nature neuroscience

ARTICLES

https://doi.org/10.1038/s41593-021-00809-5

Synaptic plasticity as Bayesian inference

Laurence Aitchison[1,2]✉, Jannes Jegminat[3,4], Jorge Aurelio Menendez[1,5], Jean-Pascal Pfister[3,4], Alexandre Pouget[1,6,7] and Peter E. Latham[1,7]

Learning, especially rapid learning, is critical for survival. However, learning is hard; a large number of synaptic weights must be set based on noisy, often ambiguous, sensory information. In such a high-noise regime, keeping track of probability distributions over weights is the optimal strategy. Here we hypothesize that synapses take that strategy; in essence, when they estimate weights, they include error bars. They then use that uncertainty to adjust their learning rates, with more uncertain weights having higher learning rates. We also make a second, independent, hypothesis: synapses communicate their uncertainty by linking it to variability in postsynaptic potential size, with more uncertainty leading to more variability. These two hypotheses cast synaptic plasticity as a problem of Bayesian inference, and thus provide a normative view of learning. They generalize known learning rules, offer an explanation for the large variability in the size of postsynaptic potentials and make falsifiable experimental predictions.

To survive, animals must accurately estimate the state of the world. This estimation problem is plagued by uncertainty: not only is information often extremely limited (for example, because it is dark) or ambiguous (for example, a rustle in the bushes could be the wind, or it could be a predator), but sensory receptors, and indeed all neural circuits, are noisy. Historically, models of neural computation ignored this uncertainty, and relied instead on the idea that the nervous system estimates values of quantities in the world, but does not include error bars[1]. However, this does not seem to be what animals do—not only does ignoring uncertainty lead to suboptimal decisions, it is inconsistent with a large body of experimental work[2,3]. Thus, the current view is that, in many if not most cases, animals keep track of uncertainty, and use it to guide their decisions[3].

Accurately estimating the state of the world is just one problem faced by animals. They also need to learn, and, in particular, they need to leverage their past experience. It is believed that learning primarily involves changing synaptic weights. But estimating the correct weights, like estimating the state of the world, is plagued by uncertainty: not only is the information available to synapses often extremely limited (in many cases just presynaptic and postsynaptic activity), but that information is highly unreliable. Historically, models of synaptic plasticity ignored this uncertainty, and assumed that synapses do not include error bars when they estimate their weights. However, uncertainty is important for optimal learning—just as it is important for optimal inference of the state of the world.

Motivated by these observations, we propose two hypotheses. The first, Bayesian plasticity (so named because it is derived using Bayes' rule), states that during learning, synapses do indeed take uncertainty into account. Under this hypothesis, synapses do not just estimate what their weight should be, they also include error bars. This allows synapses to adjust their learning rates on the fly: when uncertainty is high, learning rates are turned up; when uncertainty is low, learning rates are turned down. We show that these adjustments allow synapses to learn faster, so there is likely to be considerable evolutionary pressure for such a mechanism. And indeed, the same principle has recently been shown to recover state-of-the-art adaptive optimization algorithms for artificial neural networks[4].

Bayesian plasticity is a hypothesis about what synapses compute. It does not, however, tell synapses how to set their weights; for that, a second hypothesis is needed. Here we propose that weights are sampled from the probability distribution describing the synapse's degree of uncertainty. Under this hypothesis, which we refer to as synaptic sampling, trial-to-trial variability provides a readout of uncertainty: the larger the trial-to-trial variability in synaptic strength, the larger the uncertainty. Synaptic sampling is motivated by the observation that the uncertainty associated with a particular computation should depend on the uncertainty in the weights. Thus, to make optimal decisions, the brain needs to know something about the uncertainty; one way for synapses to communicate that is through variability in the postsynaptic potential (PSP) amplitude (see Supplementary Note 5 for an extended discussion).

Combined, these two hypotheses make several strong experimental predictions. As discussed below, one is consistent with reanalysis of existing experimental data; the others, which are feasible in the not-so-distant future, could falsify one or both hypotheses. We begin by analyzing the first hypothesis that synapses keep track of their uncertainty (Bayesian plasticity). Following this, we discuss our second hypothesis that synapses sample from the resulting distribution (synaptic sampling).

Results

Under Bayesian plasticity, each synapse computes its mean and variance, and updates both based on the pattern of presynaptic spikes. Analogous to classical learning rules, the update rule for the mean pushes it in a direction that reduces a cost function. But in contrast to classical learning rules, the amount the mean changes depends on the uncertainty: the higher the uncertainty, as measured by the variance, the larger the change in the mean. The variance thus sets the learning rate (Fig. 1). In essence, there is a rule for computing the learning rate of each synapse.

To illustrate these ideas, we considered a model of synaptic integration in which PSPs combine linearly, as given by equation (1):

[1]Gatsby Computational Neuroscience Unit, University College London, London, UK. [2]Department of Computer Science, University of Bristol, Bristol, UK. [3]Institute of Neuroinformatics, UZH/ETH Zurich, Zurich, Switzerland. [4]Department of Physiology, University of Bern, Bern, Switzerland. [5]CoMPLEX, University College London, London, UK. [6]Department of Basic Neurosciences, University of Geneva, Geneva, Switzerland. [7]These authors contributed equally: Alexandre Pouget, Peter E. Latham. ✉e-mail: laurence.aitchison@gmail.com

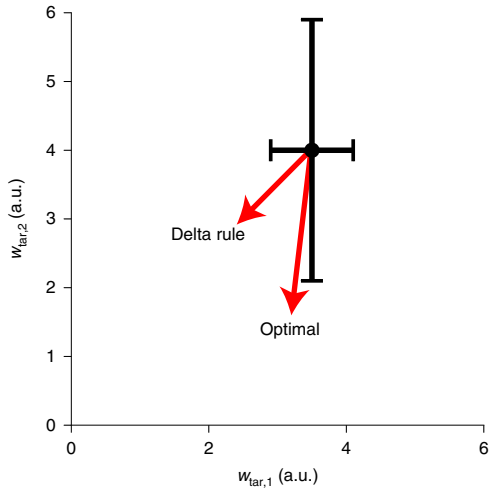

Fig. 1 | The delta rule is suboptimal. The error bars denote uncertainty (measured by the standard deviation around the mean) in estimates of target weights for two synapses, $w_{tar,1}$ and $w_{tar,2}$. The first is reasonably certain, while the second is less so. The red arrows denote possible changes in response to a negative feedback signal. The arrow labeled 'delta rule' represents an equal decrease in the first and second target weights. In contrast, the arrow labeled 'optimal' takes uncertainty into account, so there is a larger change in the second, more uncertain, target weight. a.u., arbitrary units.

$$V(t) = \sum_i w_i(t)x_i(t) + \xi_V(t) \quad (1)$$

where $V(t)$ is the membrane potential, $x_i(t)$ is the synaptic input from neuron i, $w_i(t)$ is the corresponding PSP amplitude and $\xi_V(t)$ is the membrane potential noise. For simplicity, we used discrete time, so $x_i(t)$ is either 1 (when there is a spike at time t) or 0 (when there is no spike), and we took the time step to be 10 ms, on the order of the membrane time constant[5].

We assumed that the goal of the neuron is to set its weights, w_i, so that it achieves a 'target' membrane potential (denoted V_{tar}), that is, the membrane potential that minimizes some cost to the animal. In this setting, the weights are found using a neuron-specific feedback signal, denoted f. Critically, this feedback signal contains information about the target weights through its dependence on a true error signal, δ, that is the difference between the target and actual membrane potential, given by equation (2),

$$\delta \equiv V_{tar} - V. \quad (2)$$

Our focus is on how to use the feedback signal most efficiently, not on where it comes from, which is an active area of research[6–10]. Thus, in most of our analyses, we simply assumed that the neuron receives a feedback signal, and ask how to optimally update the weights via Bayesian inference.

We considered four learning scenarios. In the first, we simply added noise, denoted ξ_δ, to δ, resulting in the error signal $f_{lin} = \delta + \xi_\delta$ (the subscript 'lin' indicates that the average feedback is linear in δ). The second scenario corresponds to cerebellar learning, in which a Purkinje cell receives a complex spike if its output is too high, thus triggering long-term depression[11]. To mimic the all-or-nothing nature of a complex spike[12], we used a cerebellar-like feedback signal: $f_{cb} = \Theta(\delta + \xi_\delta - \theta)$ where Θ is the Heaviside step function. For this feedback signal, f_{cb} is likely to be 1 if δ is above a threshold, θ, and likely to be 0 if it is below threshold. The third scenario corresponds to reinforcement learning, in which the feedback represents the reward. The reward provides the magnitude of the error signal, but not its sign, so the feedback signal is $f_{rl} = -|\delta + \xi_\delta|$. In the fourth scenario, we moved beyond analysis of single neurons, and considered learning the output weights of a recurrent neural network. In this scenario, the error signal is δ, without added noise.

The main idea behind Bayesian plasticity is most easily illustrated in the simplest possible setting, linear feedback, given by $f_{lin} = \delta + \xi_\delta$. In that case, there is a well-known learning rule, the delta rule[13,14], given by equation (3):

$$\Delta w_i = \eta x_i f_{lin} \quad (3)$$

This is most easily recognized as the delta rule in the absence of noise, so that $f_{lin} = \delta$. The change in the weight is the product of a learning rate, η; a presynaptic term, x_i; and a postsynaptic term, f_{lin}. Importantly, η is the same for all synapses, so all synapses whose presynaptic cells are active (that is, for which $x_i = 1$) change by the same amount (the red arrow labeled 'delta rule' in Fig. 1).

In the absence of any other information, the delta rule is perfectly sensible. However, suppose, based on previous information, that synapse 1 is relatively certain about its target weight, whereas synapse 2 is uncertain (error bars in Fig. 1). In that case, new information should have a larger effect on synapse 2 than synapse 1, so synapse 2 should update the estimate of its weight more than synapse 1 (Fig. 1).

