



Burst-dependent synaptic plasticity can coordinate learning in hierarchical circuits

Alexandre Payeur^{ID 1,2,3,12,13}, Jordan Guerguiev^{4,5,13}, Friedemann Zenke^{ID 6}, Blake A. Richards^{ID 7,8,9,10,14} and Richard Naud^{ID 1,2,3,11,14}

Synaptic plasticity is believed to be a key physiological mechanism for learning. It is well established that it depends on pre- and postsynaptic activity. However, models that rely solely on pre- and postsynaptic activity for synaptic changes have, so far, not been able to account for learning complex tasks that demand credit assignment in hierarchical networks. Here we show that if synaptic plasticity is regulated by high-frequency bursts of spikes, then pyramidal neurons higher in a hierarchical circuit can coordinate the plasticity of lower-level connections. Using simulations and mathematical analyses, we demonstrate that, when paired with short-term synaptic dynamics, regenerative activity in the apical dendrites and synaptic plasticity in feedback pathways, a burst-dependent learning rule can solve challenging tasks that require deep network architectures. Our results demonstrate that well-known properties of dendrites, synapses and synaptic plasticity are sufficient to enable sophisticated learning in hierarchical circuits.

The current canonical model of synaptic plasticity in the cortex is based on the co-occurrence of activity on the two sides of the synapse, pre- and postsynaptic¹. The occurrence of either long-term depression (LTD) or long-term potentiation (LTP) is controlled by specific features of pre- and postsynaptic activity^{2–9} and a more global state of neuromodulation^{10–12}. However, local learning rules by themselves do not provide a guarantee that behavioral metrics will improve. With neuromodulation driven by an external reward/punishment mechanism, this guarantee is achievable¹³. But, such learning is very slow in tasks that require large or deep networks because a global signal provides very limited information to neurons deep in a hierarchy^{14,15}. Thus, an outstanding question is (Fig. 1): how can neurons high-up in a hierarchy signal to other neurons—sometimes multiple synapses lower—whether to engage in LTP or LTD to improve behavior? This question is sometimes referred to as the ‘credit assignment problem’: essentially, how can we assign credit for any errors or successes to neurons that are multiple synapses away from the output¹⁶?

In machine learning, the credit assignment problem is typically solved with the backpropagation-of-error algorithm (backprop¹⁷), which explicitly uses gradient information in a biologically implausible manner¹⁵ to calculate synaptic weight updates. Many previous studies have attempted to capture the credit assignment properties of backprop with more biologically plausible implementations in the hope that a biological model could match backprop’s learning performance¹⁵. However, a problem with most of these models is that there is always an implicit assumption that during some phases of learning no sensory stimuli are processed, that is, the models are not ‘online’ in their learning, which is problematic for both biological plausibility and for potential future development of low-energy neuromorphic computing devices. Moreover, there are several

well-established properties of real neurons, including nonlinearities in the apical dendrites¹⁸, short-term synaptic plasticity (STP)¹⁹ and inhibitory microcircuits that are ignored. None of the previous studies successfully incorporated all of these features to perform online credit assignment (Supplementary Table 1). Furthermore, none of these models captured the frequency dependence of synaptic plasticity, which is a very well-established property of LTP/LTD^{4,6,7,20,21}.

As established in nonhierarchical systems, such as the electro-sensory lateral line lobe of the electric fish^{22,23} or the cerebellum²⁴, feedback connections on dendrites are well-poised to orchestrate learning. But for credit assignment in hierarchical networks, these connections should obey four constraints: (1) feedback must steer the sign and magnitude of plasticity. (2) Feedback signals from higher-order areas should be multiplexed with feedforward signals from lower-order areas so that credit information can percolate down the hierarchy with minimal disruption to sensory information. (3) There should be some degree of alignment between feedback connections and feedforward connections. (4) Integration of credit-carrying feedback signals should be close to linear and avoid saturation (that is, feedback signals should be linear with respect to any credit information). Experimental and theoretical work have addressed steering^{9,25}, multiplexing^{26–28}, alignment^{15,29,30} or linearity³¹ in isolation. But, it remains unclear whether a single set of cellular and subcellular mechanisms can address all four requirements for orchestrating learning in cortical hierarchies efficiently.

Here, we address the credit assignment problem with a spike-based learning rule that models how high-frequency bursts determine the sign of synaptic plasticity^{4,6,7,20,21}. Guided by the underlying philosophy first espoused by the work of Kording and König³² that the unique properties of pyramidal neurons may contain a solution to biologically plausible credit assignment, we show that

¹Department of Cellular and Molecular Medicine, University of Ottawa, Ottawa, ON, Canada. ²Ottawa Brain and Mind Institute, University of Ottawa, Ottawa, ON, Canada. ³Centre for Neural Dynamics, University of Ottawa, Ottawa, ON, Canada. ⁴Department of Biological Sciences, University of Toronto Scarborough, Toronto, ON, Canada. ⁵Department of Cell and Systems Biology, University of Toronto, Toronto, ON, Canada. ⁶Friedrich Miescher Institute for Biomedical Research, Basel, Switzerland. ⁷Mila, Montréal, QC, Canada. ⁸Department of Neurology and Neurosurgery, McGill University, Montréal, QC, Canada. ⁹School of Computer Science, McGill University, Montréal, QC, Canada. ¹⁰Learning in Machines and Brains Program, Canadian Institute for Advanced Research, Toronto, ON, Canada. ¹¹Department of Physics, University of Ottawa, Ottawa, ON, Canada. ¹²Present address: University of Montréal and Mila, Montréal, QC, Canada. ¹³These authors contributed equally: Alexandre Payeur, Jordan Guerguiev. ¹⁴These authors jointly supervised this work: Blake A. Richards, Richard Naud. e-mail: blake.richards@mila.quebec; rnaud@uottawa.ca

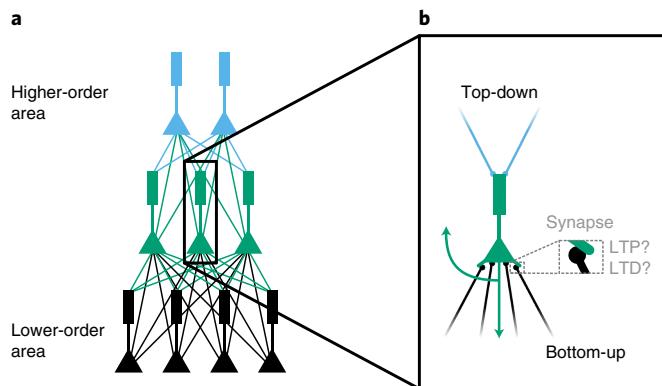


Fig. 1 | The credit assignment problem for hierarchical networks.

a, Illustration of a hierarchical neural network with feedforward and feedback connections. **b**, For an orchestration of learning in this network, the representations in higher-level neurons should steer the plasticity of connections at a lower level.

combining properties of apical dendrites¹⁸ with our burst-dependent learning rule allows feedback to steer plasticity. We further show that feedback information can be multiplexed across multiple levels of a hierarchy when feedforward and feedback connections have distinct STP^{33,34}. Using spiking simulations, we demonstrate that these mechanisms can be used to coordinate learning across a hierarchical circuit in a fully online manner. We also show that a coarse-grained equivalent of these dynamical properties will, on average, lead to learning that approximates loss-function gradients as used in backprop. We further show that this biological approximation to loss-function gradients is improved by a burst-dependent learning rule performing the alignment of feedback weights with feedforward weights, as well as recurrent connections that linearize credit signals. Finally, we show that networks trained with these mechanisms can learn to classify complex image patterns with high accuracy. Altogether, our work highlights that well-known properties of dendritic excitability, synaptic transmission, STP, inhibitory microcircuits and burst-dependent synaptic plasticity are sufficient to solve the credit assignment problem in hierarchical networks.

Results

A burst-dependent rule enables top-down steering of plasticity. Experimental work has demonstrated that the sign of plasticity can be determined by patterns of pre- and postsynaptic activity. The most common formulation of this is spike-timing-dependent plasticity (STDP), wherein the timing of pairs of pre- and postsynaptic spikes is what determines whether LTP or LTD occurs³. However, there is also evidence suggesting that in many circuits, particularly mature ones³⁵, the principal determinant of plasticity is the level of postsynaptic depolarization, with large depolarization leading to LTP and small depolarization leading to LTD^{2,4,5}, which is a direct consequence of the dynamics of NMDA receptor-dependent calcium influx³⁶. One of the easiest ways to induce large magnitude depolarization in dendrites is via high-frequency bursts of back-propagating action potentials³⁷ and, therefore, the degree of postsynaptic bursting controls plasticity^{17,21}. Since bursting may be modulated by feedback synapses on apical dendrites^{18,38}, feedback could control plasticity in the basal dendrites via control of bursting. Thus, in considering potential mechanisms for credit assignment during top-down supervised learning, the burst-dependence of synaptic plasticity appears to be a natural starting point.

To explore how high-frequency bursting could control learning in biological neural networks, we formulated a burst-dependent plasticity rule as an abstraction of the experimental data. We consider

a burst to be any occurrence of at least two spikes with a short inter-spike interval. Following ref.²⁸, we further define an event as either an isolated single spike or a burst. The learning rule states that the change over time (t) of a synaptic weight between postsynaptic neuron i and presynaptic neuron j , $d w_{ij}/dt$, results from a combination of an eligibility trace of presynaptic activity, \tilde{E}_j , and the potentiating (or depressing) effect of bursts $B_i(t)$ (or events $E_i(t)$) of the postsynaptic cell (Fig. 2a and Methods):

$$\frac{d w_{ij}}{dt} = \eta [B_i(t) - \bar{P}_i(t)E_i(t)]\tilde{E}_j(t). \quad (1)$$

The constant η is the learning rate. The variable $\bar{P}_i \in [0, 1]$ is an exponential moving average of the proportion of events that are bursts in postsynaptic neuron i , with a slow ($\tau_{avg} \approx 1–10$ s) time constant (Methods). When a postsynaptic event that is not a burst occurs, the weight decreases proportionally to $-\bar{P}_i(t)\tilde{E}_j(t) < 0$. In contrast, if a postsynaptic event is a burst, then the weight increases proportionally to $[1 - \bar{P}_i(t)]\tilde{E}_j(t) > 0$. Hence, this moving average regulates the relative strength of burst-triggered potentiation and event-triggered depression. It has been well established that such mechanisms exist in real neurons³⁹.

The plasticity rule stipulates that when a presynaptic input is paired with a postsynaptic burst LTP is induced, whereas with an isolated spike, LTD results (Fig. 2a)^{6,7,20,21,35,36}. Using this rule, we simulated a series of synaptic plasticity experiments from the experimental and computational literature. First, we examined a frequency-dependent STDP protocol⁵. We found that when the spike pairing frequency is low, LTD is produced, and when the pairing frequency is high, LTP is produced (Fig. 2b). This matches previous reports on frequency-dependent STDP and shows that a burst-dependent synaptic plasticity rule can explain these data. Then, we explored the behavior of our rule when the pre- and post-synaptic neuron fire independently according to Poisson statistics (Fig. 2c). Experimental results have established that in such a situation the postsynaptic firing rate should determine the sign of plasticity⁵. We found that a burst-dependent plasticity rule produces exactly this behavior (Fig. 2c), but with a dependence on bursting history not typically explored. Notably, contrary to the Bienenstock–Cooper–Munro model⁴⁰ where the switching point between LTD and LTP depends on a nonlinear moving average of the feedforward activity, in the present case, the adaptive threshold is a burst probability, which can be controlled independently of the feedforward activity. These results demonstrate that a burst-dependent plasticity rule is capable of linking a series of known experimental and theoretical results.

The burst-dependent rule suggests that feedback-mediated steering of plasticity could be achieved if there were a mechanism for top-down control of the likelihood of a postsynaptic burst. To illustrate this, in Fig. 2d we simulated another protocol wherein events were generated with Poisson statistics, and each event could become a burst with probability P (x axis in Fig. 2d). Manipulating this burst probability against the initial burst probability estimate ($\bar{P}_i(0) = 20\%$) controlled the occurrence of LTP and LTD, while changing the pre- and postsynaptic event rates simply modified the rate of change of the weight (but not the transition point between LTP and LTD). This shows that one way for neurons to control the sign of plasticity to ensure effective learning may be to regulate the probability of high-frequency bursts. Evidence indicates that in pyramidal neurons of sensory cortices the probability of generating high-frequency bursts is controlled by inputs to the distal apical dendrites and their activation of voltage-gated calcium channels (VGCCs)^{18,41}. Anatomical and functional data have shown that these inputs often come from higher-order cortical or thalamic regions⁴².

We wondered whether combining a burst-dependent plasticity rule with regenerative activity in apical dendrites could permit

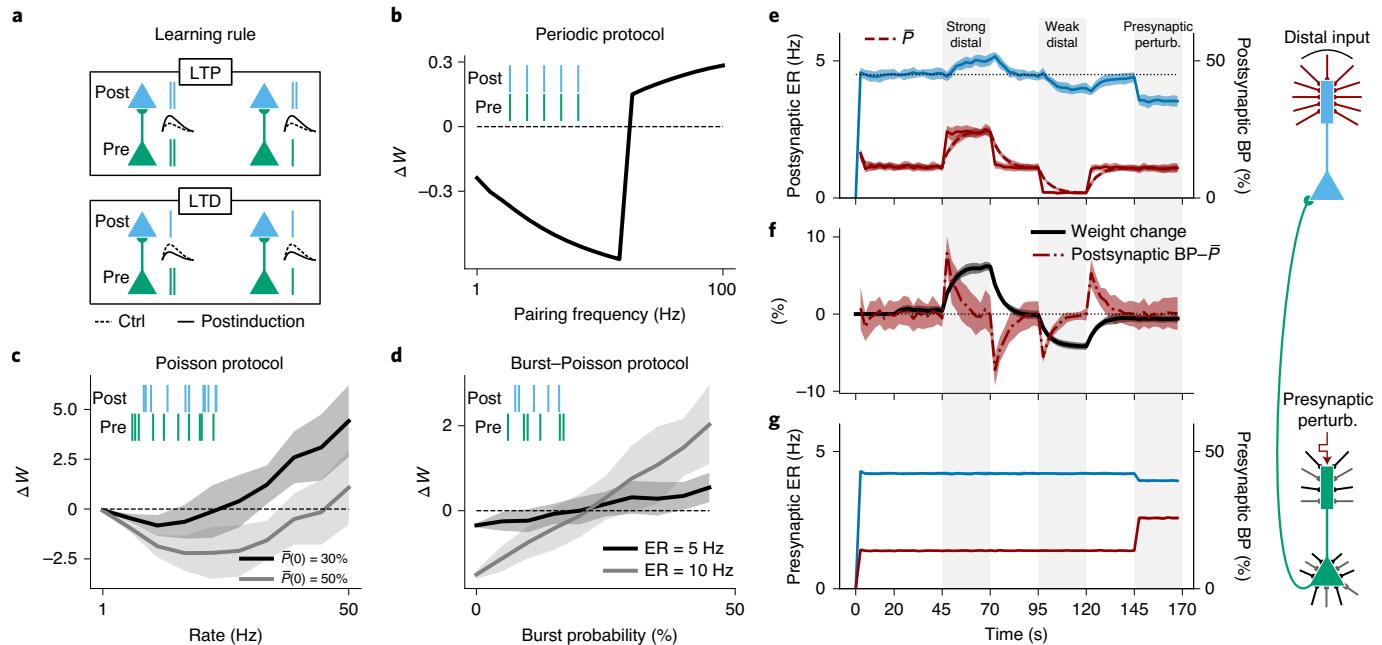


Fig. 2 | Burst-dependent plasticity rule. **a**, Schematics of the learning rule. When there is a presynaptic eligibility trace, the occurrence of a postsynaptic burst leads to potentiation (top) whereas an isolated postsynaptic spike leads to depression of the synapse (bottom). Ctrl, control. **b-d**, Net weight change for different pairing protocols. **b**, The periodic protocol consisted of 15 sequences of five pairings, separated by a 10 s interval. We used pairings with $t_{\text{post}} = t_{\text{pre}}$. **c**, For the Poisson protocol, the pre- and postsynaptic activities were Poisson spike trains with equal rates. The protocol was repeated with different initial time-averaged burst probabilities (\bar{P}). **d**, For the Burst-Poisson protocol, pre- and postsynaptic Poisson events were generated at a fixed rate (ER). For each event, a burst was produced with a probability that varied from 0 to 50%. **e-g**, Impact of distal inputs on burst probability and feedforward synaptic weights for constant presynaptic event rate. Positive distal input (45–70 s) increases burst probability (**e**) and strengthens feedforward synapses (**f**). Negative distal input (95–120 s) decreases burst probability and weakens synapses. A dendritic input to the presynaptic neuron (145–170 s) increases its burst probability and mildly affects its event rate (**g**), but does not notably change the weights (**f**). **e**, Event rate (ER, blue), burst probability (BP, solid red curve) and moving average of the BP (dashed red curve) for the postsynaptic population. The black dotted line indicates the prestimulation ER and serves as a reference for the variations of the ER with plasticity. **f**, Weight change relative to the initial average value of the weights. **g**, Same as **e**, but for the presynaptic population. For the schematic on the right-hand side, black and gray axonal terminals onto the presynaptic (green) population represent Poisson input noise; such noise is absent for the postsynaptic (light blue) population for this simulation. For **e-g**, the time constants and segment durations were selected to get a clear illustration of the learning rule. In all relevant panels, results are displayed as mean \pm s.d. over 20 realizations of the noise. Perturb., perturbation.

