potential initiation and back propagation in neurons of the mammalian central nervous system. *Trends in Neurosciences*, 20(3):125–131.

Surace, S. C., Pfister, J.-P., Gerstner, W., and Brea, J. (2020). On the choice of metric in gradient-based theories of brain function. *PLoS Computational Biology*, 16(4):e1007640.

Urbanczik, R. and Senn, W. (2014). Learning by the dendritic prediction of somatic spiking. *Neuron*, 81(3):521–528.

White, G., Levy, W. B., and Steward, O. (1990). Spatial overlap between populations of synapses determines the extent of their associative interaction during the induction of long-term potentiation and depression. *Journal of Neurophysiology*, 64(4):1186–1198.

Williams, S. R. and Stuart, G. J. (2002). Dependence of EPSP efficacy on synapse location in neocortical pyramidal neurons. *Science*, 295(5561):1907–1910.

Wöhrl, R., Eisenach, S., Manahan-Vaughan, D., Heinemann, U., and Von Haebler, D. (2007). Acute and long-term effects of MK-801 on direct cortical input evoked homosynaptic and heterosynaptic plasticity in the CA1 region of the female rat. *European Journal of Neuroscience*, 26(10):2873–2883.

Yang, H. H. and Amari, S.-i. (1998). Complexity issues in natural gradient descent method for training multilayer perceptrons. *Neural Computation*, 10(8):2137–2157.

Zenke, F. and Gerstner, W. (2017). Hebbian plasticity requires compensatory processes on multiple timescales. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1715):20160259.

Supplementary Information

S.1 Detailed derivation of learning rule

Here, we summarize the mathematical derivations underlying our natural-gradient learning rule (Eqn. 7). While all derivations in Section S.1 and Section S.2 are made for the somatic paramterization and can then be extended to other weight coordinates as described in Section 4.3, we drop the index s in $\boldsymbol{w}^{\mathrm{s}}$ for the sake of readability.

Supervised learning requires the neuron to adapt its synapses in such a way that its input-output distribution approaches a given target distribution with density p^*. For a given input spike pattern $\boldsymbol{x}$, at each point in time, the probability for a Poisson neuron to fire a spike during the interval $[t, t + \mathrm{d}t]$ (denoted as $y_t = 1$) follows a Bernoulli distribution with a parameter $\phi_t \mathrm{d}t = \phi(V_t)\,\mathrm{d}t$, depending on the current membrane potential. The probability density of the binary variable y_t on $\{0, 1\}$, describing whether or not a spike occurred in the interval $[t, t + \mathrm{d}t]$, is therefore given by

$$p_{\boldsymbol{w}}(y_t|\boldsymbol{x}_t^\epsilon) = (\phi_t \mathrm{d}t)^{y_t} (1 - \phi_t \mathrm{d}t)^{(1-y_t)} \ , \tag{S1}$$

and we have

$$p_{\boldsymbol{w}}(y_t, \boldsymbol{x}_t^\epsilon) = (\phi_t \mathrm{d}t)^{y_t} (1 - \phi_t \mathrm{d}t)^{(1-y_t)} p_{\mathrm{usp}}(\boldsymbol{x}_t^\epsilon) \ , \tag{S2}$$

where p_{usp} denotes the probability density of the unweighted synaptic potentials $\boldsymbol{x}_t^\epsilon$. Measuring the distance to the target distribution in terms of the Kullback-Leibler divergence, we arrive at

$$C(\boldsymbol{w}) = D_{\mathrm{KL}}(p^* \| p_{\boldsymbol{w}}) = \mathbb{E}\left[\log\left[\frac{p^*(y_t, \boldsymbol{x}_t^\epsilon)}{p_{\boldsymbol{w}}(y_t, \boldsymbol{x}_t^\epsilon)}\right]\right]_{p^*} = \mathbb{E}\left[\log\left[\frac{p^*(y_t|\boldsymbol{x}_t^\epsilon)}{p_{\boldsymbol{w}}(y_t|\boldsymbol{x}_t^\epsilon)}\right]\right]_{p^*} \ . \tag{S3}$$

Since the target distribution does not depend on the synaptic weights, the negative Euclidean gradient of the D_{KL} equals

$$-\nabla_{\boldsymbol{w}}^e C = -\frac{\partial C}{\partial \boldsymbol{w}} = \mathbb{E}\left[\frac{\partial}{\partial \boldsymbol{w}} \log\left[p_{\boldsymbol{w}}(y_t|\boldsymbol{x}_t^\epsilon)\right]\right]_{p^*} \ . \tag{S4}$$

We may then calculate

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{w}} \log p_{\boldsymbol{w}}(y_t|\boldsymbol{x}_t^\epsilon) &= \frac{\partial}{\partial \boldsymbol{w}}\left[y_t \log(\phi_t \mathrm{d}t) + (1 - y_t)\log(1 - \phi_t \mathrm{d}t)\right] && \text{(S5)}\\
&= \left(\frac{y_t}{\phi_t \mathrm{d}t} - \frac{1 - y_t}{1 - \phi_t \mathrm{d}t}\right)\phi_t' \mathrm{d}t \boldsymbol{x}_t^\epsilon && \text{(S6)}\\
&= (y_t - \phi_t \mathrm{d}t)\frac{\phi_t' \mathrm{d}t}{\phi_t \mathrm{d}t(1 - \phi_t \mathrm{d}t)}\boldsymbol{x}_t^\epsilon && \text{(S7)}\\
&\approx (y_t - \phi_t \mathrm{d}t)\frac{\phi_t'}{\phi_t}\boldsymbol{x}_t^\epsilon \ , && \text{(S8)}
\end{aligned}$$

where Eqn. S7 follows from the fact that $y_t \in \{0, 1\}$ and for Eqn. S8 we neglected the term of order $\mathrm{d}t^2$ which is small compared to the remainder. Plugging Eqn. S7 into Eqn. S4 leads to the Euclidean-gradient descent online learning rule, given by

$$\dot{\boldsymbol{w}}_e = \left[Y_t^* - \phi(V_t)\right]\frac{\phi_t'}{\phi_t}\boldsymbol{x}_t^\epsilon \ . \tag{S9}$$

Here, $Y_t^* = \sum_f \delta\left(t - t^f\right)$ is the teacher spike train.