Implementing this scheme leads to several features that are not present in classical learning rules. First, the variance needs to be inferred; second, the change in the weight must depend on the inferred variance; and third, because of uncertainty, the 'weight' is in fact the inferred mean weight. In Supplementary Note 1, we derived approximate learning rules that take these features into account (see 'Learning rules' for the exact rules). Using μ_i and σ_i^2 to denote the inferred mean and variance of the distribution over weights, those learning rules are given by equations (4a) and (4b):

$$\Delta\mu_i \approx \frac{\sigma_i^2}{\sigma_\delta^2} x_i f_{lin} - \frac{1}{\tau}\left(\mu_i - \mu_{prior}\right) \quad (4a)$$

$$\Delta\sigma_i^2 \approx -\frac{\sigma_i^4}{\sigma_\delta^2} x_i^2 - \frac{2}{\tau}\left(\sigma_i^2 - \sigma_{prior}^2\right) \quad (4b)$$

where σ_δ^2 is the variance of f_{lin}, and τ, μ_{prior} and σ_{prior}^2 are fixed parameters (described shortly). Note that σ_i corresponds to the length of the error bars in Fig. 1.

The update rule for the mean weight (equation (4a)) is similar to the delta rule (equation (3)). There are, however, two important differences: First, the fixed learning rate, η, that appears in equation (3) has been replaced by a variable learning rate, $\sigma_i^2/\sigma_\delta^2$, which is proportional to the synapse's uncertainty, as measured by σ_i^2. Thus, the more uncertain a synapse is about its target weight, the larger the change in its mean weight when new information arrives—exactly what we expect given Fig. 1. Moreover, as the feedback signal gets noisier (as measured by the variance of f_{lin}), and thus less informative, the learning rate falls. Second, in the absence of information ($x_i = 0$, meaning no spikes), the inferred mean weight, μ_i, moves toward the prior, μ_{prior}. That is because we are considering the realistic case in which the target weights drift randomly over time due to changes in the statistics of the world and/or surrounding circuits. (See "Target weights" for a detailed discussion.)

Unlike the update rule for the mean, the update rule for the uncertainty, σ_i^2 (equation (4b)), does not have a counterpart in classical learning rules. It does, however, have a natural interpretation. The first term in equation (4b) reduces uncertainty (note the negative sign) whenever the presynaptic cell is active ($x_i = 1$), that is, whenever the synapse receives information. The second term has the opposite effect: it continually increases uncertainty (up to the

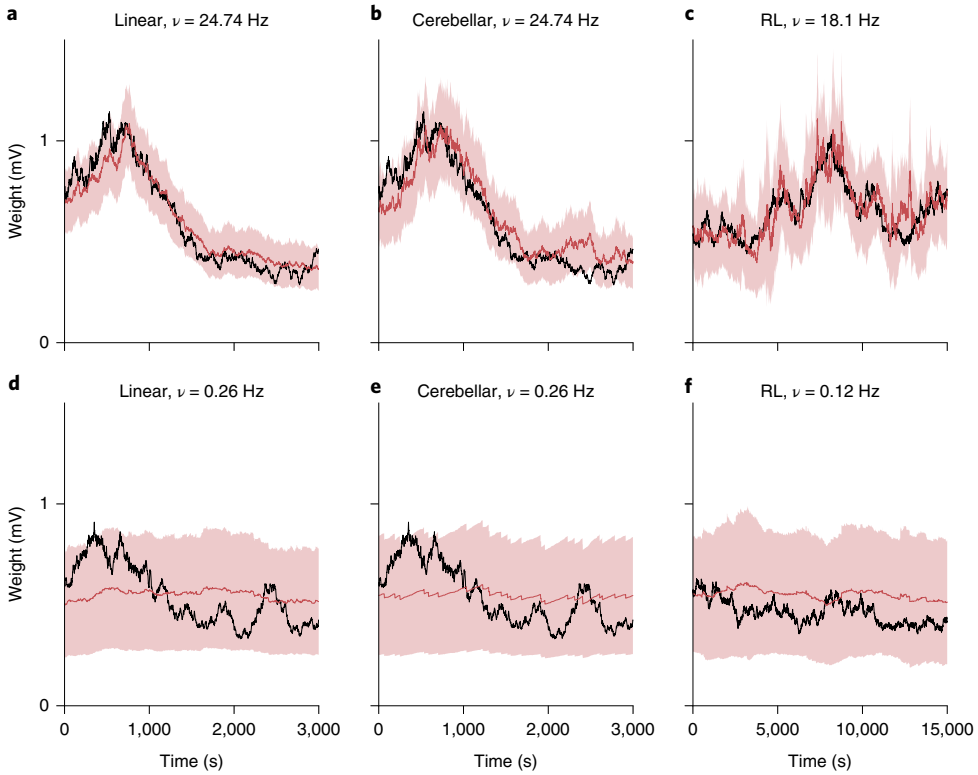

Fig. 2 | Bayesian learning rules track the target weight and estimate uncertainty. a–f, The black line is the target weight, the red line is the mean of the inferred distribution, and the red area represents 95% confidence intervals of the inferred distribution. **a–c** and **d–f** correspond to the highest and lowest presynaptic firing rates used in the simulations, respectively. Consistent with our analysis (equation (9)), higher presynaptic firing rates resulted in lower uncertainty. In **a** and **d**, linear feedback is given by $f_{\text{lin}} = \delta + \xi_\delta$. In **b** and **e**, cerebellar learning is given by $f_{\text{cb}} = \Theta(\delta + \xi_\delta - \theta)$. In **c** and **f**, reinforcement learning (RL) is given by $f_{\text{rl}} = -|\delta + \xi_\delta|$. See Supplementary Note 3 for simulation details. Note that while the red lilnes are plotted at the same thickness, the greater variability in the lower plots may make those lines appear thicker.

prior uncertainty, σ^2_{prior}), independent of presynaptic spikes. That term arises because random drift slowly reduces knowledge about the target weights.

The learning rules given in equation (4) are approximate; their form was optimized for ease of interpretation rather than accuracy. However, the more exact learning rules (see Methods and equations 41, 43 and 47 for the three feedback signals) are not that different. In particular, they retain the same flavor: they consist of a presynaptic term (x_i) and a postsynaptic term (a function of f_{lin}), and the effective learning rate is updated on each time step. Moreover, the interpretation is the same: the mean is moved, on average, toward its true value, with a rate that scales with uncertainty, and whenever there is a presynaptic spike the uncertainty is reduced.

To determine whether our Bayesian learning rules are able to accurately compute the mean and variance of the weights, we generated a set of target weights, denoted $w_{\text{tar},i}$, and used those to construct V_{tar}, given by equation (5):

$$V_{\text{tar}}(t) = \sum\nolimits_i w_{\text{tar},i}(t) x_i(t) \tag{5}$$

Simulations showed that the mean weights track the target weights very effectively (Fig. 2). Just as importantly, the synapse's estimate of its uncertainty tracks the difference between its estimate and the actual target (the black line should be inside the 95% confidence interval (Fig. 2) 95% of the time, and it is very close to that: linear, 96.1%; cerebellar learning, 95.4%; reinforcement learning, 96.8%). Note that the uncertainty was much lower at a high presynaptic firing rate than at a low rate (Fig. 2). That is because for a low firing rate, x_i is mainly zero, and so there is little decrease in σ_i^2 (equation (4b)).

The critical aspect of the learning rules in equation (4) is that the learning rate—the change in mean PSP amplitude, μ_i, per presynaptic spike—increases as the synapse's uncertainty, σ_i^2, increases. This is a general feature of our learning rules, and not specific to any one of them. Consequently, independent of the learning scenario, we expect performance to be better than that for classical learning rules, which do not take uncertainty into account. To check whether this is true, we computed the mean squared error between the actual and target membrane potential, V and V_{tar}, respectively, for classical learning rules, and compared it to the Bayesian learning rules (Fig. 3). As predicted, the Bayesian learning rules always do better than the classical ones, even if the learning rate is tuned to its optimal value. This result was robust to model mismatch (Supplementary Note 6).

For the examples so far, we considered a single neuron inferring only its own input weights. We focused on this case primarily to illustrate our method in the simplest possible setting. In reality, however, the brain needs to optimize some cost function based on a feedback signal applied to a recurrent neural network. To investigate Bayesian plasticity in this, more realistic, regime, we trained the output weights of a recurrent neural network to produce a target function, using as a feedback signal the difference between the target function and its network estimate (Fig. 4a). The learning rules are very similar to those given by equation (4). However, the target weights are not known, so we cannot compare the inferred weights to the target weights, as we did in Fig. 2. We can, however, compare the mean squared error between the target and actual membrane potential, as in Fig. 3. Bayesian plasticity indeed outperformed classical learning rules (Fig. 4b,c). Moreover, the effect was much larger than in Fig. 3; the mean squared error is about an order of

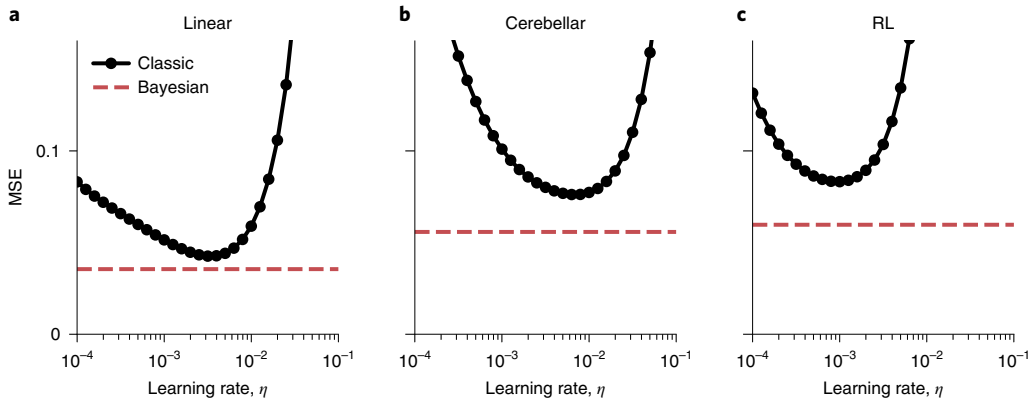

Fig. 3 | Bayesian learning rules exhibit lower error than classical ones. a–c, The red line represents the mean squared error (MSE) between the target and actual membrane potential for the Bayesian learning rules, while the black line represents the MSE for the classical rules as a function of learning rate; the former is constant because the Bayesian learning rules do not depend on the classical learning rate. **a**, Linear feedback, $f_{\text{lin}} = \delta + \xi_\delta$. **b**, Cerebellar learning, $f_{\text{cb}} = \Theta(\delta + \xi_\delta - \theta)$. **c**, Reinforcement learning, $f_{\text{rl}} = -|\delta + \xi_\delta|$. See Supplementary Note 3 for simulation details.