top-down signals to act as a ‘teaching signal’, instructing the sign of plasticity in a neuron. To explore this, we ran simulations of pyramidal neuron models with simplified VGCC kinetics in the apical dendrites (Methods). We found that by manipulating the distal inputs to the apical dendrites we could control the number of events and bursts in the neurons independently (Fig. 2e,g). The inputs to the apical dendrites in the postsynaptic neurons were what regulated the number of bursts, and this also controlled changes in the synaptic weights, through the burst-dependent learning rule. When the relative proportion of bursts increased, the synaptic weights potentiated on average, and when the relative proportion of bursts decreased, the synaptic weights depressed (Fig. 2f). Thus, in Fig. 2f, the weight increases (decreases) on average when $P - \bar{P}$ is positive (negative). Modifying the proportion of bursts in the presynaptic neurons had little effect on the weights (see the rightmost gray shaded area in Fig. 2e–g). The sign of plasticity was independent of the number of events, although the magnitude was not. Therefore, while the number of events contributed to the determination of the magnitude of changes, the top-down inputs to the apical dendrites controlled the sign of plasticity. In this way, the top-down inputs acted as a ‘teaching signal’ that determined whether LTP or LTD would occur. These results show that a burst-dependent learning rule paired with the control of bursting provided by apical dendrites enables a form of top-down steering of synaptic plasticity in an online, local and spike-based manner.

Dendrite-dependent bursting combined with short-term plasticity supports multiplexing of feedforward and feedback signals. The question that naturally arises from our finding that top-down inputs can steer synaptic plasticity via a burst-dependent rule is whether feedback can steer plasticity without affecting the communication of bottom-up signals? Using numerical simulations, we previously have demonstrated that in an ensemble of pyramidal neurons the inputs to the perisomatic and distal apical dendritic regions can be distinctly encoded using the event rate computed across the ensemble of cells and the percentage of events in the ensemble that are bursts (the ‘burst probability’), respectively²⁸. When communicated by synapses with either short-term facilitation (STF) or short-term depression (STD), this form of ‘ensemble multiplexing’ may allow top-down and bottom-up signals to be simultaneously transmitted through a hierarchy of pyramidal neurons.

To explore this possibility, we conducted simulations of two reciprocally connected ensembles of pyramidal neurons along with interneurons providing feedforward inhibition. One ensemble received currents in the perisomatic regions and projected to the perisomatic regions of the other ensemble (Fig. 3a, green ensemble). The other ensemble (Fig. 3a, light blue) received currents in the distal apical compartments and projected to the distal apical compartments of the first ensemble. As such, we considered the first ensemble to be ‘lower’ (receiving and communicating

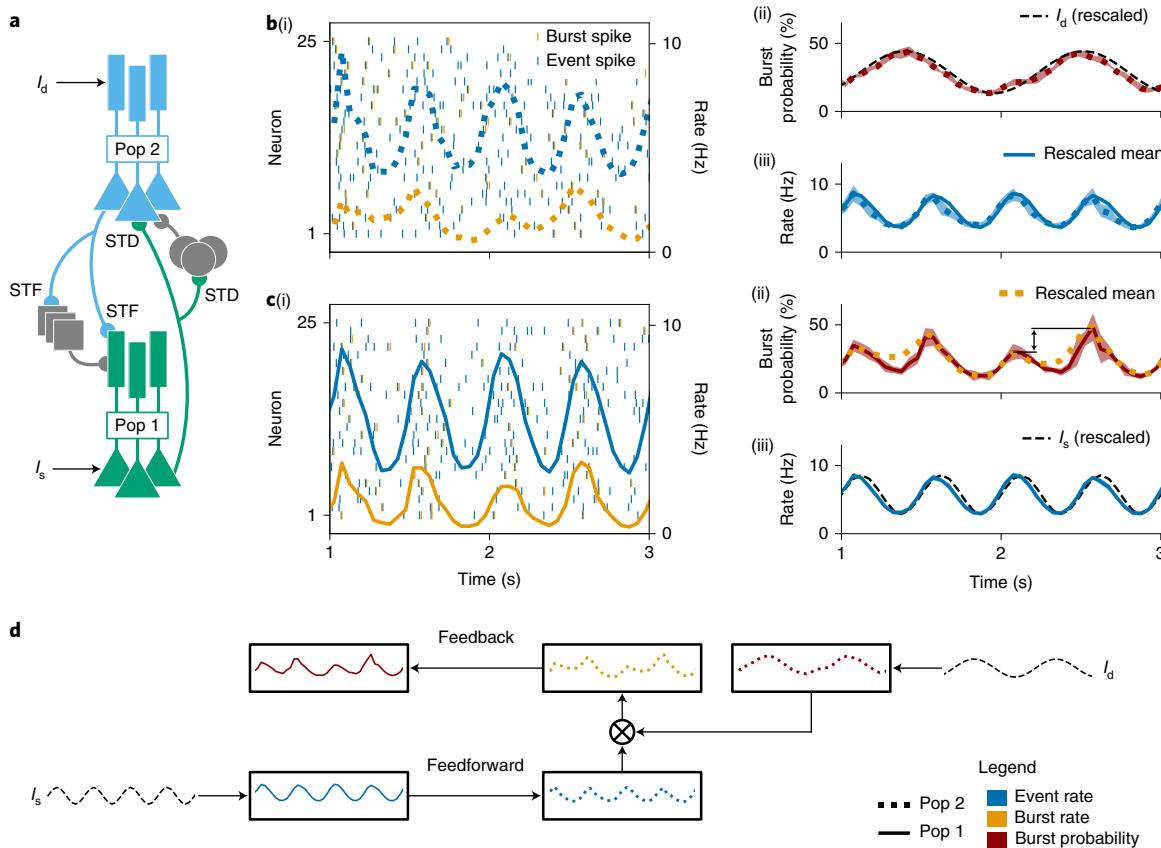


Fig. 3 | Dendrite-dependent bursting combined with short-term plasticity supports the simultaneous propagation of bottom-up and top-down signals.

a, Schematic of the network. Lower-level pyramidal neurons (pop 1, green) received a somatic current I_s and projected with STD synapses to the somatic compartments of both a higher-level pyramidal neuron population (pop 2, light blue) and to a population providing disynaptic inhibition (gray disks). The dendritic compartments of the light blue population received a current I_d . The light blue neurons innervated with STF synapses both the dendritic compartments of the green pyramidal neurons and a population providing disynaptic inhibition (gray squares). **b(i),c(i)**, Raster plots of 25 of the 4,000 neurons per pyramidal population for the light blue (**b**) and green (**c**) populations. Blue ticks show the start of an event, either a burst or an isolated spike. Orange ticks are the second spike in a burst; the remaining spikes in a burst are not shown. The corresponding population event rates (blue lines) and burst rates (orange lines) are superimposed. **b(ii),c(ii)**, Encoding performed by the light blue ensemble (pop 2). Its burst probability ((ii), dotted red line) reflects the applied dendritic current I_d (dashed black line), whereas its event rate ((iii), dotted blue line) reflects the event rate of the green population (solid blue line). **c(ii),c(iii)**, Encoding performed by the green ensemble (pop 1). Its burst probability ((ii), solid red line) reflects the burst rate (dotted orange line) of the light blue ensemble, whereas its event rate ((iii), solid blue line) reflects the applied somatic current I_s (dashed black line). Arrow highlights the amplitude modulation arising from the conjunction of top-down and bottom-up inputs. Results are displayed as mean \pm 2 s.d. over five realizations of the Poisson noise applied to all neurons in the network. In each panel, the encoded input signal has been linearly rescaled so that its range matches that of the output. For clarity, the encoded signals in **b(iii)** and **c(iii)** are displayed using their averages only (that is, without the standard deviations). The bin size used in the population averages was 50 ms. The legend applies to **b(i)-d** inclusively. **d**, Schematic illustrating information propagation in the network.

bottom-up signals), and the other to be ‘higher’ (receiving and communicating top-down signals) in the hierarchy. Furthermore, we made one key assumption in these simulations. We assumed that the synapses and cells that targeted the perisomatic regions were short-term depressing, whereas those that targeted the distal apical dendrites were short-term facilitating¹⁹. In these simulations, we observed that currents injected into the lower ensemble’s perisomatic compartments were reflected in the event rate of these neurons (Fig. 3c(iii)), although with a slight phase lead due to spike frequency adaptation. In contrast, the currents injected into the distal apical dendrites of the higher ensemble were reflected in the burst probability of these neurons (Fig. 3b(ii)). However, we also observed that these signals were simultaneously propagated up and down. Specifically, the input to the lower ensemble’s perisomatic compartments was also encoded by the higher ensemble’s event rate (Fig. 3b(iii)), whereas the burst rate of the higher ensemble was encoded by the lower ensemble’s burst probability (Fig. 3c(ii)).

In this way, the lower ensemble had access to a conjunction of the signal transmitted to the higher ensemble’s distal apical dendrites, as well as the higher ensemble’s event rate (see the arrow highlighting amplitude modulation in Fig. 3c(ii)). Thus, since the higher ensemble’s event rate is modulated by the lower ensemble’s event rate, the burst rate ultimately contains information about both the top-down and the bottom-up signals (Fig. 3d). Notably, this is important for credit assignment, as credit signals ideally are scaled by the degree to which a neuron is involved in processing a stimulus (this happens in backprop, for example). These simulations demonstrate that if bottom-up connections to perisomatic regions and perisomatic inhibition rely on STD synapses, while top-down connections to apical dendrites and distal dendritic inhibition use STF synapses, then ensembles of pyramidal neurons are capable of simultaneously processing both a top-down signal and a bottom-up signal using a combination of event rates, burst rates and burst probabilities.

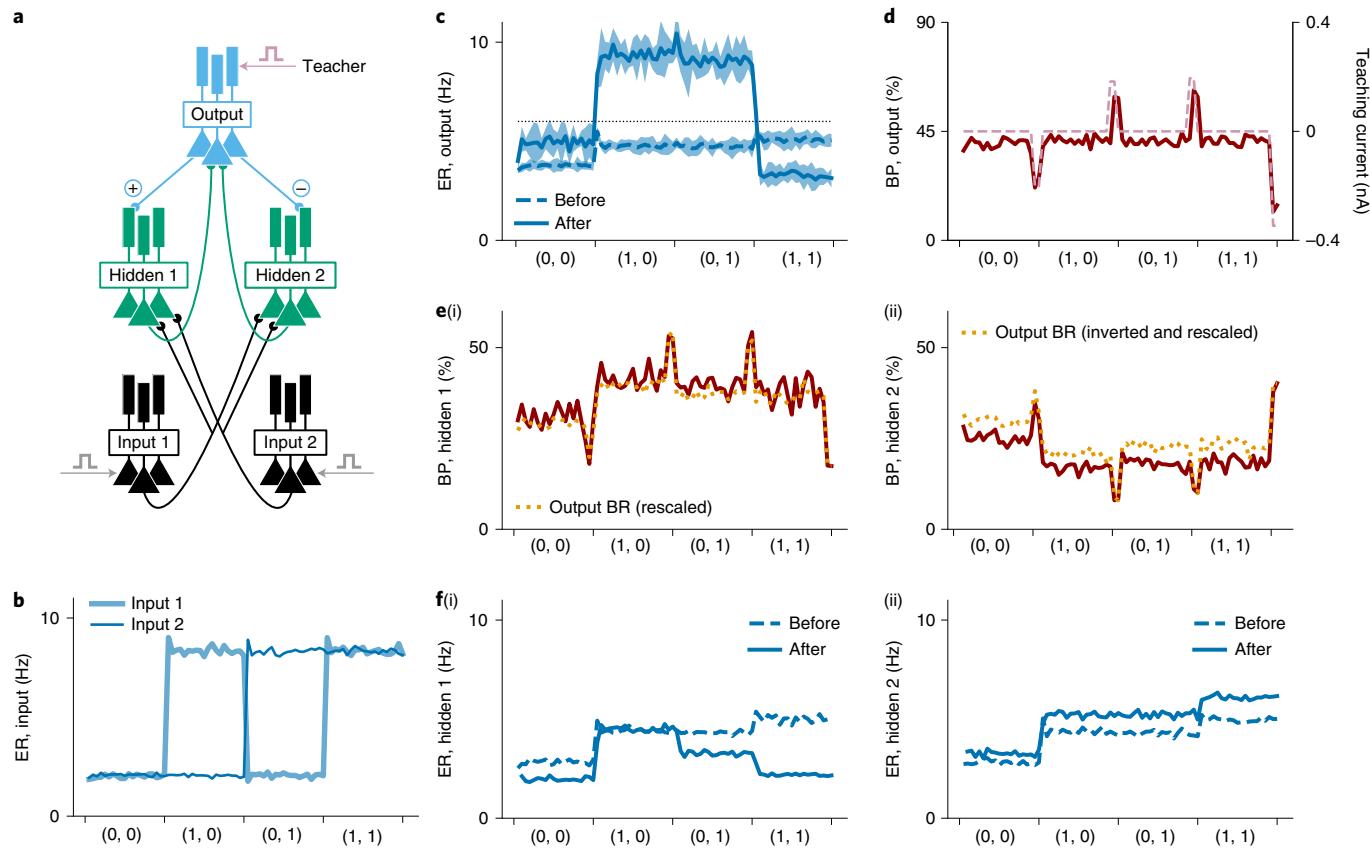


Fig. 4 | Burst-dependent plasticity can solve the credit assignment problem for the XOR task. **a**, Each neuron population contained 500 pyramidal neurons. Feedforward connections transmitted events, while feedback connections transmitted bursts. The teacher (pink arrow) was applied by injecting a hyperpolarizing current into the output ensemble's dendrites if their event rate was high in the presence of inputs that are either both active or both inactive. A depolarizing current was injected into the output ensemble's dendrites if their event rate was low when only one of the inputs was active. The activity of the input populations was controlled by somatic current injections (gray arrows). The plus and minus symbols in circles represent the initialization of the feedback synaptic weights as mainly excitatory (plus) or inhibitory (minus). **b**, Input-layer event rates (ERs) for the four input conditions presented sequentially in time. The duration of each example was 8 s. **c**, Output ER before and after learning. The output ensemble acquired strong firing (event rate above the dotted line) at the input conditions associated with ‘true’ in XOR. Results are displayed as mean \pm 2 s.d. over five random initializations of the single-neuron connectivity. In other panels, a single realization is displayed for clarity. **d**, During learning, the dendritic input (dashed pink) applied to the output ensemble's neurons controlled their burst probability (solid red) in the last 0.8 s of the input condition. **e**, During learning, the burst rate (BR) at the output layer is encoded into the BP (solid red) of the hidden layer to propagate the error. For the hidden-2 population (ii), this inherited credit is inverted with respect to that in the hidden-1 population (i). **f**, After (solid line) versus before (dashed line) learning for the hidden layer. The ER decreased in the hidden-1 population (i) but increased in the hidden-2 population (ii). The bin size used in the population averages was 0.4 s.