We obtain the negative natural gradient by multiplying Eqn. S9 with the inverse Fisher information matrix, since

$$\nabla_{\boldsymbol{w}}^n C = G(\boldsymbol{w})^{-1} \nabla_{\boldsymbol{w}}^e C \ , \tag{S10}$$

with the Fisher information matrix $G(\boldsymbol{w})$ at $\boldsymbol{w}$ being defined as

$$G(\boldsymbol{w}) = \mathbb{E}\left[\frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}} \frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}}^T\right]_{p_{\boldsymbol{w}}} \ . \tag{S11}$$

Exploiting that

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{w}} \log p_{\boldsymbol{w}}(y_t, \boldsymbol{x}_t^\epsilon) &= \frac{\partial}{\partial \boldsymbol{w}}\left[\log p_{\boldsymbol{w}}(y_t|\boldsymbol{x}_t^\epsilon) + \log p_{\mathrm{usp}}(\boldsymbol{x}_t^\epsilon)\right] && \text{(S12)}\\
&= \frac{\partial}{\partial \boldsymbol{w}} \log p_{\boldsymbol{w}}(y_t|\boldsymbol{x}_t^\epsilon) \ , && \text{(S13)}
\end{aligned}$$

since p_{usp} does not depend on $\boldsymbol{w}$, we can insert the previously derived formula Eqn. S8 for the partial derivative of the log-likelihood. Hence, using the tower property for expectation values and the definition of $p_{\boldsymbol{w}}$ (Eqn. S1, Eqn. S2), Eqn. S11 transforms to

$$G(\boldsymbol{w}) = \mathbb{E}\left[(y_t - \phi_t \mathrm{d}t)^2 \frac{\phi_t'^2}{\phi_t^2} \boldsymbol{x}_t^{\epsilon} \boldsymbol{x}_t^{\epsilon T}\right]_{p_{\boldsymbol{w}}} \tag{S14}$$

$$= \mathbb{E}\left[\mathbb{E}\left[(y_t - \phi_t \mathrm{d}t)^2 \frac{\phi_t'^2}{\phi_t^2}\right]_{p_{\boldsymbol{w}}(y|\boldsymbol{x}_t^{\epsilon})} \boldsymbol{x}_t^{\epsilon} \boldsymbol{x}_t^{\epsilon T}\right]_{p_{\mathrm{usp}}} \tag{S15}$$

$$= \mathbb{E}\left[\left(\phi_t \mathrm{d}t \left(1 - \phi_t \mathrm{d}t\right)^2 + \left(1 - \phi_t \mathrm{d}t\right)\left(\phi_t \mathrm{d}t\right)^2\right) \frac{\phi_t'^2}{\phi_t^2} \boldsymbol{x}_t^{\epsilon} \boldsymbol{x}_t^{\epsilon T}\right]_{p_{\mathrm{usp}}} \tag{S16}$$

$$= \mathbb{E}\left[\left(\phi_t \mathrm{d}t - \left(\phi_t \mathrm{d}t\right)^2\right) \frac{\phi_t'^2}{\phi_t^2} \boldsymbol{x}_t^{\epsilon} \boldsymbol{x}_t^{\epsilon T}\right]_{p_{\mathrm{usp}}} \tag{S17}$$

$$\approx \mathbb{E}\left[\mathrm{d}t \frac{\phi_t'^2}{\phi_t} \boldsymbol{x}_t^{\epsilon} \boldsymbol{x}_t^{\epsilon T}\right]_{p_{\mathrm{usp}}} . \tag{S18}$$

In order to arrive at an explicit expression for the natural gradient learning rule, we further decompose the Fisher information matrix, which will then enable us to find a closed expression for its inverse.

Inspired by the approach in Amari (1998), we exploit the fact that a positive semi-definite matrix is uniquely defined by its values as a bivariate form on any basis of $\mathbb{R}^n$. Choosing a basis for which the bilinear products with $G(\boldsymbol{w})$ are of a particularly simple form, we are able to decompose the Fisher Information Matrix by constructing a sum of matrices whose values as a bivariate form on the basis equal are equal to those of $G(\boldsymbol{w})$. Due to the structure of this particular decomposition, we may then apply well-known formulas for matrix inversion to obtain $G(\boldsymbol{w})^{-1}$.

Consider the basis $\mathcal{B} = \{\boldsymbol{w}, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_{n-1}\}$ such that the vectors $\sqrt{\Sigma_{\mathrm{usp}}}\boldsymbol{w}, \sqrt{\Sigma_{\mathrm{usp}}}\boldsymbol{b}_1, \ldots, \sqrt{\Sigma_{\mathrm{usp}}}\boldsymbol{b}_{n-1}$ are orthogonal to each other. Here, Σ_{usp} denotes the covariance matrix of the USPs which in the case of independent Poisson input spike trains is given as $\Sigma_{\mathrm{usp}} = \mathrm{diag}(c_{\epsilon}^{-1} r_i)$. In this case the matrix square root reduces to the component-wise square root. Note that for any $\boldsymbol{b}, \boldsymbol{b}' \in \mathcal{B}$ with $\boldsymbol{b} \neq \boldsymbol{b}'$, the random variables $\boldsymbol{b}^T \boldsymbol{x}_t^{\epsilon}$ and $\boldsymbol{b}'^T \boldsymbol{x}_t^{\epsilon}$ are uncorrelated, since

$$\mathrm{Cov}\left(\boldsymbol{b}^T \boldsymbol{x}_t^{\epsilon}, \boldsymbol{b}'^T \boldsymbol{x}_t^{\epsilon}\right) = \mathrm{Cov}\left(\sum_{j=1}^{n} b_j \boldsymbol{x}_{t,j}^{\epsilon}, \sum_{k=1}^{n} b_k' \boldsymbol{x}_{t,k}^{\epsilon}\right) \tag{S19}$$

$$= \sum_{j=1}^{n}\sum_{k=1}^{n} b_j b_k' \mathrm{Cov}\left(\boldsymbol{x}_{t,j}^{\epsilon}, \boldsymbol{x}_{t,k}^{\epsilon}\right) \tag{S20}$$

$$= \boldsymbol{b}^T \Sigma_{\mathrm{usp}} \boldsymbol{b}' = 0 . \tag{S21}$$

We make the mild assumptions of having small afferent populations firing at the same input rate, and that the basis $\mathcal{B}$ is constructed in such way that the basis vectors are not too close to the coordinate axes, such that the products $\boldsymbol{b}^T \boldsymbol{x}_t^{\epsilon}$ are not dominated by a single component. Then, for sufficiently large n, every linear combination of the random variables $\boldsymbol{b}^T \boldsymbol{x}_t^{\epsilon}$ and $\boldsymbol{b}'^T \boldsymbol{x}_t^{\epsilon}$ is approximately normally distributed, thus, the two random variables follow a joint bivariate normal distribution. Furthermore, uncorrelated random variables that are jointly normally distributed are independent. Since functions of independent random variables are also independent, this allows us to calculate all products of the form $\boldsymbol{b}^T G(\boldsymbol{w}) \boldsymbol{b}'$ for $\boldsymbol{b}, \boldsymbol{b}' \in \mathcal{B} \setminus \{\boldsymbol{w}\}$, as we can transform the expectation of products

$$\boldsymbol{b}^T G(\boldsymbol{w}) \boldsymbol{b}' = \mathbb{E}\left[\mathrm{d}t \frac{\phi'^2[V(t)]}{\phi[V(t)]} \left(\boldsymbol{b}^T \boldsymbol{x}_t^{\epsilon}\right)\left(\boldsymbol{x}_t^{\epsilon T} \boldsymbol{b}\right)\right]_{p_{\mathrm{usp}}} \tag{S22}$$