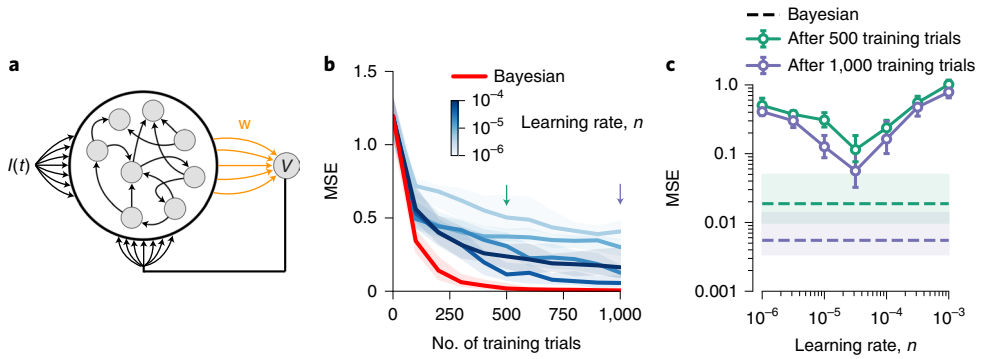

Fig. 4 | Recurrent neural network. a, Schematic of the circuit. $I(t)$ is the input (used to initialize activity) and w corresponds to the learned output weights. The feedback weights (black arrows from V to the recurrent network) are fixed, as are the recurrent weights. During learning, the output of the network, $V(t)$, is compared to the target output, $V_{\text{tar}}(t)$, and the error is used to update w. At test time, the target output is not fed back to the circuit. **b**, Learning curves, measured using mean squared error, for Bayesian and classical learning rules (red and blue, respectively, at a range of learning rates for the classical rule). Although the initial improvement in performance for the Bayesian and classical learning rules was about the same, after 100 time steps Bayesian learning became much more efficient. The arrows correspond to the number of time steps used for the comparison in **c**. **c**, Mean squared error versus the learning rate of the classical rule. Solid lines represent classical learning rules; dashed lines represent Bayesian learning rules. The mean squared error for the Bayesian learning rule was about an order of magnitude smaller than for the classical one. In **c** and **d**, we plot the median, taken over $n = 400$ network–target pairs. Error bars are 95% confidence intervals, computed using the percentile bootstrap.

magnitude smaller for the Bayesian than for the classical learning rule (note the log scale in Fig. 4c), a result that was highly robust to model mismatch (Supplementary Note 7). These simulations suggest that taking into account weight uncertainty has a much larger effect in networks than in single neurons.

Figures 3 and 4 indicate that there is a clear advantage to using uncertainty to adjust learning rates. But does the brain do this? Addressing that question will require a new generation of plasticity experiments. At present, in typical plasticity experiments, only changes in weights are measured; to test our hypothesis, it is necessary to measure changes in learning rates, and at the same time determine how those changes are related to the synapse's uncertainty. This presents two challenges: First, measuring changes in learning rates is difficult, as weights must be monitored over long periods of time and under natural conditions, preferably in vivo. Second, we cannot measure the synapse's uncertainty directly. We next discuss two approaches to overcoming these challenges.

The first approach is indirect: use neural activity measured over long periods in vivo to estimate the uncertainty a synapse should have; then, armed with that estimate, test the prediction that the learning rate increases with uncertainty. To estimate the uncertainty a synapse should have, we take advantage of a general feature of essentially all learning rules: synapses get information only when the presynaptic neuron spikes. Consequently, the synapse's uncertainty should fall as the presynaptic firing rate increases. In fact, under mild assumptions, we can derive a very specific relationship: the relative change in weight under a plasticity protocol, $\Delta\mu_i/\mu_i$, should scale approximately as $1/\sqrt{\nu_i}$ where ν_i is the firing rate of the neuron presynaptic to synapse i, given by equation (6):

$$\frac{\Delta\mu_i}{\mu_i} \propto \frac{1}{\sqrt{\nu_i}} \tag{6}$$

a relationship that held in our simulations for firing rates above about 1 Hz (Supplementary Note 3). In essence, firing rate is a proxy for uncertainty, with a higher firing rate indicating lower uncertainty and vice versa. This prediction could be tested by observing neurons in vivo, estimating the presynaptic firing rates, then

performing plasticity experiments to determine the relative change in synaptic strength, $\Delta\mu_i/\mu_i$.

The second approach is more direct, but it requires an additional hypothesis. While Bayesian plasticity tells us how to compute the mean and variance of the weights, it does not tell us what weight to use when a spike arrives. But the synaptic sampling hypothesis does; it tells us that the mean and variance of the PSP amplitude should be equal to the mean and variance of the inferred distribution over the target weight, given by equations (7a) and (7b):

$$\text{PSP mean} = \mu_i \tag{7a}$$

$$\text{PSP variance} = \sigma_i^2 \tag{7b}$$

Under our learning rules, the change in mean synaptic weight is proportional to the variance, σ_i^2 (equation (4a)). Consequently, the relative change in weight $\Delta\mu_i/\mu_i$, is proportional to σ_i^2/μ_i; combining this with equation (7) gives equation (8):

$$\frac{\Delta\mu_i}{\mu_i} \propto \frac{\text{PSP variance}}{\text{PSP mean}} \equiv \text{Normalized variability} \tag{8}$$

where we defined the normalized variability as the ratio of PSP variance to its mean. We verified that this relationship holds in simulations (Supplementary Note 2).

Equation (8) implies that when the PSP variance is high, learning rates are also high. Testing that experimentally is technically difficult, requiring monitoring the PSP mean and variance for long periods in vivo, and comparing normalized variability to changes in the mean. However, such experiments are likely to be possible in the near future.

A more indirect approach based on this idea, for which we can apply current data, makes use of equation (6) to replace the left-hand side of equation (8), $\Delta\mu_i/\mu_i$, with $1/\sqrt{\nu_i}$. This gives equation (9):

$$\text{Normalized variability} \propto \frac{1}{\sqrt{\nu_i}} \tag{9}$$

This is intuitively sensible; as discussed above, higher presynaptic firing rates means the synapse is more certain, and synaptic sampling states that higher certainty should reduce the observed variability. This relationship can be tested by estimating presynaptic firing rates in vivo, and comparing them to the normalized variability measured using paired recordings. Such data can be extracted from experiments by Ko et al.[15]. In those experiments, calcium signals in mouse visual cortex were recorded in vivo under a variety of stimulation conditions, providing an estimate of the firing rate of each imaged neuron; subsequently, whole-cell recordings of pairs of identified neurons were made in vitro, and the mean and variance of the PSPs were measured. In Fig. 5, we plot the normalized variability (the ratio of the PSP variance to the mean) versus the presynaptic firing rate on a log–log scale (data were supplied to us by Ko et al.[15]; Supplementary Note 4). On this scale, our theory predicts a slope of −1/2. The normalized variability did indeed decrease as the firing rate increased ($P<0.003$), and the slope was not significantly different from the predicted value of −1/2 ($P=0.57$). This pattern was broadly matched by simulations, at least at a sufficiently high firing rate (Supplementary Note 3).

An alternative explanation for this result is that increases in firing rate reduce the normalized variability because of short-term effects on release probability. The release probability, denoted by p_r, scales the variance of the PSP by a factor of $p_r(1-p_r)$ and the mean by a factor of p_r, so the normalized variability (the variance divided by the mean) scales as $1-p_r$. Consequently, an increase in release probability with firing rate would explain the results presented in Fig. 5. Such increases do indeed occur[16]. However, much more common—especially in rodent layers 2 and 3, where these experiments were performed—is a decrease in release probability with firing rate[17,18]. Thus, short-term synaptic plasticity would typically lead to an increase, not a decrease, in the normalized variability when firing rate increases; the opposite of what we observed experimentally.

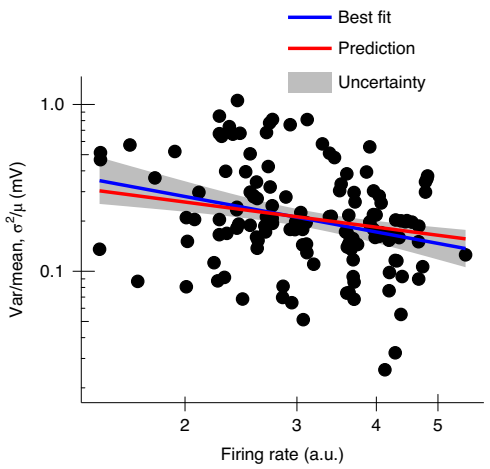

Fig. 5 | Normalized variability versus presynaptic firing rate as a diagnostic of our theory. The red line, which has a slope of −1/2, is our prediction (the intercept, for which we do not have a prediction, was chosen to give the best fit to the data). The blue line was fit by linear regression ($n=136$ points), and the gray region represents two standard errors. The slope of the blue line, −0.62, was statistically significantly different from 0 ($P<0.003$, t-test) and not significantly different from −1/2 ($P=0.57$, t-test; assumes normality, which was not formally tested). The firing rate was measured by taking the average signal from a spike deconvolution algorithm[45]. Units are arbitrary because the scale factor relating the average signal from the deconvolution algorithm and the firing rate is not exactly one[46]. Data are from layers 2 and 3 of the mouse visual cortex in ref. [15].

Discussion

We proposed that synapses do not just keep track of point estimates of their weights, as they do in classical learning rules; they also keep track of their uncertainty. They then use that uncertainty to set learning rates: the higher the uncertainty, the higher the learning rate. This allows different synapses to have different learning rates, and leads to learning rules that allow synapses to exploit all locally available information. This in turn leads to better performance, as measured by mean squared error (Figs. 3 and 4b,c), and faster learning, which is implicit in Fig. 3 (because the target weights drift, fast learning is essential for achieving low mean squared error) and explicit in Fig. 4b.

The critical difference between our learning rules and classical ones is that the learning rates themselves undergo plasticity. We derived three rules, based on three different assumptions about the feedback signal received by the neuron, and in all cases the updates for the mean had the flavor of a classical rule: the change in the mean weight was a function of the presynaptic activity and an error signal. Other assumptions about the feedback signal are clearly possible, and our method can generate a broad range of learning rules. Whether or not they can generate all rules that have been observed experimentally is an avenue for future research.