Combining a burst-dependent plasticity rule with short-term plasticity and apical dendrites can solve the credit assignment problem. To test whether STP, dendrite-dependent bursting and a burst-dependent learning rule can act simultaneously in a hierarchy to support learning, we built a simulation of ensembles of pyramidal neurons arranged in three layers, with two ensembles of cells at the input, one ensemble of cells at the output and two ensembles of cells in the middle (the ‘hidden’ layer, Fig. 4a). The distal dendrites of the top ensemble received ‘teaching’ signals indicating desired or undesired outputs. No other teaching signal was provided to the network. As such, the hidden-layer ensembles were informed of the suitability of their output only via the signals they received from the output ensemble's bursts. Currents injected into the somatic compartments of the input-layer populations controlled their activity levels in accordance with the learning task to be discussed below. Compared to Figs. 2 and 3, for this simulation we made a few modifications to synaptic transmission and pyramidal neuron dynamics to streamline the burst-event multiplexing and decoding (Methods). Also, we included a global gating term, $M(t)$, to prevent plasticity in the absence of teaching signals (Methods and equation (4)).

We trained our three-layer network on the exclusive OR (XOR) task, wherein the network must respond with a high output if only one input pool is active, and low output if neither or both input pools are active (Fig. 4a,b). We chose XOR as a canonical example of a task that requires a nonlinear hierarchy with appropriate credit assignment for successful learning. Before learning, the network was initialized such that the output pool treated any input combination as roughly equivalent (Fig. 4c, dashed line). To compute XOR, the output pool would have to learn to reduce its response to simultaneously active inputs and increase its response to a single active input.

We set up the network configuration to address a twofold question. (1) Would an error signal applied to the top-layer neurons' dendrites be propagated downward adequately? (2) Would the burst-dependent learning rule combine top-down signals with bottom-up information to make the hidden-layer neurons better feature detectors for solving XOR?

If the answers to these two questions were ‘yes’, we would expect that the two hidden ensembles would learn different features if they receive different feedback from the output. To test this, we provided hidden pool 1 with positive feedback from the output, and hidden

pool 2 with negative feedback (Fig. 4a, light blue symbols). With this configuration, adequate error propagation to the two hidden pools would make their responses diverge with learning, and the output pool would learn to take advantage of this change. Indeed, our results showed that the XOR task was solved in this manner after training (Fig. 4c, solid line).

To understand how this solution was aided by appropriate credit assignment, we examined the information about the top-down teaching signals in each layer. According to the learning rule, plasticity can be steered by controlling the instantaneous propensity to burst with respect to a moving average of the burst probability ($B_i - \bar{P}_i E_i$ in equation (1) and Fig. 2e,f). In the output pool, the error signal applied to the apical dendrites induced a temporary decrease in the burst probability when the input pools were both active or both inactive, and a temporary increase when only one input pool was active (Fig. 4d and Methods). These changes in the output burst probability modified the output burst rate, which was propagated to the hidden pools. As mentioned above, the hidden pools received top-down signals with different signs (Fig. 4e(i),(ii), orange lines), and indeed their respective burst probabilities were altered in opposite directions (Fig. 4e(i),(ii), red lines). Due to these distinct top-down signals and the adaptive threshold \bar{P}_i , the hidden pools' response diverged during learning (Fig. 4f(i),(ii)). For instance, hidden pool 1 reduced its responses to both inputs being active, while hidden pool 2 increased its response. These changes were due to the top-down control of the plasticity of synapses between the input and hidden pools. We verified that solving this task depends on the plasticity of connections from input to hidden units, but only weakly on the size of the ensembles (Extended Data Fig. 1). Also, we verified that the task was solved when the time constant of the moving average was shorter (Extended Data Fig. 2), and when the feedback pathways had the same sign of connection (Extended Data Fig. 3). These results demonstrate that the propagation of errors using burst multiplexing and the burst-dependent learning rule can combine to achieve hierarchical credit assignment in ensembles of pyramidal neurons.

Burst-dependent plasticity promotes linearity and alignment of feedback. Having demonstrated that a burst-dependent learning rule in pyramidal neurons enables online, local, spike-based solutions to the credit assignment problem, we were interested in understanding the potential relationship between this algorithm and the gradient descent-based algorithms used for credit assignment in machine learning. To do this, we wanted to derive the average behavior of the burst-dependent learning rule at the coarse-grained, ensemble-level, and determine whether it provided an estimate of a loss-function gradient. More precisely, in the spirit of mean-field theory and linear–nonlinear rate models, we developed a model where each unit represents an ensemble of pyramidal neurons, with event rates, burst probabilities and burst rates as described above (Extended Data Fig. 4 and Methods). Specifically, for an ensemble of pyramidal neurons, we defined $e(t)$, $b(t)$ and $p(t)$ as ensemble averages of the event train, burst train and burst probability, respectively. (Note that ensemble-level activity-related quantities are lowercase (e , b , p), whereas single-neuron activities are uppercase (E , B , P); the weights do not follow this convention.) We then defined the connection weight between an ensemble of presynaptic neurons and an ensemble of postsynaptic neurons, $W_{\text{post,pre}}$, as the effective impact of the presynaptic ensemble on the postsynaptic ensemble, taking into consideration potential polysynaptic interactions and inhibition (Supplementary Information).

Our goal was then to derive the ensemble-level weight updates from the burst-dependent plasticity rule (equation (1)). We assumed that any given pair of neurons were only weakly correlated on average. We further assumed that the neuron-specific moving average burst probability (\bar{P}_i) is independent of the instantaneous

occurrence of events. Using these assumptions, it can be shown (Supplementary Information) that the effective weight averaged across both pre- and postsynaptic ensembles obeys:

$$\frac{dW_{\text{post,pre}}}{dt} = \eta M(t)[p_{\text{post}}(t) - \bar{p}_{\text{post}}(t)]e_{\text{post}}(t)e_{\text{pre}}(t) \quad (2)$$

where the learning rate η is different from that appearing in equation (1), $\bar{p}_{\text{post}}(t)$ is a ratio of moving averages for the postsynaptic burst rate and event rate, and $M(t)$ is the aforementioned gating term preventing plasticity without a teacher. This learning rule can be shown to implement an approximation of gradient descent for hierarchical circuits, such as the backpropagation-of-error algorithm¹⁷. Specifically, if we assume that the burst probabilities remain in a linear regime (linearity), that the feedback synapses are symmetric to the feedforward synapses (alignment), and that error signals are received in the dendrites of the top-level ensembles, then $-[p_{\text{post}}(t) - \bar{p}_{\text{post}}(t)]e_{\text{post}}(t)$ is equivalent to the error signal sent backward in backpropagation (Supplementary Information and Supplementary Table 2).

The assumptions of feedback linearity and alignment can be supported by the presence of additional learning mechanisms. First, we examined learning mechanisms to keep the burst probabilities in a linear regime. Multiple features of the microcircuit control linearity (Extended Data Fig. 5), including distal apical inhibition^{28,43}, which is consistent with the action of somatostatin-positive (SOM+) Martinotti cells in cortical circuits^{18,31}. We used recurrent excitatory and inhibitory inputs to control the apical compartments' potential (Fig. 5a). These dendrite-targeting inputs propagated bursts from neural ensembles at the same processing stage in the hierarchy, which provided them with the necessary information to keep the burst probabilities in a linear range of the burst-probability function. We found that a simple learning rule for these connections (Methods) could learn to keep burst probabilities in a linear regime, thus improving gradient estimates (Fig. 5b). Second, we explored the issue of weight symmetry. The symmetry between feedforward and feedback weights is an implicit assumption of many learning algorithms. However, such an assumption is unnecessary, as it is possible to learn weight symmetry²⁹. In one classic algorithm, weight symmetry is obtained if feedforward and feedback weights are updated with the same error signals, plus some weight decay³⁰. We used this algorithm here because it can be implemented locally in our model using the bursting of the presynaptic cells (Fig. 5c, Extended Data Fig. 6 and Methods). But, we note that our model is not tied to this specific algorithm, and other algorithms could be used. With weight symmetry learning, the ensemble-level weight updates aligned well with the true gradient (Fig. 5d). Altogether, these results demonstrate that the burst-dependent learning rule, averaged across ensembles of pyramidal neurons, can provide a good estimate of loss-function gradients in hierarchical networks.

Ensemble-level burst-dependent plasticity in deep networks can support good performance on standard machine learning benchmarks. We wanted to determine whether the ensemble-level learning rule could perform well on difficult tasks from machine learning. Specifically, we built a deep neural network comprising pyramidal ensemble units that formed a series of convolutional layers followed by fully connected layers (Fig. 6a). We then trained these networks on two challenging image categorization datasets that previous biologically plausible algorithms have struggled with: CIFAR-10 and ImageNet⁴⁴.

The training in all components of the network used our burst-dependent plasticity rule and recurrent inputs for linearization at fully connected hidden layers. For the CIFAR-10 dataset, we observed a classification test error rate of 20.1% after 400 epochs (where an epoch is a pass through all training images), similar to

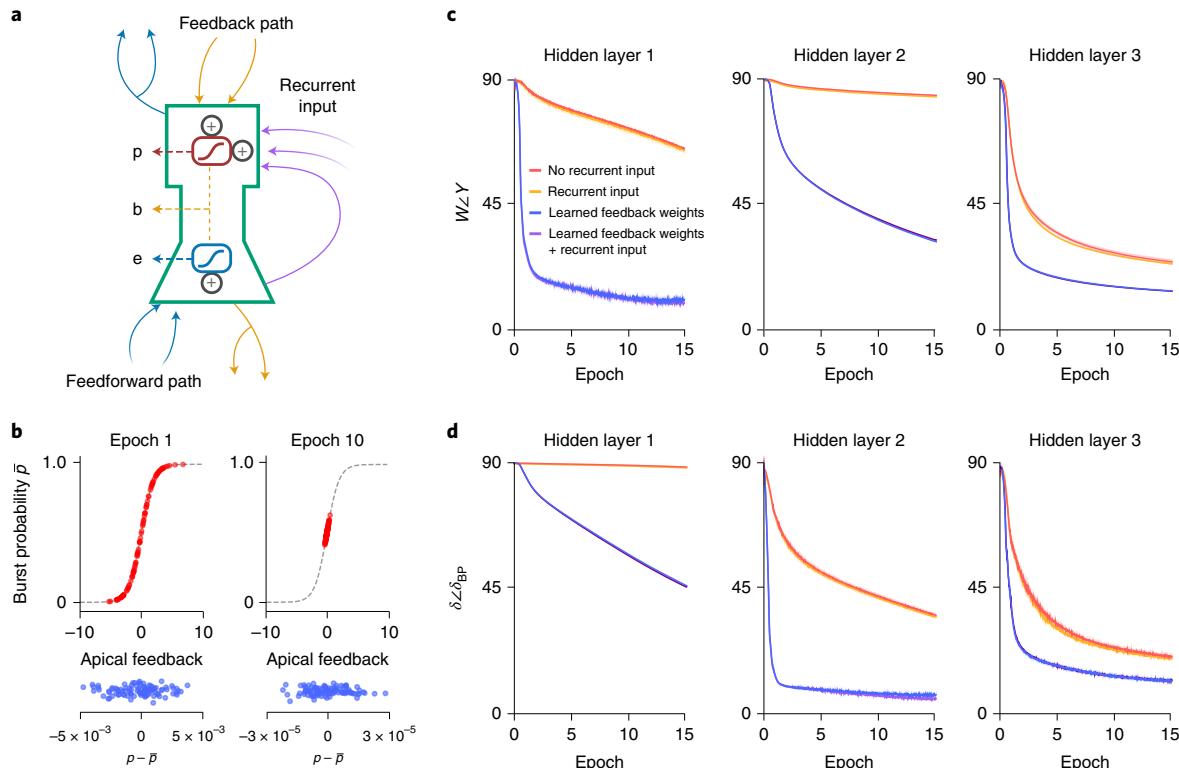


Fig. 5 | Burst-dependent plasticity of recurrent and feedback connections promotes gradient-based learning by linearizing and aligning feedback.

a, Diagram of a hidden-layer unit in the rate model, with event rate e , burst probability p and burst rate b . Each unit (green outline) in the network represents an ensemble of pyramidal neurons. Recurrent inputs (purple arrows) from all ensembles in a layer control the dendritic potential. **b**, Throughout learning, recurrent weights were updated to push the burst probabilities toward the linear regime (top). This led to an overall decrease in the magnitudes of burst probabilities, while continuing to support positive and negative values necessary for credit assignment (bottom). **c**, Alignment of feedback weights Y and feedforward weights W for three layers in a three-hidden-layer network trained on MNIST. Each hidden layer contained 500 units. Recurrent inputs slightly reduce the angle between the two sets of weights, denoted $W \angle Y$, while learning on the feedback weights dramatically improves weight alignment. Each datapoint is the angle between feedforward and feedback weights at the start of a training epoch. **d**, Angle between our weight updates (δ) and those prescribed by the backpropagation algorithm (δ_{BP}), for three layers in a three-hidden-layer network trained on MNIST. Recurrent inputs slightly improve the approximation to backpropagation, whereas learning on the feedback weights leads to a much closer correspondence. Each datapoint is the average angle between weight updates during a training epoch. In **c** and **d**, results are displayed as mean \pm s.d. over $n=5$ trials.

the error rate achieved with full gradient descent in a standard artificial neural network (Fig. 6b). Training the feedback weights was critical for enabling this level of performance on CIFAR-10, as fixed feedback weights led to much worse performance, even when the number of units was increased to match the total number of trainable parameters (Supplementary Tables 3 and 4), in line with previous results⁴⁴. Furthermore, rich unit-specific feedback signals were critical. A network trained using a global reward signal, plus activity correlations, while theoretically guaranteed to follow gradient descent on average^{13,14}, was unable to achieve good performance on CIFAR-10 in a reasonable amount of time (Fig. 6b, node perturbation). For the ImageNet dataset, we observed a classification error rate of 56.1% on the top five predicted image classes with our algorithm, which is much better than the error rate achieved when keeping the feedback weights fixed and much closer to that of full gradient descent (Fig. 6c). The remaining gap between the ensemble-level burst-dependent learning rule and backprop performance on ImageNet can likely be explained by the fact that we could not use recurrent input at convolutional layers due to memory limitations, which led to degraded linearity of feedback in the bottom layers (Extended Data Fig. 7). We also trained a network on the MNIST dataset, and achieved a similar performance of 1.1% error on the test set with all three algorithms (Extended Data Fig. 8). Therefore, these results demonstrate that the ensemble-level

burst-dependent learning rule, coupled with additional mechanisms to promote multiplexing, linearity and alignment, can solve difficult tasks that other biologically plausible learning algorithms have struggled with.