into products of expectations. Taking into account that $V(t) = \boldsymbol{w}^T \boldsymbol{x}_t^{\epsilon}$ and $\mathbb{E}\left[\boldsymbol{x}^{\epsilon}\right] = \boldsymbol{r}$, we arrive at

$$\boldsymbol{b}^T G(\boldsymbol{w}) \boldsymbol{b}' = I_1(\boldsymbol{w}) \left(\epsilon_0 \boldsymbol{r}^T \boldsymbol{b}\right)\left(\epsilon_0 \boldsymbol{r}^T \boldsymbol{b}'\right) \text{ , for } \boldsymbol{b} \neq \boldsymbol{b}' \text{ and } \boldsymbol{b}, \boldsymbol{b}' \neq \boldsymbol{w} , \tag{S23}$$

$$\boldsymbol{b}^T G(\boldsymbol{w}) \boldsymbol{b} = I_1(\boldsymbol{w}) \left(\boldsymbol{b}^T \Sigma_{\mathrm{usp}} \boldsymbol{b} + \left(\epsilon_0 \boldsymbol{r}^T \boldsymbol{b}\right)^2\right) \text{ , for } \boldsymbol{b} \neq \boldsymbol{w} , \tag{S24}$$

$$\boldsymbol{w}^T G(\boldsymbol{w}) \boldsymbol{b} = \boldsymbol{b}^T G(\boldsymbol{w}) \boldsymbol{w} = I_2(\boldsymbol{w}) \left(\epsilon_0 \boldsymbol{r}^T \boldsymbol{b}\right) \text{ , for } \boldsymbol{b} \neq \boldsymbol{w} , \tag{S25}$$

$$\boldsymbol{w}^T G(\boldsymbol{w}) \boldsymbol{w} = I_3(\boldsymbol{w}) . \tag{S26}$$

Here, I_1, I_2, I_3 denote the generalized voltage moments given as

$$I_1(\boldsymbol{w}) = \mathbb{E}\left[\frac{\phi_t'^2}{\phi_t}\right]_{p_{\mathrm{usp}}} = \frac{1}{\sqrt{2\pi\sigma_{\mathrm{v}}^2}}\int_{-\infty}^{\infty}\frac{\phi'(u)^2}{\phi(u)}\exp\frac{-(u-\mu_{\mathrm{v}})^2}{2\sigma_{\mathrm{v}}^2}du\,, \tag{S27}$$

$$I_2(\boldsymbol{w}) = \mathbb{E}\left[\frac{\phi_t'^2}{\phi_t}V_t\right]_{p_{\mathrm{usp}}} = \frac{1}{\sqrt{2\pi\sigma_{\mathrm{v}}^2}}\int_{-\infty}^{\infty}\frac{\phi'(u)^2}{\phi(u)}u\exp\frac{-(u-\mu_{\mathrm{v}})^2}{2\sigma_{\mathrm{v}}^2}du\,, \tag{S28}$$

$$I_3(\boldsymbol{w}) = \mathbb{E}\left[\frac{\phi_t'^2}{\phi_t}V_t^2\right]_{p_{\mathrm{usp}}} = \frac{1}{\sqrt{2\pi\sigma_{\mathrm{v}}^2}}\int_{-\infty}^{\infty}\frac{\phi'(u)^2}{\phi(u)}u^2\exp\frac{-(u-\mu_{\mathrm{v}})^2}{2\sigma_{\mathrm{v}}^2}du\,. \tag{S29}$$

The integral formulas follow from the fact that for a large number of input afferents, and under the mild assumption of all synaptic weights roughly being of the same order of magnitude, the membrane potential approximately follows a normal distribution with mean $\mu_{\mathrm{v}} = \epsilon_0\sum_{i=1}^n w_i r_i$ and variance $\sigma_{\mathrm{v}}^2 = \frac{1}{c_\epsilon}\sum_{i=1}^n w_i^2 r_i$.

Based on the above calculations, we construct a candidate $\tilde{G}(\boldsymbol{w})$ for a decomposition of $G(\boldsymbol{w})$. We start with the matrix $c_1\left(\epsilon_0^2\boldsymbol{r}\boldsymbol{r}^T + \Sigma_{\mathrm{usp}}\right)$, since its easy to see that

$$\boldsymbol{b}^T c_1\left(\epsilon_0^2\boldsymbol{r}\boldsymbol{r}^T + \Sigma_{\mathrm{usp}}\right)\boldsymbol{b}' = \boldsymbol{b}^T G(\boldsymbol{w})\,\boldsymbol{b}' \text{ for all } b, b' \in \mathcal{B}\,. \tag{S30}$$

Exploiting the orthogonal properties according to which we constructed $\mathcal{B}$ to carefully add more terms such that also the other identities in Eqn. S23 hold, we arrive at

$$\tilde{G}(\boldsymbol{w}) = c_1\left(\epsilon_0^2\boldsymbol{r}\boldsymbol{r}^T + \Sigma_{\mathrm{usp}}\right) + c_2\epsilon_0\left[\Sigma_{\mathrm{usp}}\boldsymbol{w}\boldsymbol{r}^T + \boldsymbol{r}\left(\Sigma_{\mathrm{usp}}\boldsymbol{w}\right)^T\right] + c_3\left(\Sigma_{\mathrm{usp}}\boldsymbol{w}\right)\left(\Sigma_{\mathrm{usp}}\boldsymbol{w}\right)^T\,. \tag{S31}$$

Here,

$$c_1 = I_1, \quad c_2 = \frac{I_2 - I_1\mu_{\mathrm{v}}}{\sigma_{\mathrm{v}}^2}, \quad c_3 = \frac{I_3 - I_1\left(\mu_{\mathrm{v}}^2 + \sigma_{\mathrm{v}}^2\right) - 2c_2\mu_{\mathrm{v}}\sigma_{\mathrm{v}}^2}{\sigma_{\mathrm{v}}^4}\,. \tag{S32}$$

To check that indeed $\tilde{G}(\boldsymbol{w}) = G(\boldsymbol{w})$, it suffices to check the values of $\tilde{G}$ as a bilinear form on the basis $\mathcal{B}$.