The hypothesis that synapses keep track of uncertainty, which we refer to as the Bayesian plasticity hypothesis, makes the general prediction that learning rates, not just synaptic strengths, are a function

of presynaptic and postsynaptic activity—something that should be testable with the next generation of plasticity experiments. In particular, it makes a specific prediction about learning rates in vivo: learning rates should vary across synapses, being higher for synapses with lower presynaptic firing rates.

We also made a second, independent, hypothesis: synaptic sampling. This hypothesis states that the variability in PSP size associated with a particular synapse matches the uncertainty in the strength of that synapse. This allows synapses to communicate their uncertainty to surrounding circuitry—information that is critical if the brain is to monitor the accuracy of its own computations. The same principle has been applied to neural activity, where it is known as the neural sampling hypothesis[19–22], which posits that variability in neural activity matches uncertainty about the state of the external world. The neural sampling hypothesis meshes well with synaptic sampling; uncertainty in the weights increases uncertainty in the current estimate of the state of the world, and likewise, variability in the weights increase variability in neural activity (Supplementary Note 5). While there is some experimental evidence for the neural sampling hypothesis[21–25], it has not been firmly established. Whether other proposals for encoding probability distributions with neural activity, such as probabilistic population codes[3,26], can be combined with synaptic sampling is an open question.

By combining our two hypotheses, we were able to make additional predictions. These focused on what we call the normalized variability, that is, the ratio of the variance in PSP size to the mean. First, we predicted that plasticity should increase with normalized variability, which remains to be tested. Second, we predicted that normalized variability should decrease with presynaptic firing rate. Reanalyzing data from Ko et al.[15], we provided evidence that this is indeed the case. In machine learning, the idea that it is advantageous to keep track of the distribution over weights has a long history[27–29]. Especially relevant is a recent study in which, as in our scheme, learning rates were reduced when certainty was high[30]. However, rather than updating the uncertainty on every time step, as we do, updating occurred only when there was a change in the task. This occurs on the timescale of minutes to hours; not the millisecond timescale on which uncertainty is updated in our model. Nevertheless, this approach worked well in settings in which deep networks had to learn multiple tasks.

In neuroscience, weight uncertainty was first explored in the context of reinforcement learning[31]. In that work, the weights related sensory stimuli to rewards, and weight correlations that developed due to Bayesian learning provided an exceptionally elegant explanation of backward blocking. The idea lay dormant for over a decade, until it was rediscovered with a slightly different focus, one in which knowledge of weight uncertainty is critical for knowledge of computational uncertainty[3]. Several theoretical studies followed. The first of those[32] bore some resemblance to ours, in that weights were sampled from a distribution. However, the timescale for sampling was hours rather than milliseconds, which is too slow to explain the spike-to-spike variability in PSP size that is ubiquitous in the brain. More recently, Hiratani and Fukai[33] postulated that the multiple synaptic contacts per connection observed in cortex provide a scaffolding for constructing a nonparametric estimate of the probability distribution over synaptic strength. Weight uncertainty has also been applied to drift diffusion models[34], using methods similar to those used in work by Dayan and Kakade[31]; the main difference was that the reward was binary (correct or incorrect) rather than continuous. Finally, recent work proposed that short-term plasticity is also governed by a Bayesian updating process[35]. It will be interesting to determine which combination of these schemes is used by the brain.

If the Bayesian plasticity hypothesis is correct, synapses would have to keep track of, and store, two variables: the mean, as is standard, but also the variance (or, equivalently, the learning rate), which is not. The complexity of synapses[36–38], and their ability to use nontrivial learning rules (for example, synaptic tagging, in which activity at a synapse 'tags' it for future long-term changes in strength[39–41], and metaplasticity, in which the learning rate can be modified by synaptic activity without changing the synaptic strength[42–44]), suggests that representing uncertainty—or learning rate—is quite possible. It will be nontrivial, but important, to work out how.

Our framework has several implications, both for the interpretation of neurophysiological data and for future work. First, under the synaptic sampling hypothesis, PSPs are necessarily noisy. Consequently, noise in synapses (for example, synaptic failures) is a feature, not a bug. We thus provide a normative theory for one of the major mysteries in synaptic physiology: why neurotransmitter release is probabilistic. Second, our approach allows us to derive local, biologically plausible learning rules, no matter what information is available at the synapse, and no matter what the statistics of the synaptic input. Thus, our approach provides the flexibility necessary to connect theoretical approaches based on optimality to complex biological reality.

In neuroscience, Bayes' theorem is typically used to analyze high-level inference problems, such as decision-making under uncertainty. Here we demonstrated that Bayes' theorem, being the optimal way to solve any inference problem, big or small, could be implemented in perhaps the smallest computationally relevant element in the brain: the synapse.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41593-021-00809-5.

Received: 19 June 2020; Accepted: 26 January 2021;
Published online: 11 March 2021

References

1. Poggio, T. A theory of how the brain might work. *Cold Spring Harb. Symp. Quant. Biol.* **55**, 899–910 (1990).
2. Knill, D. C. & Richards, W. *Perception as Bayesian Inference* (Cambridge University Press, 1996).
3. Pouget, A., Beck, J. M., Ma, W. J. & Latham, P. E. Probabilistic brains: knowns and unknowns. *Nat. Neurosci.* **16**, 1170–1178 (2013).
4. Aitchison, L. Bayesian filtering unifies adaptive and non-adaptive neural network optimization methods. *Adv. Neural Inf. Process. Syst.* https://proceedings.neurips.cc/paper/2020/file/d33174c464c877fb03e77efdab4ae804-Paper.pdf (2020).
5. Tripathy, S. J., Burton, S. D., Geramita, M., Gerkin, R. C. & Urban, N. N. Brain-wide analysis of electrophysiological diversity yields novel categorization of mammalian neuron types. *J. Neurophysiol.* **113**, 3474–3489 (2015).
6. Schiess, M., Urbanczik, R. & Senn, W. Somato-dendritic synaptic plasticity and error-backpropagation in active dendrites. *PLoS Comput. Biol.* **12**, e1004638 (2016).
7. Bono, J. & Clopath, C. Modeling somatic and dendritic spike mediated plasticity at the single neuron and network level. *Nat. Commun.* **8**, 706 (2017).
8. Sacramento, J., Ponte Costa, R., Bengio, Y. & Senn, W. Dendritic cortical microcircuits approximate the backpropagation algorithm. *Adv. Neural Inf. Process. Syst.* **31**, 8711 (2018).
9. Illing, B., Gerstner, W. & Brea, J. Biologically plausible deep learning—but how far can we go with shallow networks? *Neural Netw.* **118**, 90–101 (2019).
10. Akrout, M., Wilson, C., Humphreys, P. C., Lillicrap, T. & Tweed, D. Deep learning without weight transport. *Adv. Neural Inf. Process. Syst.* **32**, 976 (2019).
11. Ito, M., Sakurai, M. & Tongroach, P. Climbing fibre induced depression of both mossy fibre responsiveness and glutamate sensitivity of cerebellar Purkinje cells. *J. Physiol.* **324**, 113–134 (1982).
12. Eccles, J., Llinas, R. & Sasaki, K. The excitatory synaptic action of climbing fibres on the purkinje cells of the cerebellum. *J. Physiol.* **182**, 268–296 (1966).

13. Widrow, B. & Hoff, M. E. Adaptive switching circuits. Technical Report no. 1553-1. https://apps.dtic.mil/dtic/tr/fulltext/u2/241531.pdf (Office of Naval Research, 1960).
14. Dayan, P. & Abbott, L. F. *Theoretical Neuroscience* (MIT Press, 2001).
15. Ko, H. et al. The emergence of functional microcircuits in visual cortex. *Nature* **496**, 96–100 (2013).
16. Thomson, A. M. Presynaptic frequency- and pattern-dependent filtering. *J. Comput. Neurosci.* **15**, 159–202 (2003).
17. Tsodyks, M. V. & Markram, H. The neural code between neocortical pyramidal neurons depends on neurotransmitter release probability. *Proc. Natl Acad. Sci. USA* **94**, 719–723 (1997).
18. Maffei, A. & Turrigiano, G. G. Multiple modes of network homeostasis in visual cortical layer 2/3. *J. Neurosci.* **28**, 4377–4384 (2008).
19. Hoyer, P. O. & Hyvarinen, A. Interpreting neural response variability as Monte Carlo sampling of the posterior. *Adv. Neural Inf. Process. Syst.* **15**, 293–300 (2002).
20. Fiser, J., Berkes, P., Orbán, G. & Lengyel, M. Statistically optimal perception and learning: from behavior to neural representations. *Trends Cogn. Sci.* **14**, 119–130 (2010).
21. Berkes, P., Fiser, J., Orbán, G. & Lengyel, M. Spontaneous cortical activity reveals hallmarks of an optimal internal model of the environment. *Science* **331**, 83–87 (2011).
22. Orbán, G., Berkes, P., Fiser, J. & Lengyel, M. Neural variability and sampling-based probabilistic representations in the visual cortex. *Neuron* **92**, 530–543 (2016).
23. Haefner, R. M., Berkes, P. & Fiser, J. Perceptual decision-making as probabilistic inference by neural sampling. *Neuron* **90**, 649–660 (2016).
24. Aitchison, L. & Lengyel, M. The hamiltonian brain: efficient probabilistic inference with excitatory–inhibitory neural circuit dynamics. *PLoS Comput. Biol.* **12**, e1005186 (2016).
25. Lange, R. D. & Haefner, R. M. Task-induced neural covariability as a signature of approximate bayesian learning and inference. Preprint at *bioRxiv* https://doi.org/10.1101/081661 (2020).
26. Ma, W. J., Beck, J. M., Latham, P. E. & Pouget, A. Bayesian inference with probabilistic population codes. *Nat. Neurosci.* **9**, 1432–1438 (2006).
27. Buntine, W. L. & Weigend, A. S. Bayesian backpropagation. *Complex Syst.* **5**, 603–643 (1991).
28. MacKay, D. J. A practical bayesian framework for backpropagation networks. *Neural Comput.* **4**, 448–472 (1992).
29. Blundell, C., Cornebise, J., Kavukcuoglu, K. & Dean, W. Weight uncertainty in neural networks. *Proc. Mach. Learn. Res.* **37**, 1613–1622 (2015).
30. Kirkpatrick, J. et al. Overcoming catastrophic forgetting in neural networks. *Proc. Natl Acad. Sci. USA* **106**, 10296–10301 (2016).
31. Dayan, P. & Kakade, S. Explaining away in weight space. *Adv. Neural Inf. Process. Syst.* **13**, 451–457 (2001).
32. Kappel, D., Habenschuss, S., Legenstein, R. & Maass, W. Network plasticity as bayesian inference. *PLoS Comput. Biol.* **11**, e1004485 (2015).
33. Hiratani, N. & Fukai, T. Redundancy in synaptic connections enables neurons to learn optimally. *Proc. Natl Acad. Sci. USA* **115**, E6871–E6879 (2018).
34. Drugowitsch, J., Mendonça, A. G., Mainen, Z. F. & Pouget, A. Learning optimal decisions with confidence. *Proc. Natl Acad. Sci. USA* **116**, 24872–24880 (2019).
35. Pfister, J.-P., Dayan, P. & Lengyel, M. Synapses with short-term plasticity are optimal estimators of presynaptic membrane potentials. *Nat. Neurosci.* **13**, 1271–1275 (2010).
36. Kasai, H., Takahashi, N. & Tokumaru, H. Distinct initial SNARE configurations underlying the diversity of exocytosis. *Physiol. Rev.* **92**, 1915–1964 (2012).
37. Südhof, T. C. The presynaptic active zone. *Neuron* **75**, 11–25 (2012).
38. Michel, K., Müller, J. A., Oprisoreanu, A.-M. & Schoch, S. The presynaptic active zone: a dynamic scaffold that regulates synaptic efficacy. *Exp. Cell Res.* **335**, 157–164 (2015).
39. Frey, U. & Morris, R. G. Synaptic tagging and long-term potentiation. *Nature* **385**, 533–536 (1997).
40. Redondo, R. L. & Morris, R. G. M. Making memories last: the synaptic tagging and capture hypothesis. *Nat. Rev. Neurosci.* **12**, 17–30 (2011).
41. Rogerson, T. et al. Synaptic tagging during memory allocation. *Nat. Rev. Neurosci.* **15**, 157–169 (2014).
42. Abraham, W. C. & Bear, M. F. Metaplasticity: the plasticity of synaptic plasticity. *Trends Neurosci.* **19**, 126–130 (1996).
43. Abraham, W. C. Metaplasticity: tuning synapses and networks for plasticity. *Nat. Rev. Neurosci.* **9**, 387 (2008).
44. Hulme, S. R., Jones, O. D., Raymond, C. R., Sah, P. & Abraham, W. C. Mechanisms of heterosynaptic metaplasticity. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **369**, 20130148 (2014).
45. Vogelstein, J. T. et al. Fast nonnegative deconvolution for spike train inference from population calcium imaging. *J. Neurophysiol.* **104**, 3691–3704 (2010).
46. Packer, A. M., Russell, L. E., Dalgleish, H. W. P. & Häusser, M. Simultaneous all-optical manipulation and recording of neural circuit activity with cellular resolution in vivo. *Nat. Methods* **12**, 140–146 (2015).