Discussion

In this paper, we asked the following question: could high-frequency bursts in pyramidal neurons provide an instructive signal for synaptic plasticity that can coordinate learning across hierarchical circuits (Fig. 1)? We have shown that the well-known burst-dependence of plasticity combined with STP and regenerative dendritic activity turns feedback connections into a teacher (Fig. 2), which by multiplexing (Fig. 3) can coordinate plasticity across multiple synaptic jumps (Fig. 4). We then showed that, with some additional burst-dependent learning at recurrent and feedback synapses, these mechanisms provide an approximation of a loss-function gradient for supervised learning (Fig. 5) and perform well on challenging image classification tasks (Fig. 6). Together, these results demonstrate that a local, spike-based and experimentally supported learning rule that uses high-frequency bursts as an instructive signal can enable sophisticated credit assignment in hierarchical circuits.

Decades of research into biologically plausible learning have yet to produce a confluence of biological properties that permit efficient credit assignment. In this manuscript, we focused on the

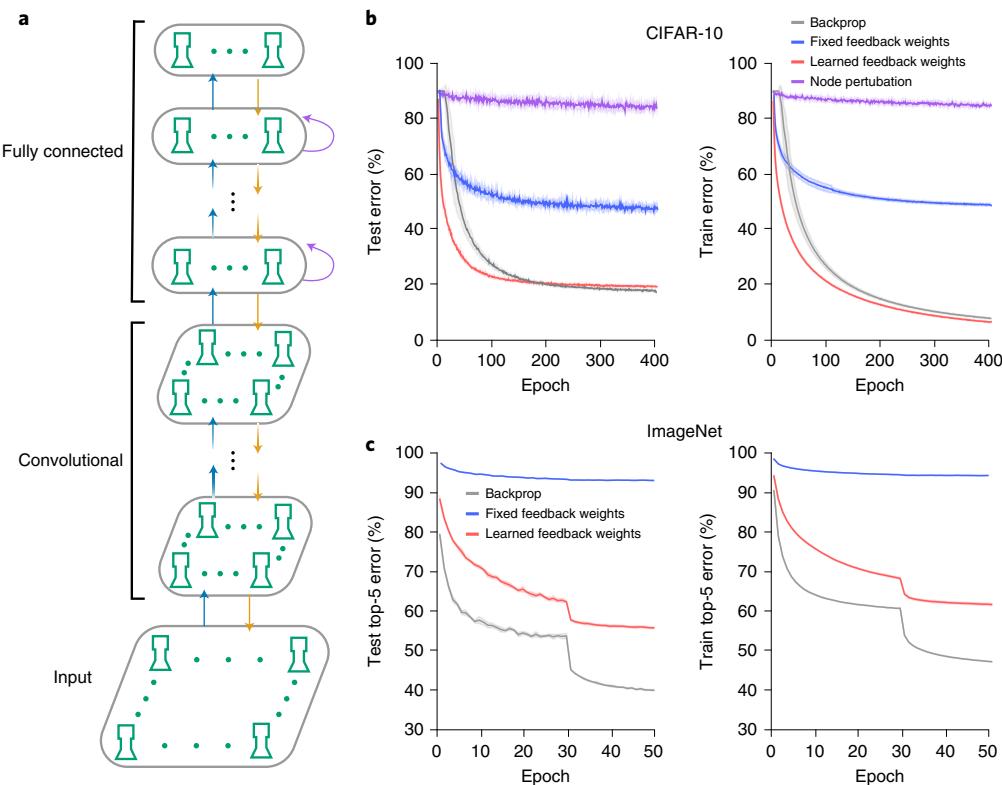


Fig. 6 | Ensemble-level burst-dependent plasticity supports learning in deep networks. **a**, The deep networks consisted of an input layer, a series of convolutional layers and a series of fully connected layers. Layers were connected with sets of feedforward weights (blue arrows) and feedback weights (orange arrows). Fully connected hidden layers contained recurrent connections (purple arrows). **b**, Our learning rule, combined with learning of the feedback weights, was able to reach the performance of the backpropagation algorithm (backprop) on the CIFAR-10 classification task. **c**, A network trained using our learning rule was able to learn to classify images in the ImageNet dataset when feedback weights were also updated. In **b** and **c**, results are displayed as mean \pm s.d. over $n = 5$ trials.

frequency dependence of LTP/LTD, STP, dendritic nonlinearities and inhibitory microcircuits. We focused on these aspects in part because the previous literature has established that these properties have important links with synaptic plasticity^{4,38}, but also because they are very well-established properties of cortical circuits. Our burst-dependent learning rule itself could readily be implemented by previously established synaptic plasticity signaling pathways³⁹. Overall, our model can be seen as a concrete implementation of a recent proposal from Lillicrap et al.¹⁵, which posited that differences in activity over time could carry gradient signals. Here, we have shown that differences in the probability of high-frequency bursts can carry gradient signals without affecting the time-dependent flow of sensory information. Therefore, one of the primary lessons from our model is that when local synaptic plasticity rules are sensitive to high-frequency bursts, then pyramidal neurons possess the necessary machinery for backprop-like top-down control of synaptic plasticity.

It is important to note that there are a number of limitations to our model. First, our ensemble-level models used ‘ensemble units’ that incorporated the activity of many pyramidal neurons. But, there is no reason that the algorithm could not in principle work with population coding⁴⁵ or with time-averaged firing of single neurons. Second, the presence of the gating term, $M(t)$, may be seen as an additional limitation in the model, since it is left in an abstract form and not directly motivated by biology. But, such gating mechanisms could be implemented by dendritic disinhibition^{31,38} (Extended Data Fig. 5b) or transient neuromodulation¹⁰. Third, we did not include soma-targeting recurrent connections between pyramidal neurons within a layer, despite the fact that such connections are known to

exist. We did this for the sake of simplicity, but again, we consider such recurrent connectivity to be fully compatible with our model and a subject for future investigations. Finally, our model makes some high-level assumptions about the structure of cortical circuitry. For example, we assumed that top-down signals are received at apical dendrites while bottom-up signals are received at basal dendrites. There is evidence for this structure⁴², but also some data showing that it is not always this way⁴⁶. Likewise, we assumed that pyramidal neurons across the cortical hierarchy project reciprocally with one another. There is some evidence that the same cells that project backward in the cortical hierarchy also project forward⁴⁷, but the complete circuitry of cortex is far from determined.

Our model makes a number of falsifiable predictions that could be examined experimentally. First, the model predicts that there should be a polarization of STP along the sensory hierarchy, with bottom-up synaptic projections being largely STD and top-down synaptic projections being largely STF. There are reports of such differences in thalamocortical projections^{33,34}, which suggests that an important missing component of our model is the inclusion of thalamic circuitry. Second, because our model proposes that burst firing carries information about errors, there should be a relationship between burst firing and progress in learning. Specifically, our model predicts that the variance in burst probabilities across a population should be correlated with the errors made during learning (Extended Data Fig. 9). Experimental evidence in other systems supports this view²⁷. Third, our model predicts that the moving average of the number of times that a neuron emits a burst when it spikes (its ‘burst fraction’) should determine the threshold for LTP in pyramidal neurons. Thus, and consistent with the fact that

synaptic weights have a finite range, our model predicts that if a neuron generally bursts whenever it spikes it will be more difficult to induce LTP in that neuron, and vice versa. Finally, our model predicts that inhibition in the distal apical dendrites serves, in part, to homeostatically regulate burst probabilities to promote learning. Thus, a fairly simple prediction from the model is that manipulations of distal dendrite-targeting interneurons, such as SOM+ interneurons, should lead to unusual levels of bursting in cortical circuits and disrupt learning. Some recent experimental evidence supports this prediction³⁸.

Linking low-level and high-level computational models of learning is one of the major challenges in computational neuroscience. Our focus on supervised learning of static inputs was motivated by recent progress in this area. However, machine learning researchers have also been making rapid progress in unsupervised learning on temporal sequences in recent years. We suspect that many of the same mechanisms we explored here (that is, burst-dependent plasticity), but also many of the mechanisms not explored here (that is, plasticity induced by cooperative synaptic inputs producing dendritic spikes⁴⁸) or bursting induced by feedforward activity escaping feedforward inhibition^{49,50} could be adapted for unsupervised learning of temporal sequences in hierarchical circuits. It is likely that the brain combines unsupervised and supervised learning mechanisms, and future research should be directed toward how neurons may combine different rules for these purposes. Ultimately, by showing that a top-down orchestration of learning is a natural result of a small set of experimentally observed physiological phenomena, our work opens the door to future approaches that use the unique physiology of cortical microcircuits to implement powerful learning algorithms on dynamic stimuli.

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/s41593-021-00857-x>.

Received: 30 March 2020; Accepted: 15 April 2021;
Published online: 13 May 2021

References

- Hebb, D. O. *The Organization of Behavior* (Wiley, New York, 1949).
- Artola, A., Bröcher, S. & Singer, W. Different voltage dependent thresholds for inducing long-term depression and long-term potentiation in slices of rat visual cortex. *Nature* **347**, 69–72 (1990).
- Markram, H., Lübke, J., Frotscher, M. & Sakmann, B. Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science* **275**, 213–215 (1997).
- Paulsen, O. & Sejnowski, T. J. Natural patterns of activity and long-term synaptic plasticity. *Curr. Opin. Neurobiol.* **10**, 172–180 (2000).
- Sjöström, P. J., Turrigiano, G. G. & Nelson, S. B. Rate, timing, and cooperativity jointly determine cortical synaptic plasticity. *Neuron* **32**, 1149–1164 (2001).
- Letzkus, J. J., Kampa, B. M. & Stuart, G. J. Learning rules for spike timing-dependent plasticity depend on dendritic synapse location. *J. Neurosci.* **26**, 10420–10429 (2006).
- Kampa, B., Letzkus, J. & Stuart, G. Requirement of dendritic calcium spikes for induction of spike-timing-dependent synaptic plasticity. *J. Physiol.* **574**, 283–290 (2006).
- Sjöström, P. J. & Häusser, M. A cooperative switch determines the sign of synaptic plasticity in distal dendrites of neocortical pyramidal neurons. *Neuron* **51**, 227–238 (2006).
- Gambino, F. et al. Sensory-evoked LTP driven by dendritic plateau potentials in vivo. *Nature* **515**, 116–119 (2014).
- Geun Hee, S. et al. Neuromodulators control the polarity of spike-timing-dependent synaptic plasticity. *Neuron* **55**, 919–929 (2007).
- Gerstner, W., Lehmann, M., Liakoni, V., Corneil, D. & Brea, J. Eligibility traces and plasticity on behavioral time scales: experimental support of neoHebbian three-factor learning rules. *Front. Neural Circuits* **12**, 53 (2018).
- Roelfsema, P. R. & Holtmaat, A. Control of synaptic plasticity in deep cortical networks. *Nat. Rev. Neurosci.* **19**, 166–180 (2018).
- Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning* **8**, 229–256 (1992).
- Werfel, J., Xie, X. & Seung, H. S. Learning curves for stochastic gradient descent in linear feedforward networks. *Neural Comput.* **17**, 2699–2718 (2005).
- Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J. & Hinton, G. Backpropagation and the brain. *Nat. Rev. Neurosci.* **21**, 335–346 (2020).
- Richards, B. A. et al. A deep learning framework for systems neuroscience. *Nat. Neurosci.* **22**, 1761–1770 (2019).
- Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986).
- Larkum, M. E., Zhu, J. & Sakmann, B. A new cellular mechanism for coupling inputs arriving at different cortical layers. *Nature* **398**, 338–341 (1999).
- Markram, H., Wang, Y. & Tsodyks, M. Differential signaling via the same axon of neocortical pyramidal neurons. *Proc. Natl. Acad. Sci. USA* **95**, 5323–5328 (1998).
- Nevian, T. & Sakmann, B. Spine Ca²⁺ signaling in spike-timing-dependent plasticity. *J. Neurosci.* **26**, 11001–11013 (2006).
- Froemke, R. C., Tsay, I. A., Raad, M., Long, J. D. & Dan, Y. Contribution of individual spikes in burst-induced long-term synaptic modification. *J. Neurophys.* **95**, 1620–1629 (2006).
- Bell, C. C., Caputi, A., Grant, K. & Serrier, J. Storage of a sensory pattern by anti-Hebbian synaptic plasticity in an electric fish. *Proc. Natl. Acad. Sci. USA* **90**, 4650–4654 (1993).
- Bol, K., Marsat, G., Harvey-Girard, E., Longtin, André & Maler, L. Frequency-tuned cerebellar channels and burst-induced LTD lead to the cancellation of redundant sensory inputs. *J. Neurosci.* **31**, 11028–11038 (2011).
- Richards, B. A. & Lillicrap, T. P. Dendritic solutions to the credit assignment problem. *Curr. Opin. Neurobiol.* **54**, 28–36 (2019).
- Bandalis, F. & Gerber, U. Mossy fiber-evoked subthreshold responses induce timing-dependent plasticity at hippocampal CA3 recurrent synapses. *Proc. Natl. Acad. Sci. USA* **111**, 4303–4308 (2014).
- Kayser, C., Montemurro, M. A., Logothetis, N. K. & Panzeri, S. Spike-phase coding boosts and stabilizes information carried by spatial and temporal spike patterns. *Neuron* **61**, 597–608 (2009).
- Herzfeld, D. J., Kojima, Y., Soetedjo, R. & Shadmehr, R. Encoding of action by the purkinje cells of the cerebellum. *Nature* **526**, 439–442 (2015).
- Naud, R. & Sprekeler, H. Sparse bursts optimize information transmission in a multiplexed neural code. *Proc. Natl. Acad. Sci. USA* **115**, 6329–6338 (2018).
- Burbank, K. S. Mirrored STDP implements autoencoder learning in a network of spiking neurons. *PLoS Comp. Biol.* **11**, e1004566 (2015).
- Akrout, M., Wilson, C., Humphreys, P. C., Lillicrap, T. & Tweed, D. Using weight mirrors to improve feedback alignment. Preprint at arXiv <https://arxiv.org/abs/1904.05391> (2019).
- Murayama, M. et al. Dendritic encoding of sensory stimuli controlled by deep cortical interneurons. *Nature* **457**, 1137–1141 (2009).
- Körding, K. P. & König, P. Supervised and unsupervised learning with two sites of synaptic integration. *J. Comput. Neurosci.* **11**, 207–215 (2001).
- Granseth, B., Ahlstrand, E. & Lindström, S. Paired pulse facilitation of corticogeniculate epscs in the dorsal lateral geniculate nucleus of the rat investigated in vitro. *J. Physiol.* **544**, 477–486 (2002).
- Sherman, S. M. Thalamocortical interactions. *Curr. Opin. Neurobiol.* **22**, 575–579 (2012).
- Meredith, R. M., Floyer-Lea, A. M. & Paulsen, O. Maturation of long-term potentiation induction rules in rodent hippocampus: role of gabaergic inhibition. *J. Neurosci.* **23**, 11142–11146 (2003).
- Inglebert, Y., Aljadeff, J., Brunel, N. & Debanne, D. Synaptic plasticity rules with physiological calcium levels. *Proc. Natl. Acad. Sci. USA* **117**, 33639–33648 (2020).
- Kampa, B. M. & Stuart, G. J. Calcium spikes in basal dendrites of layer 5 pyramidal neurons during action potential bursts. *J. Neurosci.* **26**, 7424–32 (2006).
- Doron, G. et al. Perirhinal input to neocortical layer 1 controls learning. *Science* **370**, eaaz3136 (2020).
- Mäki-Marttunen, T., Iannella, N., Edwards, A. G., Einevoll, G. & Blackwell, K. T. A unified computational model for cortical post-synaptic plasticity. *eLife* **9**, e55714 (2020).
- Bienenstock, E. L., Cooper, L. N. & Munro, P. W. Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *J. Neurosci.* **2**, 32–48 (1982).