S.2 Inverse of the Fisher Information Matrix

As the expectation of the outer product of a vector with itself, $G(\boldsymbol{w})$ is per construction symmetric and positive semidefinite. From the previous calculations, it follows that for elements b of a basis $\mathcal{B}$ of $\mathbb{R}^n$ the products $\boldsymbol{b}^T G(\boldsymbol{w})\,\boldsymbol{b}$ are strictly positive. Hence $G(\boldsymbol{w})$ is positive definite and thus invertible. We showed that

$$G(\boldsymbol{w}) = c_1\left(\epsilon_0^2\boldsymbol{r}\boldsymbol{r}^T + \Sigma_{\mathrm{usp}}\right) + c_2\epsilon_0\left[\Sigma_{\mathrm{usp}}\,\boldsymbol{w}\boldsymbol{r}^T + \boldsymbol{r}\left(\Sigma_{\mathrm{usp}}\,\boldsymbol{w}\right)^T\right] + c_3\left(\Sigma_{\mathrm{usp}}\,\boldsymbol{w}\right)\left(\Sigma_{\mathrm{usp}}\,\boldsymbol{w}\right)^T\,.$$

We introduce the notation

$$\tilde{\boldsymbol{w}} = \Sigma_{\mathrm{usp}}\boldsymbol{w} \text{ and } \tilde{\boldsymbol{r}} = \epsilon_0\Sigma_{\mathrm{usp}}^{-1}\boldsymbol{r}\,. \tag{S33}$$

Then,

$$G(\boldsymbol{w}) = \underbrace{c_1\Sigma_{\mathrm{usp}}}_{M_1} + \underbrace{\left(c_1\epsilon_0\boldsymbol{r} + c_2\tilde{\boldsymbol{w}}\right)\epsilon_0\boldsymbol{r}^T}_{M_2} + \underbrace{\left(c_2\epsilon_0\boldsymbol{r} + c_3\tilde{\boldsymbol{w}}\right)\tilde{\boldsymbol{w}}^T}_{M_3}\,. \tag{S34}$$

For the following calculations, we will repeatedly use the identities

$$\boldsymbol{w}^T\epsilon_0\boldsymbol{r} = \mu_{\mathrm{v}}\,,\ \boldsymbol{w}^T\Sigma_{\mathrm{usp}}\boldsymbol{w} = \sigma_{\mathrm{v}}^2, \text{ and } \tilde{\boldsymbol{r}}^T\epsilon_0\boldsymbol{r} = q\,. \tag{S35}$$

By the Sherman-Morrison-Woodbury formula, the inverse of an invertible rank 1 correction $(A + \boldsymbol{u}\boldsymbol{v}^T)$ of an invertible matrix A is given by

$$\left(A + \boldsymbol{u}\boldsymbol{v}^T\right)^{-1} = A^{-1} - \frac{A^{-1}\boldsymbol{u}\boldsymbol{v}^T A^{-1}}{1 + \boldsymbol{v}^T A^{-1}\boldsymbol{u}}\,. \tag{S36}$$

Applying this to invert $G(\boldsymbol{w})$, as a first step, we consider the term $M_1 + M_2$ and identify $M_1 = A$ and $M_2 = \boldsymbol{u}\boldsymbol{v}^T$. Its inverse is given by

$$\left(M_1 + M_2\right)^{-1} = M_1^{-1} - k_1\Sigma_{\mathrm{usp}}^{-1}\left(c_1\epsilon_0\boldsymbol{r} + c_2\tilde{\boldsymbol{w}}\right)\epsilon_0\boldsymbol{r}^T\Sigma_{\mathrm{usp}}^{-1} = M_1^{-1} - k_1\left(c_1\tilde{\boldsymbol{r}} + c_2\boldsymbol{w}\right)\tilde{\boldsymbol{r}}^T\,, \tag{S37}$$

with

$$k_1 = \left\{c_1^2\left[1 + \epsilon_0\boldsymbol{r}^T M_1^{-1}\left(c_1\tilde{\boldsymbol{r}} + c_2\boldsymbol{w}\right)\right]\right\}^{-1} = c_1^{-1}\left[c_1(q+1) + c_2\mu_{\mathrm{v}}\right]^{-1}\,. \tag{S38}$$

Applying the Sherman-Morrison-Woodbury formula a second time, this time with $M_1 + M_2 = A$ and $M_3 = \boldsymbol{u}\boldsymbol{v}^T$, we obtain

$$\begin{aligned}
G(\boldsymbol{w})^{-1} &= \left[(M_1 + M_2) + M_3\right]^{-1} \\
&= (M_1 + M_2)^{-1} - k_2 (M_1 + M_2)^{-1} M_3 (M_1 + M_2)^{-1} \\
&= M_1^{-1} - k_1 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T \\
&- k_2 \left[M_1^{-1} - k_1 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T\right] M_3 \left[M_1^{-1} - k_1 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T\right] \\
&= M_1^{-1} - k_1 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T - k_2 M_1^{-1} M_3 M_1^{-1} \\
&+ k_1 k_2 M_1^{-1} M_3 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T + k_1 k_2 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T M_3 M_1^{-1} \\
&+ k_1^2 k_2 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T M_3 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T \ .
\end{aligned} \tag{S39}$$

Here,

$$k_2 = \left\{1 + \tilde{\boldsymbol{w}}^T \left[M_1^{-1} - k_1 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^{\boldsymbol{T}}\right] (c_2 \epsilon_0 \boldsymbol{r} + c_3 \tilde{\boldsymbol{w}})\right\}^{-1} \tag{S40}$$

$$= \left[1 + c_1^{-1} \left(c_2 \mu_{\mathrm{v}} + c_3 \sigma_{\mathrm{v}}^2\right) - k_1 \left(c_1 \mu_{\mathrm{v}} + c_2 \sigma_{\mathrm{v}}^2\right) (c_2 q + c_3 \mu_{\mathrm{v}})\right]^{-1} \ . \tag{S41}$$

Plugging in the definitions of M_3 and M_1, and using that

$$M_1^{-1} M_3 = c_1^{-1} (c_2 \tilde{\boldsymbol{r}} + c_3 \boldsymbol{w}) \tilde{\boldsymbol{w}}^T \ , \tag{S42}$$

we arrive at

$$G(\boldsymbol{w})^{-1} = M_1^{-1} - k_1 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T - k_2 c_1^{-1} (c_2 \tilde{\boldsymbol{r}} + c_3 \boldsymbol{w}) \tilde{\boldsymbol{w}}^T M_1^{-1} \tag{S43}$$

$$+ k_1 k_2 c_1^{-1} (c_2 \tilde{\boldsymbol{r}} + c_3 \boldsymbol{w}) \tilde{\boldsymbol{w}}^T (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T + k_1 k_2 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T (c_2 \epsilon_0 \boldsymbol{r} + c_3 \tilde{\boldsymbol{w}}) \tilde{\boldsymbol{w}}^T M_1^{-1} \tag{S44}$$

$$- k_1^2 k_2 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T (c_2 \epsilon_0 \boldsymbol{r} + c_3 \tilde{\boldsymbol{w}}) \tilde{\boldsymbol{w}}^T (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T \tag{S45}$$

$$= c_1^{-1} \Sigma_{\mathrm{usp}}^{-1} - k_1 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T \tag{S46}$$

$$- k_2 c_1^{-2} (c_2 \tilde{\boldsymbol{r}} + c_3 \boldsymbol{w}) \boldsymbol{w}^T \tag{S47}$$

$$+ k_1 k_2 c_1^{-1} (c_2 \tilde{\boldsymbol{r}} + c_3 \boldsymbol{w}) \left(c_1 \mu_{\mathrm{v}} + c_2 \sigma_{\mathrm{v}}^2\right) \tilde{\boldsymbol{r}}^T + k_1 k_2 c_1^{-1} (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) (c_2 q + c_3 \mu_{\mathrm{v}}) \boldsymbol{w}^T \tag{S48}$$

$$- k_1^2 k_2 (c_1 \tilde{\boldsymbol{r}} + c_2 \boldsymbol{w}) \left(c_1 \mu_{\mathrm{v}} + c_3 \sigma_{\mathrm{v}}^2\right) (c_2 q + c_3 \mu_{\mathrm{v}}) \tilde{\boldsymbol{r}}^T \ . \tag{S49}$$

After resorting the terms and grouping, we obtain the inverse of the Fisher Information Matrix as

$$G(\boldsymbol{w})^{-1} = \frac{1}{c_1} \left[\Sigma_{\mathrm{usp}}^{-1} + (g_1 \tilde{\boldsymbol{r}} + g_2 \boldsymbol{w}) \tilde{\boldsymbol{r}}^T + (g_3 \tilde{\boldsymbol{r}} + g_4 \boldsymbol{w}) \boldsymbol{w}^T\right] \tag{S50}$$

$$= \gamma_{\mathrm{s}} \left[\Sigma_{\mathrm{usp}}^{-1} + (c_\epsilon \epsilon_0 g_1 \mathbf{1} + g_2 \boldsymbol{w}) c_\epsilon \epsilon_0 \mathbf{1}^T + (c_\epsilon \epsilon_0 g_3 \mathbf{1} + g_4 \boldsymbol{w}) \boldsymbol{w}^T\right] \ , \tag{S51}$$

with

$$g_1 = c_1 \left\{-k_1 c_1 + k_1 k_2 c_1^{-1} c_2 \left(c_1 \mu_{\mathrm{v}} + c_2 \sigma_{\mathrm{v}}^2\right) - k_1^2 k_2 c_1 \left[\left(c_1 \mu_{\mathrm{v}} + c_2 \sigma_{\mathrm{v}}^2\right) (c_2 q + c_3 \mu_{\mathrm{v}})\right]\right\} \tag{S52}$$

$$g_2 = c_1 \left\{-k_1 c_2 + c_1^{-1} k_1 k_2 c_3 \left(c_1 \mu_{\mathrm{v}} + c_2 \sigma_{\mathrm{v}}^2\right) - k_1^2 k_2 c_2 \left[\left(c_1 \mu_{\mathrm{v}} + c_2 \sigma_{\mathrm{v}}^2\right) (c_2 q + c_3 \mu_{\mathrm{v}})\right]\right\} \tag{S53}$$

$$g_3 = c_1 \left[k_1 k_2 (c_2 q + c_3 \mu_{\mathrm{v}}) - k_2 c_1^{-2} c_2\right] \tag{S54}$$

$$g_4 = c_1 \left[k_1 k_2 c_1^{-1} c_2 (c_2 q + c_3 \mu_{\mathrm{v}}) - k_2 c_1^{-2} c_3\right] \ . \tag{S55}$$

S.3 Analysis of the learning rule coefficients

In order to gain an intuitive understanding of Eqn. 7 and to judge its suitability as an in-vivo plasticity rule, we require insights into the behavior of the coefficients $\gamma_{\mathrm{s}}, \gamma_{\mathrm{u}}$, and γ_{w} under various circumstances.

S.3.1 Global scaling factor

The global scaling factor γ_{s} is given as

$$\gamma_{\mathrm{s}} = c_1^{-1} = \mathbb{E}\left[\frac{\phi'^2_t}{\phi_t}\right]_{p_{\mathrm{usp}}}^{-1} \ . \tag{S56}$$

The above formula reveals that γ_s is closely tied to the firing nonlinearity of the neuron, as well as the statistics of output. The scaling by the inverse slope of the output nonlinearity amplifies the synaptic update in regions where a small change in weight and thus in the membrane potential would not lead to a noticeable change in output distribution. A further scaling with ϕ additionally amplifies the synaptic change in high-output regimes (Fig. S1). This is in line with the spirit of natural gradient that ensures that the size of the synaptic weight update is homogeneous in terms of D_KL change rather than in absolute weight terms. Furthermore, the rescaling is based on the average output statistics, which the synapse might have access to via backpropagating action potentials (Stuart et al., 1997), rather than an instantaneous value.

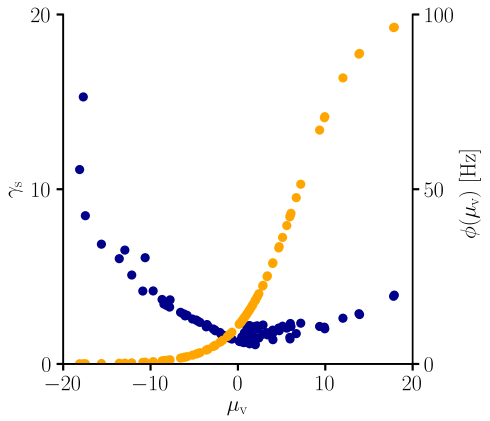

Figure S1: **Global learning rate scaling γ_s as a function of the mean membrane potential.** We sampled the global learning rate factor γ_s (blue) for various conditions. In line with Eqn. S56, γ_s is boosted in regions where the transfer function is flat, i.e., $\phi'(V)$ is small. The global scaling factor is additionally increased in regions where the transfer function reaches high absolute values.

S.3.2 Empirical analysis of γ_u and γ_w

While the formula for γ_s provides a rather good intuition of this coefficients' behavior, from the derivations in the previous sections, it becomes clear that such a straightforward interpretation is not readily available from the formulas for γ_u and γ_w. Defined as

$$\gamma_\mathrm{u} = -c_\epsilon \epsilon_0 \left(c_\epsilon \epsilon_0 g_1 \sum_{i=1}^{n} x_i^\epsilon + g_3 V \right) \tag{S57}$$

and

$$\gamma_\mathrm{w} = c_\epsilon \epsilon_0 g_2 \sum_{i=1}^{n} x_i^\epsilon + g_4 V \ , \tag{S58}$$

where $g_1, \dots g_4$ are given in Eqn. S52, these coefficients depend, apart from the membrane potential and its first and second moments, on the total input and its mean the total rate. This raises the question whether the synapse, which in general only has limited access to global quantities, can implement γ_u and γ_w. We therefore used an empirical analysis through simulations to obtain a more detailed insight.

As a starting point, we sampled $g_1, \dots, g_4$ for various input conditions (Fig. S2A-D, refer to Section 4.4.5 for simulation details). Here, we varied the afferent input rate r between 5 Hz and 55 Hz for $n = 100$ neurons and evaluated the value of the respective coefficient for a randomly sampled input weight. In a second simulation (Fig. S2E-H, Section 4.4.5), we varied the number n of afferent inputs between 10 and 200 neurons for fixed input rate 20 Hz, again with randomly chosen input weights. This revealed an approximately inverse proportional relationship between the average input rate r and g_1, g_2 and g_3 respectively. Furthermore, these coefficients seemed to be approximately inversely proportional to the number n of afferent inputs. However, this was not true for g_4, whose mean seemed to stay approximately constants across input rates, although the scattering across the mean value (for different weight samples) increased.

While the value of the membrane potential stays bounded also for a large number of inputs (due to the normalization of the synaptic weight range with $\frac{1}{n}$), the total sum of USPs increases with an increasing number of inputs. Therefore, for large enough n, the term $g_3 V$ can be neglected, so $\gamma_\mathrm{u} \approx c_\epsilon^2 \epsilon_0^2 g_1 \sum_{i=1}^n x_i^\epsilon$. A closer look at the behavior of g_1 shows that, for a sufficiently large number of neurons or high input rates, $g_1 \approx -q^{-1}$, which we verified by simulations (Fig. S2I, Section 4.4.5). In consequence,

$$\gamma_\mathrm{u} \approx s = \frac{c_\epsilon \sum_{i=1}^n x_i^\epsilon}{\sum_{i=1}^n r_i} \ . \tag{S59}$$

To verify this approximation, we sampled the values of γ_u for various conditions (Fig. S2J, Section 4.4.5) and compared them against the approximation in Eqn. S59, which confirmed our approximation. Since the input rate is the mean value of the USP, assuming large enough populations with the same input rate and a sufficient number of input afferents, by the central limit theorem we have

$$\gamma_\mathrm{u} \longrightarrow c_\mathrm{u} c_\epsilon \text{ for } n \to \infty \ , \tag{S60}$$

with $c_\mathrm{u} = \epsilon_0$. However, in practice, for $\epsilon_0 = 1\,\mathrm{mV\,ms}$, a learning behavior much closer to natural gradient was obtained when c_u was slightly smaller than 1 such as $c_\mathrm{u} = 0.95$ (cf. Section S.4).

As a starting point to approximate γ_w, we noticed that the mean of g_4 stayed approximately constant when varying the input rate or the number of input afferents. On the other hand, g_2 rapidly tends to zero in those cases, so we assumed that $g_2 \sum_{i=1}^n x_i^\epsilon$ stays either constant or goes to zero in the limit of large n. Since $c_\epsilon \epsilon_0 g_2$ seemed to be rather small

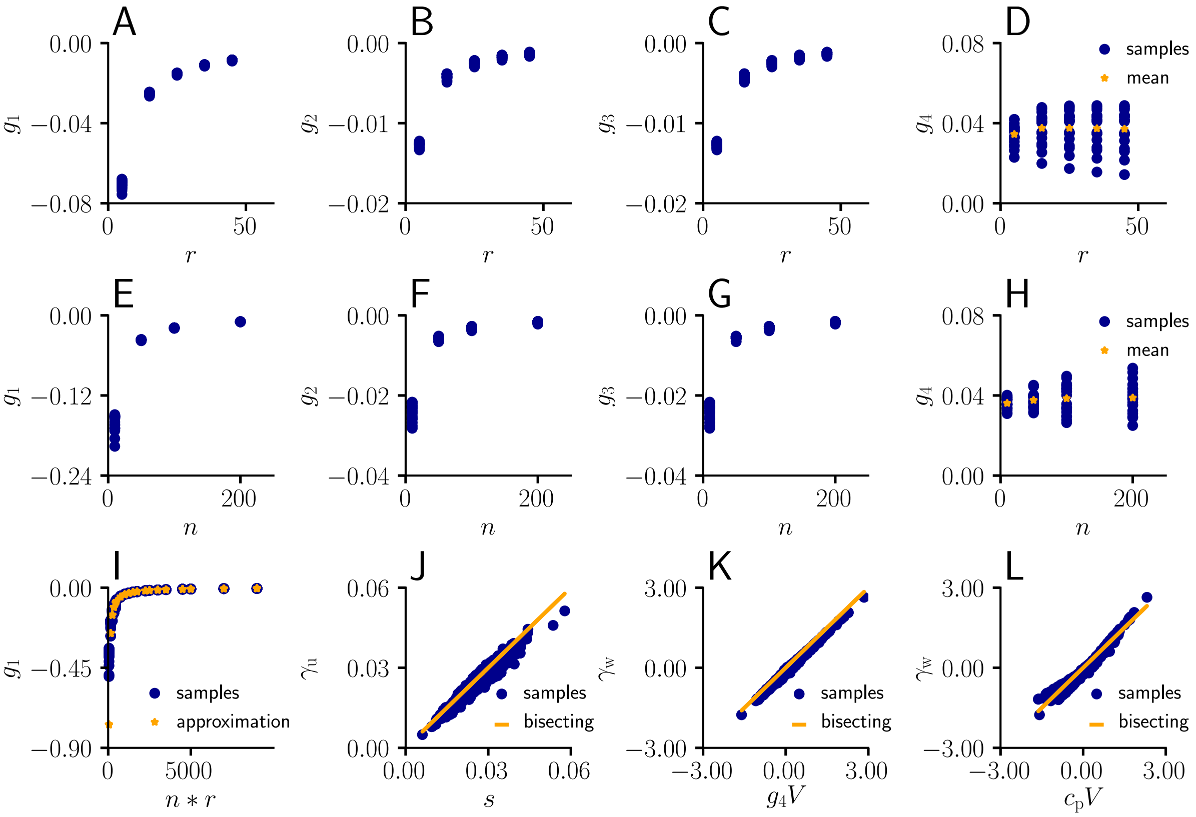

Figure S2: **Learning rule coefficients can be approximated by simpler quantities. (A)-(D)** Samples values for $g_1, \ldots, g_4$ for different afferent input rates. **(E)-(H)** In a second simulation, we varied the number n of afferent inputs. **(I)** Comparison of the sampled values of g_1 (blue) as a function of the total input rate $n * r$ to the values of the approximation given by $g_1 \approx -q^{-1}$. **(J)** Sampled values of γ_u (blue) as a function of the approximation s (Eqn. S59). The proximity of the sampled values to the diagonal indicates that s may indeed serve as an approximation for γ_u. **(K)** Sampled values of γ_w (blue) as a function of $g_4 V$. The proximity of the sampled values to the diagonal indicates that $g_4 V$ serves as an approximation for γ_w. **(L)** Same as **(K)**, but with g_4 replaced by a constant $c_w = 0.05$.

compared to g_4, we hypothesized $\gamma_\mathrm{w} \approx g_4 V$, which was confirmed by simulations (Fig. S2K, Section 4.4.5). As a second step (Fig. S2L, Section 4.4.5), since g_4 seemed to be constant in the mean, we approximated

$$\gamma_\mathrm{w} \approx c_\mathrm{w} V \, , \tag{S61}$$

where simulations with $c_\mathrm{w} = 0.05$, close to the mean of g_4 showed a learning behavior close to natural gradient learning (Fig. S3C-F). Replacing γ_u and γ_w in Eqn. 7 by the expressions in Eqn. S60 and Eqn. S61, we obtain the approximated natural gradient rule

$$\dot{\boldsymbol{w}}_\mathrm{a} = \eta \, \gamma_\mathrm{s} \left[Y^* - \phi(V) \right] \frac{\phi'(V)}{\phi(V)} f'\left(\boldsymbol{w}\right)^{-1} \left[\frac{c_\epsilon \boldsymbol{x}^\epsilon}{\boldsymbol{r}} - c_\epsilon c_\mathrm{u} + c_\mathrm{w} V f\left(\boldsymbol{w}\right) \right] . \tag{S62}$$

S.3.3 Performance of the approximated learning rule

Simulations of natural, Euclidean and approximated natural gradient weight updates for several input patterns and randomly sampled initial conditions (Section 4.4.6) showed that the average angles (both in the Euclidean metric and in the Fisher metric) between the true and approximated natural gradient weight update were small compared to the average angle between Euclidean and natural gradient weight update (Fig. S3A-B). This was confirmed by the learning curves for several tested input conditions in the setting of Fig. 3, since the performance of the approximation lay in between the natural and the Euclidean gradient's performance (Fig. S3C-F, simulation details: Section 4.4.6). It can hence be regarded as a trade-off between optimal learning speed, parameter invariance and biological implementability.

Note that plugging the above calculations into Eqn. 7 still keeps invariance under coordinate-wise smooth parameter changes.

S.4 Quadratic transfer function

While for all our simulations, we used the sigmoidal transfer function given in Eqn. 15, the derivations that lead to Eqn. S50 also hold for other choices of transfer function. A particularly simple and thereby instructive result can be obtained for the choice of a rectified quadratic transfer function

$$\phi\left(V\right) = \frac{1}{4}\left(V - \theta\right)^2 \Theta(V) \,, \tag{S63}$$

where Θ is the Heaviside step function. In this case, we have

$$\left(\phi'\right)^2 = \phi \,, \tag{S64}$$

so Eqn. S18 reduces to

$$G\left(\boldsymbol{w}\right) \approx \mathbb{E}\left(\mathrm{d}t \frac{\phi_t'^2}{\phi_t} \boldsymbol{x}_t^\epsilon \boldsymbol{x}_t^{\epsilon T}\right)_{p_{\mathrm{usp}}} \,, \tag{S65}$$

where the right side no longer depends on $\boldsymbol{w}$, thus yielding $\gamma_\mathrm{w} = 0$. Furthermore, we have

$$I_1 = 1 \,, \tag{S66}$$

$$I_2 = \mu_\mathrm{v} \,, \tag{S67}$$

$$I_3 = \mu_\mathrm{v}^2 + \sigma_\mathrm{v}^2 \,, \tag{S68}$$

and therefore $c_2 = c_3 = 0$. Plugging this into Eqn. S31, we obtain

$$G\left(\boldsymbol{w}\right) = \epsilon_0^2 \boldsymbol{r}\boldsymbol{r}^T + \Sigma_{\mathrm{usp}} \,. \tag{S69}$$

Inverting this using the Sherman-Morrison-Woodbury Formula, we arrive at

$$G\left(\boldsymbol{w}\right)^{-1} = \Sigma_{\mathrm{usp}}^{-1} - \frac{c_\epsilon^2 \epsilon_0^2}{q+1} \mathbf{1}\mathbf{1}^T \,. \tag{S70}$$

Inserting this version of the Fisher information matrix into Eqn. 6, we obtain a simplified version of the natural gradient learning rule as

$$\dot{\boldsymbol{w}} = \eta\left[Y^* - \phi(V)\right] \frac{1}{\phi'(V)} f'\left(\boldsymbol{w}\right)^{-1} \left(\frac{c_\epsilon \boldsymbol{x}^\epsilon}{\boldsymbol{r}} - \gamma_\mathrm{u} \mathbf{1}\right) \,, \tag{S71}$$

with

$$\gamma_\mathrm{u} = \frac{c_\epsilon^2 \epsilon_0^2 \sum_i x_i^\epsilon}{(q+1)} = \frac{c_\epsilon^2 \epsilon_0^2 \sum_i x_i^\epsilon}{\left(c_\epsilon \epsilon_0^2 \sum_i r_i + 1\right)} \,. \tag{S72}$$

This is in line with our empirical approximation for γ_u for the case of the sigmoidal transfer function (Eqn. S59). Note that the global scaling factor reduces to $\gamma_\mathrm{s} = 1$, which is a consequence of the balance between the slope and the absolute value of the neuronal transfer function. In summary, since for the quadratic transfer function Eqn. S63 we have $\gamma_\mathrm{s} = 1$ and $\gamma_\mathrm{w} = 0$, Eqn. 7 reduces to Eqn. S71.

S.5 Continuous-Time Limit

Under the Poisson-process assumption, firing a spike in one of the time bins is independent from the spikes in other bins, therefore the probability for firing in two disjunct time intervals $[t_1 + \mathrm{d}t]$ and $[t_2 + \mathrm{d}t]$ is given as

$$p_{\boldsymbol{w}}\left(y_{t_1}, y_{t_2} | \boldsymbol{x}_{t_1}^\epsilon, \boldsymbol{x}_{t_1}^\epsilon\right) = p_{\boldsymbol{w}}\left(y_{t_1} | \boldsymbol{x}_{t_1}^\epsilon\right) p_{\boldsymbol{w}}\left(y_{t_2} | \boldsymbol{x}_{t_1}^\epsilon\right) \,. \tag{S73}$$

Since $\mathbb{E}\left[\frac{\partial \log p_{\boldsymbol{w}}}{\partial \boldsymbol{w}}\right]_{p_{\boldsymbol{w}}} = 0$, we have

$$\mathbb{E}\left[\frac{\partial \log p_{\boldsymbol{w}}\left(y_{t_1}, y_{t_2} | \boldsymbol{x}_{t_1}^\epsilon, \boldsymbol{x}_{t_2}^\epsilon\right)}{\partial \boldsymbol{w}} \frac{\partial \log p_{\boldsymbol{w}}\left(y_{t_1}, y_{t_2} | \boldsymbol{x}_{t_1}^\epsilon, \boldsymbol{x}_{t_2}^\epsilon\right)}{\partial \boldsymbol{w}}^T\right]_{p_{\boldsymbol{w}}} \tag{S74}$$

$$= \mathbb{E}\left[\frac{\partial \log p_{\boldsymbol{w}}\left(y_{t_1} | \boldsymbol{x}_{t_1}^\epsilon\right)}{\partial \boldsymbol{w}} \frac{\partial \log p_{\boldsymbol{w}}\left(y_{t_1} | \boldsymbol{x}_{t_1}^\epsilon\right)}{\partial \boldsymbol{w}}^T\right]_{p_{\boldsymbol{w}}} + \mathbb{E}\left[\frac{\partial \log p_{\boldsymbol{w}}\left(y_{t_2} | \boldsymbol{x}_{t_2}^\epsilon\right)}{\partial \boldsymbol{w}} \frac{\partial \log p_{\boldsymbol{w}}\left(y_{t_2} | \boldsymbol{x}_{t_2}^\epsilon\right)}{\partial \boldsymbol{w}}^T\right]_{p_{\boldsymbol{w}}} \,, \tag{S75}$$

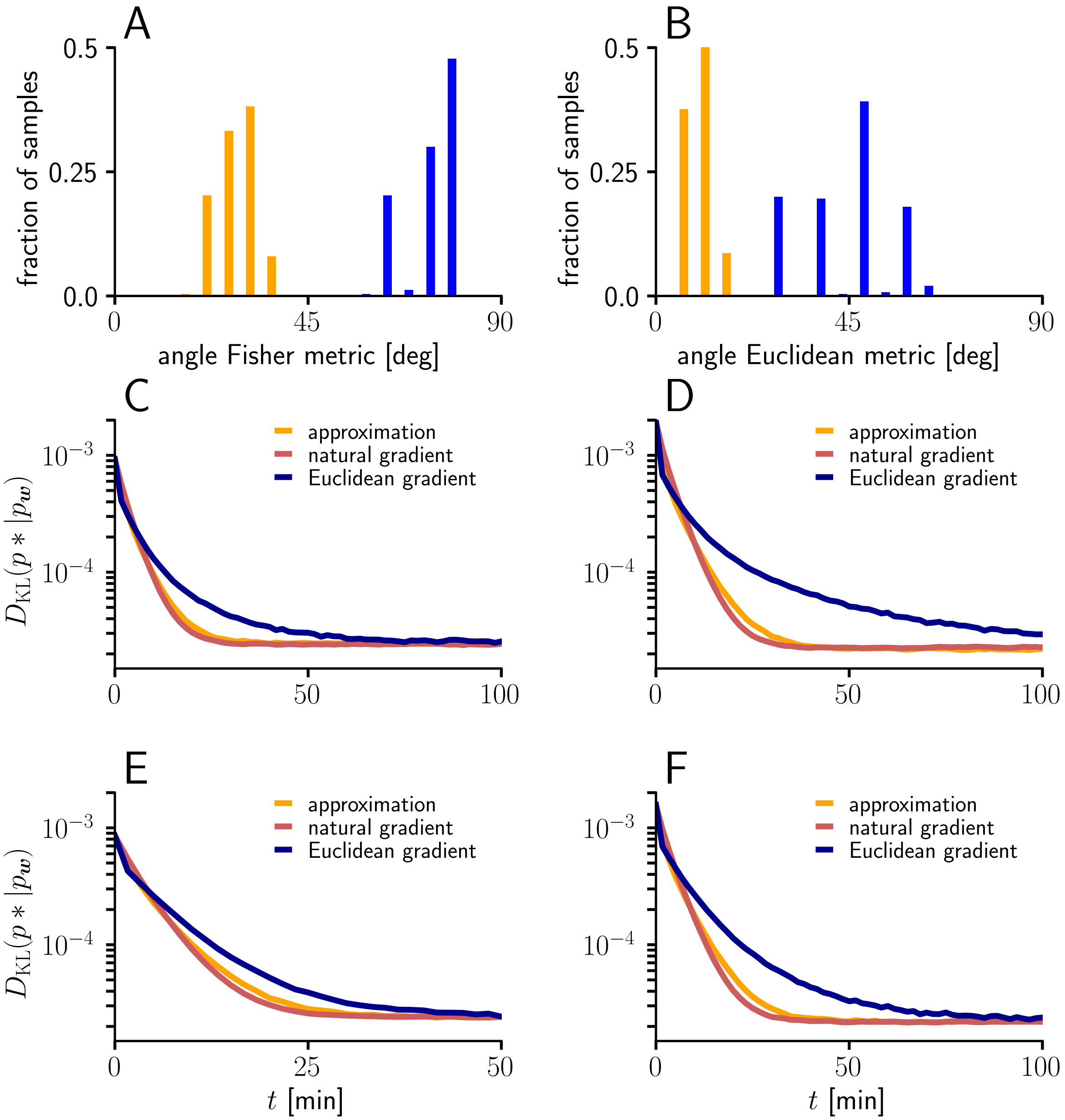

Figure S3: **Natural-gradient learning can be approximated by a simpler rule in many scenarios.** **(A)** Mean Fisher angles between true and approximated weight updates (orange) and between natural and Euclidean weight updates (blue), for $n = 100$. Results for several input patterns were pooled (group1/group2: 10 Hz/10 Hz 10 Hz/30 Hz, 10 Hz/50 Hz, 20 Hz/20 Hz, 20 Hz/40 Hz). Initial weights and input spikes were sampled randomly (100 randomly sampled initial weight vectors per input pattern; for each, angles were averaged over 100 input spike train samples per afferent). **(B)** Same as **(A)**, but angles measured in the Euclidean metric. **(C-F)** Comparison of learning curves for natural gradient (red), Euclidean gradient (blue) and approximation (orange) for $n = 100$ afferents. Simulations were performed in the setting of Fig. 3, under multiple input conditions. **(C)** Group 1 firing with 10 Hz, group 2 firing at 30 Hz. **(D)** Group 1 firing with 10 Hz, group 2 firing at 50 Hz. **(E)** Group 1 firing with 20 Hz, group 2 firing at 20 Hz. **(F)** Group 1 firing with 20 Hz, group 2 firing at 40 Hz.

so the Fisher information matrix is additive.

In the continuous-time limit, where the interval $[0,T]$ is decomposed as $[0,T] = \cup_{i=0}^{k-1}[t_i, t_{i+1}]$, with $t_0 = 0, t_k = T$ and $k \longrightarrow \infty$, we have $\mathrm{d}t = \frac{T}{k} \longrightarrow 0$. Therefore,

$$p_{\boldsymbol{w}}\left(y_{t_0}, \ldots, y_{t_{k-1}}, \boldsymbol{x}^{\epsilon}_{t_0}, \ldots, \boldsymbol{x}^{\epsilon}_{t_k}\right) = \Pi_{i=0}^{k-1} p_{\boldsymbol{w}}\left(y_i, \boldsymbol{x}^{\epsilon}_{t_i}\right) \ . \tag{S76}$$

Then, under the assumption that T is small and firing rates are approximately constant on $[0,T]$, for the Fisher information matrix, we have

$$\begin{aligned} G_{[0,T]}\left(\boldsymbol{w}\right) &= \sum_{i=0}^{k-1} G_{t_i}\left(\boldsymbol{w}\right) \approx k\frac{T}{k}\mathbb{E}\left[\frac{\phi_{t_0}'^2}{\phi_{t_0}}\boldsymbol{x}^{\epsilon}_{t_0}\boldsymbol{x}^{\epsilon}_{t_0}{}^T\right]_{p_{\mathrm{usp}}} & \text{(S77)} \\ &= T\,\mathbb{E}\left[\frac{\phi_{t_0}'^2}{\phi_{t_0}}\boldsymbol{x}^{\epsilon}_{t_0}\boldsymbol{x}^{\epsilon}_{t_0}{}^T\right]_{p_{\mathrm{usp}}} \ . & \text{(S78)} \end{aligned}$$