Methods

Description of our model. Previously, we specified how the membrane potential depends on the weights and incoming spikes (equation (1)) and how the target membrane potential depends on the target weights and incoming spikes (equation (5)), and we defined the error signal (equation (2)). Here, we describe how target weights, $w_{\mathrm{tar},i}$, the weights, w_i, and the spikes, x_i, are generated.

Target weights. The target weights are the weights that in some sense optimize the performance of the animal. We do not expect these weights to remain constant over time, for two reasons: First, both the state of the world and the organism change over time, thus changing the target weights. Second, we take a local, single-neuron view to learning, and define the target weights on a particular neuron to be the optimal weights given the weights on all the other neurons in the network. Consequently, as the weights of surrounding neurons change due to learning, the target weights on our neuron also change. While these changes may be quite systematic, to a single synapse deep in the brain they are likely to appear random.

Motivated by this last observation, in our model we assumed that the target weights evolve according to a random process. To ensure that the weights do not change sign, we worked in log space, and on each time step, we added a small amount of noise to the log of the target weights. Additionally, to ensure that the weights did not become too small or too large, we added a small drift toward a prior log weight. Specifically, defining (equation (10)):

$$\lambda_{\mathrm{tar,i}} = \log|\mathrm{w}_{\mathrm{tar,i}}| \tag{10}$$

(note the absolute value sign, which allows the weights to be either positive or negative), we let $\lambda_{\mathrm{tar},i}$ evolve according to equation (11):

$$\lambda_{\mathrm{tar,i}}(t+1) = \lambda_{\mathrm{tar,i}}(t) - \frac{\lambda_{\mathrm{tar,i}}(t) - m_{\mathrm{prior}}}{\tau} + \sqrt{\frac{2s^2_{\mathrm{prior}}}{\tau}}\,\xi_{\mathrm{tar,i}} \tag{11}$$

where m_{prior} and s^2_{prior} are the prior mean and variance of $\lambda_{\mathrm{tar},i}(t)$, τ (which is dimensionless) is the characteristic number of steps over which $\lambda_{\mathrm{tar},i}(t)$ changes, and $\xi_{\mathrm{tar},i}$ is a zero-mean, unit variance Gaussian random variable.

We chose the noise process described in equation (11) for three reasons: First, $w_{\mathrm{tar},i}$ is equal to either $+e^{\lambda_{\mathrm{tar,i}}}$ (for excitatory weights) or $-e^{\lambda_{\mathrm{tar,i}}}$ (for inhibitory weights), and thus cannot change sign as $\lambda_{\mathrm{tar},i}$ changes with learning. Consequently, excitatory weights cannot become inhibitory, and vice versa, so Dale's law is preserved. Second, spine sizes obey this stochastic process[47], and while synaptic weights are not spine sizes, they are correlated[48]. Third, this noise process gives a log-normal stationary distribution of weights, as is observed experimentally[49].

The parameters that determine how the weights drift, m_{prior} and s^2_{prior}, were set to the mean and variance of measured log weights using data from ref. [49] (Supplementary Note 4). We used a time step of 10 ms, within the range of measured membrane time constants. For the linear and cerebellar models, we set τ to 10^5; for reinforcement learning, we set τ to 5×10^5. These values were chosen so that uncertainty roughly matched observed variability (Supplementary Note 4). For the recurrent network, we do not know the target weights, so we do not know the drift rate. Nor do we know the effective drift associated with the fact that the optimal weight on one synapse changes as the surrounding circuit changes. We therefore tried different drifts in our simulations (data not shown). We found that near-zero drift was optimal, so we set τ to ∞.

Synaptic weights. Our inference algorithm computes a distribution over the target weights. Given that distribution, there is nothing in the Bayesian plasticity hypothesis that tells us how to set the weights when a spike arrives. That is where the sampling hypothesis comes in: it tells us to sample the weights, w_i, from the posterior, given by equation (12):

$$w_i = e^{m_i + s_i\xi_i} \tag{12}$$

where m_i and s_i are the mean and standard deviation of the posterior distribution over the log weights, respectively, and ξ_i is a zero-mean, unit variance Gaussian random variable. The mean and variance of w_i under equation (12), for which we use μ_i and σ_i^2, respectively, are the standard expressions for the mean and variance of a log-normal distribution, given by equations (13a) and (13b):

$$\mathbb{E}[w_i|\mathcal{D}_i] \equiv \mu_i = e^{m_i + s_i^2/2} \tag{13a}$$

$$\mathrm{Var}\,[\mathrm{w_i}|\mathcal{D}_\mathrm{i}] \equiv \sigma_\mathrm{i}^2 = \mu_\mathrm{i}^2[\mathrm{e}^{s_\mathrm{i}^2} - 1] \tag{13b}$$

where $\mathcal{D}_i$ is the data seen by the synapse so far (equation (19)).

Earlier, we compared our Bayesian learning rules to classical ones (Figs. 3 and 4). For classical rules, there is no posterior to sample from, so we could not use equation (12). Consequently, for the the classical implementation of linear and cerebellar rules, we did not sample; and for Bayesian learning, we used $w_i = \mu_i$.

The reinforcement learning rule, however, requires sampling for both Bayesian and classical learning (equations 47 and 48). We thus assumed that the variance is proportional to the mean (as is the case for Poisson statistics). To find the constant of proportionality, denoted k, we used data from ref. [49] (Supplementary Note 4). A least-squares fit to that data gave $k = 0.0877$. A naive way to implement this is to sample weights using $w_i = \mu_i + \sqrt{k\mu_i}\,\xi_i$ with $\xi_i \sim \mathcal{N}(0,1)$. However, that allows w_i to change sign; so instead, we sampled the weights using equation (14):

$$w_i = \mu_i e^{\beta_i + \gamma_i\xi_i} \tag{14}$$

and chose β_i and γ_i so that the mean and variance of w_i are μ_i and $k\mu_i$, respectively. As is straightforward to show, these conditions are satisfied when β_i and γ_i are given by equations (15a) and (15b):

$$\beta_i = -\frac{\log(1 + k/\mu_i)}{2} \tag{15a}$$

$$\gamma_i = \sqrt{\log(1 + k/\mu_i)} \tag{15b}$$

Synaptic input. For linear, cerebellar and reinforcement learning, neurons receive input from n presynaptic neurons, all firing at different rates. The firing rates, ν_i (i denotes presynaptic neuron), are drawn from a log-normal distribution, using a distribution that is intermediate between the narrow range found by some[50] and the broad range found by others[51]: a log-normal distribution with median at 1 Hz and with 95% of firing rates being between 0.1 Hz and 10 Hz, according to equation (16):

$$\log\nu_\mathrm{i} \sim \mathcal{N}\left(0, \left(\log\sqrt{10}\right)^2\right) \tag{16}$$

with ν_i measured in Hz. On each time step, x_i is drawn from a Bernoulli distribution (so it is either 0 or 1), according to equation (17):

$$P(x_i) = (\nu_i\Delta t)^{x_i}(1 - \nu_i\Delta t)^{1-x_i} \tag{17}$$

Learning rules. Here we outline how a synapse can infer a probability distribution over its target weights. This is done using a well-understood class, hidden Markov model, for which we can use a standard, two-step procedure: first, the synapse incorporates new data using Bayes' theorem; second, it accounts for random changes in the target weight.

While straightforward in principle, in practice there are two difficulties with this approach. First, it results in a joint distribution over all synaptic weights. It is unclear, however, how synapses could store such a distribution; even with a Gaussian approximation, for n synapses there are about $n^2/2$ parameters. And it is even less clear how they could compute it, as that would require communication among synapses on different dendritic branches. We thus assumed that each synapse performs probabilistic inference based only on the data available to it. This makes each synapse locally optimal, allowing us to derive local learning rules. It is potentially the most important theoretical advance of our analysis. And within the Bayesian framework it is straightforward: each synapse simply integrates over the uncertainty in the target weights of all the other synapses. Nonetheless, this is an unusual approach, and further work is necessary to understand its theoretical properties.