41. Ning-Long, X. et al. Nonlinear dendritic integration of sensory and motor input during an active sensing task. *Nature* **492**, 247–251 (2012).
42. Felleman, D. J. & van Essen, D. C. Distributed hierarchical processing in the primate cerebral cortex. *Cereb. Cortex* **1**, 1–47 (1991).
43. Sacramento, J., Costa, R. C., Bengio, Y. & Senn, W. Dendritic cortical microcircuits approximate the backpropagation algorithm. *Adv. Neural Inf. Process. Syst.* **31**, 8721–8732 (2018).
44. Bartunov, S. et al. Assessing the scalability of biologically-motivated deep learning algorithms and architectures. *Adv. Neural Inf. Process. Syst.* **31**, 9368–9378 (2018).
45. Boerlin, M., Machens, C. K. & Denève, S. Predictive coding of dynamical variables in balanced spiking networks. *PLoS Comp. Biol.* **9**, e1003258 (2013).
46. Petreanu, L., Mao, T., Sternson, S. M. & Svoboda, K. The subcellular organization of neocortical excitatory connections. *Nature* **457**, 1142–1145 (2009).
47. Ren, Si-Qiang, Li, Z., Lin, S., Bergami, M. & Shi, S.-H. Precise long-range microcircuit-to-microcircuit communication connects the frontal and sensory cortices in the mammalian brain. *Neuron* **104**, 385–401.e3 (2019).
48. Golding, N. L., Staff, N. P. & Spruston, N. Dendritic spikes as a mechanism for cooperative long-term potentiation. *Nature* **418**, 326–331 (2002).
49. Wang, X. et al. Feedforward excitation and inhibition evoke dual modes of firing in the cat's visual thalamus during naturalistic viewing. *Neuron* **55**, 465–478 (2007).
50. Owen, S. F., Berke, J. D. & Kreitzer, A. C. Fast-spiking interneurons supply feedforward control of bursting, calcium, and plasticity for efficient learning. *Cell* **172**, 683–695 (2018).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© The Author(s), under exclusive licence to Springer Nature America, Inc. 2021

Methods

Spiking model. Spiking simulations were performed using the Auryn simulator⁵¹, except for the pairing protocols of Fig. 2b–d, which used Python and NumPy.

Event and burst detection. An event was said to occur either at the time of an isolated spike or at the time of the first spike in a burst. A burst was defined as any occurrence of at least two spikes with an interspike interval (ISI) less than the threshold $b_{\text{th}} = 16$ ms (ref. ²⁸). Any additional spike with ISI < b_{th} belonged to the same burst. A neuron i kept track of its time-averaged burst probability \bar{P}_i by using exponential moving averages of its event train E_i and burst train B_i :

$$\begin{aligned}\bar{E}_i(t) &= \frac{1}{\tau_{\text{avg}}} \int_0^\infty E_i(t-\tau) e^{-\tau/\tau_{\text{avg}}} d\tau \\ \bar{B}_i(t) &= \frac{1}{\tau_{\text{avg}}} \int_0^\infty B_i(t-\tau) e^{-\tau/\tau_{\text{avg}}} d\tau \\ \bar{P}_i(t) &= \frac{\bar{B}_i(t)}{\bar{E}_i(t)},\end{aligned}\quad (3)$$

where τ_{avg} is a slow time constant (roughly 1–10 s). Also, $E_i(t) = \sum_{\text{event}} \delta(t - t_{i,\text{event}})$ and $B_i(t) = \sum_{\text{burst}} \delta(t - t_{i,\text{burst}})$, where $t_{i,\text{event}}$ and $t_{i,\text{burst}}$ indicate the timing of an event and of the second spike in a burst, respectively.

Plasticity rule. Weights were updated on the detection of a postsynaptic event or burst according to

$$\frac{dw_{ij}}{dt} = \eta M(t) [B_i(t) - \bar{P}_i(t)E_i(t)] \tilde{E}_j(t) \quad (4)$$

where $\tilde{E}_j(t) = \int_0^\infty E_j(t-\tau) e^{-\tau/\tau_{\text{pre}}} d\tau$ is a presynaptic trace with time constant τ_{pre} . Here, τ_{pre} is typically much smaller than τ_{avg} , with $\tau_{\text{pre}} \approx 10 - 100$ ms, but it could possibly be made larger to accommodate plasticity rules with slower dynamics⁵². The plasticity rule above only differs from equation (1) by the inclusion of the prefactor M , which gates plasticity during training: in the XOR task (Fig. 4), $M=1$ when the teaching signal is present and 0 otherwise. In Fig. 2, $M=1$ throughout. In Fig. 3, $M=0$ throughout.

Pairing protocols. For all pairing protocols of Fig. 2b–d, we had $\tau_{\text{pre}} = 50$ ms, $\tau_{\text{avg}} = 15$ s, $\eta = 0.1$.

- **Periodic protocol.** Five consecutive pairings were separated by a quiescent period of 10 s, 15 times. We used pairings with $\Delta t = 0$. For each pairing frequency, the starting value for the estimated burst probability was $\bar{P}_i(t=0) = 0.15$ and $\bar{E}_i(t=0) = 5$ Hz.
- **Poisson protocol.** Both the pre- and postsynaptic neurons fired spikes at a Poisson rate r with a refractory period of 2 ms. For each r , the induction lasted 100 s and we averaged over 20 independent realizations. We used $\bar{E}_i(t=0) = 5$ Hz.
- **Burst-Poisson protocol.** Both the pre- and postsynaptic neurons produced events at a Poisson rate r , including a refractory period $\tau_{\text{ref}}^{(E)} > b_{\text{th}}$. For each event, a burst was generated with probability P and an intraburst ISI was sampled from $\text{Unif}(\tau_{\text{ref}}^{(B)}, \tau_{\text{ref}}^{(B)} + t_{\text{max}})$, with $\tau_{\text{ref}}^{(B)} + t_{\text{max}} < b_{\text{th}}$. For the simulations in Fig. 2d, we used $\tau_{\text{ref}}^{(E)} = 20$ ms, $\tau_{\text{ref}}^{(B)} = 2$ ms and $t_{\text{max}} = 10$ ms. We set $\bar{P}_i(t=0) = 0.2$ and the event rate of the pre- and postsynaptic neurons were set to $r=5$ Hz and $r=10$ Hz, with corresponding values for the initial postsynaptic event rate estimates. For each r , the induction lasted 100 s and we averaged over 20 independent realizations.

Neuron models. Pyramidal neurons. The somatic compartment obeyed

$$\begin{aligned}C_s \dot{V}_s &= -(C_s/\tau_s)(V_s - E_L) + g_s f(V_d) + I_s - w_s \\ \tau_{w_s} \dot{w}_s &= -w_s + b \tau_{w_s} S(t)\end{aligned}\quad (5)$$

where V_s is the somatic membrane potential, w_s is an adaptation variable, I_s is the total current applied to the soma (includes noise and other synaptic inputs) and $S(t)$ is the spike train of the neuron. The function $f(V_d)$ in the equation for V_s takes into account the coupling with the dendritic compartment, with $f(V_d) = 1/[1 + \exp(-(V_d - E_d)/D_d)]$ and parameters $E_d = -38$ mV and $D_d = 6$ mV. A spike occurred whenever V_s crossed a moving threshold from below. The latter jumped up by 2 mV right after a spike and relaxed toward -50 mV with a time constant of 27 ms. Other somatic parameters were: the membrane time constant $\tau_s = 16$ ms, the capacitance $C_s = 370$ pF, the leak reversal potential $E_L = -70$ mV, the adaptation time constant $\tau_{w_s} = 100$ ms, the strength of spike-triggered adaptation $b = 200$ pA and the dendrosomatic coupling $g_s = 1,300$ pA. The reset voltage after a spike was $V_r = -70$ mV. The dendritic compartment obeyed

$$\begin{aligned}C_d \dot{V}_d &= -(C_d/\tau_d)(V_d - E_L) + g_d f(V_d) + c_d(K * S)(t) + I_d - w_d \\ \tau_{w_d} \dot{w}_d &= -w_d + a_w(V_d - E_L).\end{aligned}\quad (6)$$

The function $f(V_d)$ is the same as above and is responsible for the regenerative dendritic activity. The term $c_d(K * S)(t)$ (with * denoting a convolution) represents the backpropagating action potential, with the kernel K modeled as a box filter of amplitude one and duration 2 ms, delayed by 0.5 ms with respect to the somatic spike, and $c_d = 2,600$ pA scales its amplitude. Other dendritic parameters were: the membrane time constant $\tau_d = 7$ ms, the capacitance $C_d = 170$ pF, $E_L = -70$ mV, the adaptation time constant $\tau_{w_d} = 30$ ms, the subthreshold adaptation $a_w = 13$ nS and the strength of the dendritic nonlinearity $g_d = 1,200$ pA. This model and its parameters are described in more detail and compared with experimental data in ref. ²⁸.

Dendrite-targeting inhibition. We modeled SOM+ interneurons^{53–55} using the adaptive exponential integrate-and-fire (AdEx) model⁵⁶:

$$\begin{aligned}\dot{V} &= -g_L(V - E_L) + g_L \Delta_T e^{-\Delta_T} + I - w \\ \tau_w \dot{w} &= a(V - E_L) + b \tau_w S(t) - w\end{aligned}$$

where I is the total current applied to the neuron. A spike occurred whenever V crossed $V_{\text{cut}} = 24$ mV and was followed by a refractory period τ_{ref} . Parameters were the capacitance $C = 100$ pF, the leak conductance $g_L = 5$ nS, $E_L = -70$ mV, the threshold $V_T = -62$ mV, the slope factor $\Delta_T = 4$ mV, the adaptation time constant $\tau_w = 500$ ms, the subthreshold adaptation $a = 0.5$ nS, the spike-triggered adaptation $b = 10$ pA, the reset $V_r = -65$ mV and $\tau_{\text{ref}} = 2$ ms. In Fig. 3, these model neurons (gray squares in Fig. 3a) were receiving top-down excitation from higher-level pyramidal cells.

Perisomatic inhibition. We modeled parvalbumin-positive neurons⁵⁷ using the AdEx model with parameters chosen to reproduce qualitatively their typical fast-spiking phenotype. Parameter values were $C = 100$ pF, $g_L = 10$ nS, $E_L = -70$ mV, $V_T = -48$ mV, $\Delta_T = 2$ mV, $V_r = -55$ mV, $\tau_{\text{ref}} = 1$ ms and $a = b = 0$. In Fig. 3, these model neurons (gray disks in Fig. 3a) were receiving bottom-up excitation from the lower-level pyramidal cells.

Connectivity. In general, connections between distinct neural ensembles were sparse (roughly 5–20% connection probability). Pyramidal neurons within an ensemble had no recurrent connections between their somatic compartments. Within a pyramidal ensemble, burst-probability linearization was enacted by sparse STF inhibitory synapses onto the dendritic compartments (Supplementary Fig. 1). These STF connections were not illustrated in Fig. 3a for clarity. The net strength of inputs onto the apical dendrites was chosen to preserve a stationary burst probability between 10 and 50%, as in vivo experimental data report burst probability between 15 and 25% (refs. ^{58,59}).

Synapses. All synapses were conductance-based. The excitatory (or inhibitory) reversal potential was 0 mV (or -80 mV) and the exponential decay time constant was 5 ms (or 10 ms). There were no NMDA components to excitatory synapses. For a given connection between two ensembles, existing synapses had their strengths all initialized to the same value.

Noise. Each neuron (for single-compartment neurons) and each compartment (for two-compartment neurons) received its own (private) noise in the form of a high-frequency excitatory Poisson input combined with an inhibitory Poisson input. The only exception was the noise applied to the neural populations in Fig. 2e–g, where we used sparse connections from a pool of excitatory and inhibitory Poisson neurons. Noise served to decorrelate neurons within a population and to imitate in vivo conditions.

Short-term plasticity. STP was modeled following the extended Markram-Tsodyks model¹⁹. Using the notation of ref. ⁵⁰, the parameters for STF were $D = 100$ ms, $F = 100$ ms, $U = 0.02$ and $f = 0.1$. For STD, the parameters were $D = 20$ ms, $F = 1$ s, $U = 0.9$ and $f = 0.1$. These sets of parameters were chosen following ref. ²⁸ to help decode bursts (using STF) and events (using STD).

Spiking XOR gate. A XOR gate maps binary inputs (0,0) and (1,1) onto 0 and inputs (1,0) and (0,1) onto 1. In the context of our spiking network, input 0 corresponded to a low event rate (roughly 2 Hz) and input 1 to a higher event rate (roughly 10 Hz). These were obtained by applying a hyperpolarizing (resp. depolarizing) current for 0 (resp. 1) to the corresponding input-layer population. Compared to the spiking simulations described above (and as is commonly done in spiking implementations of biological credit assignment^{61,62}), our implementation of the spiking XOR gate used simplifications to reduce the dimension of the parameter search space. First, events and bursts were propagated directly instead of relying on STP (Supplementary Fig. 2). Second, disynaptic inhibition was replaced by direct inhibition coming from the pyramidal cells. Third, we used a simplified pyramidal neuron model. Below, we describe this model, as well as the initialization of the network, the error generation and the learning protocol for the XOR gate.

Simplified pyramidal neuron model. The effect of dendritic regenerative activity on the somatic compartment (controlled by g_s in equations (5) and (6)) was replaced by a conditional burst probability: whenever a somatic event occurred, a burst was produced with probability $f(V_d)$. This function is the same as that appearing in equations (5) and (6), but with $E_d = -57\text{ mV}$. This model permitted a cleaner burst-detection process and burst-ensemble multiplexing.

Initialization of the network. The feedforward synaptic strengths were initialized so that the event rates of all pyramidal ensembles in the network belonged to $[e_{\min}, e_{\max}]$ for all inputs, with $e_{\min} = 2\text{ Hz}$ and $e_{\max} = 10\text{ Hz}$. All existing feedforward excitatory synaptic strengths were equal together, and likewise for the inhibitory synapses. The feedback synaptic strengths from the output population to the hidden populations—the only existing ones—were initialized so that one coarse-grained connection would be predominantly excitatory and the other inhibitory (the one onto hidden pool 2 in Fig. 4a). As with the feedforward connections, the excitatory feedback synapses belonging to the same coarse-grained connection shared the same strength, and likewise for inhibition. A constant depolarizing current was applied to the hidden pool 2's dendritic compartments to compensate for the stronger inhibition.

Error generation. At the output layer, we specified a maximum and a minimum event rate, e_{\max} and e_{\min} (the same as above). The following linearly transformed \bar{E}_i

$$\bar{E}' = \frac{\bar{E}_i - e_{\min}}{e_{\max} - e_{\min}}$$

was then used in conjunction with a cross entropy loss function to compute the error for each neuron of the output population, assuming that $e_{\min} < \bar{E}_i < e_{\max}$. As a result, a current, $I_i^{(\text{d})}$ (where 'd' indicates 'dendritic'), was injected into every neuron so that its burst probability would increase or decrease according to the running average of its event rate and the desired output:

$$\begin{aligned} \text{if desired output } = 0 \Rightarrow I_i^{(\text{d})} &= -c(e_{\max} - \bar{E}_i) \\ \text{if desired output } = 1 \Rightarrow I_i^{(\text{d})} &= c(\bar{E}_i - e_{\min}). \end{aligned}$$

where c is approximately 1 nA Hz^{-1} . For instance, if the desired output was 0 and \bar{E}_i was large, then the injected current was strongly hyperpolarizing. The injected current was set to zero when \bar{E}_i was to within 1 Hz of the desired value.

Learning protocol. A simulation proceeded as follows. With the plasticity off, there was first a relaxation interval of duration $3\tau_{\text{avg}}$, with no input applied to the network. In Fig. 4, we have set $\tau_{\text{avg}} = 2\text{ s}$, although a faster time scale can still yield adequate learning (Extended Data Fig. 2). Then, the four different input pairs were applied consecutively to give the 'before learning' response in Fig. 4d. Afterward, the four input–output pairs were applied consecutively (for one epoch), typically in the same order (but see Extended Data Fig. 1e). For each input–output pair, first, the input alone was applied to the input populations with the plasticity off. We let the network reach its steady state for that input for the first 90% of the duration of an example. During this prediction interval, the moving average of the burst probability would converge toward the actual burst probability of the population for that given input. The duration of an example was chosen to be $4\tau_{\text{avg}} = 8\text{ s}$ to provide enough time for this steady state to be reached to a good approximation, although relaxing that assumption can still produce adequate learning (Extended Data Fig. 2). During the last 10% of the example duration, the plasticity was activated for all feedforward excitatory synapses and the teacher was applied. For computational efficiency, the error was computed once, at the very end of the prediction interval. The total number of epochs required to reach decent performance depended on the initialization of the network and the learning rate; for Fig. 4, we used 500 epochs. At the end of learning, the plasticity was switched off for good and the 'after learning' response was computed.

Deep network model for categorical learning. We now describe the deep network model that was used to learn the classification tasks reported in Figs. 5 and 6. The code used the following Python packages: PyTorch, Tensorboard and NumPy. The model can be seen as a limiting case of a time-dependent rate model, which itself can be heuristically derived from the spiking network model under simplifying assumptions (Supplementary Information), a typical assumption of studies in biological credit assignment^{15,30,43,44,63–74}.

For the fully connected layers in the network, we defined the 'somatic potentials' of units in layer l as:

$$\mathbf{v}_l = W_l \mathbf{e}_{l-1},$$

where W_l is the weight connecting layer $l-1$ to layer l . Note that in this formulation we include a bias term as a column of W_l . The event rate of layer l was given by

$$\mathbf{e}_l = f_l(\mathbf{v}_l),$$

where f_l is the activation function for layer l . In models trained on MNIST and CIFAR-10, the activation function was a sigmoid. In the model trained on ImageNet, a rectified linear unit (ReLU) activation was used for hidden layers and a softmax activation was used at the output layer.

During the feedforward pass, the burst probability at the output layer ($l=L$) was set to a constant, $p_L^{(0)}$ (in these experiments, this was set to 0.2). Our previous research²⁸ has shown that the dendritic transfer function is a sigmoidal function of its input (Extended Data Fig. 5). Therefore, the hidden-layer burst probabilities, \mathbf{p}_l , for $l < L$, were computed using a sigmoidal function of a local 'dendritic potential' \mathbf{u}_l as

$$\mathbf{p}_l = \sigma(\beta \mathbf{u}_l + \alpha),$$

where α and β are constants controlling the dendritic transfer function. In our experiments, we set $\beta=1$ and $\alpha=0$. Extended Data Fig. 5 illustrates various mechanisms affecting these parameters. The dendritic potentials were given by

$$\mathbf{u}_l = h(\mathbf{e}_l) \odot (Y_l \mathbf{b}_{l+1}), \quad (7)$$

where \odot is the elementwise product. The vector-valued function $h(\mathbf{e}_l) \equiv f'(\mathbf{v}_l) \odot \mathbf{e}_l^{-1}$ depends on the chosen activation function; of course, some caution is required when ReLU and softmax activations are used (Supplementary Information). The burst rate is given by

$$\mathbf{b}_{l+1} = \mathbf{p}_{l+1} \odot \mathbf{e}_{l+1}. \quad (8)$$

Finally, Y_l is the feedback weight matrix. For the feedback alignment algorithm, Y_l is a random matrix and is fixed throughout learning⁶⁶. In the standard backpropagation algorithm, the feedforward and feedback weight matrices are symmetric so that $Y_l = W_{l+1}^T$, where T denotes the transpose. Below, we also describe how to learn the feedback weights to make them symmetric with the feedforward weights using the Kolen–Pollack algorithm³⁰.

With the teacher present, the output-layer burst probabilities were set to a squashed version of $p_L^{(0)} - h(\mathbf{e}_l) \odot \nabla_{\mathbf{e}_l} \mathcal{L}$, where \mathcal{L} is the loss function (a mean squared error loss for Figs. 5 and 6). The squashing function was to make sure that $p_{L,i} \in [0, 1]$. The Supplementary Information provides a few examples of squashing functions. The burst probabilities of the hidden layers were then computed as above. Finally, the weights were updated according to

$$\Delta W_l = \eta_l ((\mathbf{p}_l - \bar{\mathbf{p}}_l) \odot \mathbf{e}_l) \mathbf{e}_{l-1}^T - \lambda W_l, \quad (9)$$

where \mathbf{p}_l and $\bar{\mathbf{p}}_l$ denote the burst probabilities with and without a teacher, respectively, η_l is the learning rate hyperparameter for units in layer l , and λ is a weight decay hyperparameter. Note that, for this model, $\bar{\mathbf{e}}_l$ lags \mathbf{e}_l by a single computational step (Supplementary Information). Therefore, when the teacher appears, $\bar{\mathbf{e}}_l = \mathbf{e}_l$ and we can write

$$(\mathbf{p}_l - \bar{\mathbf{p}}_l) \odot \mathbf{e}_l = \mathbf{b}_l - \bar{\mathbf{b}}_l.$$

This means that, in this model, the error is directly represented by the deviation of the burst rate with respect to a reference.

In the case of convolutional layers, the event rates of ensembles in layer l were given by

$$\mathbf{e}_l = f_l(W_l * \mathbf{e}_{l-1}),$$

where $*$ represents a convolution. Similarly, the dendritic potentials in layer l were given by $\mathbf{u}_l = Y_l * \mathbf{b}_{l+1}$ while burst probabilities were calculated as in the fully connected layers. Finally, the weights of convolutional layers were updated as

$$\Delta W_l = \eta_l \psi (\mathbf{b}_l - \bar{\mathbf{b}}_l, \mathbf{e}_{l-1}) - \lambda W_l, \quad (10)$$

where ψ combines the burst deviations and event rates to compute an approximation of the gradient with respect to the convolutional weights W_l .

Learning the recurrent weights. In certain experiments, we introduced recurrent inputs into the hidden layers that served to keep burst probabilities in the linear regime of the sigmoid function. At layer l , we set the reference dendritic potentials to

$$\bar{\mathbf{u}}_l = h(\mathbf{e}_l) \odot Y_l \hat{\mathbf{b}}_{l+1} - Z_l \hat{\mathbf{b}}_l, \quad (11)$$

where Z_l is the recurrent weight matrix and the burst rates used here are calculated as the burst rate without any recurrent inputs (represented by the hat (^) over the symbol) and without the teaching signal:

$$\hat{\mathbf{b}}_l = \sigma(\beta h(\mathbf{e}_l) \odot Y_l \hat{\mathbf{b}}_{l+1} + \alpha) \odot \mathbf{e}_l. \quad (12)$$

Otherwise, the dendritic potentials and burst rates must be solved self-consistently, slowing down computations. Recurrent weights are then updated to minimize $\bar{\mathbf{u}}_l$:

$$\Delta Z_l = -\eta_r \bar{\mathbf{u}} \hat{\mathbf{b}}_l^T, \quad (13)$$

where η_r is the learning rate. Note that, with these recurrent inputs, the updates of matrix W_l are the same as before, but now with

$$\mathbf{p}_l = \sigma[\beta(h(\mathbf{e}_l) \odot Y_l \hat{\mathbf{b}}_{l+1} - Z_l \hat{\mathbf{b}}_l) + \alpha].$$

Learning the feedback weights. Kolen and Pollack⁷⁵ found that if the feedforward and feedback weights are updated such that

$$\Delta W_l = \eta A - \lambda W_l$$

$$\Delta Y_l = \eta A - \lambda Y_l,$$

where A is any matrix with the same shape as W_l and Y_l , then Y_l and W_l will converge. This means that if the feedback weights are updated in the same direction as the feedforward weights and weight decay is applied to both sets of weights, they will eventually become symmetric. Thus, we implemented the following learning rule for the feedback weights between layer $l+1$ and layer l :

$$\Delta Y_l = \eta_l (\mathbf{b}_{l+1} - \bar{\mathbf{b}}_{l+1}) \mathbf{e}_l^T - \lambda Y_l, \quad (14)$$

where λ is a weight decay hyperparameter. In convolutional layers, we used the following weight update:

$$\Delta Y_l = \eta_l \psi(\mathbf{b}_{l+1} - \bar{\mathbf{b}}_{l+1}, \mathbf{e}_l) - \lambda Y_l. \quad (15)$$

Training the model with CIFAR-10 and ImageNet. The network architectures described in Supplementary Tables 3 and 4 were trained on standard image classification datasets, CIFAR-10 (ref.⁷⁶) and ImageNet⁷⁷. The CIFAR-10 dataset consists of 60,000 32 × 32 px training images belonging to ten classes, while the ImageNet dataset consists of 1.2 million images (resized to 224 × 224 px) split among 1,000 classes.

Each unit in these networks represents an ensemble of pyramidal neurons and has an event rate, burst probability and burst rate. For each training example, the input image is presented and a forward pass is done, where event rates \mathbf{e}_l throughout the network are computed sequentially, followed by a feedback pass where burst probabilities $\bar{\mathbf{p}}_l$ and burst rates $\bar{\mathbf{b}}_l$ are computed. Then, the teaching signal is shown at the output layer, and new burst probabilities \mathbf{p}_l and burst rates \mathbf{b}_l are computed backward through the network. Weights are then updated using our weight update rules. Networks were trained using stochastic gradient descent with mini-batches, momentum and weight decay. ReLU layers were initialized from a normal distribution using Kaiming initialization⁷⁸, whereas Xavier initialization was used in sigmoid layers⁷⁹. Hyperparameter optimization was done on all networks using validation data (Supplementary Information for details).

Training the model using node perturbation. Node perturbation is a technique that approximates gradient descent by randomly perturbing the activations of units in the network, and updating weights according to the change in the loss function^{13,14}. In the model trained using node perturbation, at each step, first the input is propagated through the network as usual, after which the global loss, \mathcal{L} , is recorded. Then, the same input is propagated again through the network but the activations of units in a single layer l are randomly perturbed:

$$\mathbf{e}_l = f_l(W_l * \mathbf{e}_{l-1} + \mathbf{n}_l), \quad (16)$$

where the elements of \mathbf{n}_l are chosen from a normal distribution with mean 0 and standard deviation σ . The new loss, \mathcal{L}_{NP} , is recorded. The weights in layer l are then updated using the following weight update rule:

$$\Delta W_l = \eta_l ((\mathcal{L}_{NP} - \mathcal{L}) \mathbf{n}_l / \sigma^2) \mathbf{e}_{l-1}^T. \quad (17)$$

The layer to be perturbed, l , is changed with each mini-batch by iterating bottom-up through all of the layers in the network.

Statistics. The only statistical test we used was a two-tailed significance test for the Pearson correlation in Extended Data Fig. 7, with a sample size of $n=1,000$ (computed via the SciPy function stats.pearsonr). No statistical methods were used to predetermine sample sizes for our simulations, but our sample sizes are similar to those reported in previous studies^{70,74}. There was no experimental randomization in our study and we had no samples, organisms or participants, because it was a computational study. When computing error bars, different results were obtained by reseeding the random number generators used in the simulations.

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

Data availability

The MNIST, CIFAR-10 (ref.⁷⁶) and ImageNet⁷⁷ datasets are publicly available from <http://yann.lecun.com/exdb/mnist/>, <https://www.cs.toronto.edu/~kriz/cifar.html> and <http://www.image-net.org>, respectively.

Code availability

The code used in this article is available at <https://github.com/apayeur/spikingburstprop> and <https://github.com/jordan-g/Burstprop>.

References

51. Zenke, F. & Gerstner, W. Limits to high-speed simulations of spiking neural networks using general-purpose computers. *Front. Neuroinf.* **8**, 76 (2014).
52. Bittner, K. C., Milstein, A. D., Grienberger, C., Romani, S. & Magee, J. C. Behavioral time scale synaptic plasticity underlies ca1 place fields. *Science* **357**, 1033–1036 (2017).
53. Tremblay, R., Lee, S. & Rudy, B. Gabaergic interneurons in the neocortex: from cellular properties to circuits. *Neuron* **91**, 260–292 (2016).
54. Nigro, M. J., Hashikawa-Yamasaki, Y. & Rudy, B. Diversity and connectivity of layer 5 somatostatin-expressing interneurons in the mouse barrel cortex. *J. Neurosci.* **38**, 1622–1633 (2018).
55. Hilscher, M. M., Leão, R. N., Edwards, S. J., Leão, K. E. & Kullander, K. ChRNA2-Martinotti cells synchronize layer 5 type a pyramidal cells via rebound excitation. *PLOS Biol.* **15**, e200139226 (2017).
56. Naud, R., Marcille, N., Clopath, C. & Gerstner, W. Firing patterns in the adaptive exponential integrate-and-fire model. *Biol. Cybern.* **99**, 335–347 (2008).
57. Packer, A. M. & Yuste, R. Dense, unspecific connectivity of neocortical parvalbumin-positive interneurons: a canonical microcircuit for inhibition? *J. Neurosci.* **31**, 13260–13271 (2011).
58. De Kock, C. P. J. & Sakmann, B. High frequency action potential bursts (>100 Hz) in l2/3 and l5b thick tufted neurons in anaesthetized and awake rat primary somatosensory cortex. *J. Physiol.* **586**, 3353–3364 (2008).
59. Womelsdorf, T., Ardid, S., Everling, S. & Valiante, T. A. Burst firing synchronizes prefrontal and anterior cingulate cortex during attentional control. *Current Biology* **24**, 2613–2621 (2014).
60. Costa, R. P., Sjöström, P. J. & Van Rossum, M. C. W. Probabilistic inference of short-term synaptic plasticity in neocortical microcircuits. *Front. Comput. Neurosci.* **7**, 75 (2013).
61. Samadi, A., Lillicrap, T. P. & Tweed, D. B. Deep learning with dynamic spiking neurons and fixed feedback weights. *Neural Comput.* **29**, 578–602 (2017).
62. Guerguiev, J., Lillicrap, T. P. & Richards, B. A. Towards deep learning with segregated dendrites. *eLife* **6**, e22901 (2017).
63. Lee, D.-H., Zhang, S., Fischer, A. & Bengio, Y. Difference target propagation. In *Proc. Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (ed. Hutter, F. et al.) 498–515 (Springer, 2015).
64. Liao, Q., Leibo, J. Z. & Poggio, T. How important is weight symmetry in backpropagation? In *Proc. Thirtieth AAAI Conference on Artificial Intelligence* (ed. Schuurmans, D. et al.) 1837–1844 (AAAI, 2016).
65. Xiao, W., Chen, H., Liao, Q. & Poggio, T. Biologically-plausible learning algorithms can scale to large datasets. Preprint at *arXiv* <https://arxiv.org/abs/1811.03567> (2018).
66. Lillicrap, T. C., Cownden, D., Tweed, D. B. & Akerman, C. J. Random synaptic feedback weights support error backpropagation for deep learning. *Nature Commun.* **7**, 13276 (2016).
67. Scellier, B. & Bengio, Y. Towards a biologically plausible backprop. Preprint at *arXiv* <https://arxiv.org/abs/1602.05179v5> (2016).
68. Yali, A. Deep learning with asymmetric connections and Hebbian updates. *Front. Comput. Neurosci.* <https://doi.org/10.3389/fncom.2019.00018> (2019).
69. Whittington, J. C. R. & Bogacz, R. Theories of error back-propagation in the brain. *Trends Cogn. Sci.* **23**, 235–250 (2019).
70. Mostafa, H., Ramesh, V. & Cauwenberghs, G. Deep supervised learning using local errors. *Front. Neurosci.* **12**, 608 (2018).
71. Nokland, A. Direct feedback alignment provides learning in deep neural networks. *Adv. Neural Inf. Process. Syst.* **29**, 1037–1045 (2016).
72. Lansdell, B. J., Prakash, P. R. & Kording, K. P. Learning to solve the credit assignment problem. Preprint at *arXiv* <https://arxiv.org/abs/1906.00889v4> (2019).
73. Pozzi, I., Bohté, S. & Roelfsema, P. A biologically plausible learning rule for deep learning in the brain. Preprint at *arXiv* <https://arxiv.org/abs/1811.01768> (2018).
74. Laborieux, A. et al. Scaling equilibrium propagation to deep convnets by drastically reducing its gradient estimator bias. *Front. Neurosci.* **15**, 129 (2021).
75. Kolen, J. F. & Pollack, J. B. Backpropagation without weight transport. In *Proc. 1994 IEEE International Conference on Neural Networks (ICNN'94)* Vol. 3, 1375–1380 (IEEE, 1994).

76. Krizhevsky, A., Nair, V. & Hinton, G. *Cifar-10 (Canadian Institute for Advanced Research) Technical Report* (Univ. Toronto, 2009).
77. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. & Fei-Fei, L. ImageNet: a large-scale hierarchical image database. In *Proc. CVPR09* 248–255 (IEEE, 2009).
78. He, K., Zhang, X., Ren, S. & Sun, J. Delving deep into rectifiers: surpassing human-level performance on imangenet classification. In *Proc. IEEE International Conference on Computer Vision* 1026–1034 (IEEE, 2015).
79. Glorot, X. & Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proc. Thirteenth International Conference on Artificial Intelligence and Statistics* (ed. Whye Teh, Y. et al.) 249–256 (Society for Artificial Intelligence and Statistics, 2010).

Acknowledgements

We thank A. Santoro and L. Maler for comments on this manuscript. We also thank M. Hilscher and M.J. Nigro for sharing data about SOM+ neurons. In addition, we thank T. Mesnard for helping with the development of the rate-based model. This work was supported by two NSERC Discovery grants (to R.N., no. 06872 and to B.A.R., no. 04947), a CIHR Project grant (no. RN383647-418955), a Fellowship from the CIFAR Learning in Machines and Brains Program (to B.A.R.), an Ontario Early Researcher Award (to B.A.R., no. ER 17-13-242), a Healthy Brains, Healthy Lives New Investigator Start-up (to B.A.R., no. 2b-NISU-8) the Novartis Research Foundation (to F.Z.).

Author contributions

All authors contributed to the burst-dependent learning rule. A.P., F.Z. and R.N. designed the spiking simulations. A.P. performed the spiking simulations. J.G. designed the recurrent plasticity rule and performed the numerical experiments on CIFAR-10 and ImageNet. B.A.R. and R.N. wrote the manuscript, with contributions from J.G. and A.P. B.A.R. and R.N. cosupervised the project.

Competing interests

R.N., B.A.R. and A.P. have a provisional patent application for a neuromorphic implementation of the algorithm described in this article. The other authors declare no competing interests.

Additional information

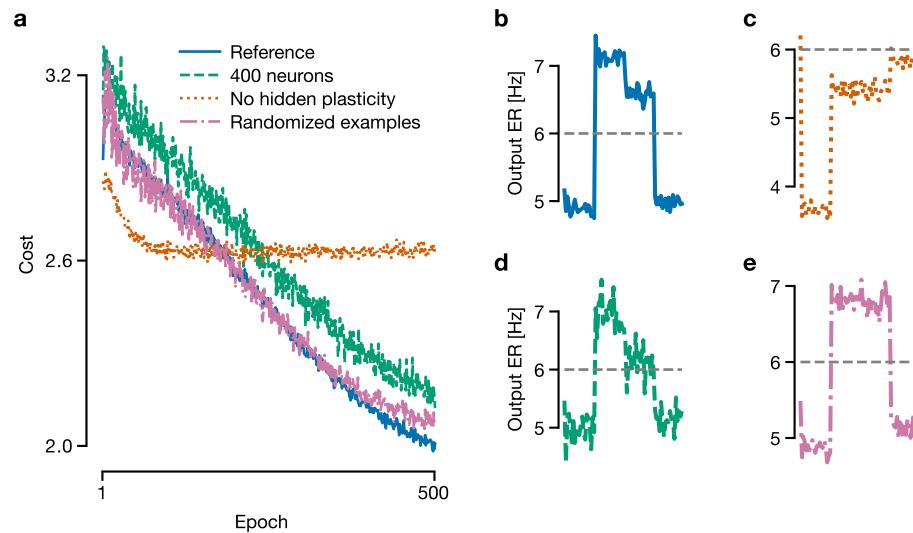
Extended data is available for this paper at <https://doi.org/10.1038/s41593-021-00857-x>.

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41593-021-00857-x>.

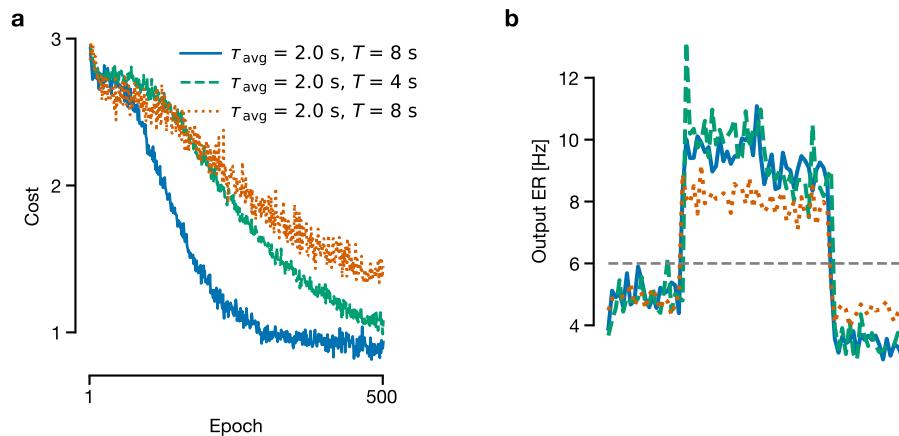
Correspondence and requests for materials should be addressed to B.A.R. or R.N.

Peer review information *Nature Neuroscience* thanks Gabriel Kreiman, Panayiota Poirazi and Nelson Spruston for their contribution to the peer review of this work.

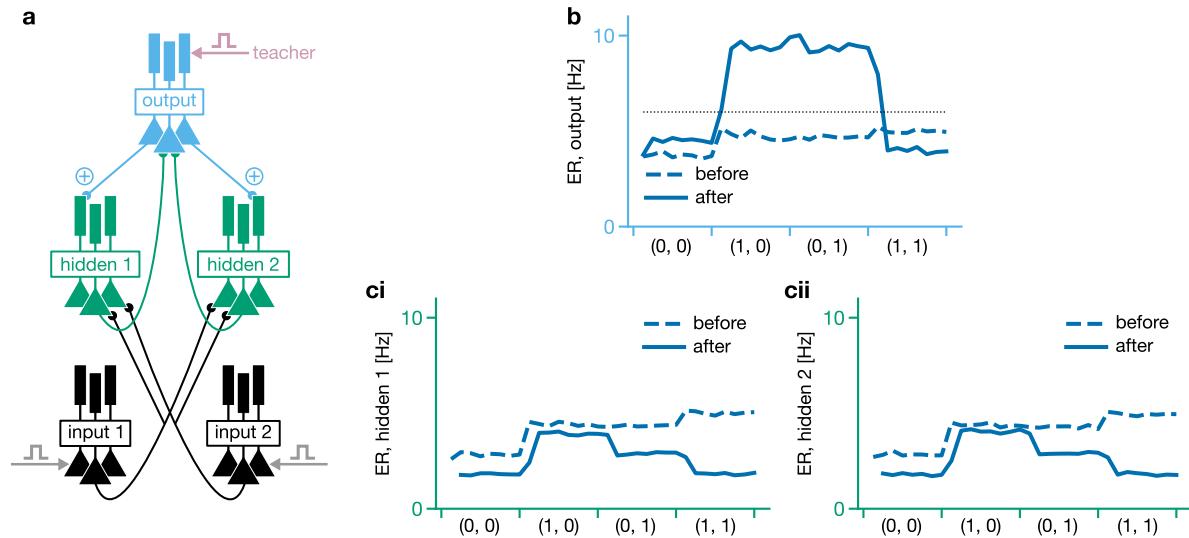
Reprints and permissions information is available at www.nature.com/reprints.



Extended Data Fig. 1 | Effects of population size, randomized examples and absence of hidden-layer plasticity on the XOR task. **a**, Comparison of costs for the XOR task. In blue is the cost for the network in Fig. 4 in the main text, but with 2000 neurons per population and slightly different parameter values. The dot-dashed pink line is for when the examples are randomly selected within an epoch. The dotted red line has no plasticity in the hidden layer. The dashed green line is for 400 neurons per population. **b-e**, Output event rate (ER) after learning. The dashed grey line separates ‘true (1)’ and ‘false (0)’ for the XOR. Only in c is XOR not solved.

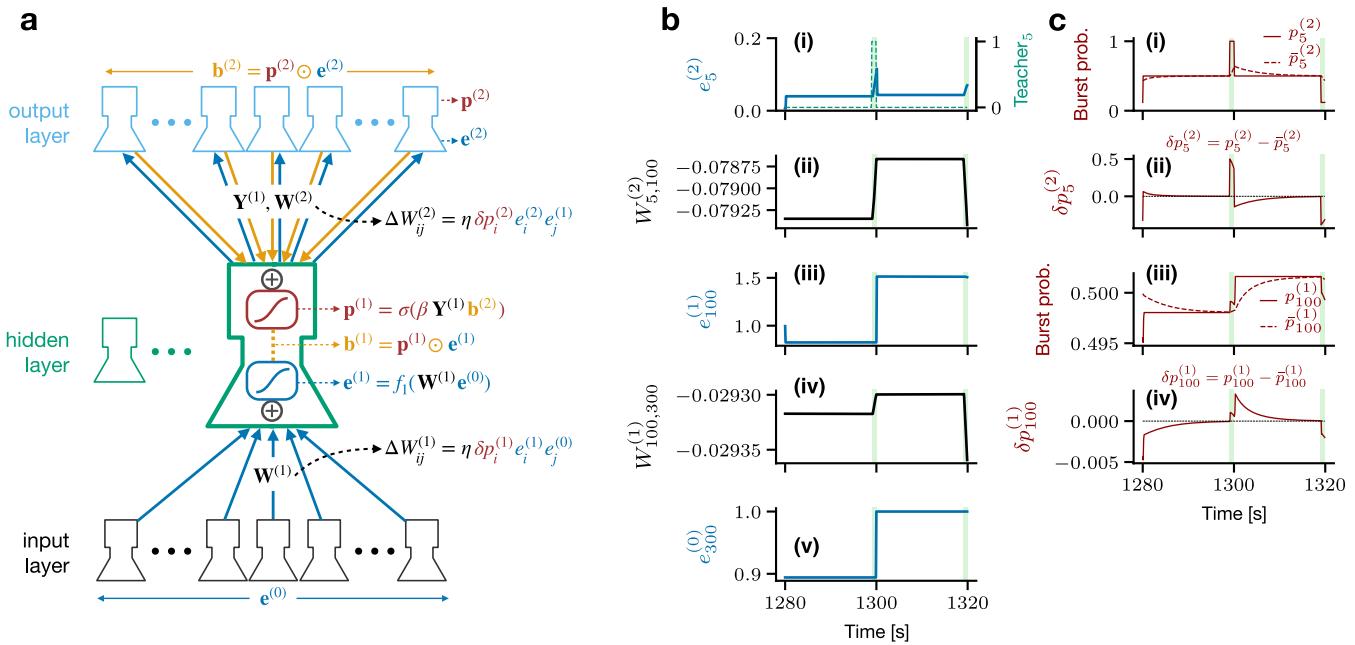


Extended Data Fig. 2 | Impact of different time scales on the XOR task. **a**, Comparison of costs for when the duration of examples T (in s) (dashed green line) and the moving average time constant τ_{avg} (in s) (dotted orange line) are changed with respect to the values used in Fig. 4 (solid blue). **b**, Output event rate (ER) after learning for the three cases in panel a. The dashed grey line separates 'true (1)' and 'false (0)' for the XOR.