The second difficulty is that even with the local approximation, inference is intractable, as it requires point-wise multiplication of probability distributions and a convolution (equations 20 and 21). To remedy this, we approximated the true distribution by a simpler one, a log normal. The log-normal distribution was chosen for two reasons: (1) it prevents synapses from changing sign, so Dale's law is respected; and (2) it matches the distribution of the target weights (equation (11)), so it produces the correct distribution in the absence of presynaptic spikes.

Single-neuron learning rules: general formalism. The goal of a synapse is to compute the probability distribution over synaptic strength given data up to the last time step. Here, the data (assumed local, as discussed above) consists of the feedback signal f (shorthand for f_{lin}, f_{cb} or f_{rl}), the presynaptic input x_i, and the actual weight w_i. To reduce clutter, we used $d_i(t)$ to denote the data at time t, according to equation (18):

$$d_i(t) \equiv \{f(t), x_i(t), w_i(t)\} \tag{18}$$

and $\mathcal{D}_i(t)$ to denote past data, according to equation (19):

$$\mathcal{D}_i(t) \equiv \{d_i(t), d_i(t-1), d_i(t-2), \ldots\} \tag{19}$$

With this notation, the goal of the synapse is to compute $P\left(\lambda_{\mathrm{tar,i}}(t+1)|\mathcal{D}_i(t)\right)$ in terms of $P\left(\lambda_{\mathrm{tar,i}}(t)|\mathcal{D}_i(t-1)\right)$. To reduce clutter even further, here and in what follows, all quantities without an explicitly specified time index were evaluated at

time step t; thus, we will derive an update rule for $P\left(\lambda_{\mathrm{tar,i}}(t+1)|\mathcal{D}_i\right)$ in terms of $P\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_i(t-1)\right)$.

As discussed above, making the approximation that synapses perform inference based only on local information, the first step in the derivation of the update rule, incorporating new data using Bayes' theorem, gives equation (20):

$$P\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_i\right) = P\left(\lambda_{\mathrm{tar,i}}|d_i,\mathcal{D}_i(t-1)\right) \propto P\left(d_i|\lambda_{\mathrm{tar,i}}\right)P\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_i(t-1)\right) \tag{20}$$

where we used the Markov property: $P\left(d_i|\lambda_{\mathrm{tar,i}},\mathcal{D}_i(t-1)\right) = P\left(d_i|\lambda_{\mathrm{tar,i}}\right)$. (Recall that $\lambda_{\mathrm{tar},i}$ is the log of the absolute value of the ith target weight, $w_{\mathrm{tar},i}$ (equation (10))). In the second step, the synapse takes into account random changes in the target weight, given by equation (21):

$$P\left(\lambda_{\mathrm{tar,i}}(t+1)|\mathcal{D}_i\right) = \int d\lambda_{\mathrm{tar,i}} P\left(\lambda_{\mathrm{tar,i}}(t+1)|\lambda_{\mathrm{tar,i}}\right)P\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_i\right) \tag{21}$$

The conditional distribution, $P\left(\lambda_{\mathrm{tar,i}}(t+1)|\lambda_{\mathrm{tar,i}}\right)$, can be extracted from equation (11). Combining both steps takes us from the distribution at time t, $P\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_i(t-1)\right)$, to the distribution at the time $t+1$, $P\left(\lambda_{\mathrm{tar,i}}(t+1)|\mathcal{D}_i\right)$.

To make progress analytically, we approximated the true distribution by a log-normal one with mean m_i and variance s_i^2; that is, we assumed equation (22):

$$\lambda_{\mathrm{tar,i}}|\mathcal{D}_i(t-1) \sim \mathcal{N}\left(m_i, s_i^2\right) \tag{22}$$

This is the quantity the synapse needs when it sets the actual weight, w_i. (Recall that quantities with no explicit time dependence are to be evaluated at time t; thus, the left-hand side is the probability distribution over $\lambda_{\mathrm{tar},i}(t)$, given data up to the previous time step).

Finalizing the calculation requires two steps: (1) insert equation (22) into equation (20) and compute $P\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_i\right)$; (2) insert the result into equation (21) and compute $P\left(\lambda_{\mathrm{tar,i}}(t+1)|\mathcal{D}_i\right)$. However, equation (20) takes us out of our log-normal model class. To remedy this, we used assumed density filtering[52], for which posteriors are taken to be log normal with mean and variance chosen to produce the distribution closest to the true one, where 'close' is measured by the Kullback–Leibler divergence between the true and log-normal distributions. This can be achieved by matching moments; the mean and variance of the 'closest' log-normal distribution are given by equations (23a) and (23b):

$$m_i = \mathbb{E}[\lambda_{\mathrm{tar,i}}|\mathcal{D}_i(t-1)] \tag{23a}$$

$$s_i^2 = \mathrm{Var}\left[\lambda_{\mathrm{tar,i}}|\mathcal{D}_{\mathrm{i}}(\mathrm{t}-1)\right] \tag{23b}$$

We will apply this first to equation (20). Taking the log of both sides of that equation gives equation (24):

$$\log \mathrm{P}\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_{\mathrm{i}}\right) = \mathrm{L}(\lambda_{\mathrm{tar,i}}) + \log \mathrm{P}\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_{\mathrm{i}}(\mathrm{t}-1)\right) + \text{ const} \tag{24}$$

where

$$L(\lambda_{\mathrm{tar,i}}) \equiv \log \mathrm{P}\left(\mathrm{d_i}|\lambda_{\mathrm{tar,i}}\right) \tag{25}$$

is the log likelihood of the data at time t given the target weight (equation (25)); we suppressed the dependence on d_i to avoid clutter. Under the log-normal assumption, the second term on the right-hand side of equation (24) is Gaussian in $\lambda_{\mathrm{tar},i}$. Motivated by the fact that new data does not provide much information, we assumed that the likelihood is a slowly varying function of the target weights, allowing us to make a Laplace approximation: we Taylor expand the log likelihood around m_i, the mean of $P\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_i(t-1)\right)$, and work only to second order in $\lambda_{\mathrm{tar},i} - m_i$. Also, using equation (22), we have equation (26):

$$\log \mathrm{P}\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_{\mathrm{i}}\right) = L'(m_i)(\lambda_{\mathrm{tar,i}} - m_i) + L''(m_i)\,\frac{(\lambda_{\mathrm{tar,i}}-m_i)^2}{2} - \frac{(\lambda_{\mathrm{tar,i}}-m_i)^2}{2s_i^2} + \text{const} \tag{26}$$

The right-hand side is now quadratic in $\lambda_{\mathrm{tar},i}$. Consequently, $P\left(\lambda_{\mathrm{tar,i}}|\mathcal{D}_i\right)$ is Gaussian, with mean and variance given by equations (27a) and (27b):

$$\mathbb{E}[\lambda_{\mathrm{tar,i}}|\mathcal{D}_i] = m_i + \frac{s_i^2 L'(m_i)}{1 - s_i^2 L''(m_i)} \approx m_i + s_i^2 L'(m_i) \tag{27a}$$

$$\mathrm{Var}\left[\lambda_{\mathrm{tar,i}}|\mathcal{D}_i\right] = \mathrm{s_i^2} + \frac{\mathrm{s_i^4 L''(m_i)}}{1 - \mathrm{s_i^2 L''(m_i)}} \approx \mathrm{s_i^2} + \mathrm{s_i^4 L''(m_i)} \tag{27b}$$

To derive the approximation expressions, we assumed $s_i^2|L''(m_i)| \ll 1$. This holds in the limit of slowly varying log likelihood, which we assumed throughout our analysis.

Equation (27) tells us how to incorporate new data; we now need to incorporate random drift, via the integral in equation (21). From equation (11), we see that $P\left(\lambda_{\mathrm{tar,i}}(t+1)|\lambda_{\mathrm{tar,i}}\right)$ is Gaussian, so the integral is straightforward, giving equations (28a) and (28b):

$$m_i(t+1) = \left(1 - \frac{1}{\tau}\right)\mathbb{E}[\lambda_{\mathrm{tar,i}}|\mathcal{D}_i] + \frac{m_{\mathrm{prior}}}{\tau} \tag{28a}$$

$$s_i^2(t+1) = \left(1 - \frac{1}{\tau}\right)^2 \mathrm{Var}\left[\lambda_{\mathrm{tar,i}}|\mathcal{D}_{\mathrm{i}}\right] + \frac{2\mathrm{s}^2_{\mathrm{prior}}}{\tau} \tag{28b}$$

Inserting equation (27) into equation (28), and working to lowest nonvanishing order in $1/\tau$, $s_i L'(m_i)$ and $s_i^2 L''(m_i)$, we arrived at our final update equations (29a) and (29b):

$$\Delta m_i = s_i^2 L'(m_i) - \frac{m_i - m_{\mathrm{prior}}}{\tau} \tag{29a}$$

$$\Delta s_i^2 = s_i^4 L''(m_i) - \frac{2(s_i^2 - s^2_{\mathrm{prior}})}{\tau} \tag{29b}$$

where $\Delta m_i \equiv m_i(t+1) - m_i$ and $\Delta s_i^2 \equiv s_i^2(t+1) - s_i^2$. Thus, to update the mean and variance, all we have to do is compute the log likelihood and take the first and second derivatives. Below, we outline how to do that; additional details are provided in Supplementary Note 1. Note that equality in these expressions (and many that follow) is shorthand for equality under the assumptions and approximations of our model.

Single-neuron learning rules for our three models. According to the above analysis (equation (29)), to determine the update rules, we just need the log likelihood of the current data, $d_i(t)$, given the error signal (f_{lin}, f_{cb} or f_{ri}). Computation of the log likelihood is nontrivial, as several approximations are required; however, it is not hard to get an intuitive understanding of its form.

Using equation (18) for the data, d_i, the likelihood (the probability of the data given $w_{\mathrm{tar},i}$) may be written as equation (30):

$$P\left(d_i|\lambda_{\mathrm{tar,i}}\right) = P\left(f|x_i,w_i,\lambda_{\mathrm{tar,i}}\right)P\left(x_i,w_i|\lambda_{\mathrm{tar,i}}\right) \propto P\left(f|x_i,w_i,\lambda_{\mathrm{tar,i}}\right) \tag{30}$$

where we are able to drop the term $P\left(x_i,w_i|\lambda_{\mathrm{tar,i}}\right)$ because without an error signal, x_i and w_i do not provide any information about $\lambda_{\mathrm{tar},i}$. For all of our feedback signals, f is a function of f_{lin}; we take advantage of this to write equation (31):

$$P\left(f|x_i,w_i,\lambda_{\mathrm{tar,i}}\right) = \int df_{\mathrm{lin}}\; P\left(f|f_{\mathrm{lin}}\right)P\left(f_{\mathrm{lin}}|x_i,w_i,\lambda_{\mathrm{tar,i}}\right) \tag{31}$$

We focus here on computing $P\left(f_{\mathrm{lin}}|x_i,w_i,\lambda_{\mathrm{tar,i}}\right)$, and describe the integral in Supplementary Note 1. Using equations (1), (2) and (5), we have equation (32):

$$f_{\mathrm{lin}} = (w_{\mathrm{tar,i}} - w_i)x_i + \sum_{j\neq i}(w_{\mathrm{tar,j}} - w_j)x_j + \xi_V + \xi_\delta \tag{32}$$

For synapse i, all the terms in the sum over j are unobserved, and so correspond to noise. By the central limit theorem (and the assumed independence of the synapses), that noise is Gaussian; we take the added noise, ξ_V and ξ_δ, to be Gaussian as well, with total variance given by equation (33):

$$\sigma_0^2 \equiv \mathrm{var}\left[\xi_\delta\right] + \mathrm{var}\left[\xi_{\mathrm{V}}\right] \tag{33}$$

Consequently, we may write equations (34) and (35):

$$f_{\mathrm{lin}}|w_i,x_i,\lambda_{\mathrm{tar,i}} \sim \mathcal{N}\left((w_{\mathrm{tar,i}} - w_i)x_i, \sigma^2_{\delta,i}\right) \tag{34}$$

where

$$\sigma^2_{\delta,i} \equiv \mathrm{var}\left[\sum_{\mathrm{j}\neq\mathrm{i}}(\mathrm{w_{tar,j}} - \mathrm{w_j})\mathrm{x_j} \mid \mathcal{D}_{\mathrm{i}}(\mathrm{t}-1)\right] + \sigma_0^2 \tag{35}$$

The quantity $\sigma^2_{\delta,i}$ depends on synapse, i. However, in the limit where there are a large number of synapses, that dependence is weak. We thus approximated this by including all terms in the sum over j, which we denoted σ^2_δ, as equation (36):

$$\sigma^2_{\delta,i} \approx \sigma^2_\delta \equiv \mathrm{Var}\left[\sum_{\mathrm{j}}(\mathrm{w_{tar,i}} - \mathrm{w_j})\mathrm{x_j} \mid \mathcal{D}_{\mathrm{i}}(\mathrm{t}-1)\right] + \sigma_0^2 \tag{36}$$

Under this approximation, we describe equation (37):

$$f_{\mathrm{lin}}|w_i,x_i,\lambda_{\mathrm{tar,i}} \sim \mathcal{N}\left((\pm e^{\lambda_{\mathrm{tar,i}}} - w_i)x_i, \sigma^2_\delta\right) \tag{37}$$

For much of our analysis, we used the value of σ^2_δ under the prior. That quantity, denoted $\sigma^2_{\delta 0}$, is given by equation (38):

$$\sigma^2_{\delta 0} \equiv (\sigma^2_{\mathrm{prior}} + \sigma^2_{w,\ \mathrm{prior}})\sum_j \nu_j \Delta t(1 - \nu_j \Delta t) + \sigma_0^2 \tag{38}$$

where the term $\nu_j\Delta t(1-\nu_j\Delta t)$ comes from the Bernoulli statistics of x_j (equation (17)), and σ^2_{prior} and $\sigma^2_{w,\text{prior}}$ are the variances of $w_{\text{tar},i}$ and w_i under the prior. The latter, $\sigma^2_{w,\text{prior}}$, depends on whether or not we are sampling, according to equation (39):

$$\sigma^2_{w,\ \text{prior}} \equiv \begin{cases} \sigma^2_{\text{prior}} & \text{Synaptic sampling} \\ k\ \mu_{\text{prior}} & \text{Variance proportional to the mean} \end{cases} \tag{39}$$

The prior mean and variance of the weights in terms of the log weights (equation (13) and Supplementary Table 1) are given by equations (40a) and (40b):

$$\mu_{\text{prior}} \equiv e^{m_{\text{prior}}+s^2_{\text{prior}}/2} \tag{40a}$$

$$\sigma^2_{\text{prior}} \equiv \mu^2_{\text{prior}}\left[e^{s^2_{\text{prior}}}-1\right] \tag{40b}$$

This analysis tells us that the distribution $P\left(f_{\text{lin}}|w_i,x_i,\lambda_{\text{tar,i}}\right)$ is Gaussian in $e^{\lambda_{\text{tar,i}}}$. To determine the learning rules, all we have to do is insert equation (37) into equation (31), perform an integral, take the log, compute the first two derivatives and evaluate them at m_i (equation (29)). These steps, described in Supplementary Note 1, are not completely straightforward, as various approximations must be made. However, from a conceptual point of view, the approximations do not add much. Thus, here we simply provide the results.

Linear feedback. The Bayesian update rules are given by equations (41a) and (41b):

$$\Delta m_i = \left(\frac{s_i^2\mu_i}{\sigma^2_{\delta 0}}\right)x_i f_{\text{lin}} - \frac{1}{\tau}\left(m_i - m_{\text{prior}}\right) \tag{41a}$$

$$\Delta s_i^2 = -\left(\frac{s_i^4\mu_i^2}{\sigma^2_{\delta 0}}\right)x_i^2 - \frac{2}{\tau}\left(s_i^2 - s^2_{\text{prior}}\right) \tag{41b}$$

For classical learning, we used the delta rule (equation (3)), according to equation (42):

$$\Delta w_i = \eta\ x_i f_{\text{lin}} \tag{42}$$

Note that we excluded weight drift in the classical learning rate (both here and below), as weight drift was derived using Bayesian analysis, and has no classical counterpart.

Cerebellar feedback. The Bayesian update rules are given in equations (43a) and (43b):

$$\Delta m_i = \left(\frac{s_i^2\mu_i}{\sigma^2_{\delta 0}}\right)x_i\sigma_{\delta 0}(2f_{\text{cb}}-1)\ \frac{\mathcal{N}(\theta_{\text{cb}})}{\Phi(\theta_{\text{cb}})} - \frac{1}{\tau}(m_i - m_{\text{prior}}) \tag{43a}$$

$$\Delta s_i^2 = -\left(\frac{s_i^4\mu_i^2}{\sigma^2_{\delta 0}}\right)x_i^2\ \frac{\mathcal{N}(\theta_{\text{cb}})}{\Phi(\theta_{\text{cb}})}\left[\theta_{\text{cb}} + \frac{\mathcal{N}(\theta_{\text{cb}})}{\Phi(\theta_{\text{cb}})}\right] - \frac{2}{\tau}\ (s_i^2 - s^2_{\text{prior}}) \tag{43b}$$

where Φ and (in a slight abuse of notation) $\mathcal{N}$ are the cumulative normal and normal functions, respectively, according to equations (44a) and (44b):

$$\Phi(z) \equiv \int_{-\infty}^{z} du\ \frac{e^{-u^2/2}}{(2\pi)^{1/2}} \tag{44a}$$

$$\mathcal{N}(z) \equiv \frac{e^{-z^2/2}}{(2\pi)^{1/2}} \tag{44b}$$

and θ_{cb} is given in terms of the threshold θ, as equation (45):

$$\theta_{\text{cb}} \equiv (1-2f_{\text{cb}})\ \frac{\theta}{\sigma_{\delta 0}} \tag{45}$$

For classical learning, we absorbed most of the prefactor in the above mean update into a fixed learning rate, described in equation (46):

$$\Delta w_i = \eta(2f_{\text{cb}}-1)x_i\ \frac{\mathcal{N}(\theta_{\text{cb}})}{\Phi(\theta_{\text{cb}})} \tag{46}$$

Reinforcement learning. The Bayesian update rules are described in equations (47a) and (47b):

$$\Delta m_i = \left(\frac{s_i^2\mu_i}{\sigma_\delta^2}\right)\left(\frac{f_{\text{rl}}^2}{\sigma_\delta^2}-1\right)x_i^2(\mu_i - w_i) - \frac{1}{\tau}(m_i - m_{\text{prior}}) \tag{47a}$$

$$\Delta s_i^2 = -\left(\frac{s_i^4\mu_i^2}{\sigma_\delta^2}\right)\ \left(1-\frac{f_{\text{rl}}^2}{\sigma_\delta^2}\right)x_i^2 - \frac{2}{\tau}\ (s_i^2 - s^2_{\text{prior}}) \tag{47b}$$

This learning rule appears nonlocal, as it depends on σ_δ^2, which in turn depends on all the synapses (equation (36)). However, we made it local by changing the feedback signal to $(1-f_{\text{rl}}^2/\sigma_\delta^2)/\sigma_\delta^2$. For classical learning, we again absorbed most of the prefactor into the learning rate, according to equations (48):

$$\Delta w_i = \eta x_i(f_{\text{rl}}\tanh((\mu_i - w_i)x_i f_{\text{rl}}/\sigma_\delta^2) - (\mu_i - w_i)x_i) \tag{48}$$

Note that 'tanh' appears in the classical, but not Bayesian, learning rules. That is because, for the Bayesian learning rules, we made the approximation $\tanh((\mu_i - w_i)x_i f_{\text{rl}}/\sigma_\delta^2) \approx (\mu_i - w_i)x_i f_{\text{rl}}/\sigma_\delta^2$. This approximation, however, made the classical learning rule unstable.

Recurrent neural network learning rules. So far, we have focused on single neurons. Here we generalize to the more realistic scenario in which the output weights of a recurrent network are trained to produce a time-dependent target function. We will assume that the network, which contains N neurons, evolves according to equations (49a), (49b) and (49c):

$$\tau_m\frac{dv_i}{dt} = -v_i + \sum_{j=1}^{N} J_{ij}x_j + A_i V(t) + I_i(t) \tag{49a}$$

$$x_j = \tanh(v_j) \tag{49b}$$

$$V(t) = \sum_{j=1}^{N} w_j x_j \tag{49c}$$

We interpret v_i as the membrane potential and x_j as the firing rate relative to baseline. The recurrent weights, J_{ij}, and feedback weights, A_i, are fixed. Parameters of the network and details of the simulations are available in Supplementary Note 3.