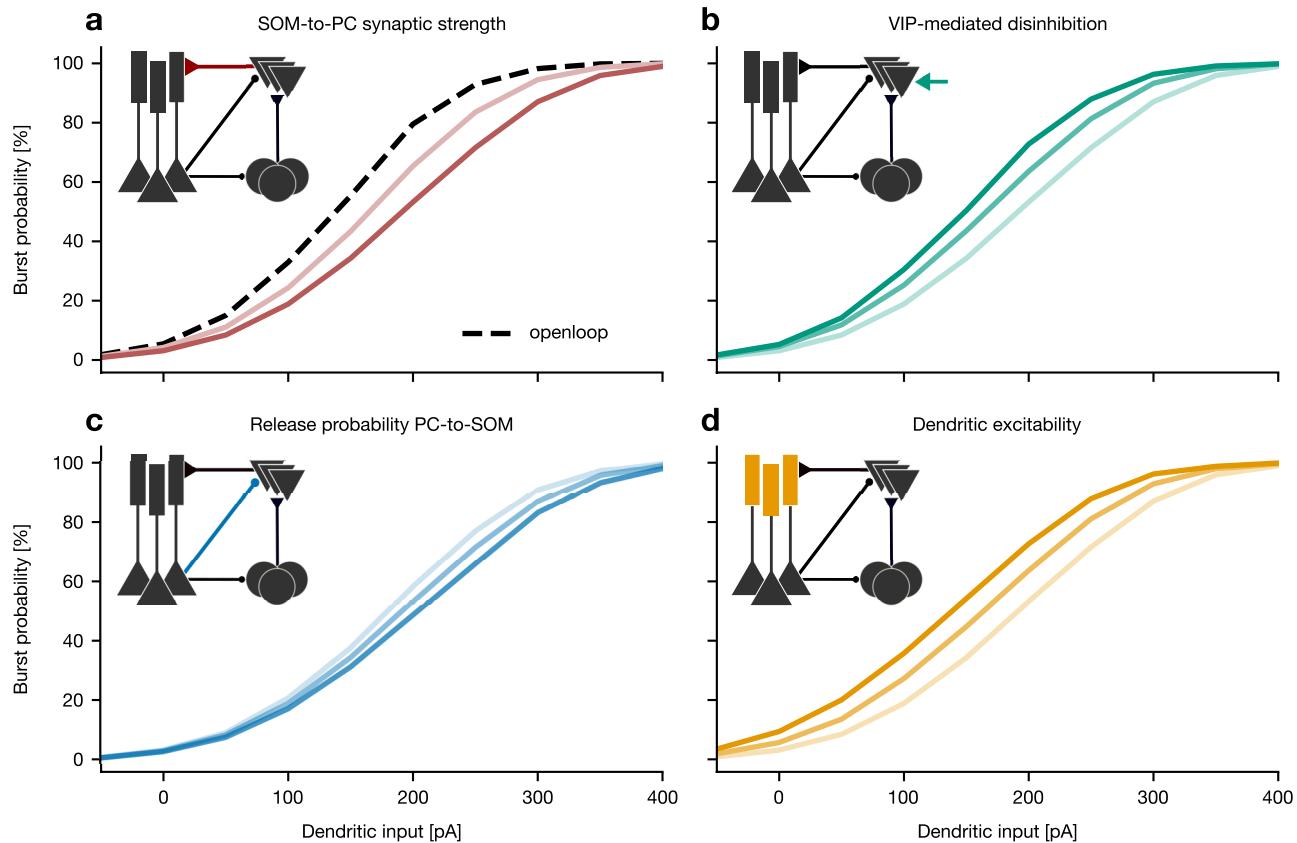


Extended Data Fig. 3 | Learning XOR with symmetric feedback pathways. **a**, Schematic diagram illustrating the symmetric feedback (\oplus and \oplus).

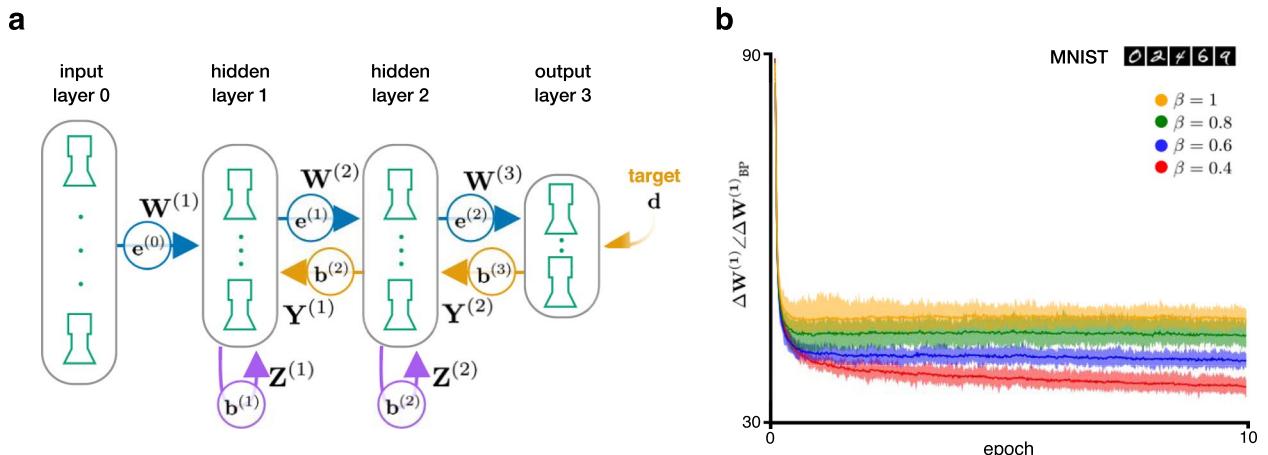
b, Output-layer activity for the XOR task. Note that the XOR task is still solved. Only a single realization is displayed here. (ci-cii) The symmetric feedback yields very similar representations at the hidden layer.



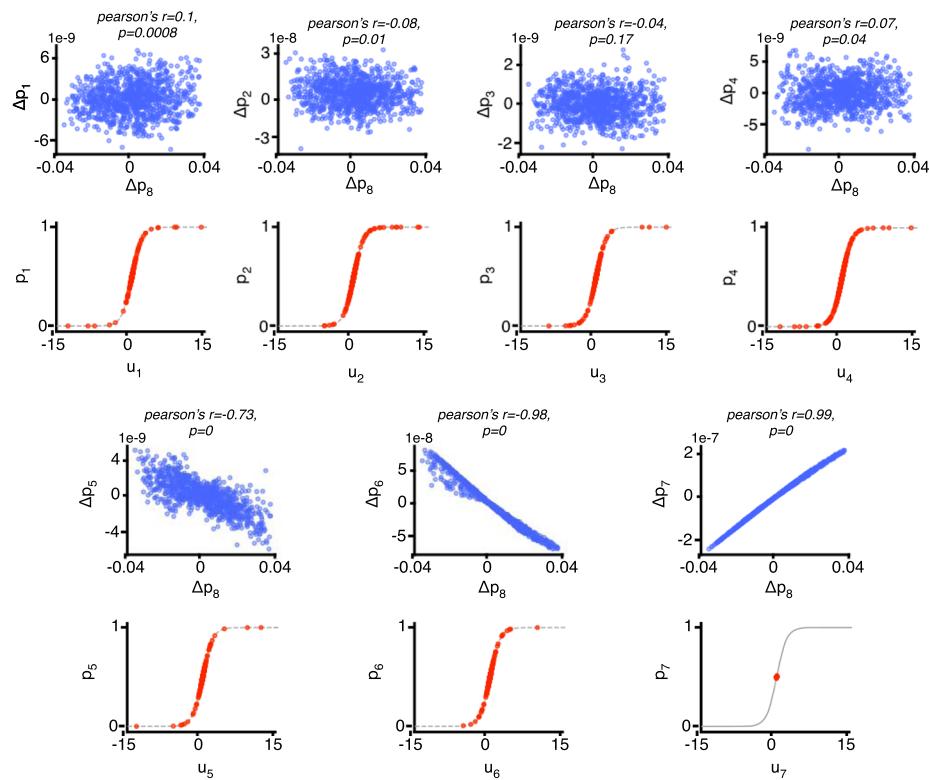
Extended Data Fig. 4 | Dynamics of the time-dependent rate model while learning MNIST. **a**, Schematic of the network. The enlarged hidden layer population stresses the fact that the burst rate is equal to the event rate times the burst probability, with the event and burst probability nonlinearly integrating the feedforward and feedback signals, respectively. **b**, Example event rates (i, iii, v) and weights (ii, iv) for two consecutive examples during the first epoch. In (i), the teacher is illustrated as a dashed line. Learning intervals are indicated by light green vertical bars. **c**, Burst probabilities (i, iii) and differences of burst probabilities (ii, iv) for the same examples as in b.



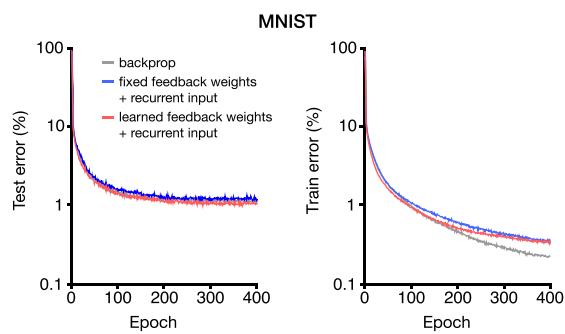
Extended Data Fig. 5 | Network mechanisms regulating the bursting nonlinearity. All panels display the burst probability of a large population of two-compartment pyramidal neurons as a function of the intensity of the injected dendritic current. The insets illustrate the microcircuit - including the PV-like neurons (disks) and the SOM-like neurons (inverted triangles) - and the parameter that is being modified is indicated by a colored circuit element. Increasing color intensities corresponds to increasing values of the parameter. **a**, Increasing the strength of inhibitory synapses from SOM neurons onto the pyramidal neurons' dendrites produces divisive burst probability control. **b**, Disinhibiting the pyramidal neurons' dendrites by applying a hyperpolarizing current to the SOM neurons - mimicking inhibition from the VIP neurons - increases the slope. **c**, Increasing the probability of release onto SOM neurons produces a small divisive gain modulation. **d**, Increasing the dendritic excitability by increasing the strength of the regenerative dendritic activity produces an additive gain control.



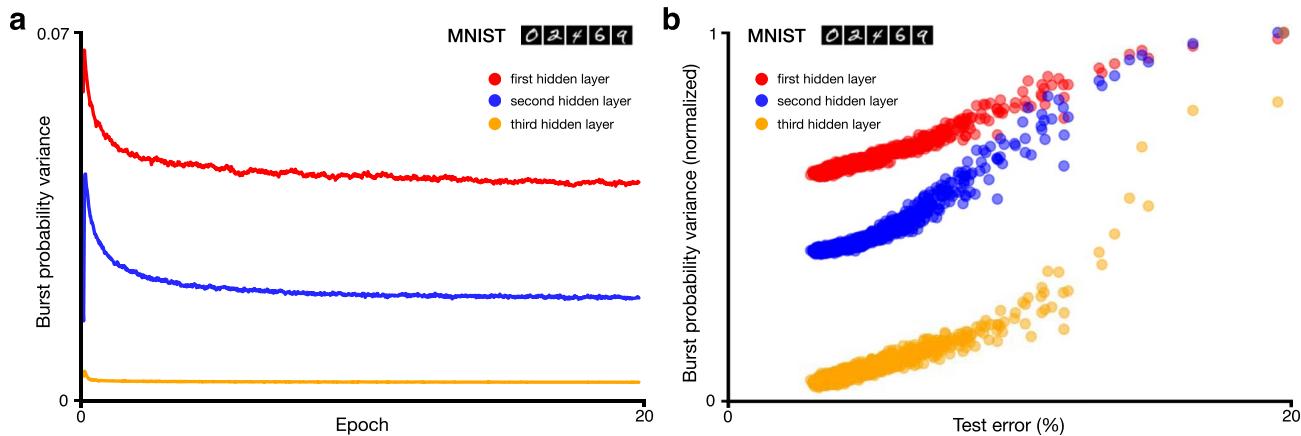
Extended Data Fig. 6 | The bursting nonlinearity controls the learning rate. **a**, Schematic of the network. Each hidden layer had 500 units. The recurrent weights ($Z^{(1)}$ and $Z^{(2)}$) and the feedback alignment weights ($Y^{(1)}$ and $Y^{(2)}$) are explicitly represented. **b**, Angle between the weight updates $\Delta W^{(1)}$ in the standard backpropagation algorithm and in burstprop for the MNIST digit recognition task. The angle is displayed for different values of the slope of the dendritic nonlinearity (β). Results are displayed as the mean +/- standard deviation over 10 realizations with randomly initialized weights.



Extended Data Fig. 7 | Linearity of feedback signals degrades with depth in deep convolutional network trained on ImageNet. Each plot shows the change in burst probability of a unit in hidden layer l , Δp_l , as the burst probability at the output layer, p_8 , is changed by Δp_8 ($n=1000$), along with the Pearson's correlation coefficient and two-tailed p-value (blue, top), as well as a random sample of 2000 burst probabilities after presentation of an input image (red, bottom).



Extended Data Fig. 8 | Learning MNIST with the simplified rate model. A convolutional network whose architecture is described in Supplementary Table 3 was trained using backprop, feedback alignment, and burstprop. As in Fig. 6a,c, recurrent input was introduced at hidden layers to keep burst probabilities linear with respect to feedback signals.



Extended Data Fig. 9 | The variance of the burst probability decreases during learning. **a**, Variance of the burst probability as a function of the epoch for the MNIST task, for each layer in a network with 3 hidden layers with 500 units each. **b**, Variance of the burst probability as a function of the test error, showing that the magnitude of the variance is correlated with the test error.

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

Spiking simulations were performed using C++ based simulator Auryn (available publicly). Rate simulations were performed using torch 1.1.0, torchvision 0.3.0, numpy 1.16.3, tensorboardX 1.9. Additional simulations for Extended Data Fig. 4 were performed in C++ using the Armadillo library (<http://arma.sourceforge.net/>). Code is available publicly at <https://github.com/apayeur/spikingburstprop> (spiking simulations) and <https://github.com/jordan-g/Burstprop> (rate simulations).

Data analysis

All data analyses were performed using Python 3 with packages numpy 1.16.3 (rate model) or 1.18.1 (spiking model), as well as scipy 1.4.1 for the statistical test.

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

The MNIST, CIFAR-10 and ImageNet datasets are publicly available from <http://yann.lecun.com/exdb/mnist/>, <https://www.cs.toronto.edu/~kriz/cifar.html> and <http://www.image-net.org>, respectively.

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

The only statistical test we used was a two-tailed significance test for the Pearson correlation in Extended Data Fig. 7, with a sample size of n=1000. No statistical methods were used to predetermine sample sizes. Our paper was a computational study and as such, sample sizes were selected so that the effects were unambiguous. For training and test errors in the large-scale rate model, sample size was n=5 and standard deviations were displayed instead of standard errors. Spiking simulations used the same n, except for Fig. 2 where n=20 was viewed as producing clearer mean traces.

Data exclusions

No data were excluded.

Replication

We used n=5 random initializations of the rate-based network and of the spiking XOR task. We also used random ordering for presentation of the examples for the spiking XOR task, and checked for robustness of the results when selectively varying simulation parameters. All the results we report were consistently found across many simulations. All attempts at replication were successful and our publicly-available code can be tested for replicability by others.

Randomization

Randomization is not relevant because our study did not involve group allocation. We had no experimental samples/organisms/participants in our study.

Blinding

Blinding is not relevant because our study did not involve group allocation.

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
<input checked="" type="checkbox"/>	<input type="checkbox"/> Antibodies
<input checked="" type="checkbox"/>	<input type="checkbox"/> Eukaryotic cell lines
<input checked="" type="checkbox"/>	<input type="checkbox"/> Palaeontology and archaeology
<input checked="" type="checkbox"/>	<input type="checkbox"/> Animals and other organisms
<input checked="" type="checkbox"/>	<input type="checkbox"/> Human research participants
<input checked="" type="checkbox"/>	<input type="checkbox"/> Clinical data
<input checked="" type="checkbox"/>	<input type="checkbox"/> Dual use research of concern

Methods

n/a	Involved in the study
<input checked="" type="checkbox"/>	<input type="checkbox"/> ChIP-seq
<input checked="" type="checkbox"/>	<input type="checkbox"/> Flow cytometry
<input checked="" type="checkbox"/>	<input type="checkbox"/> MRI-based neuroimaging