The goal of the network is to minimize the distance between $V(t)$ and some target function, denoted $V_{\text{tar}}(t)$; that is, to minimize the error $\delta(t)$, defined, as in equation (2), to be (equation (50)):

$$\delta(t) \equiv V_{\text{tar}}(t) - V(t) \tag{50}$$

As with single neurons, we used a Bayesian approach. There are, however, two important differences: First, we do not know the target weights (we do not specify them; instead, they must be learned). We assumed, however, that target weights exist, meaning we can write equation (51):

$$\delta(t) = \sum_j (w_{\text{tar,j}}(t) - w_j(t))x_j(t) \tag{51}$$

The second difference is that the feedback signal, $\delta(t)$, is a continuous function of time. Consequently, information at times t and $t+dt$ is largely redundant. To deal with this redundancy, we make several approximations. First, rather than updating the weights continuously, we update them at times separated by Δt. Bayes' theorem, described in equation (20), then becomes equation (52):

$$P\left(w_{\text{tar,i}}|\mathcal{D}_i\right) \propto P\left(d_i|w_{\text{tar,i}},\mathcal{D}_i(t-\Delta t)\right)P\left(w_{\text{tar,i}}|\mathcal{D}_i(t-\Delta t)\right) \tag{52}$$

where, as in the single-neuron case, the data for synapse i is the presynaptic input, x_i, the actual weight, w_i, and the error signal, δ. To derive this expression, we made two simplifications: (1) we did not add noise to the error signal, so the synapses see δ rather than f_{lin}, and (2) we did not enforce Dale's law, so the weights can change sign. Because of the latter simplification, we let the weights, rather than the log weights, have a Gaussian distribution; that is why equation (52) is written in terms of $w_{\text{tar},i}$ rather than $\lambda_{\text{tar},i}$.

In one respect, the analysis is simpler than it was for single neurons. Because the target weights do not evolve over time (see comments at the end of 'Target weights'), we can avoid the integral in equation (21). However, in another respect, it is more complicated; as just discussed, the likelihood (the first term on the right-hand side of equation (52)) depends on past data. An exact treatment in this regime is beyond the scope of this work. Instead, we chose the time step, Δt, so it is much larger than the correlation time of $\delta(t)$, allowing us to drop the dependence on $\mathcal{D}_i(t-\Delta t)$ in the likelihood, giving us equation (53):

$$P\left(d_i|w_{\text{tar,i}},\mathcal{D}_i(t-\Delta t)\right) \approx P\left(d_i|w_{\text{tar,i}}\right) \propto P\left(\delta|x_i,w_i,w_{\text{tar,i}}\right) \tag{53}$$

where, as in equation (30), we used the fact that without an error signal, x_i and w_i do not provide any information about $w_{\text{tar},i}$.

While this gives us a very good approximation to the likelihood if Δt is large, large Δt means that updates would be made very rarely, and so learning would be slow. We thus made a second approximation, to optimize our learning rule (via numerical simulation, as discussed below) with respect to Δt. This gives us approximate Bayesian update rules, which presumably could be improved upon. However, as we will see, the approximate update rules already outperform the classical ones by an order of magnitude. Thus, any improvement would only make the case for Bayesian plasticity stronger.

To find an expression for $P\left(\delta|x_i,w_i,w_{\text{tar,i}}\right)$, we again write δ as in equation (32) (but without noise, so $\xi_\delta = 0$, which reduces σ_0^2 (equation (33)). Now however, we

are interested in the log likelihood with respect to the target weights $w_{\text{tar},i}$, rather than the log of the target weights $\lambda_{\text{tar},i}$ (as mentioned above). Thus, the distribution over δ simplified relative to equation (37) is given by equation (54):

$$\delta|w_i, x_i, w_{\text{tar,i}} \sim \mathcal{N}\left((w_{\text{tar,i}} - w_i)x_i, \sigma_\delta^2\right) \tag{54}$$

As above, we made the approximation $\sigma_{\delta,i}^2 \approx \sigma_\delta^2$ (equation (36)). It is now straightforward to write down the log likelihood, according to equation (55):

$$L(w_{\text{tar,i}}) = -\frac{\left(\delta - \left(w_{\text{tar,i}} - w_i\right)x_i\right)^2}{2\sigma_\delta^2} + \text{ const} \tag{55}$$

The first and second derivatives evaluated at $w_{\text{tar},i} = \mu_i$ are given by equations (56a) and (56b):

$$L'(\mu_i) = \frac{(\delta - (\mu_i - w_i)x_i)x_i}{\sigma_\delta^2} \approx \frac{\delta x_i}{\sigma_\delta^2} \tag{56a}$$

$$L''(\mu_i) = -\frac{x_i^2}{\sigma_\delta^2} \tag{56b}$$

(We are justified in dropping the term $(\mu_i - w_i)x_i$ because it is a factor of $\sqrt{n}$ smaller than δ. That follows because σ_δ^2, which is the variance of δ, is $\mathcal{O}(n)$ (equation (36)).) Inserting these expressions into equation (29) (with τ taken to ∞ because, as discussed in 'Target weights', we are assuming the target weights do not drift over time), we have equations (57a) and (57b):

$$\Delta\mu_i = \frac{\sigma_i^2}{\sigma_\delta^2}\ \delta x_i \tag{57a}$$

$$\Delta\sigma_i^2 = -\frac{\sigma_i^4}{\sigma_\delta^2}\ x_i^2 \tag{57b}$$

where $\Delta\mu_i = \mu_i(t + \Delta t) - \mu_i(t)$ and similarly for $\Delta\sigma_i^2$.

Primarily for convenience, we made a third approximation, which is to update the weights continuously rather than at discrete points separated by Δt. To do that, we simply make the approximation $\Delta\mu_i \approx \Delta t\, d\mu_i/dt$, and similarly for $\Delta\sigma_i^2$. This allows us to turn the update rules into ordinary differential equations (58a) and (58b):

$$\frac{d\mu_i}{dt} = \frac{1}{\Delta t}\ \frac{\sigma_i^2}{\sigma_\delta^2}\ \delta x_i \tag{58a}$$

$$\frac{d\sigma_i^2}{dt} = -\frac{1}{\Delta t}\ \frac{\sigma_i^4}{\sigma_\delta^2}\ x_i^2 \tag{58b}$$

Then, defining equation (59):

$$\eta_i \equiv \frac{\sigma_i^2}{\sigma_\delta^2 \Delta t} \tag{59}$$

inserting this into into equation (58) and, in our fourth approximation, ignoring the time dependence in σ_δ^2, those equations simplify to equations (60a) and (60b):

$$\frac{d\mu_i}{dt} = \eta_i\ \delta x_i \tag{60a}$$

$$\frac{d\eta_i}{dt} = -\eta_i^2\ x_i^2 \tag{60b}$$

Optimizing over Δt corresponds to optimizing over the initial value of η_i, which we assumed is the same for all i. This optimization is performed via numerical simulations.

For the classical learning rules, we dropped equation (60b) and fixed η_i to the same value for all synapses.

Reporting Summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

Data are available for download at: https://github.com/Jegmi/the-bayesian-synapse/releases/tag/v2/.

Code availability

Code is available for download at: https://github.com/Jegmi/the-bayesian-synapse/releases/tag/v2/.

References

47. Loewenstein, Y., Kuras, A. & Rumpel, S. Multiplicative dynamics underlie the emergence of the log-normal distribution of spine sizes in the neocortex in vivo. *J. Neurosci.* **31**, 9481–9488 (2011).
48. Matsuzaki, M., Honkura, N., Ellis-Davies, G. C. & Kasai, H. Structural basis of long-term potentiation in single dendritic spines. *Nature* **429**, 761–766 (2004).
49. Song, S., Sjöström, P. J., Reigl, M., Nelson, S. & Chklovskii, D. B. Highly nonrandom features of synaptic connectivity in local cortical circuits. *PLoS Biol.* **3**, e68 (2005).
50. O'Connor, D. H., Peron, S. P., Huber, D. & Svoboda, K. Neural activity in barrel cortex underlying vibrissa-based object localization in mice. *Neuron* **67**, 1048–1061 (2010).
51. Mizuseki, K. & Buzsáki, G. Preconfigured, skewed distribution of firing rates in the hippocampus and entorhinal cortex. *Cell Rep.* **4**, 1010–1021 (2013).
52. Minka, T. P. A family of algorithms for approximate Bayesian inference. Dissertation, Massachusetts Institute of Technology (2001).

Acknowledgements

L.A. and P.E.L. were supported by the Gatsby Charitable Foundation. P.E.L. was also supported by the Wellcome Trust (110114/Z/15/Z). J.J. and J.-P.P. were supported by the Swiss National Science Foundation (PP00P3 150637 and 31003A 175644). J.A.M. was supported by University College London (UCL) Graduate Research and UCL Overseas Research Scholarships. A.P. was supported by a grant from the Simons Collaboration for the Global Brain and the Swiss National Foundation (31003A 165831).

Author contributions

A.P. and P.E.L. were involved in the initial formulation of the problem. L.A. and P.E.L. conducted the theoretical development. L.A., J.J. and J.A.M. performed the simulations and data analysis. P.E.L., J.-P.P. and A.P. wrote the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41593-021-00809-5.

Correspondence and requests for materials should be addressed to L.A.

Peer review information *Nature Neuroscience* thanks the anonymous reviewers for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

nature research

Corresponding author(s): Laurence Aitchison

Last updated by author(s): Dec 26, 2020

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☒	☐	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☐	☒	A description of all covariates tested
☐	☒	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☐	☒	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☐	☒	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☐	☒	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	We did not collect data
Data analysis	https://github.com/Jegmi/the-bayesian-synapse/releases/tag/v2

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

https://github.com/Jegmi/the-bayesian-synapse/releases/tag/v2

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	136; We did not collect data, only used pre-existing data.
Data exclusions	N/A; We did not collect data, only used pre-existing data.
Replication	N/A; We did not collect data, only used pre-existing data.
Randomization	N/A; We did not collect data, only used pre-existing data.
Blinding	N/A; We did not collect data, only used pre-existing data.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Human research participants
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging