A theory of inference and learning in cortex with spiking neurons and dendritic error computation

Dissertation

for the award of the degree
"Doctor rerum naturalium"
of the Georg-August-Universität Göttingen

within the doctoral programm
Theoretical and Computational Neuroscience
of the Georg-August University School of Science (GAUSS)

conducted at the
Max-Planck-Institute for Dynamics and Self-Organization, Göttingen

submitted by

Fabian A. Mikulasch

from Bamberg, Germany

Göttingen, 2023

Thesis Committee

Thesis advisory committee

Prof. Dr. Viola Priesemann
Max Planck Institute for Dynamics and Self-Organization, Göttingen

Prof. Dr. Alexander Ecker
Institute of Computer Science, University of Göttingen

Prof. Dr. Siegrid Löwel
Institute for Zoology and Anthropology, University of Göttingen

Members of the examination board

Referee

Prof. Dr. Viola Priesemann
Max Planck Institute for Dynamics and Self-Organization, Göttingen

Second referee

Prof. Dr. Alexander Ecker
Institute of Computer Science, University of Göttingen

Further members of the examination board

Prof. Dr. Siegrid Löwel
Institute for Zoology and Anthropology, University of Göttingen

Prof. Dr. Theo Geisel
Max Planck Institute for Dynamics and Self-Organization, Göttingen

Prof. Dr. Fred Wolf
Campus Institute for Dynamics of Biological Networks, Göttingen

Dr. Andreas Neef
Campus Institute for Dynamics of Biological Networks, Göttingen

Date of oral examination: October 16, 2023

Acknowledgements

There are many entities without whose help and trust this work wouldn't have been possible.

First and foremost I want to thank Viola Priesemann and her group. I'm very grateful for Viola's guidance, scientific curiosity, trust and encouragement, which have enabled this research in the first place. During the time here I also learned a lot about the scientific process, scientific communication and project management, especially from Viola. Second, I want to thank Lucas, without whom this research wouldn't have happened as well. Thank you for piquing my interest in these topics, for the many discussions, and especially for sharing (and enduring) the many sufferings of the joint scientific writing process. Another significant chunk of credit goes to Kjartan, who I had the pleasure to work with for the last year, and to whom I owe a great deal of insights. I furthermore want to thank the other members of the group, who created (and continue to create) a very wholesome working environment, which I enjoyed very much. In that sense, I also want to thank Michael Wibral and his group, with which we shared many stimulating discussions and retreats. For both groups, even though the mutual information between topics at times was small, there always was mutual interest, which I found important. I'm furthermore grateful for having met the people from the PTCN program, which was enriching both personally and scientifically. Finally, I want to thank my family and friends who rooted for me during the whole process. I'm especially indebted to my parents and my sister Katrin, for their encouragement and trust, and for providing shelter in times of an ongoing pandemic. Thank you.

I acknowledge the support of the Max-Planck-Society, as well as the German Research Foundation (DFG), which partially funded the research within SFB 1286.

I also acknowledge the help of DALL-E-2[1], which aided in creating the figures in the Introduction (chapter 1), as well as ChatGPT-3.5[2], which lifted the troublesome task of translating the Abstract into the German language off my shoulders.

[1] https://openai.com/dall-e-2
[2] https://openai.com/chatgpt

Abstract

How the cortex performs its intricate computations, and how it adapts to the world around it, is one of the central mysteries in neuroscience. It is a longstanding belief that one of the main aims of the brain, and especially the cortex, is to infer the states of the world, such as the presence of objects, that underlie the sensory observations an animal makes. Currently the most discussed theory that formalizes this idea and proposes a biological implementation is classical hierarchical Predictive Coding (hPC), which hypothesizes the existence of dedicated 'error neurons' in cortex that signal errors of the internal model. While this theory has inspired much research, it is not clear how one of its central elements—the proposed learning algorithm—can be implemented with spiking neurons, which questions its biological plausibility. In this thesis we propose an alternative theory of learning and inference with spiking neurons, where errors are computed in neural dendrites, and synaptic connections are learned with biologically plausible voltage-dependent plasticity rules. We first build on existing work of inference and learning in spiking neural networks, and show how dendritic error computation can overcome an unsolved problem for learning in these networks. Specifically, when neural activity in the network is correlated, previously assumed Hebbian-like learning leads to pathological network activity, which learning with dendritic errors prevents. We then combine this model with other theories of learning in cortex to a theory of hierarchical inference with spiking neurons, and show that this theory is isomorphic to classical hPC while overcoming its biological implausibility. Last, we employ our framework to explain how 'mismatch responses', i.e., neural responses that signal the mismatch between an internal model and observations, emerge from inference and learning in cortex. Together, this work proposes a comprehensive theory of learning, inference and their signatures in cortex, and provides a range of readily testable predictions.

Zusammenfassung

Wie der Cortex seine komplexen Berechnungen durchführt und sich an die Welt um ihn herum anpasst, ist eines der zentralen Rätsel in der Neurowissenschaft. Seit langem wird vermutet, dass eines der Hauptziele des Gehirns, insbesondere des Cortex, darin besteht, den Zustand der Welt, wie zum Beispiel das Vorhandensein von Objekten, der den sensorischen Beobachtungen eines Tieres zugrunde liegt, zu erschließen. Derzeit ist die am meisten diskutierte Theorie, die diese Idee formalisiert und eine biologische Umsetzung vorschlägt, die klassische hierarchische Prädiktive Kodierung (hPC), die die Existenz dedizierter ”Fehlerneuronen” im Cortex postuliert, die Fehler des internen Modells signalisieren. Obwohl diese Theorie viel Forschung inspiriert hat ist nicht klar wie eines ihrer zentralen Elemente - der vorgeschlagene Lernalgorithmus - mit Spiking-Neuronen implementiert werden kann, was ihre biologische Plausibilität in Frage stellt. In der vorliegenden Arbeit schlagen wir eine alternative Theorie des Lernens und der Inferenz mit Spiking-Neuronen vor, bei der Fehler in den neuronalen Dendriten berechnet werden und synaptische Verbindungen mit biologisch plausiblen spannungsabhängigen Plastizitätsregeln erlernt werden. Wir bauen zunächst auf bestehenden Arbeiten zur Inferenz und zum Lernen in Spiking-Neuronennetzen auf und zeigen, wie die Fehlerberechnung in den Dendriten ein ungelöstes Problem des Lernens in diesen Netzwerken lösen kann. Genauer führt die zuvor angenommene Hebb’sche Lernregel zu pathologischer Netzwerkaktivität, wenn die neuronale Aktivität im Netzwerk korreliert ist, was durch das Lernen mit dendritischen Fehlern verhindert wird. Im Anschluss kombinieren wir dieses Modell mit anderen Theorien des Lernens im Cortex zu einer Theorie hierarchischer Inferenz mit Spiking-Neuronen und zeigen, dass diese Theorie isomorph zur klassischen hPC ist aber die biologische Unplausibilität ihrer Lernregeln überwindet. Schließlich nutzen wir unser Framework, um zu erklären, wie ”Mismatch-Responses”, d.h., neuronale Aktivität, die den Unterschied zwischen einem internen Modell und Beobachtungen signalisiert, als Folge der Inferenz und dem Lernen im Cortex entstehen. Zusammenfassend schlägt diese Arbeit eine umfassende Theorie des Lernens, der Inferenz und ihrer Signaturen im Cortex vor und liefert eine Reihe von direkt testbaren Vorhersagen.

List of publications

Constituting parts of this dissertation

Mikulasch*, F. A., Rudelt*, L., Wibral, M., & Priesemann, V. (2023). Where is the error? Hierarchical predictive coding through dendritic error computation. *Trends in Neurosciences*, 46(1), 45-59.

Mikulasch, F. A., Rudelt, L., & Priesemann, V. (2022). Visuomotor mismatch responses as a hallmark of explaining away in causal inference. *Neural computation*, 35(1), 27-37.

Mikulasch*, F. A., Rudelt*, L., & Priesemann, V. (2021). Local dendritic balance enables learning of efficient representations in networks of spiking neurons. *Proceedings of the National Academy of Sciences*, 118(50), e2021925118.

Other work published during PhD

Jähne*, S., **Mikulasch***, F. A., Heuer, H. G., Truckenbrodt, S., Agüi-Gonzalez, P., Grewe, K., Vogts, A., Rizzoli, S. O., & Priesemann, V. (2021). Presynaptic activity and protein turnover are correlated at the single-synapse level. *Cell Reports*, 34(11).

Previous work

Ecke, G. A., Bruijns, S. A., Hoelscher, J., **Mikulasch**, F. A., Witschel, T., Arrenberg, A. B., & Mallot, H. A. (2020). Sparse coding predicts optic flow specificities of zebrafish pretectal neurons. *Neural Computing and Applications*, 32, 6745-6754.

*These authors contributed equally

Contents

Acknowledgements v

Abstract vii

List of publications ix

1 Introduction and Background 1
- 1.1 Theory-driven approaches to brain function . . . 1
- 1.2 Models of successful brains . . . 2
- 1.3 A formal basis for world models, inference and learning . . . 4
- 1.4 Spiking neurons—the computational building blocks of cortex . . . 6
- 1.5 Learning with spiking neurons . . . 7
- 1.6 Motivation . . . 9
- 1.7 Thesis overview . . . 10
- References . . . 12

2 Local dendritic balance enables learning of efficient representations in networks of spiking neurons 16
- 2.1 Introduction . . . 17
- 2.2 Results . . . 18
- 2.3 Discussion . . . 22
- 2.4 Materials and Methods . . . 25
- References . . . 25

3 Where is the error? Hierarchical predictive coding through dendritic error computation 28
- 3.1 Neural models of inference in cortex . . . 29
- 3.2 Dendritic predictive coding in balanced spiking neural networks . . . 30
- 3.3 Is dendritic predictive coding biologically plausible? . . . 35
- 3.4 How can error responses arise in prediction neurons? . . . 38
- 3.5 Testable predictions . . . 39

3.6 Concluding remarks . . . 40
References . . . 41

4 Visuomotor mismatch responses as a hallmark of explaining away in causal inference 44
4.1 Introduction . . . 45
4.2 Theory . . . 46
4.3 Results . . . 47
4.4 Discussion . . . 49
4.5 Methods . . . 52
References . . . 53

5 Prediction mismatch responses arise as corrections of a predictive spiking code 56
5.1 Introduction . . . 57
5.2 Theory . . . 58
5.3 Results . . . 59
5.4 Discussion . . . 61
5.5 Methods . . . 64
References . . . 65

6 Overall Discussion 73
6.1 Unifying and extending theories of cortical computation . . . 74
6.2 Missing bits and open avenues . . . 75
6.3 Practical applications for theories of cortical computation . . . 77
6.4 Testing theories of cortical computation . . . 79
6.5 Conclusion . . . 80
References . . . 81

A Appendix 84
A.1 Supplementary Material to: Local dendritic balance enables learning of efficient representations in networks of spiking neurons . . . 85
A.2 A comment about scaling time-steps . . . 109
A.3 Generalization to arbitrary spike kernels and hierarchical and recurrent predictions . . . 110
A.4 Reinforced spiking algorithm . . . 115

1 Introduction and Background

1.1 Theory-driven approaches to brain function

"A wing would be a most mystifying structure if one did not know that birds flew. One might observe that it could be extended a considerable distance, that it had a smooth covering of feathers with conspicuous markings, that it was operated by powerful muscles, and that strength and lightness were prominent features of its construction. These are important facts, but by themselves they do not tell us that birds fly. Yet without knowing this, and without understanding something of the principles of flight, a more detailed examination of the wing itself would probably be unrewarding. I think that we may be at an analogous point in our understanding of the sensory side of the central nervous system."

More than 60 years later, and despite significant progress in our understanding of neural processing, this statement by Barlow (1961) is still relevant for large parts of the the central nervous system, especially the cortex. By now there is a great deal of data about the molecular, sub-cellular, cellular, circuit level and area level structure and mechanisms in cortex, but how this in the end leads to the intelligent behaviour we observe in animals—how the brain flies, so to speak—mostly remains a mystery. As Barlow argues, this question will not be solved by gathering more data alone, since this will not immediately reveal how the brain performs its intricate computations.

Perhaps the biggest challenge when trying to understand the underlying principles of cortex from observation is the complexity of the associated data. Recent efforts have recorded several 10.000 cortical cells simultaneously and reconstructed their connectivity consisting of hundreds of millions of synapses (MICrONS-Consortium et al. 2021), but understanding the computations in cortex from these vast datasets proves highly nontrivial. Current approaches make use of advanced machine learning techniques with the goal to extract the essential computational motives that are at play (Wang et al. 2023). This, for example, might allow us to understand the inductive biases[1] and invariances that are utilized by neural systems (Baroni et al. 2023; Sinz et al. 2019). Ultimately, while these

[1] Roughly, inductive biases are prior assumptions of a learning algorithm about the structure of the data, ideally helping it to learn quickly and generalize well (Goyal et al. 2022; Sinz et al. 2019).

methods might provide us bias free insights into the algorithms that are used by neural circuits, it remains a massive undertaking to collect and curate the required data and implement the necessary data analysis tools to reach an understanding of computations and plasticity in the brain.

The complementary approach, which Barlow advocates for, is to develop theories of neural computation from first principles. A general recipe is to follow the hierarchy of Marr's three levels (Chater et al. 2011; Marr 2010): One first specifies the *computational goal* the system has to solve (level 1), finds an *algorithm* that reaches this goal (level 2), and lastly develops a theory how this algorithm might be *implemented in neural hardware* (level 3), yielding a direct understanding of neural dynamics in terms of function. Typically such theories focus on one or few computational aspects, such as inference (Rao and Ballard 1999), attention (Reynolds et al. 2009), memory (Whittington, Muller, et al. 2020) or criticality (Zeraati et al. 2021). Most of these theories are therefore by construction incomplete and thus cannot be expected to make quantitative predictions in neural circuits that likely have many more requirements than what was specified in the theory. On the bright side, they directly offer a qualitative understanding of what functions could be implemented by specific neural mechanism, and they can similarly provide qualitative predictions for how neural systems behave in experiments.

This thesis follows this theory driven approach to neural function. More specifically, we develop a theory of how *inference in an internal model* of the animal can be *implemented by spiking neurons* in cortex, and how this internal model is *improved via synaptic plasticity*.

1.2 Models of successful brains

The reason why many animals and especially humans have such an enormous brain is to produce behaviour in order to survive and proliferate in a complex world. But what does this require? From our own experience we know that we handle this to a large extent by having access to a representation of the world, which we ultimately use to judge consequences of our actions. While this is only a vague intuition gained by introspection, there exists a formal equivalent in control theory: The good regulator theorem (Conant et al. 1970), which states that "Every good regulator of a system must be a model of that system". Applied to the brain, the theorem implies that in order to achieve behavioral goals the brain must model the world, or to be more specific, at least the animal itself and the ecologically relevant aspects of its environment.

While a model of the world around the animal can in principle be implicit in the action generation (e.g., a reflex arc can be understood as modeling the fact that a certain stimulus

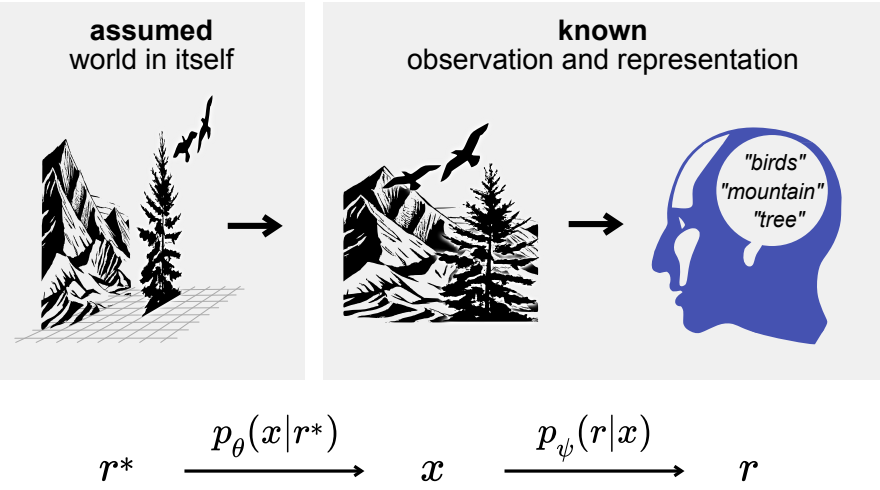

Figure 1.1: A world model $p_\theta(x, r^*)$ can be thought of as an assumption of the animal how unobservable world states r^* are distributed ($p_\theta(r^*)$) and how observations x are generated from them ($p_\theta(x|r^*)$) (Jones et al. 2011). Given an observation, the goal is then to compute the possible underlying world states, via an inverse function $p_\psi(r|x)$. The goodness of the model can only be evaluated on observations x. Here, and in the remainder of the thesis, we ignore the temporal aspect of the evolution of world states r, which in principle is an integral part of any world model. We will come back to this in the discussion (Chapter 6.2).

implies harm), higher level processing and reasoning likely requires more explicit and separate general purpose models. Intuitively, the value in such models is that, on the one hand they can enable predicting future world states and therewith anticipatory actions and planning (Moerland et al. 2023); on the other hand it is known that a good representation of sensory states greatly reduces the complexity of finding desired input-output mappings (Bengio, Courville, et al. 2013)[2] and appropriate actions, especially under sparse feedback regarding the value of these actions (Dayan 1993; Yarats et al. 2021). World models might therefore form a solid basis for intelligent behavior even in dynamic, uncertain and unfamiliar situations (Friston et al. 2021; Moerland et al. 2023).

From an outside perspective a successful world model can be defined by requiring that the internal states of the animal somehow correspond to the states of the environment (e.g., through an isomorphism; Conant et al. 1970). In practice, however, such a definition can become problematic. In mammals there is ample evidence that a big part of the model, that is, how sensory stimuli are represented in cortex, is learned during the life of the animal (Dosher et al. 2017; Löwel et al. 1992; Wiesel et al. 1963), for which only indirect access to world

[2] This principle is also part of the recent success of foundation models (Bommasani et al. 2021).

states through the sensory faculties is available. Typically, theories of brain function and machine intelligence thus do not refer to objective external states, but require that the internal representations of world models explain the *sensory observations* (which result from external states, Fig 1.1; Friston et al. 2021; Ha et al. 2018).[3] While these models therefore do not necessarily learn internal states that directly correspond to any unobservable world states, they nevertheless can result in useful representations that enable solving real-world tasks.

Having outlined the basic theory of what the brain aims to achieve (Marr's level 1), two immediate follow-up questions arise: What does it take to implement a world model (level 2)? And how might this be achieved in neural hardware (level 3)? There have been several heroic attempts at answering these two questions, such as Adaptive Resonance Theory (Grossberg 2013), the Hierarchical Temporal Memory model (Hawkins et al. 2016), Predictive Coding and Active Inference (Friston 2010; Rao and Ballard 1999), or more straightforward Bayesian inference in recurrent neural networks (Deneve 2008; Kappel et al. 2014; Nessler et al. 2013; Rao 2004b), just to name a few. The algorithmic side of many of these attempts can be subsumed under the umbrella of approximate Bayesian inference and the Expectation-Maximization (**EM**) algorithm (Goodfellow et al. 2016a), which provide an elegant and principled approach to inference and learning in the brain. Because these methods will also be the formal underpinning of the models in this thesis, I will shortly introduce their main ideas (section 1.3). I will then briefly outline approaches to understand how these algorithms might be implemented with spiking neurons in the brain (sections 1.4 & 1.5).

1.3 A formal basis for world models, inference and learning

Bayesian inference formalizes how an animal extracts the probable state of the world from noisy and incomplete observations. To illustrate this we can look at a very general form of a world model, defined by probability distributions

$$p_\theta(x, r) = p_\theta(x|r)p_\theta(r), \tag{1.1}$$

[3] This distinction between estimating objective world states and estimating states under some model is not only important on a conceptual level. The Bayesian framework from the ground up operates with beliefs, and it would be wrong to assume that in this framework animals always have to optimally infer external states. Rather, the animal estimates states under some given model, which also means that experiments can only test whether an animal performs inference under some model or not. This is often a source of confusion, since in most models the (very reasonable) core assumption is that the internal model of the animal is in some sense optimal, which, however, does not necessarily have to be the case. See also the discussions in (Jones et al. 2011; Rahnev et al. 2018) and especially (Ma et al. 2023).

where θ are model parameters, x is the sensory observation and r the representation in the animal, which stands for the inaccessible underlying state of the world (Fig 1.1). Here, $p_\theta(x|r)$ can be understood as the *assumed* physical process that generates observations from world states and $p_\theta(r)$ as a prior over world states. Bayesian inference aims to invert this generative model, and estimates the world state r given observations x, which can be achieved using Bayes' theorem

$$p_\theta(r|x) = \frac{p_\theta(x|r)p_\theta(r)}{\sum_r p_\theta(x|r)p_\theta(r)}. \tag{1.2}$$

This estimate of the distribution of possible world states can be considered optimal under the given model, e.g., for decision making (Lindig-León et al. 2022), and the idea that the brain typically is 'Bayes-optimal' has long been pervasive throughout theoretical neuroscience (Fiser et al. 2010; Knill et al. 2004).

Nevertheless, there are also many cases where perception does not seem to be straightforward Bayesian inference (Rahnev et al. 2018), and the concept of Bayes-optimal perception has been criticised on several grounds[4], importantly for often being computationally intractable (Jones et al. 2011). Formally, this relates to the problem that the normalization of the model posterior, which requires a summation over all world states (Eq 1.2), becomes impossible to compute for most interesting models. One solution to this issue is to only approximate the posterior $p_\theta(r|x)$ with a simpler and easily normalizable distribution $p_\psi(r|x)$. In this view, the simple distribution then corresponds to the inference function performed by the brain (Fig 1.1), where ψ are 'physical' neural parameters. To fit this approximation to the actual posterior one defines the function (Goodfellow et al. 2016a)

$$\begin{aligned}\mathcal{F}(\psi,\theta) =& \langle \log p_\theta(x) - D_{KL}[p_\psi(r|x)||p_\theta(r|x)]\rangle_{p^*(x)} && (1.3)\\ =& \langle \log p_\theta(x,r) - \log p_\psi(r|x)\rangle_{p_\psi(r|x)p^*(x)}, && (1.4)\end{aligned}$$

which is maximized when the Kullback-Leibler distance between approximate and actual posterior $D_{KL}[p_\psi(r|x)||p_\theta(r|x)]$ is as small as possible (Eq 1.3). The trick here is that $\mathcal{F}$ is easy to find, as we don't have to compute the actual posterior and its normalization (Eq 1.4). A good approximation $p_\psi(r|x)$ of $p_\theta(r|x)$ can then be found using established optimization methods, such as gradient ascent on $\mathcal{F}$. This formulation of approximate Bayesian inference using variational methods thus allows for tractable models of inference in the brain, and

[4] Another critical point is that models of Bayesian inference in the brain are extremely flexible, and often observations can be explained using arbitrary prior or likelihood distributions (Jones et al. 2011). A similar problem as that for these abstract models of cognition applies also the mapping of the algorithm to the biophysical implementation (Sprevak 2021). We will come back to this in the discussion (chapter 6).

might in some cases also explain why decision making can be Bayes-suboptimal (Lindig-León et al. 2022).

Aside from performing inference, the brain also has to update its internal model if it doesn't explain the sensory observations well. The performance of the internal model can be measured by comparing the model distribution $p_\theta(x)$ to the distribution of actual observations $p^*(x)$

$$D_{KL}[p^*(x)||p_\theta(x)] \propto \langle \log p_\theta(x) \rangle_{p^*(x)}. \tag{1.5}$$

Conveniently, this model log-likelihood is already part of the performance measure $\mathcal{F}$ we defined previously to implement variational inference (Eq 1.3). One of the important insights for world model learning has been that a world model that is consistent with sensory observations can be found by performing inference using an approximate posterior $p_\psi(r|x)$ and maximizing the lower bound on the log-likelihood $\mathcal{F}$ in respect to both θ and ψ (Neal et al. 1998). This is a generalization of the classical EM algorithm (Goodfellow et al. 2016a), which solves the chicken and egg problem of having to both improve a model of sensory observations given their representations, and find these representations of sensory observations given the model (Friston 2018).

1.4 Spiking neurons—the computational building blocks of cortex

Variational inference and EM-learning provide an elegant framework to understand the world model algorithm that could be implemented in the brain, but the question remains how this algorithm can be realized in neural hardware. Since the brain is constrained to use spiking neurons that communicate with discrete pulses, to understand this connection it is important to formalize how the required computations could be implemented using this form of neural communication. One specific example where such a direct connection to spiking neurons is missing is classical hierarchical predictive coding (**hPC**), which has only been formulated within a rate-based framework (Millidge et al. 2021). As we will also discuss in Chapter 3, this has prevented a clear mapping to single cell physiology which makes testing of this theory quite difficult (Kogo et al. 2015). This example highlights the importance of building theories with spiking neurons, also considering that many of the features of cortical dynamics and plasticity might be a result of this particular neural design.

A major open question encountered when building theories with spiking neurons is how exactly information is encoded within spike trains (Fiser et al. 2010; Gerstner et al. 2002). Contributing to this issue, it is not even clear why the brain uses spiking neurons to

compute in the first place, especially since in rare cases neural systems can also employ neurons with graded potentials (Juusola et al. 1996). Suggestions range from the idea that transmitting information over long cables is more efficient via spikes (Manwani et al. 1999), that the sparseness of activity generally improves energy efficiency, as for example targeted by neuromorphic computing (Marković et al. 2020), or it might also be that the all-or-nothing nature of spiking provides a powerful inductive bias (Bengio, Courville, et al. 2013) which is very appropriate to model the sparse and event-based statistics of the world. Whatever the reason, it is evident that information in spike trains has to be encoded such that downstream neurons can read out and use this information via plausible neural mechanisms (Gerstner et al. 2002). Formally, these readout mechanisms are described in the model distribution $p_\theta(x|r)$, since it dictates how information about sensory states can be recovered from spike representations. This allows for a dual interpretation of $p_\theta(x|r)$ as both part of a world model (Fig 1.1) and a decoder, as it is often understood in machine learning theory (Goodfellow et al. 2016b). It could be worthwhile to try to separate these conflated readout and world modeling aspects to improve the interpretability of the theory, but this dual interpretation also has its merits, as it makes explicit that the inductive biases imparted by the spike encoding and by the actual world model might not be easily disentangled. Ultimately, while these theoretical considerations are important, in practice many theories, including the theory presented in this thesis, make use of highly simplified, i.e., linear model distributions (e.g., Brendel et al. 2020). Not only does this afford the analytical tractability of these theories, but it also matches how in most computational models neurons read out information from other neurons, that is, through a linear combination of inputs.

1.5 Learning with spiking neurons

Another difficulty when working with spiking neurons, and the main reason why bridging from rate to spiking models is not easy, is that learning with spiking neurons is not straightforward. Typically, rate-based artificial neural networks are trained using gradient based methods (Goodfellow et al. 2016a), but unfortunately the gradient of the often discontinuous spiking process of most spiking neuron models is not well defined (Neftci et al. 2019). On top of the gradient issue, biological spiking neurons are constrained to employ local learning rules, which means plasticity can only rely on locally available information.

There is no general recipe to obtain biologically plausible local learning rules that are able to optimize any given goal-function. However, there are several ideas how this problem can be tackled, most of which can be categorized into three general approaches:

i) Computing gradients using surrogate gradients (Neftci et al. 2019), which heuristically introduces a well-defined derivative of the spiking mechanism. This allows to employ backpropagation algorithms as used in machine learning models. However, these algorithms have to be forcibly localized, for example by leaving away inaccessible information or projecting it to other neurons with random connections (Bellec et al. 2020).[5]

ii) Computing gradients through stochastic spike mechanisms, which is possible since the likelihood of a spiketrain is typically differentiable even if the discrete spike mechanism is not (Bengio, Léonard, et al. 2013; Pfister et al. 2006). Some of these learning algorithms are local by design, like REINFORCE learning (Williams 1992), which, however, due to its slow convergence is not suited to learn world models in recurrent neural networks (Appendix A.4).

iii) Abstracting away the spiking mechanism (e.g., through neural sampling or minimizing an energy function; Brendel et al. 2020; Buesing et al. 2011) and only comparing a readout to a target (Brendel et al. 2020; Kappel et al. 2014; Nicola et al. 2017). The earliest well known model that applied this approach are Hopfield networks, which store memories with Hebbian plasticity and retrieve them via discrete gradient descent on an energy function (Hopfield 1982). For simplicity, in the following this general approach will be referred to as 'energy-based networks'.[6] Difficulties are that the required neural architectures typically can only be derived for relatively simple energy functions, and the derived learning rules also are not necessarily local.

Overall, energy-based networks so far have proven most fruitful in deriving biologically plausible spike-timing-dependent plasticity rules for world model learning (Brendel et al. 2020; Kappel et al. 2014). Perhaps the biggest advantage this approach brings is that it can dissolve the clear distinction between the world model $p_\theta(x, r)$ and the approximate posterior $p_\psi(r|x)$. While in the other approaches we discussed (i & ii) these two

[5] Note, that there are also several other approaches to implement backpropagation with local learning rules, most of which update their weights after an equilibration phase (Lillicrap et al. 2020; Whittington and Bogacz 2019), which is related most closely to approach (iii). However, except for few notable examples (e.g., Guerguiev et al. 2017), these ideas have typically been only demonstrated with rate-based neurons.

[6] To define an energy for networks sampling from a distribution $p(r)$ we can take inspiration from statistical physics. Writing $p(r)$ as a Boltzmann distribution $p(r) = 1/Z \exp(-\beta E(r))$ (assuming $\forall r : p(r) > 0$), we can define $E(r) \propto -\log p(r)$ as the energy.

distributions in general have to be considered separately, which leaves the question how their interaction can be thought of in biological terms, in energy-based networks the posterior $p_\psi(r|x)$ is found through a network whose parameters directly relate to model parameters θ.[7] Therefore, updating the model parameters θ and updating the neural network parameters ψ can be considered equivalent, which allows to directly understand biological synaptic plasticity as learning a model of the world.

1.6 Motivation

The theory of energy-based networks has enabled great progress over the last decades in providing answers to the question what mechanisms spiking neurons in cortex might use to learn and perform inference in a model of the world (Brendel et al. 2020; Denève et al. 2016; Földiak 1990; Kappel et al. 2014; Nessler et al. 2013). In many of these models common themes have emerged, like the proposal that strong lateral inhibition helps neurons cooperate to explain the sensory data well (Denève et al. 2016; Földiak 1990; Nessler et al. 2013), to name one central example. Yet, despite the clear similarities between those models, there exists no general picture how exactly they could map to cortical physiology, which hinders the direct testing of their predictions in experiment.

At the same time, classical hPC—even though it is not known how it can be implemented by spiking neurons—is widely considered to be the most promising unifying theory of inference and learning in cortex (Friston 2018). Much of this sentiment derives from the ease with which classical hPC explains the influence of the ubiquitous top-down connections in cortex, which project from higher-level cortical areas to lower levels (Walsh et al. 2020). One core feature of these hierarchical interactions are for example mismatch responses, which seem to signal the mismatch between an internal model of the animal and sensory observations (Fig 1.2), and commonly classical hPC is invoked to make sense of these observations (De Lange et al. 2018). In contrast, spiking neuron models of inference and learning in cortex have only rarely considered the role of top-down connections (e.g., Rao 2004a), and a connection of these models to the experimental results that are explained by classical hPC is missing. It remains to be concluded that in both classical hPC, as well as spiking models of inference, significant explanatory gaps persist.

[7] One alternative to the direct relation of parameters θ and ψ in many energy-based networks worth mentioning is the wake-sleep algorithm (Hinton et al. 1995), where recognition weights ψ are trained to align with generative weights θ in a separate learning phase.

Figure 1.2: Mismatch responses are a prominent feature of cortical processing and might help us distinguish between different theories of cortical computation. They have been observed in several forms, such as strong neural activity in response to the omission of an expected stimulus, to a mismatch of information in different sensory modalities, to an unexpected stimulus, or reduced activity in response to expected stimuli (Chapter 3; De Lange et al. 2018). In this thesis we will specifically look at multi-modal mismatch responses (Chapter 4) and prediction mismatch responses (Chapter 5).

1.7 Thesis overview

In this thesis we aim to address some of these explanatory gaps of existing theories. The individual chapters will each address specific shortcomings and open questions, outlined below. Together, they collect a broad range of theoretical considerations and empirical findings into a coherent framework of world model learning and inference in cortex.

In Chapter 2, we will investigate how local learning for a simple model of sensory data can be implemented by a population of spiking neurons. While this is an old question in theoretical neuroscience (Brendel et al. 2020; Földiak 1990), there remains the fundamental open question how learning can be achieved with neurons that show correlated activity. The solution we propose is that neurons compute coding errors via a balance of excitation and inhibition on neural dendrites, and exploit them for both inference and learning. This provides directly testable predictions for the interaction of excitation and inhibition for synaptic plasticity.

In Chapter 3 we will then combine this solution with several other previously unconnected models of neural computation to propose a comprehensive theory of *hierarchical* inference and learning in cortex. We will show that this theory is isomorphic to

classical hPC (Rao and Ballard 1999), but overcomes its central open questions, which mostly regard its biological plausibility. Furthermore, towards making the proposed theory testable, we suggest how the algorithm might be embedded into the cortical microcircuit and lay out specific computational roles for clearly defined neuron types in cortex.

In the remainder of the thesis (Chapter 4 & 5) we will look into how specific experimental observations can be modeled in our framework, and what experimental predictions this generates. Concretely, we address one of the most important questions raised when comparing any theory of inference in cortex without explicit error neurons to classical hPC: Why do we observe strong neural responses in cortex when prediction errors are large (Fig 1.2)? Because theories with or without error neurons give different answers, this will allow us to propose various experiments which could be conducted to distinguish between them.

References

Barlow, H. B. (1961). "Possible principles underlying the transformation of sensory messages". *Sensory communication* 1, pp. 217–233.

Baroni, L., M. Bashiri, K. F. Willeke, J. Antolík, and F. H. Sinz (2023). "Learning invariance manifolds of visual sensory neurons". In: *NeurIPS Workshop on Symmetry and Geometry in Neural Representations*. PMLR, pp. 301–326.

Bellec, G., F. Scherr, A. Subramoney, E. Hajek, D. Salaj, R. Legenstein, and W. Maass (2020). "A solution to the learning dilemma for recurrent networks of spiking neurons". *Nature communications* 11, p. 3625.

Bengio, Y., A. Courville, and P. Vincent (2013). "Representation learning: A review and new perspectives". *IEEE transactions on pattern analysis and machine intelligence* 35, pp. 1798–1828.

Bengio, Y., N. Léonard, and A. Courville (2013). "Estimating or propagating gradients through stochastic neurons for conditional computation". *arXiv preprint arXiv:1308.3432*.

Bommasani, R., D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, et al. (2021). "On the opportunities and risks of foundation models". *arXiv preprint arXiv:2108.07258*.

Brendel, W., R. Bourdoukan, P. Vertechi, C. K. Machens, and S. Denéve (2020). "Learning to represent signals spike by spike". *PLoS computational biology* 16, e1007692.

Buesing, L., J. Bill, B. Nessler, and W. Maass (2011). "Neural dynamics as sampling: a model for stochastic computation in recurrent networks of spiking neurons". *PLoS computational biology* 7, e1002211.

Chater, N., N. Goodman, T. L. Griffiths, C. Kemp, M. Oaksford, and J. B. Tenenbaum (2011). "The imaginary fundamentalists: The unshocking truth about Bayesian cognitive science". *Behavioral and Brain Sciences* 34, p. 194.

Conant, R. C. and W. Ross Ashby (1970). "Every good regulator of a system must be a model of that system". *International journal of systems science* 1, pp. 89–97.

Dayan, P. (1993). "Improving generalization for temporal difference learning: The successor representation". *Neural computation* 5, pp. 613–624.

De Lange, F. P., M. Heilbron, and P. Kok (2018). "How do expectations shape perception?" *Trends in cognitive sciences* 22, pp. 764–779.

Deneve, S. (2008). "Bayesian spiking neurons I: inference". *Neural computation* 20, pp. 91–117.

Denève, S. and C. K. Machens (2016). "Efficient codes and balanced networks". *Nature Neuroscience* 19, pp. 375–382.

Dosher, B. and Z.-L. Lu (2017). "Visual perceptual learning and models". *Annual review of vision science* 3, pp. 343–363.

Fiser, J., P. Berkes, G. Orbán, and M. Lengyel (2010). "Statistically optimal perception and learning: from behavior to neural representations". *Trends in cognitive sciences* 14, pp. 119–130.

Földiak, P. (1990). "Forming sparse representations by local anti-Hebbian learning". *Biological cybernetics* 64, pp. 165–170.

Friston, K. (2010). "The free-energy principle: a unified brain theory?" *Nature reviews neuroscience* 11, pp. 127–138.
– (2018). "Does predictive coding have a future?" *Nature neuroscience* 21, pp. 1019–1021.
Friston, K., R. J. Moran, Y. Nagai, T. Taniguchi, H. Gomi, and J. Tenenbaum (2021). "World model learning and inference". *Neural Networks* 144, pp. 573–590.
Gerstner, W. and W. M. Kistler (2002). *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge university press.
Goodfellow, I., Y. Bengio, and A. Courville (2016a). *Deep learning*. MIT press. Chap. 19.
– (2016b). *Deep learning*. MIT press. Chap. 14.
Goyal, A. and Y. Bengio (2022). "Inductive biases for deep learning of higher-level cognition". *Proceedings of the Royal Society A* 478, p. 20210068.
Grossberg, S. (2013). "Adaptive Resonance Theory: How a brain learns to consciously attend, learn, and recognize a changing world". *Neural networks* 37, pp. 1–47.
Guerguiev, J., T. P. Lillicrap, and B. A. Richards (2017). "Towards deep learning with segregated dendrites". *Elife* 6, e22901.
Ha, D. and J. Schmidhuber (2018). "World models". *arXiv preprint arXiv:1803.10122*.
Hawkins, J. and S. Ahmad (2016). "Why neurons have thousands of synapses, a theory of sequence memory in neocortex". *Frontiers in neural circuits*, p. 23.
Hinton, G. E., P. Dayan, B. J. Frey, and R. M. Neal (1995). "The" wake-sleep" algorithm for unsupervised neural networks". *Science* 268, pp. 1158–1161.
Hopfield, J. J. (1982). "Neural networks and physical systems with emergent collective computational abilities." *Proceedings of the national academy of sciences* 79, pp. 2554–2558.
Jones, M. and B. C. Love (2011). "Bayesian fundamentalism or enlightenment? On the explanatory status and theoretical contributions of Bayesian models of cognition". *Behavioral and brain sciences* 34, pp. 169–188.
Juusola, M., A. S. French, R. O. Uusitalo, and M. Weckström (1996). "Information processing by graded-potential transmission through tonically active synapses". *Trends in neurosciences* 19, pp. 292–297.
Kappel, D., B. Nessler, and W. Maass (2014). "STDP installs in winner-take-all circuits an online approximation to hidden Markov model learning". *PLoS computational biology* 10, e1003511.
Knill, D. C. and A. Pouget (2004). "The Bayesian brain: the role of uncertainty in neural coding and computation". *Trends in Neurosciences* 27, pp. 712–719.
Kogo, N. and C. Trengove (2015). "Is predictive coding theory articulated enough to be testable?" *Frontiers in computational neuroscience*, p. 111.
Lillicrap, T. P., A. Santoro, L. Marris, C. J. Akerman, and G. Hinton (2020). "Backpropagation and the brain". *Nature Reviews Neuroscience* 21, pp. 335–346.
Lindig-León, C., N. Kaur, and D. A. Braun (2022). "From Bayes-optimal to heuristic decision-making in a two-alternative forced choice task with an information-theoretic bounded rationality model". *Frontiers in Neuroscience* 16.
Löwel, S. and W. Singer (1992). "Selection of intrinsic horizontal connections in the visual cortex by correlated neuronal activity". *Science* 255, pp. 209–212.

Ma, W. J., K. P. Kording, and D. Goldreich (2023). *Bayesian Models of Perception and Action: An Introduction*. Chap. 15.

Manwani, A. and C. Koch (1999). "Detecting and estimating signals in noisy cable structures, II: information theoretical analysis". *Neural Computation* 11, pp. 1831–1873.

Marković, D., A. Mizrahi, D. Querlioz, and J. Grollier (2020). "Physics for neuromorphic computing". *Nature Reviews Physics* 2, pp. 499–510.

Marr, D. (2010). *Vision: A computational investigation into the human representation and processing of visual information*. MIT press.

MICrONS-Consortium, J. A. Bae, M. Baptiste, C. A. Bishop, A. L. Bodor, D. Brittain, J. Buchanan, D. J. Bumbarger, M. A. Castro, B. Celii, et al. (2021). "Functional connectomics spanning multiple areas of mouse visual cortex". *BioRxiv*.

Millidge, B., A. Seth, and C. L. Buckley (2021). "Predictive Coding: a Theoretical and Experimental Review". *arXiv:2107.12979*. arXiv: 2107.12979.

Moerland, T. M., J. Broekens, A. Plaat, C. M. Jonker, et al. (2023). "Model-based reinforcement learning: A survey". *Foundations and Trends® in Machine Learning* 16, pp. 1–118.

Neal, R. M. and G. E. Hinton (1998). "A view of the EM algorithm that justifies incremental, sparse, and other variants". In: *Learning in graphical models*. Springer, pp. 355–368.

Neftci, E. O., H. Mostafa, and F. Zenke (2019). "Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks". *IEEE Signal Processing Magazine* 36, pp. 51–63.

Nessler, B., M. Pfeiffer, L. Buesing, and W. Maass (2013). "Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity". *PLoS computational biology* 9, e1003037.

Nicola, W. and C. Clopath (2017). "Supervised learning in spiking neural networks with FORCE training". *Nature communications* 8, p. 2208.

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

Rahnev, D. and R. N. Denison (2018). "Suboptimality in perceptual decision making". *Behavioral and brain sciences* 41, e223.

Rao, R. P. (2004a). "Hierarchical Bayesian inference in networks of spiking neurons". *Advances in neural information processing systems* 17.

Rao, R. P. (2004b). "Bayesian computation in recurrent neural circuits". *Neural computation* 16, pp. 1–38.

Rao, R. P. and D. H. Ballard (1999). "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects". *Nature neuroscience* 2, pp. 79–87.

Reynolds, J. H. and D. J. Heeger (2009). "The normalization model of attention". *Neuron* 61, pp. 168–185.

Sinz, F. H., X. Pitkow, J. Reimer, M. Bethge, and A. S. Tolias (2019). "Engineering a less artificial intelligence". *Neuron* 103, pp. 967–979.

Sprevak, M. (2021). "Predictive coding IV: The implementation level". *[Preprint]*.

Walsh, K. S., D. P. McGovern, A. Clark, and R. G. O'Connell (2020). "Evaluating the neurophysiological evidence for predictive processing as a model of perception". *Annals of the new York Academy of Sciences* 1464, p. 242.

Wang, E. Y., P. G. Fahey, K. Ponder, Z. Ding, A. Change, T. Muhammad, S. Patel, Z. Ding, D. T. Tran, J. Fu, et al. (2023). "Towards a foundation model of the mouse visual cortex". *bioRxiv*, pp. 2023–03.

Whittington, J. C. and R. Bogacz (2019). "Theories of error back-propagation in the brain". *Trends in cognitive sciences* 23, pp. 235–250.

Whittington, J. C., T. H. Muller, S. Mark, G. Chen, C. Barry, N. Burgess, and T. E. Behrens (2020). "The Tolman-Eichenbaum machine: unifying space and relational memory through generalization in the hippocampal formation". *Cell* 183, pp. 1249–1263.

Wiesel, T. N. and D. H. Hubel (1963). "Single-cell responses in striate cortex of kittens deprived of vision in one eye". *Journal of neurophysiology* 26, pp. 1003–1017.

Williams, R. J. (1992). "Simple statistical gradient-following algorithms for connectionist reinforcement learning". *Reinforcement learning*, pp. 5–32.

Yarats, D., A. Zhang, I. Kostrikov, B. Amos, J. Pineau, and R. Fergus (2021). "Improving sample efficiency in model-free reinforcement learning from images". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35, pp. 10674–10681.

Zeraati, R., V. Priesemann, and A. Levina (2021). "Self-organization toward criticality by synaptic plasticity". *Frontiers in Physics* 9, p. 619661.

2 Local dendritic balance enables learning of efficient representations in networks of spiking neurons

Published at	Proceedings of the National Academy of Sciences 118.50 (2021): e2021925118
DOI	10.1073/pnas.2021925118
Supplementary Material	Appendix A.1
Source Code	github.com/Priesemann-Group/dendritic_balance

Contributions Conceptualization, Investigation, Writing - Original Draft

This work also constitutes a chapter in LR's PhD thesis. LR performed initial work on the theory, together we further developed the model to its final state. I implemented the code and created the figures. Both were involved in designing the research and writing the manuscript.

Local dendritic balance enables learning of efficient representations in networks of spiking neurons

Fabian A. Mikulasch[a,1], Lucas Rudelt[a,1], and Viola Priesemann[a,b,2]

[a]Max Planck Institute for Dynamics and Self-Organization, 37077 Göttingen, Germany; and [b]Bernstein Center for Computational Neuroscience Göttingen, 37077 Göttingen, Germany

Edited by Terrence J. Sejnowski, Salk Institute for Biological Studies, La Jolla, CA, and approved October 29, 2021 (received for review October 26, 2020)

How can neural networks learn to efficiently represent complex and high-dimensional inputs via local plasticity mechanisms? Classical models of representation learning assume that feedforward weights are learned via pairwise Hebbian-like plasticity. Here, we show that pairwise Hebbian-like plasticity works only under unrealistic requirements on neural dynamics and input statistics. To overcome these limitations, we derive from first principles a learning scheme based on voltage-dependent synaptic plasticity rules. Here, recurrent connections learn to locally balance feedforward input in individual dendritic compartments and thereby can modulate synaptic plasticity to learn efficient representations. We demonstrate in simulations that this learning scheme works robustly even for complex high-dimensional inputs and with inhibitory transmission delays, where Hebbian-like plasticity fails. Our results draw a direct connection between dendritic excitatory–inhibitory balance and voltage-dependent synaptic plasticity as observed in vivo and suggest that both are crucial for representation learning.

efficient coding | synaptic plasticity | balanced state | neural sampling | dendritic computation

Many neural systems have to encode high-dimensional and complex input signals in their activity. It has long been hypothesized that these encodings are highly efficient; that is, neural activity faithfully represents the input while also obeying energy and information constraints (1–3). For populations of spiking neurons, such an efficient code requires two central features: First, neural activity in the population has to be coordinated, such that no spike is fired superfluously (4); second, individual neural activity should represent reoccurring patterns in the input signal, which reflect the statistics of the sensory stimuli (2, 3). How can this coordination and these efficient representations emerge through local plasticity rules?

To coordinate neural spiking, choosing the right recurrent connections between coding neurons is crucial. In particular, recurrent connections have to ensure that neurons do not spike redundantly to encode the same input. An early result was that in unstructured networks the redundancy of spiking is minimized when excitatory and inhibitory currents cancel on average in the network (5–7), which is also termed loose global excitatory–inhibitory (E-I) balance (8). To reach this state, recurrent connections can be chosen randomly with the correct average magnitude, leading to asynchronous and irregular neural activity (5) as observed in vivo (4, 9). More recently, it became clear that recurrent connections can ensure a much more efficient encoding when E-I currents cancel not only on average, but also on fast timescales and in individual neurons (4), which is also termed tight detailed E-I balance (8). Here, recurrent connections have to be finely tuned to ensure that the network response to inputs is precisely distributed over the population. To achieve this intricate recurrent connectivity, different local plasticity rules have been proposed, which robustly converge to a tight balance and thereby ensure that networks spike efficiently in response to input signals (10, 11).

To efficiently encode high-dimensional input signals, it is additionally important that neural representations are adapted to the statistics of the input. Often, high-dimensional signals contain redundancies in the form of reoccurring spatiotemporal patterns, and coding neurons can reduce activity by representing these patterns in their activity. For example, in an efficient code of natural images, the activity of neurons should represent the presence of edges, which are ubiquitous in these images (3). Early studies of recurrent networks showed that such efficient representations can be found through Hebbian-like learning of feedforward weights (12, 13). With Hebbian learning the repeated occurrence of patterns in the input is associated with postsynaptic activity, causing neurons to become detectors of these patterns. Recurrent connections indirectly guide this learning process by forcing neurons to fire for distinct patterns in the input. Recent efforts rigorously formalized this idea for models of spiking neurons in balanced networks (11) and spiking neuron sampling from generative models (14–17). The great strength of these approaches is that the learning rules can be derived from first principles and turn out to be similar to spike-timing–dependent plasticity (STDP) curves that have been measured experimentally.

However, to enable the learning of efficient representations, these models have strict requirements on network dynamics. Most crucially, recurrent inhibition has to ensure that neural responses are sufficiently decorrelated. In the neural sampling

Significance

Neurons have to represent an enormous amount of sensory information. To represent this information efficiently, neurons have to adapt their connections to the sensory inputs. An unresolved problem is how this learning is possible when neurons fire in a correlated way. Yet, these correlations are ubiquitous in neural spiking, either because sensory input shows correlations or because perfect decorrelation of neural spiking through inhibition fails due to physiological transmission delays. We derived from first principles that neurons can, nonetheless, learn efficient representations if inhibition modulates synaptic plasticity in individual dendritic compartments. Our work questions pairwise Hebbian plasticity as a paradigm for representation learning and draws a link between representation learning and a dendritic balance of input currents.

Author contributions: F.A.M., L.R., and V.P. designed research; F.A.M. and L.R. performed research; F.A.M. implemented the simulations and created the figures; and F.A.M., L.R., and V.P. wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission.

[1]F.A.M. and L.R. contributed equally to this work.

[2]To whom correspondence may be addressed. Email: viola.priesemann@ds.mpg.de.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2021925118/-/DCSupplemental.

Published December 7, 2021.

approaches, learning therefore relies on strong winner-take-all dynamics (14–17). In the framework of balanced networks, transmission of inhibition has to be nearly instantaneous to ensure strong decorrelation (18). These requirements are likely not met in realistic situations, where neural activity often shows positive correlations (19–22).

We here derive a learning scheme that overcomes these limitations. First, we show that when neural activity is correlated, learning of feedforward connections has to incorporate nonlocal information about the activity of other neurons. Second, we show that recurrent connections can provide this nonlocal information by learning to locally balance specific feedforward inputs on the dendrites. In simulations of spiking neural networks we demonstrate the benefits of learning with dendritic balance over Hebbian-like learning for the efficient encoding of high-dimensional signals. Hence, we extend the idea that tightly balancing inhibition provides information about the population code locally and show that it can be used not only to distribute neural responses over a population, but also for an improved learning of feedforward weights.

Results

The goal in this paper is to efficiently encode a continuous high-dimensional input signal by neural spiking. In the following, we explain how neurons can learn efficient representations of these inputs through local plasticity mechanisms. We first show how a tight somatic balance can guide neural spiking to distribute the encoding over the population. We then show how a tight balance on the level of dendrites can guide the learning of efficient representations in the feedforward weights.

Background: Efficient Encoding by Spiking Neurons with Tight E-I Balance.

Setup. Continuous spatiotemporal inputs $\mathbf{x}(t)$ drive a recurrently connected spiking neural network, which encodes the inputs through responses $\mathbf{z}(t)$ (Fig. 1*A*). Feedforward weights F_{ji} indicate how strongly inputs $x_i(t)$ couple to neuron j, and recurrent weights W_{jk} provide coupling between the neurons. Inputs $x_i(t)$ are always positive, to ensure that single synapses act either excitatory or inhibitory, but not both. Neurons in the network encode the inputs by emitting spikes, which then elicit postsynaptic potentials (PSPs) $\mathbf{z}(t)$. The PSPs are modeled as a sum of exponentially decaying depolarizations $z_j(t) = \sum_{t_s^j \le t-\delta} \exp(-\frac{t-\delta-t_s^j}{\tau})$ with decay time τ for each spike of neuron j at times t_s^j. PSPs arrive after one timestep δ, which we interpret as a finite transmission delay of neural communication.

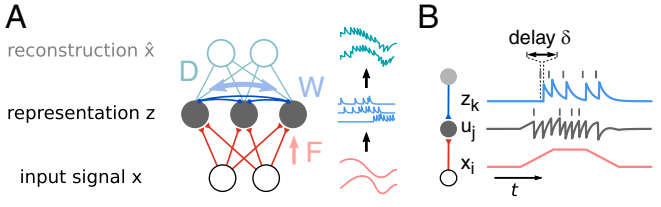

Fig. 1. The task is to efficiently encode analog input signals $\mathbf{x}$ by the response of a population of spiking neurons $\mathbf{z}$. (*A*) To that end, neurons couple to the input via feedforward weights F (dominated by excitation) and to each other via recurrent weights W (dominated by inhibition). From the encoding an external observer can decode an approximation $\hat{\mathbf{x}}$ of the original input signal $\mathbf{x}$ via a linear transformation D. (*B*) The membrane potential u_j of neuron j is a linear sum of continuous inputs x_i and spike traces z_k. Spikes cause an immediate self-inhibition, which can be seen as an approximate reset of u_j. Spikes of other neurons are transmitted with a delay δ. When recurrent weights are learned such that recurrent input z_k cancels feedforward input x_i, u_j is balanced and reflects the global encoding error $\mathbf{x} - \hat{\mathbf{x}}$. In that case, spikes are fired only when the encoding error is high, so that the spike encoding is efficiently distributed over the population.

Our model is similar to those in previous studies of balanced spiking networks (11, 23).

The goal is to find the most efficient spike encoding that enables the best reconstruction of the input, while the average firing rate of individual neurons is held fixed (see *SI Appendix*, section B for details). To test the reconstruction of the input, we consider the best linear readout $\hat{\mathbf{x}}(t) = D\mathbf{z}(t)$ from the neural response and quantify the mean decoder loss

$$\mathcal{L} = \frac{1}{2N_x} \left\langle ||\mathbf{x}(t) - \hat{\mathbf{x}}(t)||^2 \right\rangle_t = \frac{1}{2N_x} \left\langle ||\mathbf{x}(t) - D\mathbf{z}(t)||^2 \right\rangle_t, \quad [1]$$

where N_x is the number of inputs. It is important to note that the readout is not part of the network, but serves only as a guidance to define a computational goal. Hence, learning an efficient code amounts to minimizing $\mathcal{L}$ via local plasticity rules on F_{ji} and W_{jk}, given the best decoder D and a fixed firing rate.

Spiking neuron model. Spiking neurons are modeled as stochastic leaky integrate-and-fire (LIF) neurons. More precisely, the model employed here is a special case of the spike response model with escape noise, which is a phenomenological noise model that summarizes effects of biophysical channel noise as well as stochastic input on neural spiking (24). This stochasticity of spiking is required, since deterministic neurons in balanced networks with transmission delays lead to erratic network behavior (18), and it allows a direct link to neural sampling and unsupervised learning via expectation–maximization (*SI Appendix*, section B). A neuron j emits spikes with a probability that depends on its membrane potential $u_j(t)$ according to

$$p_{\text{spike}}(u_j(t)) = \text{sig}\left(\frac{u_j(t) - T_j}{\Delta u}\right), \quad [2]$$

where $\text{sig}(x) = [1 + \exp(-x)]^{-1}$ is a sigmoid function. When the membrane potential approaches the firing threshold T_j, the firing probability increases rapidly. To fix the number of spikes for an efficient code, T_j is adapted to control the average firing rate of each neuron (Fig. 2*C*). Furthermore, Δu regulates the stochasticity of spiking. For increasing Δu the spike emission becomes increasingly noisy, whereas for $\Delta u \to 0$ one recovers the standard LIF neuron with sharp threshold. The membrane potential itself is modeled as a linear sum of the feedforward inputs $x_i(t)$ and recurrent inputs $z_k(t)$; i.e.,

$$u_j(t) = \underbrace{\textstyle\sum_i F_{ji} x_i(t)}_{\text{feedforward input}} + \underbrace{\textstyle\sum_k W_{jk} z_k(t)}_{\text{recurrent input}}. \quad [3]$$

Note that, for simplicity, in this model coding neurons are directly coupled by inhibitory connections, but similar dynamics and learning behavior can be implemented in networks with inhibitory interneurons (11).

Learning an efficient spike encoding with recurrent plasticity. Spiking neurons can efficiently distribute neural responses to the input signals over the population, by tightly balancing feedforward and recurrent input at the soma (4, 11) (Fig. 1*B*). In fact, a tight balance of inputs is a direct consequence of learning an efficient encoding via gradient descent on the decoder loss (see *SI Appendix*, section B for derivation). To learn a tight balance recurrent weights adapt according to

$$\Delta W_{jk} \propto -z_k u_j \quad \text{(somatic balance)}. \quad [4]$$

Hence, when neuron k is active and the somatic potential of neuron j is out of balance, i.e., $u_j(t) \neq 0$, the weight W_{jk} changes to balance $u_j(t)$. Note that all neurons have an autapse that learns to balance their own membrane potentials, which can alternatively be interpreted as an approximate membrane potential reset after spiking.

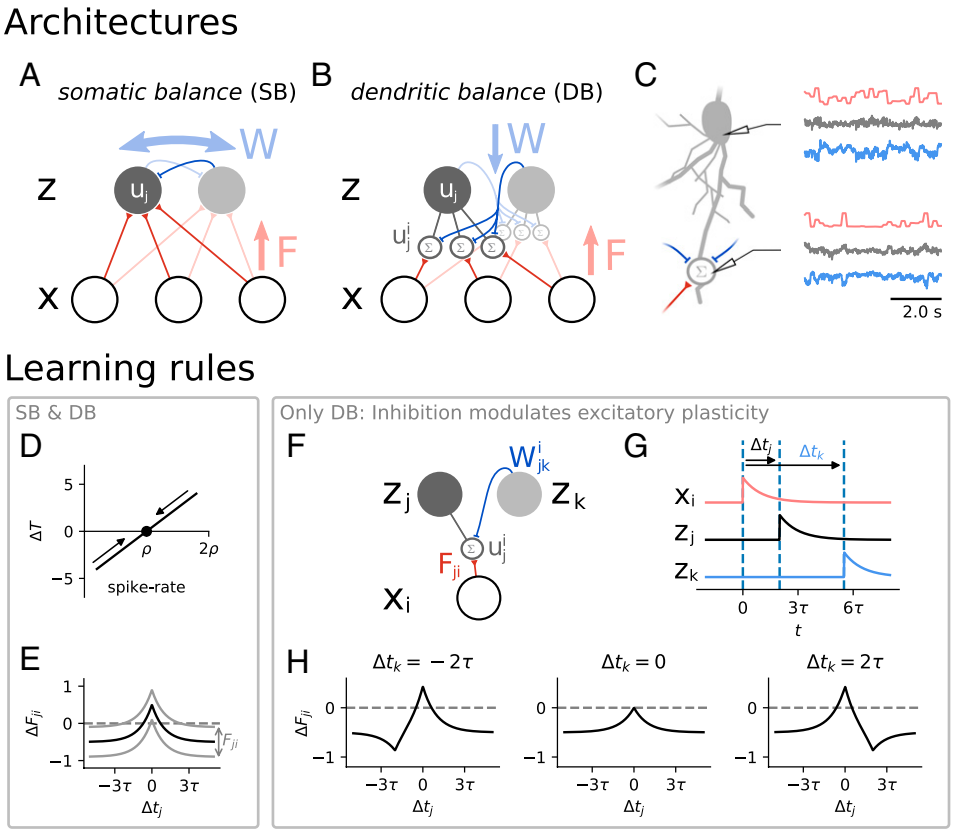

Fig. 2. We compare learning in two network models, a point neuron model with somatic balance, and a model with dendritic balance. (*A*) In the model with SB, neurons (gray circles) with outputs **z** receive feedforward network inputs **x** (white circles) and are coupled via recurrent connections. Recurrent weights W are adapted to balance other inputs to the somatic membrane potential u_j, which ensures an efficient spike encoding. (*B*) In our proposed model with DB, neurons receive inputs at specific dendritic compartments. Recurrent connections learn to balance input currents locally at the dendrites. This leads to dendritic potentials u_j^i that are proportional to the coding error for specific feedforward inputs and therefore can be used to learn feedforward weights. (*C*) After learning, local feedforward (red) and recurrent (blue) currents have adapted to tightly balance each other in individual dendritic compartments (*Bottom*). This dendritic balance also results in a somatic balance of inputs (*Top*). Here we show a neuron from a network with 80 neurons coding for natural images. (*D*) In both models a rapid compensatory mechanism ensures that every neuron fires with rate ρ. If any neuron spikes too rarely, its threshold T_j is lowered; if it spikes too often, T_j is increased. (*E–H*) Illustration of learning rules in terms of experimental STDP rules. For easier interpretability we plot weight changes for spiking inputs x_i, whereas in the remainder of this paper, x_i are analog input signals. (*E*) For learning feedforward weights in the point neuron model (SB) a Hebbian-like STDP rule increases or decreases weights F_{ji} depending on the time difference between pre- and postsynaptic spikes Δt_j and the weight F_{ji} itself. If F_{ji} is high or low, this shifts plasticity toward depression or potentiation, respectively. The same learning rule applies to the DB model, if a neuron does not simultaneously receive any recurrent input. (*F–H*) Illustration of how inhibition modulates feedforward plasticity in the proposed model for a network of two coding neurons z_j (with one dendritic compartment) and z_k and one input neuron x_i. (*F*) The excitatory weight F_{ij} and the inhibitory weight W_{jk}^i attach to the same dendritic potential u_j^i. (*G*) We consider the following example where three spikes are fired: x_i at $t = 0$, z_j at $t = \Delta t_j$, and z_k at $t = \Delta t_k$. (*H*) The total change in the weight F_{ji} depends not only on the spike time difference Δt_j between the input and the postsynaptic neuron, but also on the relative inhibitory spike time Δt_k. In general, if z_j and z_k spike close together, F_{ji} will tend to be depressed. All weight changes were calculated with $F_{ji} = -W_{jk}^i = 0.5$.

This tight balance enables an efficient encoding, since once an input signal is encoded by the spike of a coding neuron, this spike will approximately cancel the excitatory feedforward input to all other neurons and therefore discourage further spiking. More technically, learning a balance with recurrent plasticity leads to recurrent weights that "decode" the population activity onto the membrane potential of each individual neuron $W_{jk} = -\sum_i F_{ji} D_{ik}$ (where D_{ik} is the optimal decoder). The membrane potentials thus reflect the coding error $u_j(t) = \sum_i F_{ji}\,(x_i(t) - \hat{x}_i(t))$, i.e., the global coding goal, and subsequently drive spiking only when the global encoding is not capturing the signal well.

Learning Efficient Representations with Feedforward Plasticity. To enable an efficient encoding of high-dimensional signals, feedforward weights F should be adapted to the statistics of the input signal. To that end, it is possible to derive a plasticity rule for weights F_{ji} that minimizes the decoder loss $\mathcal{L}$ via gradient descent (*SI Appendix*, section B), which yields

$$\Delta F_{ji} \propto z_j(x_i - \hat{x}_i) = z_j(x_i - \textstyle\sum_k D_{ik} z_k). \qquad [5]$$

Intuitively, this rule drives neuron j to correlate its output z_j to input x_i, except if the population is already encoding it. To extract the latter information, the plasticity rule requires a decoding $\hat{x}_i = \sum_k D_{ik} z_k$, which contains information about the neural code for input i of all other neurons in the population.

We thus conclude that an efficient code relies on information about other neurons in two ways: 1) Neurons need to know what is already encoded to avoid redundancy in spiking (dynamics), and 2) plasticity of feedforward connections requires to know what neurons encode about specific inputs to avoid redundancy in representation (learning). While recurrent weights W_{jk} for efficient spiking dynamics 1) can be learned locally (Eq. **4**), learning feedforward synapses F_{ji} correctly 2) is not feasible locally for point neurons, since they lack knowledge about the population code for single inputs x_i.

In the following, we introduce the main result of this paper: Similar to efficient spiking through a tight balance of all

feedforward and recurrent inputs at the soma, local learning of efficient representations can be realized by tightly balancing specific feedforward inputs with recurrent input. Physiologically, we argue that this corresponds to spatially separated inputs at different dendritic compartments, where recurrent connections balance the local membrane potential. We contrast this local implementation of the correct gradient of the decoder loss with a common local approximation of the gradient, which is necessary for point neurons with somatic balance only.

Somatic balance alone requires an approximation for local learning. Since synapses for point neurons have no access to the population code for single inputs, previous approaches used a local approximation to ΔF_{ji} where only pre- and postsynaptic currents are taken into account (Fig. 2*E*):

$$\Delta F_{ji} \propto z_j (x_i - F_{ji} z_j) \quad \text{(Hebbian-like learning)}. \qquad \textbf{[6]}$$

We refer to this learning scheme, consisting of Eqs. **4** and **6**, as somatic balance (SB). A practically identical setup has been proposed in ref. 11. We take this setup as a paradigmatic example of a larger group of Hebbian-like learning rules, which have been used to model representation learning (for a more detailed discussion of related models and learning rules in the literature see *SI Appendix*, section C).

Such Hebbian-like learning rules follow the correct gradient when neurons do not code simultaneously, and thus nonlocal dependencies during learning are not present. This is the case when only a single PSP $z_j(t)$ is nonzero at a time, e.g., in winner-take-all circuits with extremely strong inhibition (15), or when the PSP is extremely short (14). The learning rule becomes also approximately exact when neural PSPs $\mathbf{z}(t)$ in the encoding are uncorrelated (11, 12). However, these are strong demands on the dynamics of the network, which ultimately limit its coding versatility and are likely not met under realistic conditions.

Dendritic balance allows local learning of efficient representations. When neural PSPs $\mathbf{z}(t)$ in the population are correlated, learning efficient representations requires that information about the population code is available at the synapses. To this end, we introduce local dendritic potentials u_j^i at synapses F_{ji} and couple neurons k via dendritic recurrent connections W_{jk}^i to these membrane potentials (Fig. 2*B*). The somatic membrane potential is then realized as the linear sum of the local dendritic potentials

$$\begin{aligned} u_j(t) &= \textstyle\sum_i u_j^i(t) \\ u_j^i(t) &= \underbrace{F_{ji} x_i(t)}_{\text{feedforward input}} + \underbrace{\textstyle\sum_k W_{jk}^i z_k(t)}_{\text{recurrent input}}. \end{aligned} \qquad \textbf{[7]}$$

Note that this amounts only to a refactoring of the equation for the somatic membrane potential and does not change the computational power of the neuron. Given such a network with recurrent weights W_{jk}^i, a SB network with recurrent weights $W_{jk} = \sum_i W_{jk}^i$ has equivalent dynamics. Hence, any improvement in the neural code in this setup is due to an improvement in the learning of feedforward weights. In the discussion, we address how the compartmentalization in Eq. **7** could be realized in biological neurons and how one can reduce the amount of recurrent dendritic connections W_{jk}^i without losing the central benefits of this model.

Introducing dendritic compartments for individual inputs allows us to use the same trick as before: By enforcing a tight E-I balance locally, recurrent connections will try to cancel the input as well as possible. Thereby, recurrent weights W_{jk}^i will automatically learn the best possible decoding of the population activity $\mathbf{z}$ to the input $F_{ji} x_i$. This leads to a local potential that is proportional to the coding error $u_j^i = F_{ji}(x_i - \hat{x}_i)$. In terms of recurrent synaptic plasticity, this is realized by

$$\Delta W_{jk}^i \propto -z_k u_j^i \quad \text{(dendritic balance)}. \qquad \textbf{[8]}$$

Thus, the dendritic membrane potential u_j^i can be used to find the correct gradient ΔF_{ji} from Eq. **5** locally:

$$\Delta F_{ji} \propto \frac{1}{F_{ji}} z_j u_j^i \quad \text{(learning by errors)}. \qquad \textbf{[9]}$$

We refer to this learning scheme as dendritic balance (DB). As can be seen, the learning rules for feedforward and recurrent weights both rely on the local dendritic potential, which they also influence. This enables recurrent inputs to locally modulate feedforward plasticity. However, this also requires the cooperation of feedforward and recurrent weights during learning. We propose three different implementations that ensure this cooperation, by learning recurrent weights on a faster or on the same timescale as feedforward weights (*SI Appendix*, section B.3). We show that these three approaches yield similar results, which equal the analytical solution (Eq. **5**) in performance (*SI Appendix*, Figs. S2 and S3).

It is possible to integrate the learning rules that depend on membrane potentials over time and obtain learning rules that depend on the relative spike timings of multiple neurons. If we consider only one input neuron and one coding neuron, learning with dendritic balance and somatic balance yields the same spike-timing–dependent plasticity rule. This rule is purely symmetric and strengthens the connection when both neurons fire close in time (Fig. 2*E*). However, if the spike of the excitatory input neuron is accompanied by an inhibitory spike in the coding population, the spike-timing–dependent rule breaks symmetry (Fig. 2*H*). This shows how learning with dendritic balance can take more than pairwise interactions into account to enable the neuron to find its place in the population code.

Simulation Experiments. To illustrate the differences that arise between the networks using SB and DB during learning, we set up several coding tasks of increasing complexity. Most centrally, we will show that two aspects of realistic neural dynamics make learning especially difficult: 1) correlated occurrences of the patterns that are represented by coding neurons and 2) transmission delays of recurrent inhibition. Both aspects lead to correlations in the activity of coding neurons $\mathbf{z}$, which, as we demonstrate, have a detrimental effect on the representations learned by Hebbian-like learning.

Learning an efficient encoding with recurrent and feedforward synaptic plasticity. In a first test we performed a comparison on the MNIST dataset of handwritten digits (Fig. 3*C*). We restricted the dataset to the digits 0, 1, and 2, which were encoded by nine coding neurons. Networks were initialized with random feedforward weights and with zero recurrent weights. To demonstrate the effects of recurrent and feedforward plasticity, we separated learning into two stages: First, recurrent plasticity learned to balance feedforward input to the neurons, which leads to a decorrelation of neural responses to the input signals (Fig. 3*B*), and reduced the decoder loss (Fig. 3*A*). Later, feedforward plasticity was turned on, which aligned feedforward weights with reoccurring patterns in the input (Fig. 3*D*). This further reduced the decoder loss and led to better reconstructions (Fig. 3*C*). Since images were rarely encoded by more than one or two neurons (Fig. 3*B*), interactions in the population were small and thus both setups found similar solutions.

Dendritic balance can disentangle complex correlations. Our theoretical results suggest that DB networks should find a better encoding than SB networks when correlations between learned representations are present in the stimuli. To test this, we devised a variation of Földiak's bar task (12), which is a classic independent component separation task. In the original task neurons encode images of independently occurring but overlapping vertical and horizontal bars. Since the number of neurons is equal to the number of possible bars in the images, each neuron should learn to represent a single bar to enable a good encoding. We

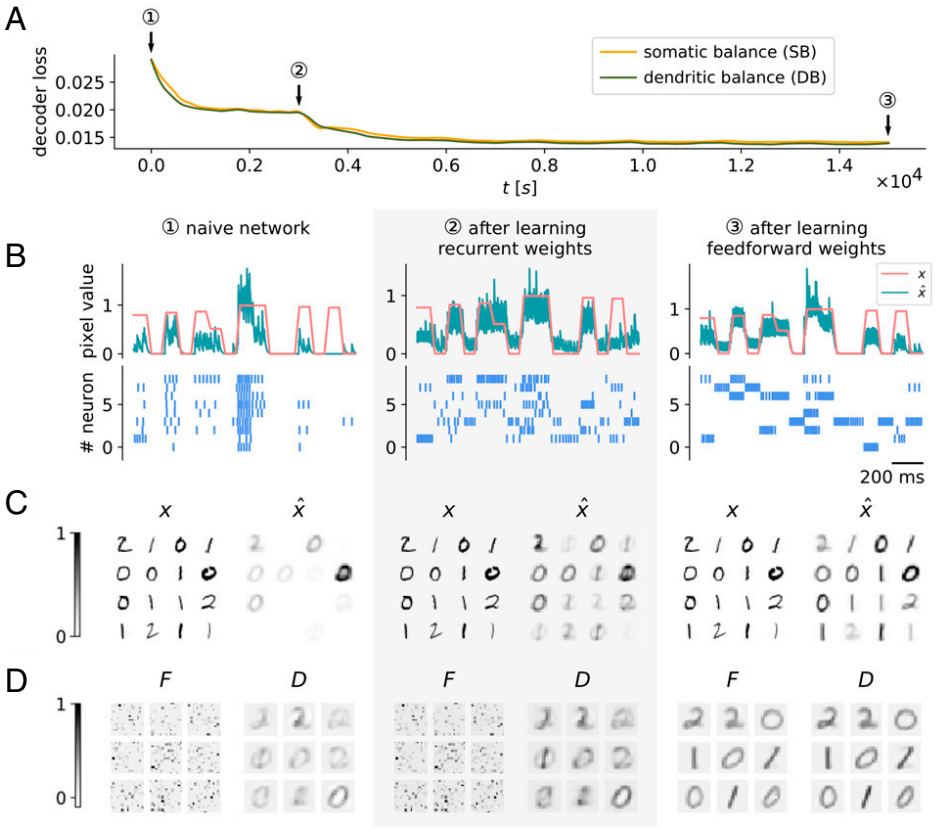

Fig. 3. Learning an efficient encoding with recurrent and feedforward synaptic plasticity. In this simulation experiment, networks consisting of nine coding neurons encoded 16 × 16 images of digits 0, 1, and 2 from the MNIST dataset. (*A*) Decoder loss decreases with neural plasticity for both models using either SB or DB. A naive network with random feedforward and zero recurrent weights shows a large decoder loss (*1*). Learning recurrent connections results in a drop in decoder loss (*2*). Later, feedforward plasticity was turned on, also resulting in an improvement of performance (*3*). Final performances and encodings of SB and DB are very similar. (*B–D*) Results of the DB network for different moments in time during learning. (*B*) Input signal x_i and decoded signal $\hat{x}_i$ for a single pixel i in the center of the image. MNIST digits were presented as constant input signals for 70 ms and faded for 30 ms to avoid discontinuities. After learning, the decoded signal tracks the input reasonably well given the very limited capacity of the network. Below are the spike trains of all neurons in the network in response to the input signal. Learning recurrent weights decorrelates neural responses; learning feedforward weights makes neural responses more specific for certain inputs. (*C*) Sample of input images $\mathbf{x}$ from the MNIST dataset and reconstructions $\hat{\mathbf{x}}$ of the input images. The reconstructions presented here are calculated by averaging the decoded signal during input signal presentation over 70 ms. (*D*) Feedforward weights F and the optimal decoder D. Weights F are first initialized randomly; after learning every neuron becomes specific for a certain prototypical digit. Learning also causes feedforward and decoder weights to align.

kept this basic setup, but additionally we introduced between-bar correlations for selected pairs of bars (Fig. 4*A*). We then could vary the correlation strength p between the bars within the pairs to render them easier or harder to separate.

The simulation results indeed showed that the performance of the SB, but not of the DB model, deteriorates when learned representations are correlated (Fig. 4*B*). The decoder loss for SB grows for increasing p and reaches its maximum at about $p = 0.8$. This is because Hebbian-like learning (as used in SB) correlates a neuron's activity with the appearance of patterns in the input signal, irrespective of the population activity. The correlation between two bars therefore can lead a neuron that initially is coding for only one of the bars to incorporate also the second bar into its receptive field (Fig. 4*B*). Hence, for increasing correlation p neurons start to represent two bars, which does not reflect the true statistics of the input, where single bars may still occur. For $p > 0.8$ the decoder loss decreases, as here the occurrence of the correlated pairs of bars becomes so likely that the representations reflect the statistics of the images again. In contrast, DB enables neurons to communicate which part of the input signal they encode and hence they consistently learn to code for single bars. Accordingly, the decoder loss for DB is smaller than for SB for every correlation strength of bars (Fig. 4*B*).

We expected to see a similar difference between SB and DB networks when complex stimuli are to be encoded. In a third experiment we therefore tested the performance of the networks encoding images of natural scenes (Fig. 4*C*). To also test whether the amount of compression (number of inputs vs. number of coding neurons) would affect SB and DB networks differently, we varied the number of coding neurons while keeping the population rate fixed at 1,000 Hz. This way, only the compression, and not also the total number of spikes, has an effect on the performance of the networks.

The simulations showed that for natural images, DB networks learn more efficient representations than SB networks. The difference in performance becomes larger the higher the compression of the input signal by the network is (Fig. 4*D*). This effect seems to be related to the observations we made in the bar task: Networks with few coding neurons have to learn correlated representations (*SI Appendix*, Fig. S10), which renders SB less appropriate. We found that SB networks consistently needed about twice as many neurons to achieve a similar coding performance to that of DB networks (Fig. 4*D*).

Dendritic balance can cope with inhibitory transmission delays. Correlations between coding neurons can also be introduced by transmission delays of inhibition (18). We therefore expected to find that DB networks are much more robust to long transmission

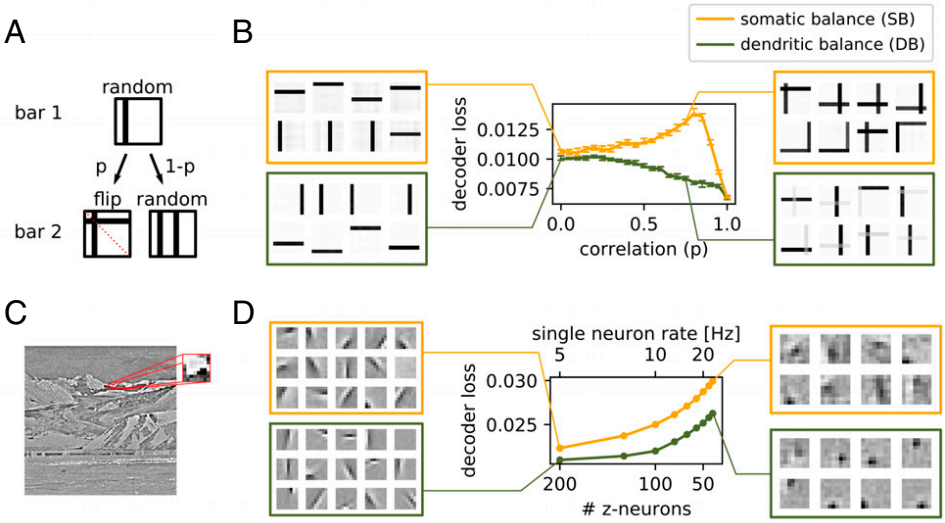

Fig. 4. Dendritic balance improves learning for complex correlations in the input signal. (*A* and *B*) In one simulation experiment, 16 neurons code for 8×8 images containing 2 random of 16 possible bars. Thus, optimally, every neuron codes for a single bar. (*A*) Creation of input images with correlation between reoccurring patterns. Two bars are selected in succession and added to the image. With probability p the bars are symmetric around the top left to bottom right diagonal axis. With probability $1 - p$ the two bars are chosen randomly. (*B*) Decoder loss after learning for different correlation strengths for networks with SB and DB. Displayed is the median decoder loss for 50 different realizations for each datapoint; error bars denote 95% bootstrap confidence intervals. On the sides, 8 of all 16 converged feedforward weights are shown for representative networks. When correlations between bars are present, the representations learned by SB overlap, while DB still learns efficient single-bar representations. (*C* and *D*) Similarly, for complex natural stimuli DB finds better representations when coding neurons are correlated. (*C*) We extracted 16 $\times$ 16-pixel images from a set of whitened pictures of natural scenes (3), scaled them down to 8 $\times$ 8 pixels, and applied a nonlinearity (*SI Appendix*, section D). (*D*) Decoder loss after learning of SB and DB networks featuring varying numbers of coding neurons, while keeping the population rate constant at 1,000 Hz. On the sides we show exemplary converged feedforward weights. For a large number of coding neurons (*Left*) both learning schemes yield similar representations, but performance is slightly better for DB. For a small number of neurons (*Right*) DB learns more refined representations with substantially reduced decoder loss compared to SB. The reason is that for a small number of neurons the learned representations are more correlated and consequently are harder to disentangle. Notably, different amounts of neurons result in different coding strategies.

delays than SB networks. To investigate this, we simulated networks of 200 neurons with a range of timesteps δ, which we interpret as transmission delays. We varied the delay from $\delta = 0.1$ ms to $\delta = 10$ ms and observed how the delay affected coding performance for natural images. Indeed, performance of SB networks drastically broke down to a baseline level when transmission delays became longer than 0.3 ms (Fig. 5*A*). All neurons had learned the same feedforward weights (Fig. 5*B*). In contrast, DB networks continued to perform well even for much longer delays. While long delays for DB also lead to a decrease in coding performance, DB prevented the sudden collapse of the population code.

To illustrate the mechanism that caused the breakdown in performance for SB, we also ran simulations of networks learning to code for MNIST images with longer transmission delays (Fig. 5*C*). After learning with Hebbian-like plasticity, neurons showed highly synchronized activity (Fig. 5*D*) and had learned overly similar feedforward weights (Fig. 5*E*). When transmission delays become long, inhibition will often fail to prevent that multiple neurons with similar feedforward weights spike to encode the same input. Hebbian-like plasticity can exacerbate this effect, since it will adapt feedforward weights of simultaneously spiking neurons in the same direction. In contrast, neurons learning with DB use the information that inhibition provides for learning, even if it arrives too late to prevent simultaneous spiking. Hence DB manages to learn distinct representations also in the face of long transmission delays.

Finally, this difference in the two learning schemes is still present for input signals with fast and complex temporal dynamics. To show this we repeated an experiment from ref. 11, where natural speech sounds were encoded by a population of 100 neurons (Fig. 5 *F*–*H*). In this scenario SB learned only a proper encoding with instantaneous transmission, i.e., when simultaneous spiking was prohibited by removing the least likely spikes in the case of multiple spikes per time bin. However, even for extremely short transmission delays of $\delta = 0.05$ ms, Hebbian-like plasticity led to pathological network behavior (Fig. 5*H*). In contrast, DB learned a similarly efficient encoding in both conditions (Fig. 5*G*).

Discussion

In the past, the formation of neural representations has often been modeled with pairwise Hebbian-like learning rules (11, 12, 14–17, 25–27). However, the learning rules that are derived directly from neural coding models typically require not only information about pre- and postsynaptic activity, but also the coding error of the whole population. Commonly it is maintained that this information is not locally available to the synapse and it is left out of the equation, yielding pairwise Hebbian-like learning rules. Here, we found that omitting this information about the population code can have a detrimental effect on learning when neural activity is correlated, which is the case in realistic conditions. In this case, Hebbian-like learning leads to a highly inefficient encoding in comparison to the derived learning by errors or even in comparison to random connections (Figs. 4 and 5). To overcome this problem, we showed how learning by errors can be implemented locally by neurons with dendritic balance and a voltage-dependent plasticity rule. This suggests that dendritic balance could play a crucial role in synaptic plasticity for the formation of efficient representations.

Why does Hebbian-like learning fail when neural activity is correlated, and how does learning by errors prevent this? When the activity of neurons is correlated, Hebbian-like learning adapts the feedforward weights of these neurons into a similar direction. This even further strengthens the correlations between neurons—a vicious cycle, which ultimately

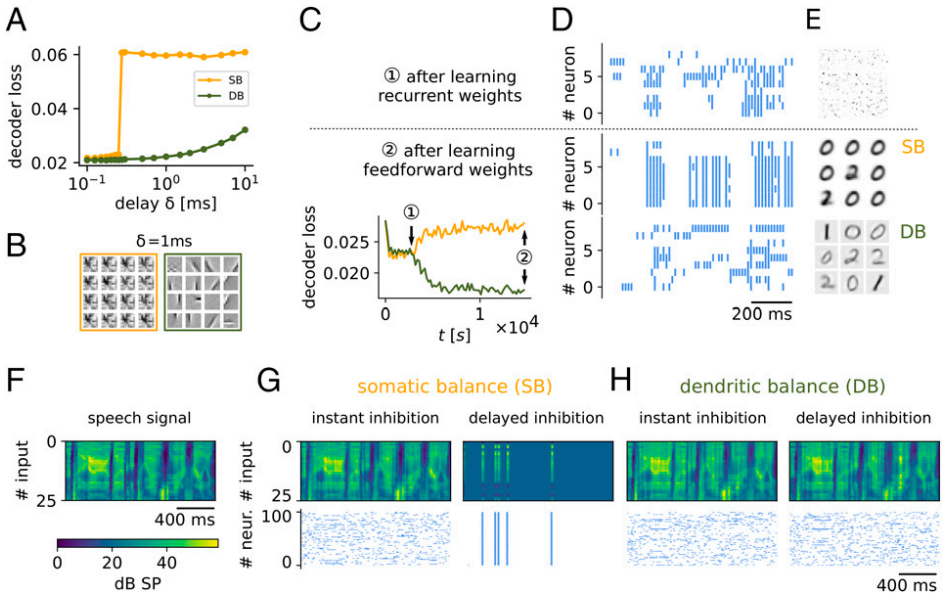

Fig. 5. Dendritic balance prevents learning of redundant representations for inhibitory transmission delays. (*A*) Decoder loss of networks of 200 neurons coding for natural scenes for different inhibitory transmission delays δ. For transmission delays longer than 0.3 ms, Hebbian-like learning in SB networks leads to highly inefficient representations and large decoder loss. In contrast, for networks learning with DB, the decoder loss increases only moderately even for long transmission delays. The results are robust with respect to the stochasticity of firing Δu and the firing rate ρ (*SI Appendix*, Fig. S8). (*B*) Selection of learned weights for a transmission delay of 1 ms. DB learns similar weights as before (Fig. 4*D*), while SB leads to a collapse of representations. (*C*–*E*) To illustrate the effect of feedforward plasticity, we repeated the MNIST experiment in Fig. 3 with long transmission delays of 3 ms (before, 0.1 ms in Fig. 3). (*C*) First, only recurrent connections were learned (*1*); later, feedforward weights were learned (*2*). As before, recurrent plasticity decorrelates responses and decreases the decoder loss. When feedforward plasticity was turned on, Hebbian-like plasticity (SB) learned worse representations than random feedforward weights, which is indicated by the increase in decoder loss. In contrast, our model with DB learned improved representations with substantially reduced decoder loss. (*D*) The poor performance of the SB model is a consequence of highly synchronous spiking responses to the inputs, whereas neurons fire asynchronously in the model with DB. (*E*) Neurons in the SB model learn overly similar feedforward weights, whereas neurons with dendritic balance learn feedforward weights that capture the input space well. (*F*–*H*) This effect is still present when input signals show fast changes in time. Here, 100 coding neurons firing at 5 Hz encode a speech signal. (*F*) Spectrogram of the signal presented in 25 frequency channels. (*G*) As can be seen in the reconstructed signal (*Top*), SB finds a good encoding for instant inhibition (loss = 0.06), but even for extremely small delays of 0.05 ms the learned representations collapse, leading to pathological network behavior and bad encoding performance (loss = 0.23). (*H*) In contrast, DB finds a similar encoding for both instant inhibition (loss = 0.057) and inhibitory delays of 0.05 ms (loss = 0.06).

can lead to highly redundant representations and extremely correlated spiking. Strong correlations between coding neurons typically mean that certain inputs are overrepresented and others underrepresented in the population, which is indicated by negative or positive coding errors, respectively. In contrast to Hebbian-like learning, learning by errors selectively weakens connections to overrepresented inputs and thereby helps to reduce the correlations between coding neurons. In our model, correlations between coding neurons can arise through either correlations of the learned representations in the input signal or transmission delays of recurrent inhibition. Correlated firing due to correlations in the input can in principle always be addressed by increasing the number of coding neurons, as this will increase the independence of the learned representations (Fig. 4*D* and *SI Appendix*, Fig. S10). Correlations due to transmission delays of recurrent inhibition, on the other hand, are a fundamental problem that arises in balanced networks (18, 23, 28). Here, the exact point of breakdown of Hebbian learning depends on the specific type of input and network size and might occur for longer transmission delays in simplistic scenarios. However, already in the case of moderately large networks receiving complex input signals the effect is severe—even for submillisecond delays Hebbian-like learning can lead to a collapse of neural representations and almost perfectly correlated spiking of the whole population (Fig. 5). In contrast, learning by errors consistently avoids this breakdown, and we therefore argue that it becomes indispensable when transmission delays are present.

To make coding errors available for single synapses locally, we introduced balanced dendritic potentials that are proportional to these errors. This can be achieved by learning a balance through recurrent plasticity on the dendrites, as then the network automatically finds an optimal decoding of neural activity to the feedforward inputs. Yet, presenting an error through a balance of inputs is a quite general principle, and theoretically it would also be possible to present the coding error elsewhere. Rate-based models of predictive and sparse coding for example suggest that coding errors are presented in the activity of other neural populations (29–32). However, this idea cannot be easily transferred to spiking neurons, where coding errors would be rectified by neural spiking mechanisms; hence, it is not directly possible to present negative and positive errors in the same unit. Neural learning in these theories, however, relies on this, and indeed, still no conclusive experimental evidence for such error units exists (33, 34). Another theory therefore suggests that prediction errors are presented by voltage differences between soma and dendrite in two-compartment neurons (35–37). In contrast, our work shows that a coding error, which is calculated from the mismatch between excitation and inhibition locally in each dendritic compartment, can act as a very precise learning cue for single synapses. What supports this idea is that a local dendritic balance of inputs, which is maintained by plasticity, has indeed been observed experimentally (8, 38–40). Furthermore, this balance on single neurons can also explain central characteristics of cortical dynamics (4), such as highly irregular spiking (41, 42), but correlated membrane potentials of similarly tuned neurons (43, 44).

An apparent downside of implementing dendritic balance is the large increase in the number of recurrent inhibitory

connections. Connecting every neuron to each feedforward synapse on the dendrites of other neurons would even for moderately sized networks prove extremely costly. However, we found that only a small fraction of the inhibitory connections in our model are required for learning, namely strong connections between neurons whose firing is correlated. We demonstrated this in the example of the bars task, where 90% of dendritic connections can be pruned without changing the learning outcome (*SI Appendix*, Fig. S9). Moreover, in our model inhibition is mediated by direct recurrent connections between coding neurons, but fewer connections would be required if inhibition was mediated via interneurons. By incorporating inhibitory interneurons with broad feature selectivity, it is possible to merge inhibitory connections that provide largely the same information (11). We therefore expect that the main benefits of the proposed learning scheme can be achieved also with relatively few connections.

Biological Plausibility. Our model presents the simplest extension of existing point-neuron models (11), which allows us to formally derive and isolate the effect that dendritic balance can have on representation learning. While more complex models of dendritic structure and nonlinear dynamics can elucidate their role for neural computation (45), nonlinear dynamics would also alter the computational capacity (45), thus hampering a direct comparison to previous models of learning. Nevertheless, the question remains whether the proposed learning based on dendritic balance can be implemented by biological neurons. In the following we discuss the main requirements of the proposed learning scheme.

A central element of the dendritic balance model is the dependence of synaptic plasticity on local membrane potentials. Indeed, it has been argued that the local membrane potential is a critical factor determining synaptic plasticity (46–49). Such voltage-dependent plasticity is thought to be mediated mainly by the local calcium concentration, which closely follows the local membrane potential (50, 51) and locally modulates neural plasticity (52). As required by our model, this voltage dependence implies that inhibition can have a large impact on excitatory synaptic plasticity locally (53, 54), which also has been found experimentally (8, 55). Yet, it remains a major open question what the precise functional role of these voltage-dependent plasticity mechanisms could be (8, 45). Our work proposes that a central feature of voltage-dependent synaptic plasticity is to base the plasticity of single synapses not only on pre- and postsynaptic activity, but also on the activity of other neurons in the population.

How are the proposed learning rules related to experimentally observed voltage-dependent plasticity? Many experiments show that excitatory plasticity requires a strong depolarization of the membrane potential, which for example happens during postsynaptic spiking (46). Our feedforward plasticity rule can be reconciled with plasticity rules that are inspired by these experiments (56, 57) (see Fig. 6*B* for details). The voltage dependence of inhibitory (recurrent) plasticity has only recently started being investigated (8). Recent experimental evidence suggests that this inhibitory plasticity, like excitatory plasticity, is calcium dependent and also requires postsynaptic spiking (58, 59). Our recurrent plasticity rule is similar to previous models of voltage-dependent inhibitory plasticity (11, 60), which set a target value for the postsynaptic membrane potential. Like our rule, these rules have not considered the requirement of postsynaptic spiking for plasticity induction explicitly. We speculate that such a requirement enables the network to preferentially select connections between neurons with correlated activity, which are especially relevant for learning (*SI Appendix*, section B.3). Further experimental and theoretical research is required to understand the precise mechanism and purpose of this type of inhibitory plasticity.

Another requirement of the learning scheme is that different compartments on the dendritic tree are well isolated, so that recurrent inputs can modulate the plasticity of specific synapses (Fig. 6*A*). In biological neurons, dendrites are electronically distributed elements, where strong voltage gradients may exist across the dendritic tree (61, 62). These voltage differences are the result of strong attenuation of input currents, meaning that individual synapses can have very localized effects (63). Thus, the required isolation between compartments exists in biological neurons if they are sufficiently separated and especially for compartments on different dendritic branches (38). This isolation between spatially separated compartments also reduces nonlinear interactions between them. As a result, the integration of any net excitation from different compartments at the soma is approximately linear (63), as required by our model. In contrast to excitation, though, inhibition on distant dendrites mainly acts locally by gating excitation, so that dendritic inhibition can have a very weak effect on the somatic membrane potential (64). Propagating dendritic inhibition to the soma is, however, not required for network function, because any remaining net excitation can also be balanced by plastic inhibitory synapses close to the soma. Therefore, the model's key requirements for learning and network function could also be met in biological neurons.

However, how synapses in biological neurons are organized on these dendritic compartments seems to be at odds with our model: First, while in our model individual feedforward inputs (which are mostly excitatory) have isolated dendritic potentials, it is well known that correlated excitatory synapses often cluster on dendrites (65–67); second, while in our model we generally find more inhibitory than excitatory synapses, excitatory synapses outnumber inhibitory synapses on dendritic branches, e.g., 4:1 on the dendrites of cultured rat hippocampal neurons (38). We argue that these two disparities can be resolved, if the individual continuous inputs provided to our model are seen as the resulting currents of clustered, correlated synapses. How this clustering could be organized by synaptic plasticity is a matter of ongoing research (68), and it will have to be the subject of future work to reconcile these plasticity mechanisms with representation learning.

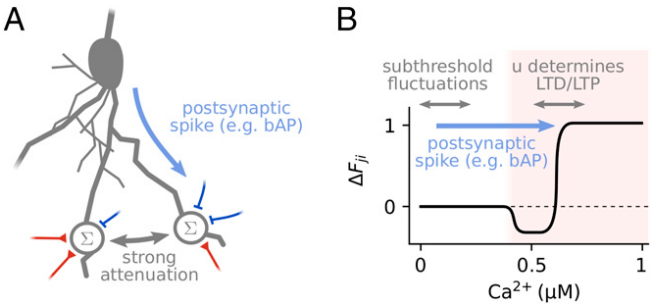

Fig. 6. Biologically feasible implementation of the proposed feedforward learning rule. (*A*) The proposed learning scheme requires the following distribution of information in the dendritic tree: First, synapses need to know when a postsynaptic spike occurred. This information could be provided, e.g., by backpropagating action potentials (bAPs). Second, the potentials of the dendritic compartments that sum specific excitatory and inhibitory inputs have to be sufficiently decoupled. Such a strong attenuation of inputs exists for example between dendritic branches (38). (*B*) The inputs u_j^i to the local potential and the postsynaptic spike signal z_j can be used by a voltage-dependent plasticity rule to implement the proposed learning scheme. Typically such rules assume that plasticity happens in a strongly depolarized regime that is associated with large calcium concentrations (shaded red area, compare to ref. 57). To reconcile our model with such rules, we assume the postsynaptic spike z_j shifts the local potential into the strongly depolarized regime, e.g., through bAPs or dendritic plateau potentials (72), and local input u_j^i determines whether long-term depression (LTD) or long-term potentiation (LTP) occurs.

Experimental Predictions. Ultimately, we can generate two directly measurable experimental predictions from our model: First, if input currents to a neuron's dendrites are locally unbalanced, recurrent plasticity will learn to establish a local E-I balance. Second, our model predicts that the strength of local inhibition determines the sign of synaptic plasticity: During plasticity induction at excitatory feedforward synapses, activating inhibitory neurons that target the same dendritic loci should lead to long-term depression of the excitatory synapses. We would expect this effect to persist, even if the inhibitory signal arrives shortly after the pre- and postsynaptic spiking. These predictions mainly apply to populations of sensory coding neurons, but models similar to the somatic balance model have been proposed to solve other tasks as well (69, 70), suggesting that dendritic balance could be of more general relevance for learning. Indeed, indications of inhibitory modulated plasticity can be found not only in visual cortex (55), but also in hippocampus (38) and possibly other areas (8, 71).

To conclude, we here presented a learning scheme that facilitates highly cooperative population codes for complex stimuli in neural populations. Our results question pairwise Hebbian learning as a paradigm for representation learning and suggest that there exists a direct connection between dendritic balance and synaptic plasticity.

Materials and Methods

Neural activity was simulated in discrete timesteps of length δ. Images were presented as continuous inputs for 100 ms each, that is, as constant inputs for 70 ms, after which they were linearly interpolated over 30 ms to the next image to avoid discontinuities in the input signal. In the speech task, audio signals were encoded in 25 frequency channels, sampled at 200 Hz, and presented with linear interpolation between datapoints. For every experiment a learning set and a test set were created. The networks learned online on the training set; in regular intervals the learning rules were turned off and the performance was evaluated on the test set. Performance was measured via the instantaneous decoder loss (Eq. **1**) by learning the decoder D alongside the network. The respective update rule for the decoder is given by

$$\Delta D_{ij} \propto z_j(x_i - \textstyle\sum_k D_{ik} z_k). \qquad [10]$$

For DB networks we propose three learning schemes with fast or slow recurrent plasticity (detailed in *SI Appendix*, section B.3). To reduce computation time for large networks, the analytical solution of optimal recurrent weights $W^i_{jk} = -F_{ij}F_{ik}$ was used as an approximation of the proposed learning schemes. For Figs. 3 and 5 *C–E* the dendritic balance learning scheme with fast recurrent plasticity and the weight decay trick (DB decay in *SI Appendix*) is displayed. For Fig. 4, as well as Fig. 5 *F–H*, we used the analytical solution. When comparing the proposed learning schemes to the analytical solution on reference simulations (*SI Appendix*, Figs. S2 and S3), they consistently found very similar network parameters and reached the same performance.

In early simulations we observed that coding performance is largely affected by the population rate, i.e., how many spikes can be used to encode the input signal. To avoid this effect when comparing the two learning schemes, we additionally introduced a rapid compensatory mechanism to fix the firing rates, which is realized by changing the thresholds T_j. We emphasize again that this adaptation is in principle not necessary to ensure stable network function. In fact, error-correcting balanced state inhibition can already be sufficient for a network to develop into a slow firing regime (11). The fixed firing rate is enforced by adapting the threshold T_j according to

$$\Delta T_j \propto (s_j - \rho\,\delta),$$

such that neurons are firing with a target firing rate ρ. Here, $\rho\,\delta$ is the mean number of spikes in a time window of size δ if a neuron would spike with rate ρ, and s_j is a spike indicator that is 1 if neuron j spiked in the last time δ; otherwise $s_j = 0$.

Furthermore, in the simulations of correlated bars and natural scenes (Fig. 4), we aided the learning process by starting with a high stochasticity in spiking and slowly decreasing it toward the desired stochasticity. While similar results were obtained without using this method, we observed that convergence of the networks to an efficient solution was more reliable with it, as it helped in avoiding local minima of the goal function in early phases of learning. Specifically, we started with a stochasticity of $\Delta u = 1.0$. We then exponentially annealed it toward the final value Δu^* by applying every timestep

$$\Delta u(t+1) = \Delta u(t) - \eta_{\Delta u}(\Delta u(t) - \Delta u^*).$$

Data Availability. Full derivations of the network dynamics and learning rules, more details about the relation of our model to previous models in the literature, and supplementary figures containing additional information for simulation experiments, as well as simulation parameters, are provided in *SI Appendix*. Code for reproducing the main simulations is available in GitHub at https://github.com/Priesemann-Group/dendritic_balance (73). Computer programs data have been deposited in Zenodo at https://zenodo.org/record/4133446.

ACKNOWLEDGMENTS. We thank Friedemann Zenke, Christian Machens, and Sebastian Bruijns for their comments on the manuscript, as well as the Priesemann group, especially Matthias Loidolt and Daniel González Marx, for valuable comments and for reviewing the manuscript. F.A.M., L.R., and V.P. received support from the Max-Planck-Society. L.R. acknowledges funding by SMARTSTART, the joint training program in computational neuroscience by the VolkswagenStiftung and the Bernstein Network.

1. H. B. Barlow, "Possible principles underlying the transformations of sensory messages" in *Sensory Communication*, W. A. Rosenblith, Ed. (The MIT Press, 2012), pp. 216–234.
2. J. J. Atick, A. N. Redlich, Towards a theory of early visual processing. *Neural Comput.* **2**, 308–320 (1990).
3. B. A. Olshausen, D. J. Field, Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature* **381**, 607–609 (1996).
4. S. Denève, C. K. Machens, Efficient codes and balanced networks. *Nat. Neurosci.* **19**, 375–382 (2016).
5. C. van Vreeswijk, H. Sompolinsky, Chaotic balanced state in a model of cortical circuits. *Neural Comput.* **10**, 1321–1371 (1998).
6. N. Brunel, Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *J. Comput. Neurosci.* **8**, 183–208 (2000).
7. B. Sengupta, S. B. Laughlin, J. E. Niven, Balanced excitatory and inhibitory synaptic currents promote efficient coding and metabolic efficiency. *PLoS Comput. Biol.* **9**, e1003263 (2013).
8. G. Hennequin, E. J. Agnes, T. P. Vogels, Inhibitory plasticity: Balance, control, and codependence. *Annu. Rev. Neurosci.* **40**, 557–579 (2017).
9. B. Haider, A. Duque, A. R. Hasenstaub, D. A. McCormick, Neocortical network activity in vivo is generated through a dynamic balance of excitation and inhibition. *J. Neurosci.* **26**, 4535–4545 (2006).
10. T. P. Vogels, H. Sprekeler, F. Zenke, C. Clopath, W. Gerstner, Inhibitory plasticity balances excitation and inhibition in sensory pathways and memory networks. *Science* **334**, 1569–1573 (2011).
11. W. Brendel, R. Bourdoukan, P. Vertechi, C. K. Machens, S. Denève, Learning to represent signals spike by spike. *PLoS Comput. Biol.* **16**, e1007692 (2020).
12. P. Földiák, Forming sparse representations by local anti-Hebbian learning. *Biol. Cybern.* **64**, 165–170 (1990).
13. R. Linsker, Local synaptic learning rules suffice to maximize mutual information in a linear network. *Neural Comput.* **4**, 691–702 (1992).
14. B. Nessler, M. Pfeiffer, L. Buesing, W. Maass, Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity. *PLoS Comput. Biol.* **9**, e1003037 (2013).
15. J. Bill *et al.*, Distributed Bayesian computation and self-organized learning in sheets of spiking neurons with local lateral inhibition. *PLoS One* **10**, e0134356 (2015).
16. B. Nessler, M. Pfeiffer, W. Maass, "STDP enables spiking neurons to detect hidden causes of their inputs" in *Advances in Neural Information Processing Systems* (2009), pp. 1357–1365.
17. D. Kappel, B. Nessler, W. Maass, STDP installs in Winner-Take-All circuits an online approximation to hidden Markov model learning. *PLoS Comput. Biol.* **10**, e1003511 (2014).
18. C. E. Rullán Buxó, J. W. Pillow, Poisson balanced spiking networks. *PLoS Comput. Biol.* **16**, e1008261 (2020).
19. A. Torrado Pacheco *et al.*, Rapid and active stabilization of visual cortical firing rates across light-dark transitions. *Proc. Natl. Acad. Sci. U.S.A.* **116**, 18068–18077 (2019).
20. A. Kohn, M. A. Smith, Stimulus dependence of neuronal correlation in primary visual cortex of the macaque. *J. Neurosci.* **25**, 3661–3673 (2005).
21. D. P. Schulz, M. Sahani, M. Carandini, Five key factors determining pairwise correlations in visual cortex. *J. Neurophysiol.* **114**, 1022–1033 (2015).
22. W. A. Freiwald, A. K. Kreiter, W. Singer, Synchronization and assembly formation in the visual cortex. *Prog. Brain Res.* **130**, 111–140 (2001).
23. M. Boerlin, C. K. Machens, S. Denève, Predictive coding of dynamical variables in balanced spiking networks. *PLoS Comput. Biol.* **9**, e1003258 (2013).
24. W. Gerstner, W. M. Kistler, R. Naud, L. Paninski, *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition* (Cambridge University Press, 2014).
25. S. Y. Kung, K. I. Diamantaras, "A neural network learning algorithm for adaptive principal component extraction (apex)" in *International Conference on Acoustics, Speech, and Signal Processing* (IEEE, New York, NY, 1990), pp. 861–864.
26. A. Tavanaei, T. Masquelier, A. Maida, Representation learning using event-based STDP. *Neural Netw.* **105**, 294–303 (2018).

NEUROSCIENCE

27. Y. Bahroun, A. Soltoggio, "Online representation learning with single and multi-layer Hebbian networks for image classification" in *International Conference on Artificial Neural Networks*, A. Lintas, S. Rovetta, P. F. M. J. Verschure, A. E. P. Villa, Eds. (Springer, New York, NY, 2017), pp. 354–363.
28. M. Chalk, B. Gutkin, S. Denève, Neural oscillations as a signature of efficient coding in the presence of synaptic delays. *eLife* **5**, e13824 (2016).
29. R. P. Rao, D. H. Ballard, Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nat. Neurosci.* **2**, 79–87 (1999).
30. R. Bogacz, A tutorial on the free-energy framework for modelling perception and learning. *J. Math. Psychol.* **76** (Pt B), 198–211 (2017).
31. K. Friston, A theory of cortical responses. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **360**, 815–836 (2005).
32. B. A. Olshausen, D. J. Field, Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Res.* **37**, 3311–3325 (1997).
33. M. Heilbron, M. Chait, Great expectations: Is there evidence for predictive coding in auditory cortex? *Neuroscience* **389**, 54–73 (2018).
34. K. S. Walsh, D. P. McGovern, A. Clark, R. G. O'Connell, Evaluating the neurophysiological evidence for predictive processing as a model of perception. *Ann. N. Y. Acad. Sci.* **1464**, 242–268 (2020).
35. J. Brea, A. T. Gaál, R. Urbanczik, W. Senn, Prospective coding by spiking neurons. *PLoS Comput. Biol.* **12**, e1005003 (2016).
36. J. Guerguiev, T. P. Lillicrap, B. A. Richards, Towards deep learning with segregated dendrites. *eLife* **6**, e22901 (2017).
37. J. Sacramento, R. P. Costa, Y. Bengio, W. Senn, Dendritic cortical microcircuits approximate the backpropagation algorithm. arXiv [Preprint] (2018). https://arxiv.org/abs/1810.11393 (Accessed 29 November 2021).
38. G. Liu, Local structural balance and functional interaction of excitatory and inhibitory synapses in hippocampal dendrites. *Nat. Neurosci.* **7**, 373–379 (2004).
39. D. M. Iascone *et al.*, Whole-neuron synaptic mapping reveals spatially precise excitatory/inhibitory balance limiting dendritic and somatic spiking. *Neuron* **106**, 566–578.e8 (2020).
40. J. N. Bourne, K. M. Harris, Coordination of size and number of excitatory and inhibitory synapses results in a balanced structural plasticity along mature hippocampal CA1 dendrites during LTP. *Hippocampus* **21**, 354–373 (2011).
41. D. J. Tolhurst, J. A. Movshon, A. F. Dean, The statistical reliability of signals in single neurons in cat and monkey visual cortex. *Vision Res.* **23**, 775–785 (1983).
42. A. Wohrer, M. D. Humphries, C. K. Machens, Population-wide distributions of neural activity during perceptual decision-making. *Prog. Neurobiol.* **103**, 156–193 (2013).
43. J. F. Poulet, C. C. Petersen, Internal brain state regulates membrane potential synchrony in barrel cortex of behaving mice. *Nature* **454**, 881–885 (2008).
44. J. Yu, D. Ferster, Membrane potential synchrony in primary visual cortex during sensory stimulation. *Neuron* **68**, 1187–1201 (2010).
45. J. Bono, K. A. Wilmes, C. Clopath, Modelling plasticity in dendrites : From single cells to networks. *Curr. Opin. Neurobiol.* **46**, 136–141 (2017).
46. J. Lisman, N. Spruston, Postsynaptic depolarization requirements for LTP and LTD: A critique of spike timing-dependent plasticity. *Nat. Neurosci.* **8**, 839–841 (2005).
47. J. Lisman, N. Spruston, Questions about STDP as a general model of synaptic plasticity. *Front. Synaptic Neurosci.* **2**, 140 (2010).
48. A. Artola, S. Bröcher, W. Singer, Different voltage-dependent thresholds for inducing long-term depression and long-term potentiation in slices of rat visual cortex. *Nature* **347**, 69–72 (1990).
49. A. Ngezahayo, M. Schachner, A. Artola, Synaptic activity modulates the induction of bidirectional synaptic changes in adult mouse hippocampus. *J. Neurosci.* **20**, 2451–2458 (2000).
50. R. W. Tsien, R. Y. Tsien, Calcium channels, stores, and oscillations. *Annu. Rev. Cell Biol.* **6**, 715–760 (1990).
51. Y. Kanemoto *et al.*, Spatial distributions of GABA receptors and local inhibition of Ca2+ transients studied with GABA uncaging in the dendrites of CA1 pyramidal neurons. *PLoS One* **6**, e22652 (2011).
52. G. J. Augustine, F. Santamaria, K. Tanaka, Local calcium signaling in neurons. *Neuron* **40**, 331–346 (2003).
53. L. Bar-Ilan, A. Gidon, I. Segev, The role of dendritic inhibition in shaping the plasticity of excitatory synapses. *Front. Neural Circuits* **6**, 118 (2013).
54. A. Saudargienė, B. P. Graham, Inhibitory control of site-specific synaptic plasticity in a model CA1 pyramidal neuron. *Biosystems* **130**, 37–50 (2015).
55. L. Wang, A. Maffei, Inhibitory plasticity dictates the sign of plasticity at excitatory synapses. *J. Neurosci.* **34**, 1083–1093 (2014).
56. C. Clopath, W. Gerstner, Voltage and spike timing interact in STDP—A unified model. *Front. Synaptic Neurosci.* **2**, 25 (2010).
57. H. Z. Shouval, M. F. Bear, L. N. Cooper, A unified model of NMDA receptor-dependent bidirectional synaptic plasticity. *Proc. Natl. Acad. Sci. U.S.A.* **99**, 10831–10836 (2002).
58. M. Udakis, V. Pedrosa, S. E. L. Chamberlain, C. Clopath, J. R. Mellor, Interneuron-specific plasticity at parvalbumin and somatostatin inhibitory synapses onto CA1 pyramidal neurons shapes hippocampal output. *Nat. Commun.* **11**, 4395 (2020).
59. J. A. D'amour, R. C. Froemke, Inhibitory and excitatory spike-timing-dependent plasticity in the auditory cortex. *Neuron* **86**, 514–528 (2015).
60. V. Pedrosa, C. Clopath, Voltage-based inhibitory synaptic plasticity: Network regulation, diversity, and flexibility. bioRxiv [Preprint] (2020). https://doi.org/10.1101/2020.12.08.416263 (Accessed 17 November 2021).
61. I. Segev, "Dendritic processing" in *The Handbook of Brain Theory and Neural Networks*, M. A. Arbib, Ed. (MIT Press, 1995), pp. 324–332.
62. I. Segev, M. London, Untangling dendrites with quantitative models. *Science* **290**, 744–750 (2000).
63. W. Rall, "Theoretical significance of dendritic trees for neuronal input-output relations" in *The Theoretical Foundation of Dendrite Function*, I. Segev, J. Rinzel, G. M. Shephard, Eds. (MIT Press, Cambridge, MA, 1995), pp. 122–146.
64. J. J. B. Jack, D. Noble, R. W. Tsien, "Interaction between excitation and postsynaptic inhibition" in *Electric Current Flow in Excitable Cells* (Clarendon Press, Oxford, UK, 1983), pp. 197–213.
65. G. Kastellakis, D. J. Cai, S. C. Mednick, A. J. Silva, P. Poirazi, Synaptic clustering within dendrites: An emerging theory of memory formation. *Prog. Neurobiol.* **126**, 19–35 (2015).
66. J. L. Chen *et al.*, Clustered dynamics of inhibitory synapses and dendritic spines in the adult neocortex. *Neuron* **74**, 361–373 (2012).
67. T. Kleindienst, J. Winnubst, C. Roth-Alpermann, T. Bonhoeffer, C. Lohmann, Activity-dependent clustering of functional synaptic inputs on developing hippocampal dendrites. *Neuron* **72**, 1012–1024 (2011).
68. J. H. Kirchner, J. Gjorgjieva, Emergence of local and global synaptic organization on cortical dendrites. *Nat. Commun.* **12**, 4005 (2021).
69. K. Soman, S. Chakravarthy, M. M. Yartsev, A hierarchical anti-Hebbian network model for the formation of spatial cells in three-dimensional space. *Nat. Commun.* **9**, 4046 (2018).
70. C. Pehlevan, S. Mohan, D. B. Chklovskii, Blind nonnegative source separation using biological neural networks. *Neural Comput.* **29**, 2925–2954 (2017).
71. G. Koch, V. Ponzo, F. Di Lorenzo, C. Caltagirone, D. Veniero, Hebbian and anti-Hebbian spike-timing-dependent plasticity of human cortico-cortical connections. *J. Neurosci.* **33**, 9725–9733 (2013).
72. B. A. Milojkovic, M. S. Radojicic, S. D. Antic, A strict correlation between dendritic and somatic plateau depolarizations in the rat prefrontal cortex pyramidal neurons. *J. Neurosci.* **25**, 3940–3951 (2005).
73. F. A. Mikulasch, Code for "Local dendritic balance enables learning of efficient representations in networks of spiking neurons." GitHub. https://github.com/Priesemann-Group/dendritic_balance. Deposited 3 November 2021.

3 Where is the error? Hierarchical predictive coding through dendritic error computation

Published at Trends in Neurosciences 46.1 (2023): 45-59.
DOI 10.1016/j.tins.2022.09.007
Supplementary Material
Source Code

Contributions Conceptualization, Investigation, Writing - Original Draft

This work also constitutes a chapter in LR's PhD thesis. I proposed the initial idea to connect our model in chapter 2 to hierarchical predictive coding. Together with LR we then developed the details of the proposed model, created the figures and wrote the manuscript.

Trends in
Neurosciences

CellPress
OPEN ACCESS

Review

Where is the error? Hierarchical predictive coding through dendritic error computation

Fabian A. Mikulasch,[1,5,*] Lucas Rudelt,[1,5] Michael Wibral,[2] and Viola Priesemann[1,3,4]

Top-down feedback in cortex is critical for guiding sensory processing, which has prominently been formalized in the theory of hierarchical predictive coding (hPC). However, experimental evidence for error units, which are central to the theory, is inconclusive and it remains unclear how hPC can be implemented with spiking neurons. To address this, we connect hPC to existing work on efficient coding in balanced networks with lateral inhibition and predictive computation at apical dendrites. Together, this work points to an efficient implementation of hPC with spiking neurons, where prediction errors are computed not in separate units, but locally in dendritic compartments. We then discuss the correspondence of this model to experimentally observed connectivity patterns, plasticity, and dynamics in cortex.

Highlights

Hierarchical predictive coding has been considered one of the most promising unifying theories of cortical computation. Yet, in its classical form, it remains difficult to connect to single neuron physiology.

We review work that shows that hierarchical predictive coding can be implemented by neurons with dendritic compartments, where prediction errors are computed by the local voltage dynamics in the dendrites.

This connects the theories of predictive coding and efficient coding in balanced networks and provides a solution to the open problem of implementing predictive coding with spiking neurons.

This also links predictive coding to cortical physiology and voltage-dependent plasticity, which offers new ways to test for predictive coding in cortex.

Neural models of inference in cortex

A central feature of perception is that our internal expectations to a large degree shape how we perceive the world [1]. A long line of research aims to describe these expectation-guided computations in our brain by Bayesian **inference** (see Glossary) (i.e., statistically optimal perception) and, subsequently, could show that Bayesian inference often captures perception extraordinarily well [2,3] (for a critical discussion see also [4]). In light of these results, it has been proposed that the primary computation that is performed by the cortex is a hierarchically organized inference process, where cortical areas combine bottom-up sensory information and top-down expectations to find a consistent explanation of sensory data [5–8].

While the general idea of hierarchical inference in cortex found considerable experimental support [7,9,10], it is less clear how exactly this inference could be implemented by cortical neurons. A popular theory to describe the neural substrate of inference in cortex is classical **hierarchical predictive coding (hPC)** [6,11]. A central proposition of this theory is the existence of error units, which are thought to compare top-down predictions with bottom-up inputs, and guide neural computation and plasticity. However, classical hPC for the most part remains on the level of firing-rate dynamics of neural populations and it has proven difficult to connect the theory to the properties of single neurons with spiking dynamics [12,13].

Here we point towards a different, emerging theory of hierarchical inference in cortex, which relies on the local membrane dynamics in neural dendrites. The core idea of this theory, which we will refer to as dendritic hPC, is to shift error computation from separate neural populations into the dendritic compartments of **pyramidal neurons**. We will first discuss how this shift in perspective enables a biologically plausible implementation of hPC with spiking neurons, and how it connects hPC to theories of efficient coding in **balanced spiking networks** [14] and **neural sampling** [2]. In the second part, we will discuss the biological plausibility of dendritic hPC, explain how several experimental observations of hierarchical cortical computation fit into the picture, and highlight the experimental predictions that can be generated from the theory.

[1]Max-Planck-Institute for Dynamics and Self-Organization, Göttingen, Germany
[2]Göttingen Campus Institute for Dynamics of Biological Networks, Georg-August University, Göttingen, Germany
[3]Bernstein Center for Computational Neuroscience (BCCN), Göttingen, Germany
[4]Department of Physics, Georg-August University, Göttingen, Germany
[5]These authors contributed equally to this work

*Correspondence:
fabian.mikulasch@ds.mpg.de
(F.A. Mikulasch).

Dendritic predictive coding in balanced spiking neural networks

Classical models of predictive coding

Hierarchical predictive coding (hPC) describes the processing of sensory information as inference in a hierarchical model of sensory data (see Box 1 for mathematical details, which are not needed to understand the main text). The central idea of hPC is that activity of **prediction units** in one level of the hierarchy:

(i) should accurately predict sensory data or the prediction unit activity in a lower level, and
(ii) should be consistent with top-down predictions generated by higher levels in the hierarchy.

hPC tries to understand how these properties of neural activity can be ensured by neural dynamics on short timescales, and neural learning and plasticity on long timescales. The theory predicts that to this end, the prediction units in every level of the hierarchy need access to two types of errors:

Box 1. Mathematical details of classical predictive coding

The goal in hPC is to maximize the model log-likelihood [11] (for a detailed tutorial see [134])

$$\mathcal{L} = \sum_{i=1}^{N} \log p_\theta\left(\mathbf{r}^{i-1} | \mathbf{r}^{i}\right), \qquad [\mathrm{I}]$$

where θ are the model parameters, $\mathbf{r}^i$ is neural activity of a neural network at level i, and inputs are provided by the previous level $\mathbf{r}^{i-1}$. This defines a hierarchy of processing stages that, for example, can be associated with different visual cortical areas (e.g., V1, V2, etc.), where $\mathbf{r}^0$ are visual inputs from LGN [11]. Typically, a linear model is assumed, where inputs are modeled according to

$$\mathbf{r}^{i-1} = \mathbf{D}^i \mathbf{r}^i + \mathbf{n}^{i-1}, \qquad [\mathrm{II}]$$

with decoding matrix $\mathbf{D}^i$ and Gaussian white noise $\mathbf{n}^{i-1}$ with zero mean and variance σ_{i-1}^2. With this model, for a single level i, the relevant contributions of the negative log-likelihood $-\mathcal{L}^i$ take the intuitive form of the square sum of coding errors for bottom-up inputs and errors of top-down predictions:

$$\begin{aligned} &\text{bottom-up error}: \quad \mathbf{e}^{i-1} = \mathbf{r}^{i-1} - \mathbf{D}^i \mathbf{r}^i, \\ &\text{top-down error}: \quad \mathbf{e}^{i} = \mathbf{r}^{i} - \mathbf{D}^{i+1} \mathbf{r}^{i+1}, \end{aligned} \qquad [\mathrm{III}]$$

$$-\mathcal{L}^i = \frac{1}{2\sigma_{i-1}^2} \mathbf{e}^{i-1\top} \mathbf{e}^{i-1} + \frac{1}{2\sigma_i^2} \mathbf{e}^{i\top} \mathbf{e}^{i}. \qquad [\mathrm{IV}]$$

The goal is then to minimize the sum of coding errors on a fast timescale τ_r via neural dynamics $\frac{d}{dt}\mathbf{r}^i$, and with a slow learning rate η_D via neural plasticity on the weights $\mathbf{D}^i$, by performing gradient ascent on $\mathcal{L}$:

$$\text{dynamics}: \quad \tau_r \frac{d}{dt}\mathbf{r}^i = \frac{1}{\sigma_{i-1}^2} \mathbf{D}^{i\top} \mathbf{e}^{i-1} - \frac{1}{\sigma_i^2} \mathbf{e}^i \qquad [\mathrm{V}]$$

$$\text{plasticity}: \quad \eta_D^{-1} \frac{d}{dt}\mathbf{D}^i = \frac{1}{\sigma_{i-1}^2} \mathbf{e}^{i-1} \mathbf{r}^{i\top}. \qquad [\mathrm{VI}]$$

To yield a neural implementation, the key innovation in classical hPC was to represent prediction errors within a distinct neural population of error units. Error units integrate inputs of prediction units within the same level and subtract top-down predictions according to

$$\tau_e \frac{d}{dt}\mathbf{e}^i = -\mathbf{e}^i + \mathbf{r}^i - \mathbf{D}^{i+1} \mathbf{r}^{i+1}, \qquad [\mathrm{VII}]$$

where decoding weights $\mathbf{D}^i$ now correspond directly to weights of neural connections [134]. Together with the dynamics of prediction units, this results in the hierarchical neural circuit shown in Figure 1A in the main text.

Glossary

Balanced spiking networks: recurrent networks of spiking neurons with E-I balance; these networks show asynchronous irregular spiking activity and can efficiently encode dynamic variables.
E-I balance: excitatory and inhibitory currents are 'balanced', when their magnitude approximately matches.
Hierarchical predictive coding (hPC): a theory of hierarchical inference in cortex.
Inference: in hPC, inference is the process of finding the underlying causes of sensory data; these underlying causes can be used to predict (or similarly, 'explain away') the sensory input or the activity in lower levels of the hierarchy.
Lateral inhibition: pyramidal neurons in a population compete via lateral inhibition through interneurons, which can be used to both increase the efficiency of the neural code and to distinguish between competing explanations of sensory data.
Neural sampling: instead of computing a single best explanation of sensory data, neural activity can sample possible explanations according to their likelihood.
Prediction neuron: pyramidal neuron that aims to predict the activity of other neurons, as proposed by dendritic hPC.
Prediction unit: abstract unit of neurons that aims to predict the activity of other units, as proposed by classical hPC.
Pyramidal neuron: the primary excitatory neuron in cortex, typically with a characteristic long 'apical' dendrite.
Tight balance: if the E-I balance is present not only on average, but also on short timescales, it is 'tight'.
Voltage-dependent plasticity (VDP): changes in synaptic strength that depend on the postsynaptic membrane potential in the vicinity of the synapse.

(i) bottom-up errors (i.e., the mismatch between activity in lower levels and predictions generated within the level);
(ii) top-down errors (i.e., the mismatch between activity within the level and top-down predictions from higher levels).

In classical hPC [11], the key innovation was to represent these errors in distinct populations of error units that compare top-down predictions with the activity within a level (Figure 1A, Key figure). The elegance of this approach is that the same error units can mediate both, bottom-up errors to update prediction units in the next level, as well as top-down errors to neurons of the same level. Another central result of classical hPC is that the learning rules that improve the hierarchical model take the form [error × prediction], which turns out to be classic Hebbian plasticity (i.e., the multiplication of pre- and postsynaptic activity).

A functionally equivalent formulation of predictive coding with dendritic error computation

Although the idea of error units is undeniably elegant, it is not the only way to compute prediction errors in a neural circuit. More recent models showed that error computation can also be performed in the voltage dynamics of individual dendritic compartments [14–16] and, thus, without specialized error units. Combining these models allows for a reinterpretation of hPC, which we term dendritic hPC, where every **prediction neuron** will represent the two types of errors we discussed before in different sections of its dendritic tree (Figure 1B, see Box 2 for mathematical details):

Key figure

Implementation of predictive coding with dendritic error computation and spiking neurons

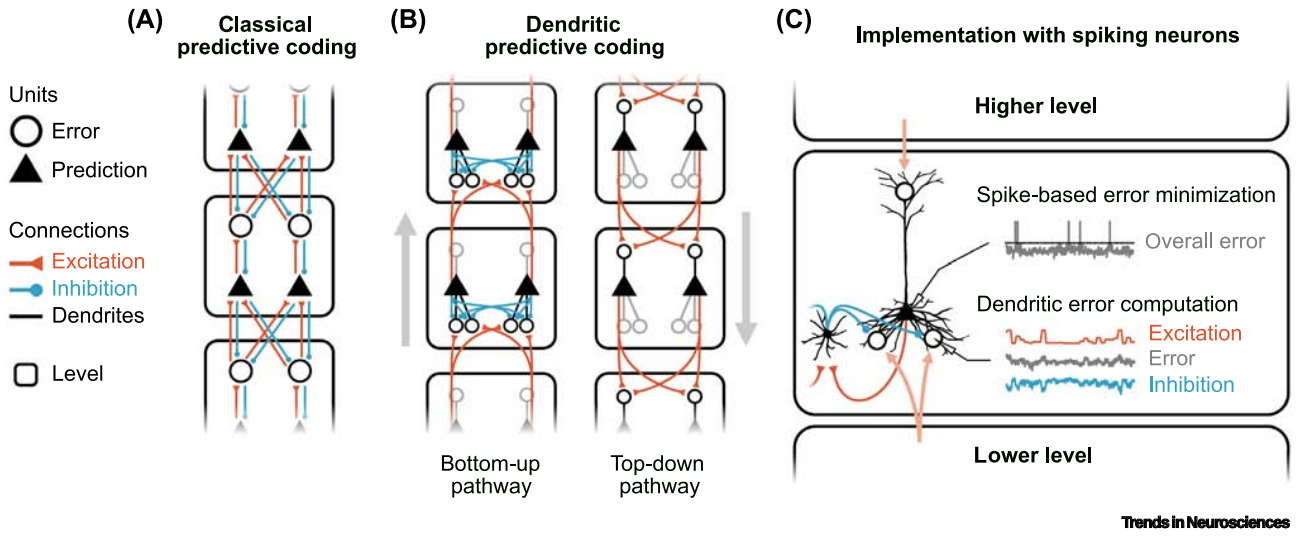

Figure 1. (A) Illustration of the classical model of hierarchical predictive coding (hPC). Errors and predictions are computed in different neural populations within one level of the hierarchy. Errors are sent up the hierarchy, while predictions are sent downwards. (B) In dendritic hPC, prediction neurons implement the same function, but errors are computed in neural dendrites. Predictions are sent up the hierarchy to basal dendrites, where they are balanced by lateral connections to compute bottom-up prediction errors (left). At the same time, predictions are sent down the hierarchy to apical dendrites, where they try to predict somatic spiking and guide the inference process (right). The pathways are shown separately for better visibility. (C) Dendritic hPC can be implemented with spiking neurons. The errors that are computed in the dendritic membrane potentials are integrated at the soma to form an overall error signal of the neuron's encoding. A spike is emitted when the somatic error potential grows too large and a spike would lead to a reduction in the overall error.

Box 2. Mathematical details of dendritic predictive coding

In dendritic hPC, the computation of errors in Equation VII is accomplished by the leaky voltage dynamics of dendritic compartments. Different models have explored this idea separately for basal dendrites [16,25,40] and apical dendrites (also with nonlinearities, which we here omit) [15,28], which we here combine to form a model that is equivalent to classical hPC. To this end, for each prediction neuron j, one introduces basal dendritic compartments $b^i_{jk} \approx D^i_{kj} e^{i-1}_k$, which are each innervated by a single synapse of a prediction neuron k of the previous level [16], as well as an apical compartment $a^i_j \approx -e^i_j$ that is innervated by prediction neurons of a higher level [15] (see Figure 1B in the main text). The error computation is then performed by voltage dynamics according to

$$\tau_b \frac{d}{dt} b^i_{jk} = -b^i_{jk} + D^i_{kj} r^{i-1}_k - \sum_l W^i_{jkl} r^i_l, \qquad \text{[VIII]}$$

$$\tau_a \frac{d}{dt} a^i_j = -a^i_j - r^i_j + \sum_l D^{i+1}_{jl} r^{i+1}_l, \qquad \text{[IX]}$$

where bottom-up inputs are balanced with lateral connections W^i_{jkl} (connection of neuron r^i_l to the kth dendritic compartment of neuron r^i_j), and top-down predictions are matched by the neurons own predictions r^i_j. The latter has been proposed to be implemented via the backpropagating action potential [15], solving the one-to-one connections problem of classical hPC [135]. To compute bottom-up errors, lateral weights have to be chosen as $W^i_{jkl} = D^i_{kj} D^i_{lj}$. Such weights can be found through a voltage-dependent plasticity rule, which enforces a tight balance in the kth dendritic compartment [16]

$$\eta_W^{-1} \frac{d}{dt} W^i_{jkl} = \frac{1}{\sigma^2_{i-1}} b^i_{jk} r^i_l. \qquad \text{[X]}$$

The dynamics of prediction neurons are then simply driven by the dendritic error potentials

$$\tau_r \frac{d}{dt} r^i_j = \frac{1}{\sigma^2_{i-1}} \sum_k b^i_{jk} + \frac{1}{\sigma^2_i} a^i_j, \qquad \text{[XI]}$$

and weights for bottom-up and top-down inputs can be learned with voltage-dependent rules (Equation XII proposed in [16], Equation XIII proposed in a generalized form in [15])

$$\eta_D^{-1} \frac{d}{dt} D^i_{kj} = \frac{1}{\sigma^2_{i-1}} \frac{1}{D^i_{kj}} b^i_{jk} r^i_j, \qquad \text{[XII]}$$

$$\eta_D^{-1} \frac{d}{dt} D^{i+1}_{jl} = -\frac{1}{\sigma^2_i} a^i_j r^{i+1}_l. \qquad \text{[XIII]}$$

Here, learning of bottom-up weights requires that lateral and bottom-up weights always align via $W^i_{jkl} = D^i_{kj} D^i_{lj}$, which in classical hPC is known as the weight transport problem [49,135]. For dendritic hPC a solution based on weight decay has been proposed in [16], which was demonstrated in a single-level model and is similar to a solution proposed for classical hPC [49]. Together, these equations yield an equivalent formulation of hPC for both learning and inference, where prediction errors are computed locally in dendritic compartments.

(i) bottom-up errors in basal dendritic compartments [16], where input from lower-level cortical areas is integrated [17];
(ii) the top-down prediction error (for the neuron's own activity) in an apical compartment [15], where higher-level cortical feedback arrives [17].

Besides the absence of error units, two additional central differences arise between the architectures of classical and dendritic hPC. First, in dendritic hPC both bottom-up and top-down signals are predictions, a possibility that has been discussed before [18]. Second, and more importantly, while prediction units in classical hPC inhibit each other through error units, prediction neurons in dendritic hPC directly compete through **lateral inhibition** on basal dendrites. Such networks with strong lateral inhibition (or similarly, winner-take-all-like dynamics [19]) have a long tradition in theoretical neuroscience, as models for the sparse and efficient encoding of sensory data [16,20–25] and as divisive normalization models of cortical computation [26,27]. Dendritic hPC

is closely related to these models, except that in these models it was not considered how exactly top-down connections could guide neural computations with predictions. In a more general context it has been proposed that top-down connections could provide these predictions by targeting apical dendrites [15,28–31]. Dendritic hPC combines these ideas of lateral competition and top-down predictions into a coherent theory of hierarchical inference in cortex.

Since in dendritic hPC error computation takes place in the voltage dynamics of basal and apical dendritic compartments, these local potentials play an important role for synaptic plasticity. For basal dendrites, dendritic hPC predicts that plastic lateral connections compute the errors for bottom-up inputs by establishing a **tight balance** locally in individual dendritic compartments (i.e., trying to closely match excitatory with inhibitory currents [32]). The intuitive explanation for this computation is that in a tightly balanced state, every input that can be predicted from other neurons is effectively canceled and the remaining unpredictable input constitutes the prediction error [14,16]. These errors can then be exploited by another voltage-dependent rule for bottom-up connections, in order to find an optimal encoding of inputs [16]. This learning rule is Hebbian-like (i.e., pairing postsynaptic firing with presynaptic input will induce potentiation of the synapse). At the same time, strong local inhibition during the postsynaptic spike would signal an over-prediction of the input and consequently should lead to long-term depression of the synapse. For apical dendrites, it has been proposed that error computation relies on the mismatch between apical prediction and somatic spiking [15]. In this theory of apical learning, plasticity of top-down connections is Hebbian-like as well, but synapses are depressed for a depolarization of the apical dendritic potential in the absence of somatic spiking. By employing these **voltage-dependent plasticity (VDP)** rules, dendritic hPC implements the same learning algorithm as classical hPC, but in prediction neurons with dendritic error computation (Box 2).

Dendritic error computation has also been used in a different context to implement the backpropagation algorithm in a cortical microcircuit [33–36]. Although this model of dendritic error backpropagation and dendritic hPC employ similar ideas, they ultimately pursue different goals and thus make distinct predictions for plasticity and **E-I balance** in basal and apical dendritic compartments (Figure 2).

Dendritic errors enable an efficient implementation of hPC with spiking neurons

Dendritic errors do not only yield an equivalent formulation of hPC, they also enable inference with spiking neurons. Here, the inferred variables have to be efficiently represented by spikes, which is

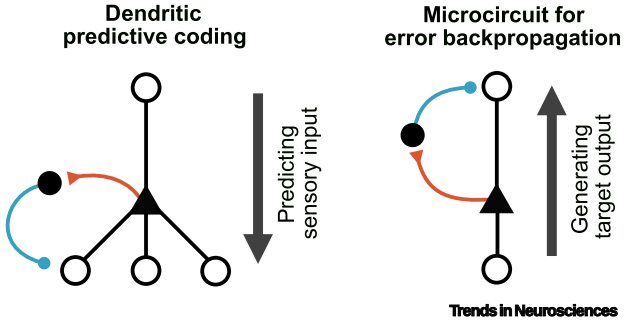

Figure 2. Relation of dendritic predictive coding to dendritic microcircuits for error backpropagation. (Left) In dendritic hierarchical predictive coding (hPC) the goal is to generate predictions of bottom-up sensory inputs. Here prediction errors are computed via balancing inhibition to basal dendrites and the mismatch of top-down predictions and somatic spiking at apical dendrites. (Right) In models that employ backpropagation the goal is to generate a target output at the highest level (e.g., a label) [33]. To this end an 'inverted' model of hPC is employed [35], where balancing inhibition at the apical dendrite is used to compute the backpropagated error of the output. While thorough testing of both theories remains to be conducted, a recent study indicates that pyramidal neurons learn predictive (and not balanced) apical activity [31], more consistent with dendritic hPC. However, this particular observation of course would not rule out that cortical networks could make use of both proposed mechanisms in different modes of operation or different neural populations.

Box 3. Mathematical details of dendritic predictive coding with spikes

Spike-based predictions of sensory data

A popular choice to mathematically formalize the prediction generated by a spike at time t_{sp} is via spike traces $\kappa(t, t_{sp}) = \exp(-(t - t_{sp})/\tau)$ that decay exponentially with some time constant τ [16,40]. Predictions of a neuron then change upon a spike according to $r(t) \rightarrow r(t) + \kappa(t, t_{sp})$, which approximately corresponds to the way spikes are read out in the membranes of postsynaptic neurons. With these predictions $r(t)$, the same formalism as before can be used to compute the instantaneous log-likelihood (see Box 1 in the main text):

$$\mathcal{L}(t) = \sum_{i=1}^{N} \log p_\theta\left(\mathbf{r}^{i-1}(t) | \mathbf{r}^i(t)\right). \quad [XIV]$$

However, due to the discontinuous nature of spikes, inference can no longer be implemented by simple gradient ascent.

Efficient spiking implementation of predictive coding with dendritic errors

One straightforward approach to implement inference with spikes is to deterministically fire a spike at time t if it instantly improves bottom-up and top-down errors, that is, the log-likelihood $\mathcal{L}(t)$ [40]:

$$\mathcal{L}(t|\text{neuron } j \text{ spikes at time } t) > \mathcal{L}(t|\text{no spike at time } t). \quad [XV]$$

This can be seen as a discrete implementation of gradient ascent to find the instantaneous maximum a posteriori (MAP) estimate for predictions r_j^i. From this principle it can be derived that a neuron should spike if its balanced membrane potential $u_j^i(t)$ surpasses a firing threshold T_j [40], that is, if

$$u_j^i(t) = \frac{1}{\sigma_{i-1}^2} \sum_k b_{jk}^i + \frac{1}{\sigma_i^2} a_j^i > T_j. \quad [XVI]$$

This equation is analogous to Equation XI, where $b_{jk}^i(t)$ are the balanced dendritic potentials of basal dendrites and $a_j^i(t)$ the potential of the apical dendrite.

Predictive coding with neural sampling

A more general approach to inference with spikes is to sample a (binary) spike train $\mathbf{S}_{0:T} = \{\mathbf{s}^i(t) | i \in \{1, \ldots, N\}, t \in \{0, \ldots, T\}\}$ from the posterior distribution of the generative model $\mathbf{S}_{0:T} \sim p_\theta(\mathbf{S}_{0:T} | \mathbf{r}_{0:T}^0)$ [16,136]. The posterior is implicitly defined via the model $p_\theta(\mathbf{r}^{i-1}(t) | \mathbf{r}^i(t))$, a prior on spiking $p_\theta(\mathbf{s}^N(t))$ and spike traces $\mathbf{r}^i(t) = \sum_{t'=0}^{t} \mathbf{s}^i(t') \kappa(t, t')$. While computing the posterior distribution exactly is intractable [16,136], approximate online sampling can be implemented with the same membrane potentials $u_j^i(t)$ and threshold T_j as before (up to a constant factor) and a soft spiking threshold mechanism

$$p(\text{neuron } j \text{ spikes at time } t) = \text{sig}\left(u_j^i(t) - T_j\right), \quad [XVII]$$

where $\text{sig}(x) = 1/(1 + \exp(-x))$ is the logistic function [16]. Note, that $u_j^i(t)$ and T_j are scaled by the precisions of errors $\frac{1}{\sigma_i^2}$ (Equation XVI) and thus the stochasticity of spiking will capture the uncertainty in inference. This model is a special case of the spike response model with escape noise [137] and can be implemented by a leaky-integrate-and-fire neuron with a noisy membrane potential. Equations XI, XVI, and XVII highlight the intimate relation that exists between the theories of hPC, efficient coding with spikes, and neural sampling.

possible if spikes are only fired if they reduce the overall prediction error [14,37,38] (see Box 3 for the mathematical details of dendritic hPC with spiking neurons). Since in dendritic hPC prediction errors are represented in the balanced membrane potentials, an efficient spike encoding can be found with a simple threshold mechanism that generates a spike when the error potential grows too large (Figure 1C), as demonstrated in single-level models [39,40]. Predictive coding thus serves a dual purpose in dendritic hPC, by enabling both inference in a hierarchical model and an efficient spike encoding of dynamical variables.

A central role in this inference scheme with spikes is played by noise in the neural dynamics, for two reasons. First, noise enables an efficient spike encoding in the face of transmission delays. With deterministic neurons, even a small delay of inhibition can lead to erratic network behavior, since inhibition will often arrive too late to prevent synchronous spiking of large parts of a

population [41]. Noise relaxes this constraint on the speed of feedback, since it effectively decouples and desynchronizes neural spiking [37,41,42]. Second, noise in spiking neural networks enables neural sampling [2,43–46]. Here, the idea is that neural activity samples possible predictions according to their likelihood, instead of computing a single best estimate as in classical hPC (Box 3). Neural sampling therefore is a principled way to represent uncertainty in inference via neural activity and has, for example, been used to explain variability in neural responses [47,139] and the origin of multistability in perception [48]. Recent models show that neural sampling and efficient spike coding with tight E-I balance can be combined in a single model with dendritic error computation [16,43], relating these concepts to the proposed model of dendritic hPC (Box 3).

In addition to neural inference, dendritic errors also enable learning in populations of spiking neurons. This is not straightforward, since the switch from rate-based to spike-based models typically requires a modification of the learning algorithms. For example, when using spiking error units, as in classical hPC, it is not directly possible to represent both positive and negative errors by non-negative activity [49]. To resolve this, it was proposed that errors are represented by deviations relative to a baseline firing rate [49], but this would require high firing rates and therefore seems implausible considering the low firing rates in neocortex [50]. An alternative is to represent positive and negative errors in separate populations [11,50], but it is unclear how in this case biological plasticity can recombine the positive and negative parts, which are both required for the learning of single synapses. Due to these difficulties, to date, no complete implementation of hPC that uses spiking error units has been proposed [13]. By contrast, in dendritic hPC the same learning algorithm as for rate-based units can be straightforwardly applied to spiking neurons. The reason is that dendritic membrane potentials remain continuous quantities, despite the spiking nature of neural activity, and thus can easily represent the prediction errors that are required for the learning of bottom-up and top-down connections (Box 2), which has been successfully applied in [15,16].

Is dendritic predictive coding biologically plausible?

In the previous section we have introduced the two main assumptions of dendritic hPC, which are: (i) cortex implements inference in a hierarchical probabilistic model, and (ii) errors of the resulting predictions are computed in the local voltage dynamics of basal and apical dendrites. The implications of the first assumption have been discussed at length in the context of classical hPC and were found to align well with experimental observations [7,10,51]. In the following we will discuss the biological plausibility of the second assumption. Ultimately, we will argue that dendritic hPC can indeed be closely connected to many properties of pyramidal neurons and inhibitory connectivity in cortex.

Dendritic error computation and synaptic plasticity in pyramidal neurons

To compute errors in basal dendrites, a tight and local E-I balance is required. Indeed, it has been found in several instances that inhibitory and excitatory currents are tightly correlated, with inhibition trailing excitation by few milliseconds [14,52,53]. This tight balance leaves neurons only with a brief window of opportunity for spiking, which effectively decorrelates neural responses to inputs and thereby ensures an efficient neural code [25]. A tight E-I balance can therefore explain the origin of the irregular spiking patterns of neurons that have been observed throughout cortex [14,54]. Although models with a tight balance can reproduce irregular firing on the single neuron level, incorporating realistic synaptic transmission delays in these models can lead to oscillations on the population level [37]. Oscillations in cortical activity in the gamma frequency band have therefore been discussed as signatures of efficient coding in balanced networks [42] (and might also support efficient neural sampling [45,55]).

Consistent with dendritic hPC, this balance has also been found to extend to individual dendritic compartments [32,56,57]. Crucially, this local balance can be observed down to the scale of

(at least) single dendritic branches [56], since the attenuation of dendritic currents prevents that inhibitory postsynaptic potentials spread into other dendritic branches and influence the E-I balance there [58,59]. Experiments could also show that this local balance is maintained through localized synaptic plasticity, which re-establishes the balance after a perturbation and coordinates excitatory and inhibitory plasticity locally [56,60–65]. Overall, these findings are compatible with the idea that a local balance can compute prediction errors for specific synaptic contacts at basal dendrites.

Another prediction of dendritic hPC, which has been consistently observed in a range of experiments, is that the local membrane potential is a central determinant of synaptic plasticity [61,65–68]. This VDP is thought to be mainly mediated by the local calcium concentration, which follows the local membrane potential and modulates synaptic plasticity [59,69,70]. Based on these observations, VDP rules have been proposed that can reproduce several experiments of spike-timing-dependent plasticity in a unified picture [71–73]. An especially important consequence of locally organized VDP, which is also required by dendritic hPC, is that inhibition can strongly modulate synaptic plasticity in a very localized manner [32,65,74–76].

Are the VDP rules that can be derived from dendritic hPC consistent with these experimentally observed VDP rules? A distinction has to be made here between VDP rules in basal dendrites, which should enable the learning of neural representations [16], and VDP in apical dendrites, which should enable the prediction of somatic spiking [15]. For representation learning in basal dendrites, we have argued in [16] that previously proposed VDP rules [71,72] can be reconciled with the VDP rules derived from dendritic hPC. One prediction of these derived VDP rules is that strong local inhibition should promote the depression of excitatory synapses, an effect that has been observed in proximal dendrites of hippocampal pyramidal neurons [75] (similarly found in [77]). By contrast, for the learning of apical connections, an explicit correspondence to experimental VDP still has to be proposed. Experiments show that synaptic plasticity close to and far from the soma behaves vastly differently [31,78–80], which could support the different requirements for basal and apical synaptic plasticity in dendritic hPC. While more experimental and theoretical work is needed to clarify the connections between dendritic hPC and experimental VDP, these results suggest that cortical pyramidal neurons in principle are suited to implement the learning algorithm proposed by dendritic hPC.

A diversity of inhibitory interneurons is required for dendritic predictive coding

Since pyramidal neurons in general only excite other cells, additional inhibitory interneurons are required to implement the dendritic hPC model. The central inhibitory motif of dendritic hPC requires interneurons that balance bottom-up inputs to basal dendrites via lateral connections [16,25]. These interneurons show strong similarities to parvalbumin-expressing (PV) interneurons in cortex, which implement a precisely adjusted competition between pyramidal neurons [24,81–84]. PV positive, fast-spiking basket cells alone make up around 30–50% of all interneurons in the cortical microcircuit [85] and are especially adapted to tightly control pyramidal neuron spiking and the cortical E-I balance via very fast inhibition to somata and basal dendrites [86,87]. PV interneurons also seem to be responsible for the gamma oscillations that similarly arise through lateral inhibition in dendritic hPC [87–89]. Dendritic hPC is therefore closely linked to one of the defining inhibitory motifs of cortex.

Next to PV interneurons, most other interneurons in cortex can be classified as either somatostatin-expressing (SST) interneurons, which preferentially target the apical dendrites of pyramidal neurons, or vasoactive intestinal peptide-expressing (VIP) interneurons, which mainly inhibit other interneurons, especially SST [86,90]. SST and VIP interneurons, for example, have been observed to be responsible for top-down inhibitory control [91], which is also required in

dendritic hPC when top-down input predicts a decrease in activity. However, not all of the major connectivity patterns of SST and VIP cells can be straightforwardly explained by dendritic hPC: SST interneurons, for example, also mediate short-range lateral inhibition to apical dendrites, which allows them to contribute to surround suppression [92] and to gate top-down input [93,94]. The disinhibitory circuit of VIP contributes to this gating mechanism by specifically suppressing SST neurons during active behavior [93,95,96]. SST and VIP neurons have also been found to be crucial for gating apical plasticity, for example, during reward-based learning [97–99]. These connectivity motifs thus play a central role in how predictions are processed by apical dendrites, but precisely what functions they could implement, especially in the context of dendritic hPC, has yet to be understood [65].

Dendritic predictive coding in neocortical lamination

Neocortex employs multiple types of pyramidal neurons that reside on different cortical layers and exhibit specific connectivity [17]. We here propose that dendritic hPC in particular describes the computations of layer 2/3 neurons (Figure 3D). That layer 2/3 neurons are central in the hierarchical integration of information and the interpretation of sensory data has been proposed before, for example, based on cortical physiology [19] or in theories of classical hPC, where errors and predictions are first computed in layer 2/3 [6] (Figure 3C). There are several arguments for why dendritic hPC is particularly well suited to describe layer 2/3: first, like in dendritic hPC, layer 2/3 neurons combine bottom-up signals (sent from layer 4 to their basal dendrites) with top-down

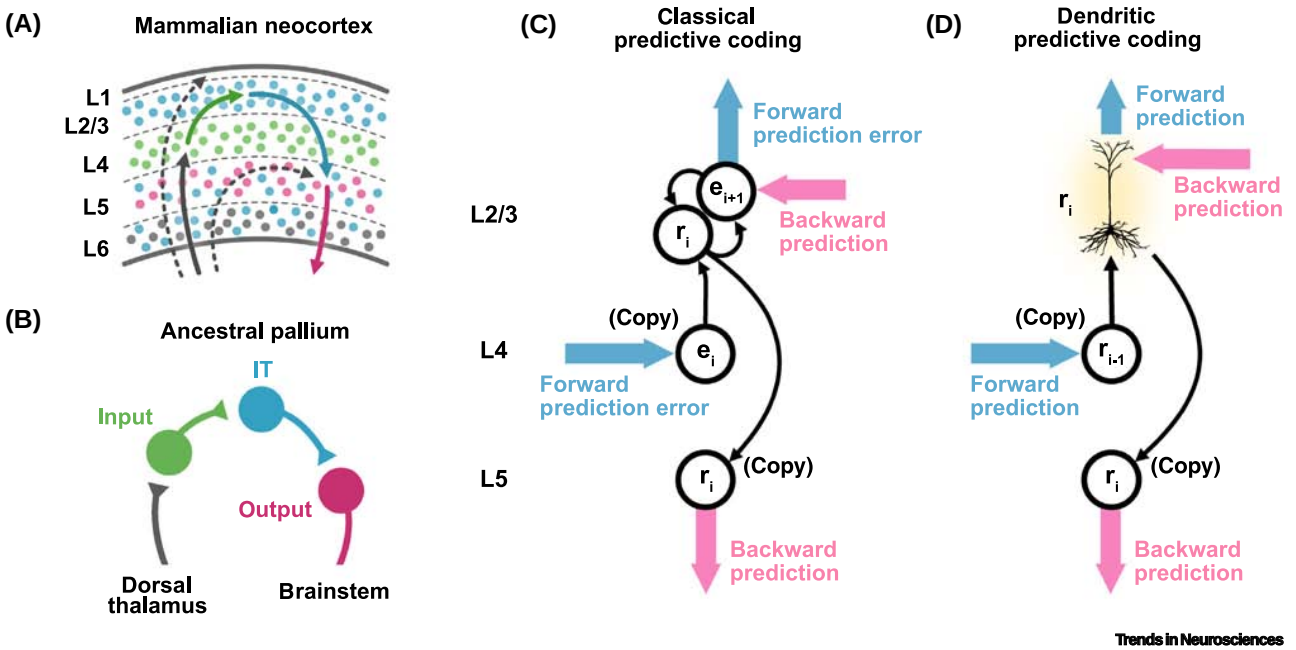

Figure 3. How could dendritic predictive coding be embedded into neocortical microcircuits and lamination? (A) Core circuitry of mammalian neocortex, as shown in [102,104]. Input neurons in layer 4 (green) receive sensory information from the dorsal thalamus, layer 2/3 intratelencephalic (IT) neurons (blue) further process this information, and output neurons in layer 5 (red) project to the brainstem and other areas. Additional connections, for example, from thalamus to layer 1 (mostly relayed from other cortical areas [94]) or layer 5 (broken lines), or within layer 2/3 between areas exist [17,138], but will be omitted in the following for simplicity. (B) Theories of cortical evolution hypothesize that these input, IT, and output cells are homologous to cells that existed in the ancestral amniote pallium [104]. Also, in birds and non-avian reptiles, homologous cell types exist, but are organized in architectures that differ from the laminar organization of mammalian neocortex. (C) The predictive coding microcircuit as proposed by [6] (here presented in a simplified form) follows the organization of the neocortical microcircuit. Predictions (r_i) and prediction errors (e_i) are computed in layer 2/3. Deeper layers mainly act as communication hubs by copying signals from layer 2/3. (D) Speculative microcircuit for dendritic predictive coding. Here, deeper layers fulfill the same role as communication hubs (and possibly complementary functions [19]), but layer 2/3 only computes predictions.

signals (sent from layer 5 or layer 2/3 to their apical dendrites) [6,17,19]. Second, layer 2/3 neurons exhibit sparse activity, which is mainly enforced by lateral inhibition via PV interneurons [83,84,100], a motif that is present in dendritic hPC but not in other theories of hPC [6,35]. Last, superficial cortical layers show pronounced gamma oscillations [6,88,89] that are expected to arise through lateral inhibition in dendritic hPC [37,42].

Importantly, these properties implied by dendritic hPC are not general features of pyramidal neurons, which in other layers likely implement different functions. Layer 5 neurons, for example, employ a dense and not a sparse code [100] and show less gamma oscillations [6,89]. These properties, together with the position of layer 5 neurons as downstream elements in the microcircuit [17], have led to the suggestion that layer 5 might be employed in long-range communication [100] and output selection [19]. Layer 4 in turn shows an abundance of PV interneurons [86] and could implement a preprocessing of bottom-up inputs [17]. These different roles of deeper layers are also in line with theories of cortical evolution, which hypothesize that deeper layers have migrated from previously separate 'input' and 'output' neural populations to neocortex in order to integrate cortical neurons more deeply with the rest of the brain and other cortical areas [101–104] (Figure 3A,B). Hence, the different functions of deeper layers could complement the computations of dendritic hPC in important ways, but how exactly such an interaction could look has yet to be formulated.

Another aspect of cortical lamination that could support the computations of dendritic hPC are neuromodulators. Neuromodulators act on a wide range of scales [105] and can target specific cortical layers, where they might modulate computations in specific dendritic domains of pyramidal cells [94,106–108]. For example, acetylcholine (ACh), which is associated with attention and learning, has been found to promote (dis-)inhibition of apical or basal dendrites through distinct mechanisms, possibly in a very targeted manner [96,99,107–109]. In the context of hPC, ACh and other neuromodulators have been proposed to set the precisions of the internal model and thereby determine the influence of sensory and top-down information [110–112]. The separation of top-down and bottom-up inputs across cortical layers, as in dendritic hPC, could therefore be a central factor to enable the targeted modulation of these pathways. This might not only apply to the effects of ACh on neural gain, but also to the various other effects ACh and other neuromodulators have on cortical dynamics and plasticity [105].

How can error responses arise in prediction neurons?

One of the central features of classical hPC is its ability to explain a variety of experimental observations through the concept of error neurons. Error neurons have, for example, been used to explain extra-classical receptive field effects in visual cortex [11], as well as mismatch responses in cortex, which are neural responses that appear to signal the mismatch between an internal model and sensory data [10]. Thus, an important question for dendritic hPC is if and how these experimental observations can arise in a model without error neurons.

The first experimental observation that has been explained with error neurons in hPC is the extra-classical receptive field effect of endstopping [11]. In endstopping it is found that, first, the response of a neuron in V1 to a bar stimulus decreases when the bar extends over its receptive field, and second, this effect is reduced when feedback from higher-level areas is disabled [113,114]. Recent theoretical work showed that endstopping behavior, as well as other extra-classical receptive field effects, also occur in prediction neurons, where top-down connections strengthen these effects [7,115,116]. Here, endstopping is mainly mediated by lateral inhibition between neurons with overlapping receptive fields [116]. Top-down connections from higher-level areas predict the activity patterns that arise from these lateral interactions and enhance

them, which strengthens endstopping behavior [115]. This cooperation of lateral and top-down interactions could be important to help the network to cope with noise in the inputs and improve visual processing [115,117] and has been widely observed in visual cortex [114,117–119].

Mismatch responses have been observed in different forms, such as responses to the omission of expected stimuli [10], responses to a mismatch between information in different modalities (e.g., visual and motor information) [120–122], strong responses to unexpected stimuli [7,123], or suppressed responses to expected stimuli [1,124]. Omission responses can already occur in straightforward prediction neuron responses, as prediction neurons can be active even without the expected input [10]. Recent work from our group has also shown that multimodal mismatch responses can naturally arise in prediction neurons, when different cortical areas jointly infer a consistent explanation of sensory data [125]. This joint inference aims to find single causes that underlie stimuli in multiple modalities, meaning cortical areas should suppress predictable activity in other areas (as in [122,126]), but might also drive activity in case of a prediction mismatch (as in [120,121,127]). Strong/suppressed responses to unexpected/expected stimuli in turn have so far not been explained with pure prediction responses, but it has been argued that they might be mediated by other mechanisms, such as attention to interesting stimuli, the variance in neural sampling, or adaptation mechanisms [7,124,128]. In conclusion, the observed mismatch responses can be explained by a variety of plausible mechanisms in prediction neurons, which, however, in some cases might not be directly relatable to the computations of dendritic hPC.

Testable predictions

To better assess the potential as well as the limitations of dendritic hPC to describe inference in cortex, we here propose experiments that: (i) test predictions for specific neural mechanisms, and (ii) aim to distinguish between the different implementations of hPC with and without error neurons.

Predictions for specific neural mechanisms

- Bottom-up excitation to basal dendrites of layer 2/3 pyramidal cells should be locally matched and balanced with lateral inhibition, likely via PV interneurons (an indication that such a precise matching is possible, e.g., in dendritic spines, has been found in [129]). This could be tested in detail, for example, using large-scale connectomics datasets [130].
- Plasticity for excitatory bottom-up connections is predicted to be modulated by local inhibitory input, which is expected to turn long-term potentiation into depression. While such modulation of plasticity has been found (e.g., in hippocampal neurons in a spike-timing-dependent plasticity experiment [56]), it would be interesting to test this more specifically in layer 2/3 basal dendrites, with a particular focus on the predicted impact of the strength and timing of inhibition on plasticity [16].
- Similar experiments could be conducted for top-down connections to apical dendrites, where plasticity should be Hebbian, but switch to depression when presynpaptic spikes depolarize the dendrite while the neuron remains silent. Also, here it would be interesting to explicitly test for the predicted dependence of plasticity on the dendritic membrane potential [15].
- As a consequence of these plasticity mechanisms, activity in basal dendrites is expected to be decreased (‘explained away’) in the course of learning, whereas activity in apical dendrites should increase and become predictive of somatic spiking (similar to what was found in [31]). An important experiment would be to test explicitly if apical activity indeed becomes predictive on a single neuron level, which would also distinguish dendritic hPC from theories of dendritic error backpropagation that predict a clear decrease of apical activity (Figure 2).

Distinguishing between hPC with and without error neurons

The central challenge in distinguishing between different implementations of hPC is that their underlying mathematical framework is the same, hence they predict the same computations in prediction neurons. Thus, since classical hPC as yet does not make clear predictions on the single neuron level, the main distinguishing characteristic between classical and dendritic hPC is the presence or absence of error units. For specific computations, this might be used to rule out one of the models:

- As we discussed, mismatch responses are explained via distinct mechanisms in models with or without error neurons, which could be tested on a case-by-case basis. For example, mismatch responses in multimodal mismatch experiments are transient [131], where classical hPC predicts this decrease to be caused by top-down inhibition, while in dendritic hPC one would expect the origin in adaptation or other bottom-up mechanisms [125] (for other experiments, see also discussion in [7,124]).
- Another, more direct approach would be to map out the functional circuits in cortex, where classical hPC expects a clear separation between error and prediction units (i.e., error units only receive predictions and vice versa), but dendritic hPC expects no such separation. For example, in several experiments reporting 'error' and 'prediction' neurons, their populations appear intermixed [123,132] and it would be important to clarify whether or not there exists a clear feedforward–feedback circuit motif between these populations (e.g., if bottom-up excitation and inhibition always arrives first in one of the populations).

For these experiments it is important to note that dedicated error neurons (or even classical hPC) might coexist with dendritic hPC for complementary computations. For example, it is well known that dopaminergic neurons code for reward prediction errors to guide behavioral learning [133]. However, it is unclear whether there exists an advantage to implement the same computation, such as inference in sensory cortex, simultaneously with two different implementations of hPC.

Concluding remarks

Since its conception over 20 years ago, hPC has been considered one of the most promising unifying theories of cortical computation, but – in its classical form – it is still facing substantial questions regarding its biological plausibility. Here, we outlined an emerging hPC scheme based on dendritic error computation, which is functionally equivalent, but provides solutions to the most pressing open problems of the established theory of classical hPC: first, it can explain the lack of clear empirical evidence for the coexistence of error and prediction neurons [10,51], and second, it overcomes the unresolved question of how learning can be efficiently implemented with spiking error neurons [13]. Moreover, we explained how dendritic hPC could connect the microscopic properties of neural dendrites, such as the local E-I balance [14,32,57] and VDP [72,75], to neural dynamics [14] and learning [15,16,115] in the cortical hierarchy.

These advances open up several interesting paths for future research. Next to experimentally testing for the predicted mechanisms of inference and learning in cortex (see section 'Testable predictions'), there are a number of open theoretical challenges, especially concerning the details of the biological implementation (see Outstanding questions). Going forward, it will also be important to understand how the learning of a hierarchical model of sensory data interacts with complementary mechanisms, such as attention and behavioral learning, not only for dendritic hPC, but also for hPC and other theories of inference in cortex more generally.

Outstanding questions

Dendritic hPC has been derived under the assumption of linear dendrites for a linear encoding of sensory data, but dendrites often show nonlinear behavior. How can the ideas of dendritic hPC be transported to a model with nonlinear dendrites and could this allow for a nonlinear and thus more versatile encoding?

Pyramidal cells show extensive lateral excitatory connectivity, which could be used to learn and predict temporal sequences within a single level. Can these mechanisms interact purposefully with the learning of predictions in a hierarchical model?

When cortical areas communicate there might be substantial challenges, such as long transmission delays or sparse activity in both areas. Are there additional mechanisms that could improve neural communication under these conditions, such as communication through coherence, and how could they be integrated into dendritic hPC?

Pyramidal cells are not a uniform class of cells, for example, the different physiology of layer 2/3 and layer 5 apical dendrites leads to different integration of top-down inputs, but also layers 2 and 3 contain slightly different subtypes of pyramidal cells. What are the functional reasons for these properties and how are they related to dendritic hPC?

We have suggested that dendritic hPC describes the computations of layer 2/3 pyramidal neurons. Under this assumption, what are the roles of deeper cortical layers and how can they be integrated into the framework?

Inference has not only been used to model sensory processing, but also computations in hippocampus, and some of the core predictions of dendritic hPC also seem to apply to hippocampal pyramidal cells. Are principles of dendritic hPC also employed by different brain regions, or different neuron types?

Often indirect measures of neural activity (e.g., electroencephalography, fMRI) have been used to search for evidence of classical hPC. How would

error computation in specialized neurons and in dendrites differ in these measures?

Acknowledgments

We would like to thank Abdullah Makkeh, Beatriz Belbut, Caspar Schwiedrzik, David Ehrlich, Georg Keller, and members of the Priesemann Lab, especially Andreas Schneider, Kjartan van Driel, and Matthias Loidolt, for helpful discussions and comments on the manuscript. F.A.M. and L.R. were funded by the German Research Foundation (DFG), SFB 1286. V.P. and M.W. received support from the German Research Foundation (DFG), SFB 1528, Cognition of Interaction. F.A.M., L.R., and V.P. acknowledge support by the Max Planck Society.

Declaration of interests

The authors declare no competing interests in relation to this work.

References

1. De Lange, F.P. *et al.* (2018) How do expectations shape perception? *Trends Cogn. Sci.* 22, 764–779
2. Fiser, J. *et al.* (2010) Statistically optimal perception and learning: from behavior to neural representations. *Trends Cogn. Sci.* 14, 119–130
3. Knill, D.C. and Pouget, A. (2004) The Bayesian brain: the role of uncertainty in neural coding and computation. *Trends Neurosci.* 27, 712–719
4. Rahnev, D. and Denison, R.N. (2018) Suboptimality in perceptual decision making. *Behav. Brain Sci.* 41
5. Lee, T.S. and Mumford, D. (2003) Hierarchical Bayesian inference in the visual cortex. *JOSA A* 20, 1434–1448
6. Bastos, A.M. *et al.* (2012) Canonical microcircuits for predictive coding. *Neuron* 76, 695–711.1
7. Aitchison, L. and Lengyel, M. (2017) With or without you: predictive coding and Bayesian inference in the brain. *Curr. Opin. Neurobiol.* 46, 219–227.1
8. Heeger, D.J. (2017) Theory of cortical function. *Proc. Natl. Acad. Sci.* 114, 1773–1782
9. Cao, Y. *et al.* (2019) Causal inference in the multisensory brain. *Neuron* 102, 1076–1087
10. Walsh, K.S. *et al.* (2020) Evaluating the neurophysiological evidence for predictive processing as a model of perception. *Ann. N. Y. Acad. Sci.* 1464, 242
11. Rao, R.P.N. and Ballard, D.H. (1999) Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nat. Neurosci.* 2, 79–87.1
12. Kogo, N. and Trengove, C. (2015) Is predictive coding theory articulated enough to be testable? *Front. Comput. Neurosci.* 9, 111
13. Millidge, B. *et al.* (2021) Predictive coding: a theoretical and experimental review. *arXiv* https://doi.org/10.48550/arXiv.2107.12979
14. Denève, S. and Machens, C.K. (2016) Efficient codes and balanced networks. *Nat. Neurosci.* 19, 375–382.X
15. Urbanczik, R. and Senn, W. (2014) Learning by the dendritic prediction of somatic spiking. *Neuron* 81, 521–528
16. Mikulasch, F.A. *et al.* (2021) Local dendritic balance enables learning of efficient representations in networks of spiking neurons. *Proc. Natl. Acad. Sci.* 118, e2021925118
17. Harris, K.D. and Shepherd, G.M.G. (2015) The neocortical circuit: themes and variations. *Nat. Neurosci.* 18, 170–181
18. Spratling, M.W. (2008) Predictive coding as a model of biased competition in visual attention. *Vis. Res.* 48, 1391–1408.1
19. Douglas, R.J. and Martin, K.A.C. (2004) Neuronal circuits of the neocortex. *Annu. Rev. Neurosci.* 27, 419–451.X
20. Földiak, P. (1990) Forming sparse representations by local anti-Hebbian learning. *Biol. Cybern.* 64, 165–170
21. Olshausen, B.A. and Field, D.J. (1996) Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature* 381, 607–609.1
22. Boerlin, M. *et al.* (2013) Predictive coding of dynamical variables in balanced spiking networks. *PLoS Comput. Biol.* 9, e1003258.1
23. Bill, J. *et al.* (2015) Distributed Bayesian computation and self-organized learning in sheets of spiking neurons with local lateral inhibition. *PLoS One* 10, e0134356
24. Chettih, S.N. and Harvey, C.D. (2019) Single-neuron perturbations reveal feature-specific competition in V1. *Nature* 567, 334–340
25. Brendel, W. *et al.* (2020) Learning to represent signals spike by spike. *PLoS Comput. Biol.* 16, e1007692
26. Carandini, M. and Heeger, D.J. (2012) Normalization as a canonical neural computation. *Nat. Rev. Neurosci.* 13, 51–62
27. Burg, M.F. *et al.* (2021) Learning divisive normalization in primary visual cortex. *PLoS Comput. Biol.* 17, e1009028
28. Brea, J. *et al.* (2016) Prospective coding by spiking neurons. *PLoS Comput. Biol.* 12, e1005003
29. Larkum, M. (2013) A cellular mechanism for cortical associations: an organizing principle for the cerebral cortex. *Trends Neurosci.* 36, 141–151
30. Aru, J. *et al.* (2020) Cellular mechanisms of conscious processing. *Trends Cogn. Sci.* 24, 814–825
31. Gillon, C.J. *et al.* (2021) Learning from unexpected events in the neocortical microcircuit. *bioRxiv* https://doi.org/10.1101/2021.01.15.426915
32. Hennequin, G. *et al.* (2017) Inhibitory plasticity: balance, control, and codependence. *Annu. Rev. Neurosci.* 40, 557–579
33. Sacramento, J. *et al.* (2018) Dendritic cortical microcircuits approximate the backpropagation algorithm. In *Advances in Neural Information Processing Systems*, pp. 8721–8732
34. Richards, B.A. and Lillicrap, T.P. (2019) Dendritic solutions to the credit assignment problem. *Curr. Opin. Neurobiol.* 54, 28–36
35. Whittington, J.C.R. and Bogacz, R. (2019) Theories of error back-propagation in the brain. *Trends Cogn. Sci.* 23, 235–250
36. Haider, P. *et al.* (2021) Latent equilibrium: a unified learning theory for arbitrarily fast computation with arbitrarily slow neurons. *Adv. Neural Inf. Proces. Syst.* 34, 17839–17851
37. Kadmon, J. *et al.* (2020) Predictive coding in balanced neural networks with noise, chaos and delays. *Adv. Neural Inf. Proces. Syst.* 33
38. Yoon, Y.C. (2016) LIF and simplified SRM neurons encode signals into spikes via a form of asynchronous pulse sigma–delta modulation. *IEEE Trans. Neural Netw. Learn. Syst.* 28, 1192–1205
39. Mancoo, A. *et al.* (2020) Understanding spiking networks through convex optimization. *Adv. Neural Inf. Proces. Syst.* 33, 8824–8835
40. Boerlin, M. and Denève, S. (2011) Spike-based population coding and working memory. *PLoS Comput. Biol.* 7, e1001080
41. Rullán Buxó, C.E. and Pillow, J.W. (2020) Poisson balanced spiking networks. *PLoS Comput. Biol.* 16, e1008261
42. Chalk, M. *et al.* (2016) Neural oscillations as a signature of efficient coding in the presence of synaptic delays. *Elife* 5, e13824
43. Savin, C. and Deneve, S. (2014) Spatio-temporal representations of uncertainty in spiking neural networks. *Adv. Neural Inf. Proces. Syst.* 27, 2024–2032
44. Buesing, L. *et al.* (2011) Neural dynamics as sampling: a model for stochastic computation in recurrent networks of spiking neurons. *PLoS Comput. Biol.* 7, e1002211
45. Aitchison, L. and Lengyel, M. (2016) The Hamiltonian brain: efficient probabilistic inference with excitatory-inhibitory neural circuit dynamics. *PLoS Comput. Biol.* 12, e1005186
46. Petrovici, M.A. *et al.* (2016) Stochastic inference with spiking neurons in the high-conductance state. *Phys. Rev. E* 94, 042312
47. Orbán, G. *et al.* (2016) Neural variability and sampling-based probabilistic representations in the visual cortex. *Neuron* 92, 530–543
48. Gershman, S.J. *et al.* (2012) Multistability and perceptual inference. *Neural Comput.* 24, 1–24

49. Alonso, N. and Neftci, E. (2021) Tightening the biological constraints on gradient-based predictive coding. In *International Conference on Neuromor- phic Systems 2021*, pp. 1–9
50. Keller, G.B. and Mrsic-Flogel, T.D. (2018) Predictive processing: a canonical cortical computation. *Neuron* 100, 424–435.X
51. Heilbron, M. and Chait, M. (2018) Great expectations: is there evidence for predictive coding in auditory cortex? *Neuroscience* 389, 54–73.X
52. Wehr, M. and Zador, A.M. (2003) Balanced inhibition underlies tuning and sharpens spike timing in auditory cortex. *Nature* 426, 442–446
53. Okun, M. and Lampl, I. (2008) Instantaneous correlation of excitation and inhibition during ongoing and sensory-evoked activities. *Nat. Neurosci.* 11, 535–537
54. Vogels, T.P. *et al.* (2011) Inhibitory plasticity balances excitation and inhibition in sensory pathways and memory networks. *Science* 334, 1569–1573.1
55. Korcsak-Gorzo, A. *et al.* (2022) Cortical oscillations support sampling-based computations in spiking neural networks. *PLoS Comput. Biol.* 18, e1009753
56. Liu, G. (2004) Local structural balance and functional interaction of excitatory and inhibitory synapses in hippocampal dendrites. *Nat. Neurosci.* 7, 373–379.1
57. Iascone, D.M. *et al.* (2020) Whole-neuron synaptic mapping reveals spatially precise excitatory/inhibitory balance limiting dendritic and somatic spiking. *Neuron* 106, 566–578
58. Spruston, N. *et al.* (2016) Principles of dendritic integration. *Dendrites* 351, 361–364
59. Müllner, F.E. *et al.* (2015) Precision of inhibition: dendritic inhibition by individual GABAergic synapses on hippocampal pyramidal cells is confined in space and time. *Neuron* 87, 576–589
60. Field, R.E. *et al.* (2020) Heterosynaptic plasticity determines the set point for cortical excitatory-inhibitory balance. *Neuron* 106, 842–854
61. Chen, S.X. *et al.* (2015) Subtype-specific plasticity of inhibitory circuits in motor cortex during motor learning. *Nat. Neurosci.* 18, 1109–1115
62. Hu, H.Y. *et al.* (2019) Endocannabinoid signaling mediates local dendritic coordination between excitatory and inhibitory synapses. *Cell Rep.* 27, 666–675
63. Bourne, J.N. and Harris, K.M. (2011) Coordination of size and number of excitatory and inhibitory synapses results in a balanced structural plasticity along mature hippocampal CA1 dendrites during LTP. *Hippocampus* 21, 354–373.1
64. D'amour, J.A. and Froemke, R.C. (2015) Inhibitory and excitatory spiketiming-dependent plasticity in the auditory cortex. *Neuron* 86, 514–528
65. Herstel, L.J. and Wierenga, C.J. (2021) Network control through coordinated inhibition. *Curr. Opin. Neurobiol.* 67, 34–41
66. Artola, A. *et al.* (1990) Different voltage-dependent thresholds for inducing long-term depression and long-term potentiation in slices of rat visual cortex. *Nature* 347, 69–72
67. Lisman, J. and Spruston, N. (2005) Postsynaptic depolarization requirements for LTP and LTD: a critique of spike timing-dependent plasticity. *Nat. Neurosci.* 8, 839–841
68. Lisman, J. and Spruston, N. (2010) Questions about STDP as a general model of synaptic plasticity. *Front. Synaptic Neurosci.* 2, 140
69. Higley, M.J. (2014) Localized GABAergic inhibition of dendritic Ca 2+ signalling. *Nat. Rev. Neurosci.* 15, 567–572
70. Augustine, G.J. *et al.* (2003) Local calcium signaling in neurons. *Neuron* 40, 331–346
71. Shouval, H.Z. *et al.* (2002) A unified model of NMDA receptor-dependent bidirectional synaptic plasticity. *Proc. Natl. Acad. Sci.* 99, 10831–10836
72. Clopath, C. and Gerstner, W. (2010) Voltage and spike timing interact in STDP–a unified model. *Front. Synaptic Neurosci.* 2, 25
73. Clopath, C. *et al.* (2010) Connectivity reflects coding: a model of voltage-based STDP with homeostasis. *Nat. Neurosci.* 13, 344
74. Meredith, R.M. *et al.* (2003) Maturation of long-term potentiation induction rules in rodent hippocampus: role of GABAergic inhibition. *J. Neurosci.* 23, 11142–11146
75. Hayama, T. *et al.* (2013) GABA promotes the competitive selection of dendritic spines by controlling local Ca 2+ signaling. *Nat. Neurosci.* 16, 1409–1416
76. Wang, L. and Maffei, A. (2014) Inhibitory plasticity dictates the sign of plasticity at excitatory synapses. *J. Neurosci.* 34, 1083–1093
77. Steele, P.M. and Mauk, M.D. (1999) Inhibitory control of LTP and LTD: stability of synapse strength. *J. Neurophysiol.* 81, 1559–1566
78. Sjöström, P.J. and Hausser, M (2006) A cooperative switch determines the sign of synaptic plasticity in distal dendrites of neocortical pyramidal neurons. *Neuron* 51, 227–238
79. Letzkus, J.J. *et al.* (2006) Learning rules for spike timing-dependent plasticity depend on dendritic synapse location. *J. Neurosci.* 26, 10420–10429
80. Froemke, R.C. *et al.* (2010) Dendritic synapse location and neocortical spiketiming-dependent plasticity. *Front. Synaptic Neurosci.* 2, 29
81. Yoshimura, Y. and Callaway, E.M. (2005) Fine-scale specificity of cortical networks depends on inhibitory cell type and connectivity. *Nat. Neurosci.* 8, 1552–1559
82. Znamenskiy, P. *et al.* (2018) Functional selectivity and specific connectivity of inhibitory neurons in primary visual cortex. *bioRxiv* https://doi.org/10.1101/294835
83. Petersen, C.C.H. and Crochet, S. (2013) Synaptic computation and sensory processing in neocortical layer 2/3. *Neuron* 78, 28–48
84. Avermann, M. *et al.* (2012) Microcircuits of excitatory and inhibitory neurons in layer 2/3 of mouse barrel cortex. *J. Neurophysiol.* 107, 3116–3134
85. Kubota, Y. (2014) Untangling GABAergic wiring in the cortical microcircuit. *Curr. Opin. Neurobiol.* 26, 7–14
86. Tremblay, R. *et al.* (2016) GABAergic interneurons in the neocortex: from cellular properties to circuits. *Neuron* 91, 260–292
87. Ferguson, B.R. and Gao, W.-J. (2018) PV interneurons: critical regulators of E/I balance for prefrontal cortex-dependent behavior and psychiatric disorders. *Front. Neural Circ.* 12, 37
88. Cardin, J.A. *et al.* (2009) Driving fast-spiking cells induces gamma rhythm and controls sensory responses. *Nature* 459, 663–667
89. David, F. *et al.* (2022) Layer-specific stimulations of parvalbumin-positive cortical interneurons in mice entrain brain rhythms to different frequencies. *bioRxiv* https://doi.org/10.1101/2021.03.31.437894
90. Markram, H. *et al.* (2004) Interneurons of the neocortical inhibitory system. *Nat. Rev. Neurosci.* 5, 793–807
91. Zhang, S. *et al.* (2014) Long-range and local circuits for top-down modulation of visual cortex processing. *Science* 345, 660–665
92. Adesnik, H. *et al.* (2012) A neural circuit for spatial summation in visual cortex. *Nature* 490, 226–231
93. Gentet, L.J. *et al.* (2012) Unique functional properties of somatostatin-expressing GABAergic neurons in mouse barrel cortex. *Nat. Neurosci.* 15, 607–612
94. Schuman, B. *et al.* (2021) Neocortical layer 1: an elegant solution to top-down and bottom-up integration. *Annu. Rev. Neurosci.* 44, 221–252
95. Lee, S. *et al.* (2013) A disinhibitory circuit mediates motor integration in the somatosensory cortex. *Nat. Neurosci.* 16, 1662–1670
96. Yu, F. *et al.* (2014) A cortical circuit for gain control by behavioral state. *Cell* 156, 1139–1152
97. Pi, H.-J. *et al.* (2013) Cortical interneurons that specialize in disinhibitory control. *Nature* 503, 521–524
98. Garrett, M. *et al.* (2020) Experience shapes activity dynamics and stimulus coding of VIP inhibitory cells. *Elife* 9, e50340
99. Letzkus, J.J. *et al.* (2015) Disinhibition, a circuit mechanism for associative learning and memory. *Neuron* 88, 264–276
100. Harris, K.D. and Mrsic-Flogel, T.D. (2013) Cortical connectivity and sensory coding. *Nature* 503, 51–58
101. Shepherd, G.M. and Rowe, T.B. (2017) Neocortical lamination: insights from neuron types and evolutionary precursors. *Front. Neuroanat.* 11, 100

102. Dugas-Ford, J. *et al.* (2012) Cell-type homologies and the origins of the neocortex. *Proc. Natl. Acad. Sci.* 109, 16974–16979
103. Karten, H.J. (2013) Neocortical evolution: neuronal circuits arise independently of lamination. *Curr. Biol.* 23, R12–R15
104. Briscoe, S.D. and Ragsdale, C.W. (2018) Homology, neocortex, and the evolution of developmental mechanisms. *Science* 362, 190–193
105. Mei, J. *et al.* (2022) Informing deep neural networks by multiscale principles of neuromodulatory systems. *Trends Neurosci.* 45, 237–250
106. Allaway, K.C. *et al.* (2020) Cellular birthdate predicts laminar and regional cholinergic projection topography in the forebrain. *Elife* 9, e63249
107. Urban-Ciecko, J. *et al.* (2018) Precisely timed nicotinic activation drives SST inhibition in neocortical circuits. *Neuron* 97, 611–625
108. Brombas, A. *et al.* (2014) Activity-dependent modulation of layer 1 inhibitory neocortical circuits by acetylcholine. *J. Neurosci.* 34, 1932–1941
109. Kruglikov, I. and Rudy, B. (2008) Perisomatic GABA release and thalamocortical integration onto neocortical excitatory cells are regulated by neuromodulators. *Neuron* 58, 911–924
110. Moran, R.J. *et al.* (2013) Free energy, precision and learning: the role of cholinergic neuromodulation. *J. Neurosci.* 33, 8227–8236
111. Iglesias, S. *et al.* (2021) Cholinergic and dopaminergic effects on prediction error and uncertainty responses during sensory associative learning. *Neuroimage* 226, 117590
112. Barron, H.C. *et al.* (2020) Prediction and memory: a predictive coding account. *Prog. Neurobiol.* 192, 101821
113. Bolz, J. and Gilbert, C.D. (1986) Generation of end-inhibition in the visual cortex via interlaminar connections. *Nature* 320, 362–365
114. Nassi, J.J. *et al.* (2013) Corticocortical feedback contributes to surround suppression in V1 of the alert primate. *J. Neurosci.* 33, 8504–8517
115. Boutin, V. *et al.* (2021) Sparse deep predictive coding captures contour integration capabilities of the early visual system. *PLoS Comput. Biol.* 17, e1008629
116. Spratling, M.W. (2010) Predictive coding as a model of response properties in cortical area V1. *J. Neurosci.* 30, 3531–3543
117. Liang, H. *et al.* (2017) Interactions between feedback and lateral connections in the primary visual cortex. *Proc. Natl. Acad. Sci.* 114, 8637–8642
118. Marques, T. *et al.* (2018) The functional organization of cortical feedback inputs to primary visual cortex. *Nat. Neurosci.* 21, 757–764
119. Nurminen, L. *et al.* (2018) Top-down feedback controls spatial summation and response amplitude in primate visual cortex. *Nat. Commun.* 9, 1–13
120. Zmarz, P. and Keller, G.B. (2016) Mismatch receptive fields in mouse visual cortex. *Neuron* 92, 766–772
121. Jordan, R. and Keller, .B. (2020) Opposing influence of top-down and bottom-up input on excitatory layer 2/3 neurons in mouse primary visual cortex. *Neuron* 108, 1194–1206
122. Eliades, S.J. and Wang, X. (2008) Neural substrates of vocalization feedback monitoring in primate auditory cortex. *Nature* 453, 1102–1106
123. Fiser, A. *et al.* (2016) Experience-dependent spatial expectations in mouse visual cortex. *Nat. Neurosci.* 19, 1658–1664
124. Feuerriegel, D. *et al.* (2021) Evaluating the evidence for expectation suppression in the visual system. *Neurosci. Biobehav. Rev.* 126, 368–381
125. Mikulasch, F.A. *et al.* (2022) Visuomotor mismatch responses as a hallmark of explaining away in causal inference. *bioRxiv* https://doi.org/10.1101/2022.04.07.486697
126. Garner, A.R. and Keller, G.B. (2022) A cortical circuit for audio-visual predictions. *Nat. Neurosci.* 25, 98–105
127. Keller, G.B. and Hahnloser, R.H.R. (2009) Neural processing of auditory feedback during vocal practice in a songbird. *Nature* 457, 187–190
128. Młynarski, W.F. and Hermundstad, A.M. (2018) Adaptive coding for dynamic sensory inference. *Elife* 7, e32055
129. Kubota, Y. *et al.* (2007) Neocortical inhibitory terminals innervate dendritic spines targeted by thalamocortical afferents. *J. Neurosci.* 27, 1139–1150
130. Alexander Bae, J. *et al.* (2021) Functional connectomics spanning multiple areas of mouse visual cortex. *bioRxiv* https://doi.org/10.1101/2021.07.28.454025
131. Keller, G.B. *et al.* (2012) Sensorimotor mismatch signals in primary visual cortex of the behaving mouse. *Neuron* 74, 809–815
132. Hamm, J.P. *et al.* (2021) Cortical ensembles selective for context. *Proc. Natl. Acad. Sci.* 118, e2026179118
133. Schultz, W. (2016) Dopamine reward prediction-error signalling: a two-component response. *Nat. Rev. Neurosci.* 17, 183–195
134. Bogacz, R. (2017) A tutorial on the free-energy framework for modelling perception and learning. *J. Math. Psychol.* 76, 198–211
135. Millidge, B. *et al.* (2020) Relaxing the constraints on predictive coding models. *arXiv* https://doi.org/10.48550/arXiv.2010.01047
136. Kappel, D. *et al.* (2014) STDP installs in winner-take-all circuits an online approximation to hidden Markov model learning. *PLoS Comput. Biol.* 10, e1003511.X
137. Gerstner, W. *et al.* (2014) *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*, Cambridge University Press
138. Constantinople, C.M. and Bruno, R.M. (2013) Deep cortical layers are activated directly by thalamus. *Science* 340, 1591–1594
139. Echeveste, R. *et al.* (2020) Cortical-like dynamics in recurrent circuits optimized for sampling-based probabilistic inference. *Nat. Neurosci.* 23, 1138–1149

4 Visuomotor mismatch responses as a hallmark of explaining away in causal inference

Published at Neural computation 35.1 (2022): 27-37.
DOI 10.1162/neco_a_01546
Supplementary Material
Source Code github.com/Priesemann-Group/mismatch_responses

Contributions Conceptualization, Investigation, Writing - Original Draft

This work also constitutes a chapter in LR's PhD thesis. I proposed the project idea, wrote the code, created the figures and wrote the initial manuscript. Together with LR we refined the model and finalized the manuscript.

LETTER Communicated by Georg Keller

Visuomotor Mismatch Responses as a Hallmark of Explaining Away in Causal Inference

Fabian A. Mikulasch
fabian.mikulasch@ds.mpg.de
Lucas Rudelt
lucas.rudelt@ds.mpg.de
Max-Planck-Institute for Dynamics and Self-Organization, Göttingen 37077, Germany

Viola Priesemann
viola.priesemann@ds.mpg.de
Max-Planck-Institute for Dynamics and Self-Organization, Göttingen 37077, Germany; Bernstein Center for Computational Neuroscience, Göttingen 37073, Germany; and Department of Physics, Georg-August University, Göttingen, Germany

How are visuomotor mismatch responses in primary visual cortex embedded into cortical processing? We here show that mismatch responses can be understood as the result of a cooperation of motor and visual areas to jointly explain optic flow. This cooperation requires that optic flow is not explained redundantly by both areas, meaning that optic flow inputs to V1 that are predictable from motor neurons should be canceled (i.e., explained away). As a result, neurons in V1 represent only external causes of optic flow, which could allow the animal to easily detect movements that are independent of its own locomotion. We implement the proposed model in a spiking neural network, where coding errors are computed in dendrites and synaptic weights are learned with voltage-dependent plasticity rules. We find that both positive and negative mismatch responses arise, providing an alternative to the prevailing idea that visuomotor mismatch responses are linked to dedicated neurons for error computation. These results also provide a new perspective on several other recent observations of cross-modal neural interactions in cortex.

1 Introduction

In recent years, several experiments confirmed the surprising result that locomotion has a considerable impact on neural activity in visual cortex (Keller, Bonhoeffer, & Hübener, 2012; Niell & Stryker, 2010; Saleem, Ayaz, Jeffery, Harris, & Carandini, 2013). Some of these experiments managed to show that pyramidal cells in layer 2/3 of primary visual cortex (V1)

Neural Computation **35, 27–37** (2023)
https://doi.org/10.1162/neco_a_01546

compute a visuomotor mismatch, that is, the difference between presented optic flow and optic flow predicted from the animal's locomotion (Jordan & Keller, 2020; Zmarz & Keller, 2016).

An important question to ask is what purpose these computations fulfill in cortex and how they should be interpreted. A widespread idea is that these mismatch responses are indicative for the canonical computation of "error neurons" (Jordan & Keller, 2020) that occurs in a hierarchical predictive coding model of cortex (Rao & Ballard, 1999). However, as of yet, no formal model of how exactly these mismatch responses could be embedded into cortical hierarchical processing has been presented.

Here we argue for a different interpretation of mismatch responses in visual cortex as a result of explaining away. First, we explain the idea of this effect and demonstrate it in simulations, where spiking neurons learn to encode simple optic flow stimuli and locomotion. We then discuss how this interpretation differs from the prevailing interpretation in terms of dedicated error neurons.

2 Theory

The core idea of our model is that visual and motor neurons jointly explain the optic flow the animal perceives. This does not necessarily require that motor neurons are actively driven by optic flow stimuli; it means only that the explanation of optic flow (in the internal model of the animal) is distributed over different populations. Formally, we can state this with the following model (see Figure 1A), where it is assumed that optic flow $\mathbf{r}^{\text{flow}}$ can be reconstructed as a linear sum of the activity of a visual population $\mathbf{r}^{\text{V1}}$ and a motor population $\mathbf{r}^{\text{M2}}$ plus some gaussian noise $\mathbf{n}^{\text{flow}}$ with variance σ^2_{flow}:

$$\mathbf{r}^{\text{flow}} \overset{!}{=} D^{\text{flow}\leftarrow\text{V1}}\mathbf{r}^{\text{V1}} + D^{\text{flow}\leftarrow\text{M2}}\mathbf{r}^{\text{M2}} + \mathbf{n}^{\text{flow}}. \tag{2.1}$$

Here, $D^{\text{b}\leftarrow\text{a}}$ are decoding matrices that decode activity from a to b. To constrain the activity of the motor population $\mathbf{r}^{\text{M2}}$, we furthermore require that it encodes the locomotion of the animal $\mathbf{r}^{\text{move}}$ according to a linear model,

$$\mathbf{r}^{\text{move}} \overset{!}{=} D^{\text{move}\leftarrow\text{M2}}\mathbf{r}^{\text{M2}} + \mathbf{n}^{\text{move}}. \tag{2.2}$$

From the model of optic flow, equation 2.1, it is directly visible that (on average) activity in visual neurons should be proportional to the difference between optic flow and predictions from motor neurons:

$$\mathbf{r}^{\text{V1}} \overset{!}{\propto} \mathbf{r}^{\text{flow}} - D^{\text{flow}\leftarrow\text{M2}}\mathbf{r}^{\text{M2}}. \tag{2.3}$$

Hence, as a result of the cooperation of motor and visual neurons to explain optic flow, motor neurons should cancel predictable (i.e., self-generated) optic flow in visual neurons via efference copies. In terms of inference, this is called explaining away, since activity of one area must not explain aspects of the input that are already explained by the other area (Moreno-Bote & Drugowitsch, 2015).

3 Results

To illustrate the emergence of mismatch responses via explaining away, we simulated inference and learning in this model (see equations 2.1 and 2.2) in a previously proposed framework of population coding with spiking neurons (Mikulasch, Rudelt, & Priesemann, 2021) (However, any neural implementation of the model should yield similar results). In this framework, explaining away is implemented via connections between and within neural populations (see Figure 1B), which cancel (i.e., balance) inputs on a neuron's dendrites that can be predicted from the activity of other neurons. For the encoding of optic flow as in equation 2.3, this framework therefore requires that motor neurons cancel optic flow inputs on the dendrites of visual neurons. This can be achieved by learning a balance on neural dendrites via voltage-dependent plasticity (Mikulasch et al., 2021). Visual neurons can then learn to efficiently encode the residual visual flow via another voltage-dependent plasticity rule.

Using this model, we recreated the visuomotor mismatch experiment of Jordan and Keller (2020) in a simplified manner (see Figure 1C). The task of the network was to encode locomotion and optic flow, which were presented simultaneously (for details about the data creation, see section 5). We used simple locomotion signals that indicated a turn to the left or right. Optic flow consisted of the activity of three receptors, which indicated speed and direction of optic flow and were correlated with the locomotion signal (see Figure 1D). The idea of this setup is that locomotion can partly predict the optic flow; hence, motor neurons should cancel this predictable component in the dendrites of visual neurons in V1.

In the simulations, the network first learned to represent locomotion and optic flow by adapting feedforward weights from sensory inputs, as well as weights within and between populations. After learning, we tested the responses of V1 neurons with conflicting stimuli (where optic flow in the center receptor did not match the prediction from locomotion) or nonconflicting stimuli. As we expected, we indeed found neurons that specifically reacted to a mismatch between optic flow and prediction (see Figure 1E). These neurons were mostly silent when motor prediction and optic flow input matched, but during a prediction mismatch, they corrected the erroneous joint representation and encoded either a positive or a negative deviation.

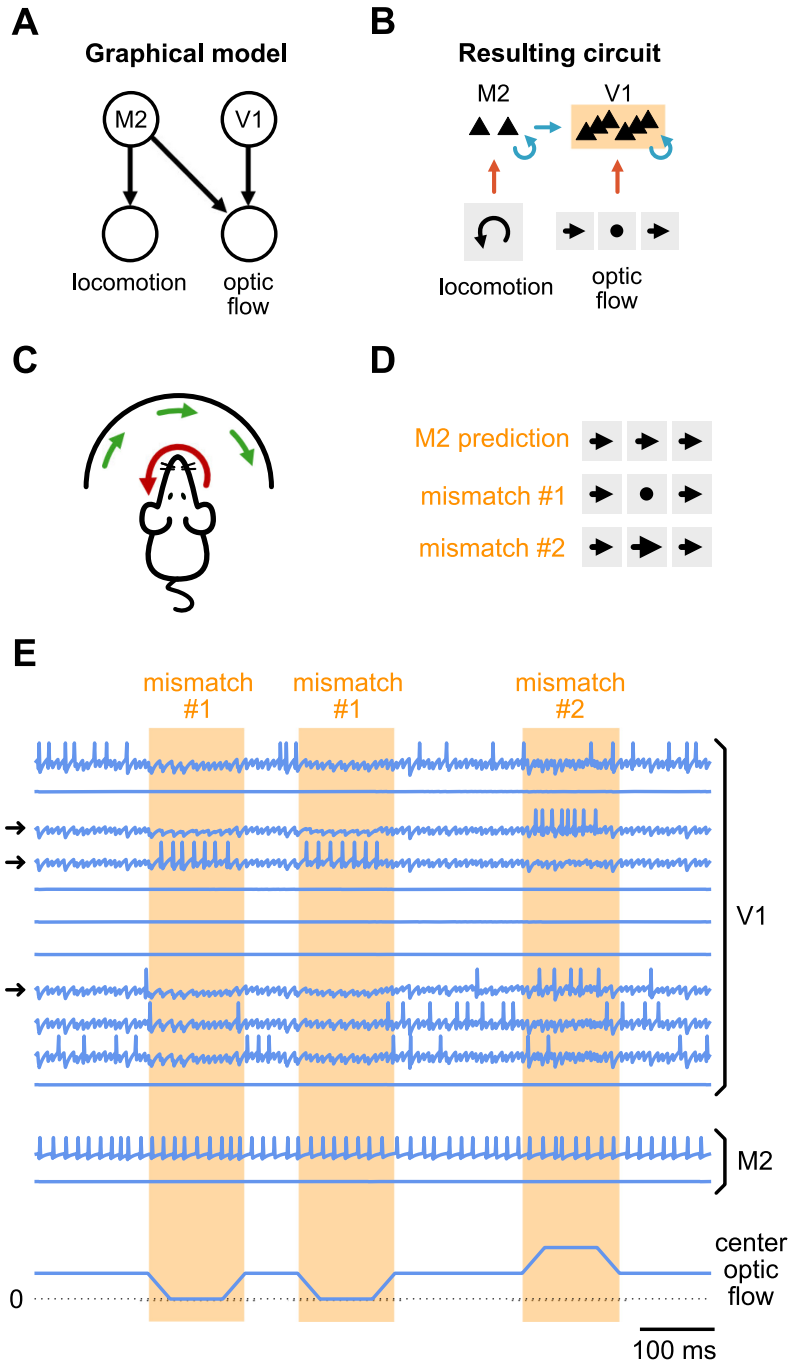

Figure 1: Mismatch responses emerge in spiking neural networks from explaining away. (A) Graphical model representation of the assumed model of optic flow and locomotion (see equations 2.1 and 2.2). Motor neurons (M2) and visual neurons (V1) explain optic flow together, while M2 also explains the locomotion of the animal. This introduces explaining-away effects between motor and visual neurons. (B) Neural circuit that implements causal inference in the model in panel A. Here, explaining away is implemented via connections between and within neural populations (blue arrows), which learn to cancel (i.e., balance) sensory inputs that can already be explained from the activity of other neurons. Because motor neurons also explain optic flow, connections from M2

4 Discussion

In any causal inference problem where multiple competing explanations exist for the same observation, having evidence for one of the explanations will reduce the probability of the other explanations because the probable causes explain away the data. Here, we showed that explaining away manifests as visuomotor mismatch responses in a model where motor and visual areas jointly infer the underlying causes of optic flow. In particular, if an animal is moving, this can partially explain the optic flow, and thus locomotion competes with other (external) explanations that are represented in visual areas. In this case, these external causes are less probable, and hence activity in visual areas should be suppressed if it is predictable from locomotion. Vice versa, a mismatch between prediction and optic flow indicates an external motion and should result in additional activity in visual neurons that encodes this difference. Disentangling the potential explanations of the perceived optic flow in this way could allow the animal to rapidly identify objects that move independently of its own locomotion (e.g., as similarly found in Schneider, Sundararajan, & Mooney, 2018).

Importantly, the results presented here do not critically depend on the specific model we chose. Here, we assumed that optic flow can be linearly decoded from a spiking representation in visual and motor areas, which allows us to derive analytically the interactions that are expected from the cooperation between areas. However, as outlined above, explaining away is a very general effect that occurs in inference in graphical models with converging arrows (Bill et al., 2015; see Figure 1A). Thus, qualitatively

---

to V1 learn to cancel optic flow inputs in V1. To find an efficient encoding, connections from sensory inputs (red arrows) are learned via voltage-dependent plasticity (Mikulasch et al., 2021). (C) Experimental setup to induce mismatch responses (Jordan & Keller, 2020). A mouse is placed in a virtual environment, while the head is fixed. Egomotion of the mouse (red arrow) results in visual flow (green arrows) that is displayed on a screen. Note that for simplicity of presentation, we here depict locomotion of the mouse as rotations, while the original experiment employed translations, which makes no conceptual difference on our level of modeling. (D) Sample optic flow stimuli that are presented in our model. A rotation to the left would predict uniform visual flow to the right. Two mismatch conditions are also presented, where center optic flow is slower (mismatch #1) or faster (mismatch #2) than expected. (E) Simulation of optic flow mismatch responses with spiking neurons. The mouse turns to the left, which is encoded by motor neurons (M2). Visual neurons (V1) with mismatch responses are indicated by arrows. After learning, neurons emerge that are active for faster or slower optic flow than expected, which is similarly found in experiment (Jordan & Keller, 2020). These responses correct the joint representation of optic flow in case it is not fully predictable from locomotion.

similar results are also expected in other models of neural coding that are not linear (Bill et al., 2015; Heeger, 2017).

Our model specifically applies to layer 2/3 pyramidal cells in V1, where mismatch responses first seem to emerge (Jordan & Keller, 2020). This is compatible with the previously proposed idea that layer 2/3 neurons compute a representation of sensory inputs that is consistent with the representation in other cortical areas (Douglas & Martin, 2004; Mikulasch et al., 2022). In this proposal, deeper cortical layers would have distinctly different purposes: layer 4 neurons would relay optic flow input from thalamus (possibly implementing a pre-processing), while layer 5 neurons are speculated to participate in output selection and long-range communication of the microcircuit (Douglas & Martin, 2004). Indeed, while V1 layer 5 neurons are responsive to both optic flow and locomotion, they do not compute a difference, but rather a weighted sum of visual and motor signals (Jordan & Keller, 2020; Saleem et al., 2013).

Previously, the observed mismatch responses have been interpreted as the responses of "error" neurons in a predictive processing context (Keller & Mrsic-Flogel, 2018). Our model and this previous interpretation mainly differ in how they expect the neural output to be processed further. While in our model the output of representation neurons could be used directly as input for other computations, the concept of error neurons implies additional representation neurons that integrate their outputs for further processing (Bastos et al., 2012; Keller & Mrsic-Flogel, 2018). So far, no formal predictive coding model for mismatch responses has been presented, but two plausible graphical models can be considered (explained in Figure 2). An open problem for these predictive coding-based explanations is that conclusive evidence for the additional representation neurons they imply is still missing (see Figures 2B and 2D for details). Alternatively, mismatch neurons in V1 could be employed in supervised behavioral learning using a predictive processing scheme (Jordan & Rumelhart, 1992; Keller & Mrsic-Flogel, 2018). However, also for this theory a formal model here of how the observed mismatch responses could be embedded into the learning algorithm has yet to be presented. Thus, while mismatch responses can be interpreted as error neuron responses in predictive processing, it is less clear what specific role they could play in cortical computations.

An important difference that arises between models with error neurons and our model lies in their expectations of how mismatch responses would evolve over time. In error neurons, further processing of error neuron output by representation neurons should lead to a top-down mediated suppression of activity (Bastos et al., 2012). While Jordan and Keller (2020) did not present mismatches long enough to determine such a decay, in a similar experiment, movement onset type feedback mismatch responses have been observed to decay over time, on the order of hundreds of milliseconds (Keller et al., 2012). This is consistent with the idea of error neurons, but would imply a relatively slow processing of visual information in mice. In

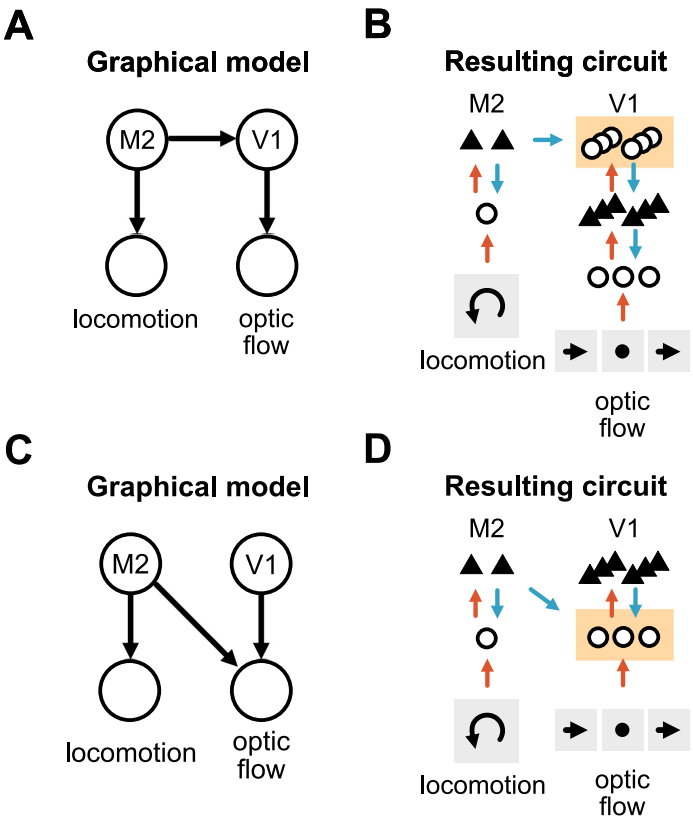

Figure 2: How could mismatch responses of dedicated error neurons be embedded into a graphical model? (A) One possibility would be that in a hierarchical model of inference, motor information in M2 improves the representation of visual flow in V1 by providing predictions. (B) This model implies the existence of error neurons in V1 (orange box) that compare representation neuron activity in V1 to predictions from locomotion. Error neurons would respond to visuomotor mismatch, while V1 representation neurons would encode optic flow. Neurons that integrate both visual and motor information like these representation neurons indeed exist in layer 5 (Jordan & Keller, 2020). However, contrary to this picture, layer 5 neurons are typically expected to receive bottom-up input from layer 2/3 (i.e., mismatch neurons) and should not be the main driver of activity in layer 2/3 (Bastos et al., 2012; Jordan & Keller, 2020). (C) Another possibility is that, as in our model, both V1 and M2 explain optic flow jointly. (D) In this case the model implies error neurons in V1 that directly compare optic flow and motor information (orange box). The difference to our model is that, additionally, representation neurons would be required that integrate and cancel the activity of error neurons (similar to the circuit proposed in Keller & Mrsic-Flogel, 2018). Note that these representation neurons would perform the same computation as the neurons in our model (computing a representation of the residual flow), but would be updated on the timescale of error neuron responses, that is, hundreds of milliseconds (Keller et al., 2012), compared to a few milliseconds in our model. In panels B and D, only connections are shown that are essential for mismatch responses.

contrast, our model does not show decaying mismatch responses, and additional mechanisms, such as synaptic depression (Chance, Nelson, & Abbott, 1998) or other adaptation processes (Pozzorini, Naud, Mensi, & Gerstner, 2013), would be required to explain this finding. Interestingly, this might allow distinguishing these two theories experimentally, as they expect different sources for the decay of mismatch responses: error neuron theories would expect a top-down mediated suppression, while in our theory, we would expect the decay to result from population-internal processes or the preprocessing of bottom-up inputs.

To summarize, we have shown how mismatch responses emerge in causal inference in a plastic spiking neural network. Mechanistically, our model is similar to previous models, which showed that mismatch responses emerge when connections between neural populations learn to establish a balance on neural dendrites (Hertäg & Sprekeler, 2020). Based on our results, we argue that experimentally observed mismatch responses are not a signature of dedicated error neurons, but instead arise in representation neurons as a hallmark of explaining away, when multiple areas explain the same sensory inputs. This interpretation of mismatch responses can also be applied to other such observations, for example audiovisual suppression in V1 (Garner & Keller, 2021), suppression of auditory responses predictable from locomotion (Schneider et al., 2018), or mismatch responses in the tactile (Ayaz et al., 2019) and auditory modality (Eliades & Wang, 2008; Keller & Hahnloser, 2009). If correct, this could signify that cortex, already at the earliest levels of processing, encodes sensory stimuli as a whole in order to integrate sensory information in multiple modalities.

5 Methods

Spiking neurons were modeled with the model presented in Mikulasch, Rudelt, and Priesemann (2021). Neurons were updated in discrete time steps $\delta = 0.2\,\text{ms}$. Feedforward weights $F^{b\leftarrow a} \approx D^{a\leftarrow b^T}$ from signals to populations were learned online with voltage-dependent learning rules (Mikulasch et al., 2021). For weights $W^{b\leftarrow a}$ within and between populations, we used an analytical solution for simplicity:

$$W^{M2\leftarrow M2} = -F^{M2\leftarrow move} D^{move\leftarrow M2}, \tag{5.1}$$

$$W^{V1\leftarrow M2} = -F^{V1\leftarrow flow} D^{flow\leftarrow M2}, \tag{5.2}$$

$$W^{V1\leftarrow V1} = -F^{V1\leftarrow flow} D^{flow\leftarrow V1}. \tag{5.3}$$

Previously we showed that this analytical solution can be well approximated by learning a tight balance on neural dendrites (Mikulasch et al., 2021). Note that in this model, neurons inhibit each other directly, but similar neural codes can be obtained by learning a balance that is mediated by inhibitory interneurons (Brendel, Bourdoukan, Vertechi, Machens, &

Denéve, 2020; Hertäg & Sprekeler, 2020). Spiking rates of V1 and M2 neurons were homeostatically regulated to 8 Hz and 30 Hz, respectively. After learning, we probed neural activity with plasticity turned off to produce the results in Figure 1. Note that plasticity in our model only acts on long timescales, and thus can be ignored when testing the behavior of the network on a few input patterns.

To simulate the experiment in Jordan and Keller (2020), we created pairs of locomotion and optic flow signals. Signals were each presented for 100 ms before switching to the next pair. Locomotion is represented by a one-dimensional signal, where -1 indicated a turn to the left, 0 no movement, and 1 a turn to the right, each occurring with probability $p = 1/3$. Optic flow is represented by a three-dimensional signal, where all values were initially set to 1, 0, or -1 for each movement condition, respectively. We then added optic flow that could not be predicted from locomotion. Every dimension had the chance to be increased by 1 or decreased by 1 (with probability $p = 0.1$ each). After the creation of these two vectors, we ensured that only positive values were presented to the network, by doubling dimensions, making a negative copy to the new dimensions, and rectifying the signal.

Code for reproducing the simulations can be found at https://github.com/Priesemann-Group/mismatch_responses.

Acknowledgments

We thank Georg Keller and the Priesemann Lab, especially Matthias Loidolt, for helpful discussions and comments on the manuscript. F.A.M. and L.R. were funded by the German Research Foundation (DFG), SFB1286. V.P. received support from the SFB1528, Cognition of Interaction.

Author Contributions

F.A.M.: Conceptualization, investigation, writing (original draft). L.R.: Investigation, writing (review and editing). V.P.: Supervision, writing (review and editing).

References

Ayaz, A., Stäuble, A., Hamada, M., Wulf, M.-A., Saleem, A. B., & Helmchen, F. (2019). Layer-specific integration of locomotion and sensory information in mouse barrel cortex. *Nature Communications*, *10*(1), 1–14. 10.1038/s41467-019-10564-8

Bastos, A. M., Usrey, W. M., Adams, R. A., Mangun, G. R., Fries, P., & Friston, K. J. (2012). Canonical microcircuits for predictive coding. *Neuron*, *76*(4), 695–711. 10.1016/j.neuron.2012.10.038

Bill, J., Buesing, L., Habenschuss, S., Nessler, B., Maass, W., & Legenstein, R. (2015). Distributed Bayesian computation and self-organized learning in sheets of

spiking neurons with local lateral inhibition. *PLOS One*, *10*(8), e0134356. 10.1371/journal.pone.0134356

Brendel, W., Bourdoukan, R., Vertechi, P., Machens, C. K., & Denéve, S. (2020). Learning to represent signals spike by spike. *PLOS Computational Biology*, *16*(3), e1007692. 10.1371/journal.pcbi.1007692

Chance, F. S., Nelson, S. B., & Abbott, L. F. (1998). Synaptic depression and the temporal response characteristics of V1 cells. *Journal of Neuroscience*, *18*(12), 4785–4799. 10.1523/JNEUROSCI.18-12-04785.1998

Douglas, R. J., & Martin, K. A. (2004). Neuronal circuits of the neocortex. *Annual Review of Neuroscience*, *27*(1), 419–451. 10.1146/annurev.neuro.27.070203.144152

Eliades, S. J., & Wang, X. (2008). Neural substrates of vocalization feedback monitoring in primate auditory cortex. *Nature*, *453*(7198), 1102–1106. 10.1038/nature06910

Garner, A. R., & Keller, G. B. (2021). A cortical circuit for audio-visual predictions. *Nature Neuroscience*, *25*, 1–8. 10.1038/s41593-021-00974-7

Heeger, D. J. (2017). Theory of cortical function. In *Proceedings of the National Academy of Sciences*, *114*(8), 1773–1782. 10.1073/pnas.1619788114

Hertäg, L., & Sprekeler, H. (2020). Learning prediction error neurons in a canonical interneuron circuit. *eLife*, *9*, e57541. 10.7554/eLife.57541

Jordan, M. I., & Rumelhart, D. E. (1992). Forward models: Supervised learning with a distal teacher. *Cognitive Science*, *16*(3), 307–354. 10.1207/s15516709cog1603_1

Jordan, R., & Keller, G. B. (2020). Opposing influence of top-down and bottom-up input on excitatory layer 2/3 neurons in mouse primary visual cortex. *Neuron*, *108*(6), 1194–1206. 10.1016/j.neuron.2020.09.024

Keller, G. B., Bonhoeffer, T., & Hübener, M. (2012). Sensorimotor mismatch signals in primary visual cortex of the behaving mouse. *Neuron*, *74*(5), 809–815. 10.1016/j.neuron.2012.03.040

Keller, G. B., & Hahnloser, R. H. (2009). Neural processing of auditory feedback during vocal practice in a songbird. *Nature*, *457*(7226), 187–190. 10.1038/nature07467

Keller, G. B., & Mrsic-Flogel, T. D. (2018). Predictive processing: A canonical cortical computation. *Neuron*, *100*(2), 424–435. 10.1016/j.neuron.2018.10.003

Mikulasch, F. A., Rudelt, L., & Priesemann, V. (2021). Local dendritic balance enables learning of efficient representations in networks of spiking neurons. In *Proceedings of the National Academy of Sciences*, *118*(50). 10.1073/pnas.2021925118

Mikulasch, F. A., Rudelt, L., Wibral, M., & Priesemann, V. (2022). *Dendritic predictive coding: A theory of cortical computation with spiking neurons*. arXiv:2205.05303.

Moreno-Bote, R., & Drugowitsch, J. (2015). Causal inference and explaining away in a spiking network. *Scientific Reports*, *5*(1), 1–18. 10.1038/srep17531

Niell, C. M., & Stryker, M. P. (2010). Modulation of visual responses by behavioral state in mouse visual cortex. *Neuron*, *65*(4), 472–479. 10.1016/j.neuron.2010.01.033

Pozzorini, C., Naud, R., Mensi, S., & Gerstner, W. (2013). Temporal whitening by power-law adaptation in neocortical neurons. *Nature Neuroscience*, *16*(7), 942–948. 10.1038/nn.3431

Rao, R. P. N., & Ballard, D. H. (1999). Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, *2*(1), 79–87. 10.1038/4580

Saleem, A. B., Ayaz, A., Jeffery, K. J., Harris, K. D., & Carandini, M. (2013). Integration of visual motion and locomotion in mouse visual cortex. *Nature Neuroscience*, *16*(12), 1864–1869. 10.1038/nn.3567

Schneider, D. M., Sundararajan, J., & Mooney, R. (2018). A cortical filter that learns to suppress the acoustic consequences of movement. *Nature*, *561*(7723), 391–395. 10.1038/s41586-018-0520-5

Zmarz P., & Keller, G. B. (2016). Mismatch receptive fields in mouse visual cortex. *Neuron*, *92*(4), 766–772. 10.1016/j.neuron.2016.09.057

Received May 19, 2022; accepted July 22, 2022.

5 Prediction mismatch responses arise as corrections of a predictive spiking code

Published at [Article in preparation]
DOI
Supplementary Material
Source Code

Contributions Conceptualization, Investigation, Supervision, Writing - Original Draft

Prediction mismatch responses arise as corrections of a predictive spiking code

Kjartan van Driel[1,2], Lucas Rudelt[1], Viola Priesemann[1,3], Fabian A. Mikulasch[1]

1 Max-Planck-Institute for Dynamics and Self-Organization, Göttingen, Germany

2 University of Amsterdam, Amsterdam, Netherlands

3 Institute for the Dynamics of Complex Systems, University of Göttingen, Göttingen, Germany

Prediction mismatch responses in cortex seem to signal the difference between an internal model of the animal and sensory observations. Often these responses are interpreted as evidence for the existence of error neurons, which guide inference in models of hierarchical predictive coding. Here we show that prediction mismatch responses also arise naturally in a spiking encoding of sensory signals, where spikes predict the future signal. In this model, the predictive representation has to be corrected when a mispredicted stimulus appears, which requires additional neural activity. This adaptive correction could explain why mismatch response latency can vary with mismatch detection difficulty, as the network gathers sensory evidence before committing to a correction. Prediction mismatch responses thus might not reflect the computation of errors per se, but rather the reorganization of the neural code when new information is incorporated.

1 Introduction

Strong neural responses in cortex to unexpected events are a common observation. For example, an early finding was that oddball stimuli result in elevated activity in Electroencephalography recordings, which now is well-known as mismatch negativity (MMN) [1]. A long standing question is if MMN, and prediction mismatch responses (PMRs) in general, are a result of (bottom-up) neural adaptation or (top-down) prediction processes, and both possibilities have found experimental support [2, 3, 4]. Nevertheless, there is increasing evidence for predictions being the primary driver of PMRs in many cases [5, 6, 7, 8]. For example, in several experiments PMRs to deviant stimuli seem to be modulated by top-down connections [4, 9, 10, 11], and other experiments show PMRs for events that become predictable only given a wider context, which speaks against adaptation as the underlying mechanism [12, 13, 14, 15, 16]. These results suggest that top-down predictions play a central role in cortical processing, and that PMRs are an important characteristic of their effect on neural dynamics.

What could be the computational function that underlies PMRs? The perhaps most discussed answer is given by classical hierarchical Predictive Coding (hPC) theories of cortical processing, which propose that neurons in cortex perform inference in a hierarchical model of sensory data [17]. Classical hPC argues that PMRs are generated by dedicated error neurons, which enable inference and learning by comparing top-down predictions to sensory observations [18], and thus it connects the observed top-down modulated PMRs to a specific cortical function. Following this theory, several models of spiking neural networks have demonstrated possible mechanisms that could lead to the presence of error neurons in cortical circuits [19, 20, 21, 22]. However, while these models give mechanistic accounts of PMRs, so far there is no direct connection of spiking neuron PMRs to the formal model of inference that is at the heart of the predictive coding theory [23]. This issue is connected to the more general

open question of how the inference and learning algorithm of classical hPC might be implemented by spiking neurons [23, 24].

In this paper we offer an alternative account of PMRs, based on theories of hierarchical inference in cortex that operate without error neurons [24] (see also [25, 26, 27, 28]). In summary, we assume that the neural spiking code in cortex is predictive, and a mispredicted stimulus requires a correction of the code, which leads to additional neural activity. This will allow us to connect PMRs to biologically plausible theories of inference and learning in spiking pyramidal cells [24, 29]. Finally, we will discuss how PMRs might provide an opportunity to experimentally distinguish between models of hierarchical inference with or without error neurons.

2 Theory

To illustrate our ideas with a particular example, our aim is to model an experiment by Fiser and colleagues [13]. In this experiment a mouse traverses a virtual tunnel with landmarks (i.e., the location in the tunnel is known to the animal). At specific locations the mouse is then presented with visual patterns that are either predicted by the location, or mispredicted (Fig 1a). In the experiment, mispredicted stimuli lead to heightened neural activity in visual cortex V1 [13].

The basic assumption of our model is that V1 aims to find a predictive spike encoding of visual stimuli. This assumption has two motivations: i) in experiment neurons have been found that learn to predict upcoming stimuli [13, 30], and ii) it might be advantageous to predict stimuli to facilitate rapid stimulus recognition, anticipatory behaviour and counteract processing delays in the neural system [30, 31, 32]. We realize this idea formally, by assuming that the mouse has an internal (generative) model of how sensory stimuli are generated (Fig 1b,d), which is inverted by the neural circuits in cortex (Fig 1c). More specifically, the generative model states that the location is predictive of the encoding in V1, and the encoding is predictive of the perceived stimulus. Therefore, to find the encoding in V1, neurons have to combine sensory signals and location information (e.g., provided by the anterior cingulate cortex [13]).

An important additional component of the generative model is a binary context variable $c \in \{0, 1\}$, which determines if the location is indeed predictive for the encoding ($c = 1$) or not ($c = 0$). This context variable is necessary to enable the generative model to capture the sensory data in cases where the location mispredicts the observed stimulus. Intuitively, this enables the mouse to realize that a predicted stimulus (e.g., an object) is not present at a certain location, and to integrate this information into the stimulus representation. Otherwise, the mouse would continue to naively combine the wrong prediction from location with the sensory observation, resulting in a wrongfully biased representation no matter how much evidence for the inadequacy of the prediction is available. Finding the context variable while inverting the generative model is a nontrivial problem [33, 34]. To tackle this, we here build on previous work [33] and deterministically switch from $c = 1$ to $c = 0$ if the location prediction deviates significantly from the encoding in V1 over an extended period, which can be motivated from the generative model (i.e., this implements a maximum a-posteriori estimate of c, see Methods). In the neural circuit, the realization that the location is mispredictive ($c = 0$) then leads neurons in V1 to ignore inputs from location neurons (Fig 1c). We will outline possible biological mechanisms for this computation in the Discussion.

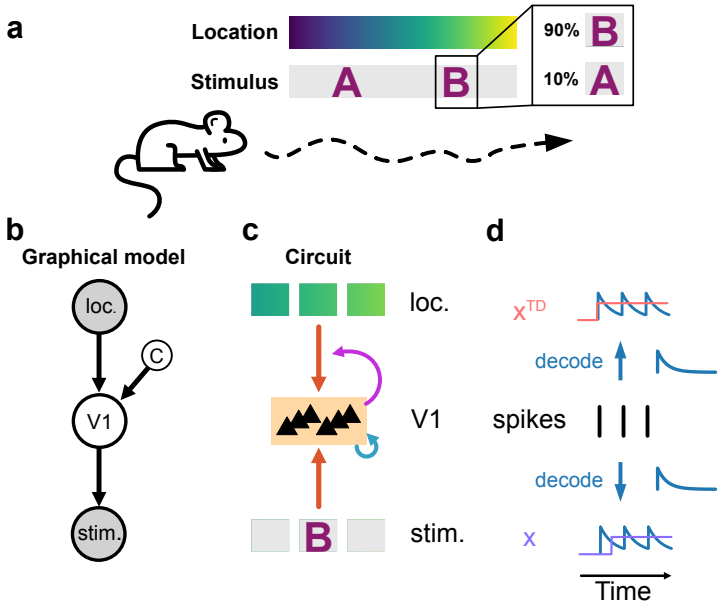

Fig 1. Model of neural coding for spatially predicted visual stimuli. **(a)** Neurons aim to encode visual stimuli (A,B) that are associated with specific given locations of the animal. In rare cases the association can be violated, i.e., 10% of the trials the predicted stimulus B is replaced by A. **(b)** We assume the animal forms a generative model of visual data. The location (loc.) predicts the stimulus representation in V1, and the stimulus representation predicts the sensory observation (stim.). An additional context variable (c) indicates if the prediction from location is valid. Shaded grey circles denote signals that are given (fixed) by the experiment. Variables in white circles have to be inferred to model the given data. **(c)** Cortical circuits invert the generative model. Depending on the inferred context variable c the influence of top-down location information is enabled or disabled. **(d)** Illustration of how spikes in V1 aim to track visual (x) and location (x^{TD}) input signals. Spikes encode the future signals via an exponential kernel, or in other words, they predict the future signal. Spikes are fired such that they simultaneously conform to visual and location signals.

3 Results

To demonstrate how this model leads to heightened neural activity during prediction mismatch, we simulated the responses of a small network to simple experimental stimuli. A network of 6 neurons encodes two patterns (A and B) and an inter-stimulus signal, denoted by low-dimensional orthogonal vectors. Stimulus encoding weights and top-down connections were set fixed for simplicity, but could in principle also be learned via voltage-dependent plasticity rules [24, 29, 35, 36]. The location generally perfectly predicts the stimulus, except for pattern B, where it is predictive only 90% of trials.

We now explain how a prediction mismatch leads to a correction of the population code, and with that to a burst of neural activity. Because a location is predictive, neurons coding for the predicted pattern (B) will be driven by top-down location inputs, and fire in anticipation to encode the future signal (Fig 2). When the actual pattern (A) appears, the network starts to find an encoding of the stimulus (A) that is biased towards the prediction (B). The emerging mismatch between prediction and encoding leads the network to realize that the prediction is invalid (i.e., c switches from 1 to 0). In consequence, the top-down prediction is ignored and the encoding in the population fully switches from

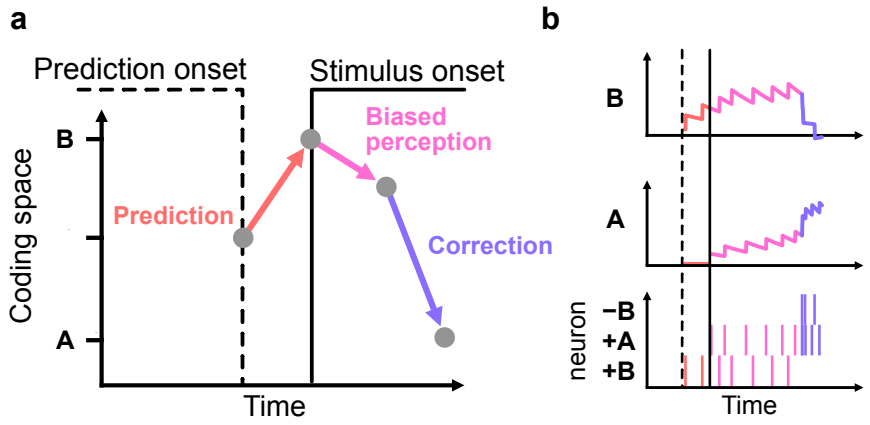

Fig 2. A mechanism for PMRs through corrections of a predictive spiking code. **(a)** Schematic illustration of the evolution of the population code over time in case of a prediction mismatch. After prediction onset the population begins coding for the predicted pattern (B). With stimulus onset the network starts to encode the observed pattern strongly biased towards the prediction. When the prediction mismatch is detected (c switches from 1 to 0), the code is corrected by removing the wrongly predicted pattern and adding the observed pattern. Moving in coding space (arrows) requires neural spiking. **(b)** The same evolution of the population code but in simulation. Top panels show the decoded stimulus code, bottom panel shows network spiking activity. Because past spikes predict the future signal, the rapid switch during the correction requires neurons that actively pull the population code for the predicted pattern (B) down.

the predicted pattern (B) to the observed (A). This switch requires activity of two types of neurons: i) neurons coding for pattern A, which are driven by bottom-up input, and ii) neurons coding for pattern '-B', which are driven by neurons coding for B and released from top-down inhibition (Suppl. Fig S1). These 'correction' neuron (ii) are required because past spikes predicted pattern B, and this prediction has to be removed from the predictive encoding. Intuitively, a predictive code has 'momentum', and a rapid correction requires strong network action in the form of spikes.

In our simulations, these two processes together resulted in a burst of activity of the population in response to mispredicted stimuli (Fig 3a). This increase in activity was even more pronounced for correction neurons ('-B' neurons), which only become active when the population code over-predicts a pattern (B) (Fig 3b). Since correction neurons remove wrong predictions from the population code, they become more active in trials where the activity of mispredictive neurons before the stimulus was stronger (Fig 3c), which was also found experimentally [13]. In that sense, these neurons can be considered dedicated 'mismatch' or 'error' neurons, although they simply keep the population code in check by removing the over-predicted pattern from the code.

Finally, we aimed to analyze the proposed context detection mechanism (switching of variable c) in more detail. c was estimated by selecting the context ('correct prediction' or 'misprediction') which better captures the relation of encoding and prediction. Mechanistically, this was implemented by comparing the time-window averaged prediction error (context criterion) to a switching threshold (Fig 4c). We found that this implies a longer time delay before a context switch if the mismatch is harder to detect (e.g., if it is smaller), which we verified in our simulations (Fig 4). Intuitively, in situations where it is hard to detect the inadequacy of the prediction the mouse has to deliberate longer, and gather more evidence, before the internal predictions can be ignored.

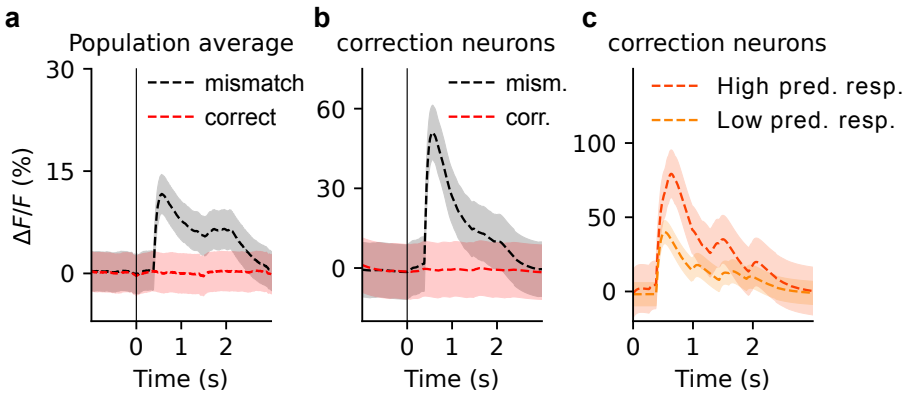

Fig 3. The model recovers several effects of PMRs as observed in experiment [13]. All panels show simulated fluorescence recordings based on simulated spike trains (see Methods). Stimulus onset is at Time = 0s. **(a)** The average population response to a mispredicted pattern (A) is higher than to a correctly predicted pattern (B). **(b)** This effect is even more pronounced in neurons that remove the over-prediction of the mispredicted pattern ('-B' neurons), and therefore appear as dedicated 'correction neurons'. **(c)** Same as the mismatch condition in panel b, but with trials separated based on the strength of predictive activity that precedes stimulus onset. Stronger predictive activity requires a stronger response of correction neurons. Note that in the modeled experiment these effects (**a** - **c**) are found for the omission of a predicted stimulus (Fig 4 in [13]). To ease interpretation we here consider a misprediction (B instead of A), but in our model this is equivalent to a omission (B instead of inter-stimulus signal). Panels **a** and **b** show mean and standard deviation for 50 trials in each condition. Panel **c** is based on the same data and shows top and bottom 10% of trials sorted by pre-stimulus activity.

4 Discussion

Building on previous work on efficient spike coding [24, 37], we here proposed that prediction mismatch responses (PMRs) can result from the correction of a predictive spiking code. In our model, the correction is initiated when the prediction from other areas is incompatible with the activity in the coding population over an extended time. When this happens, the old prediction is removed from the population code and the representation of the perceived stimulus is added, both of which requires additional neural activity (Fig 2). These dynamics are consistent with experimentally measured responses of V1 neurons during prediction mismatch (Fig 3).

The correction dynamics were derived from a generative model view of perception (Fig 1). Specifically, we assumed that the internal model of the mouse distinguishes between two cases: One, where the location is predictive of the encountered pattern, and one where it is not. Which one of these two possibilities is the case has to be inferred from observations, and we showed that this process takes longer in cases where the observations are more ambiguous (Fig 4). As an illustrative example, consider that you have left an object (e.g., a bottle) in a room, and another person removed it later. Entering the room you have the expectation to observe the bottle (and are more inclined to perceive it), but after a short look you realize that the location-object prediction you held was incorrect and you correct your internal model. If, instead, the room would be only dimly lit, it would take longer for this realization to occur, and other bottle-shaped objects might deceive your perception in the meantime. Based on our model, we argue that this realization and the subsequent correction of the representation of sensory stimuli is what underlies the PMRs observed in experiment.

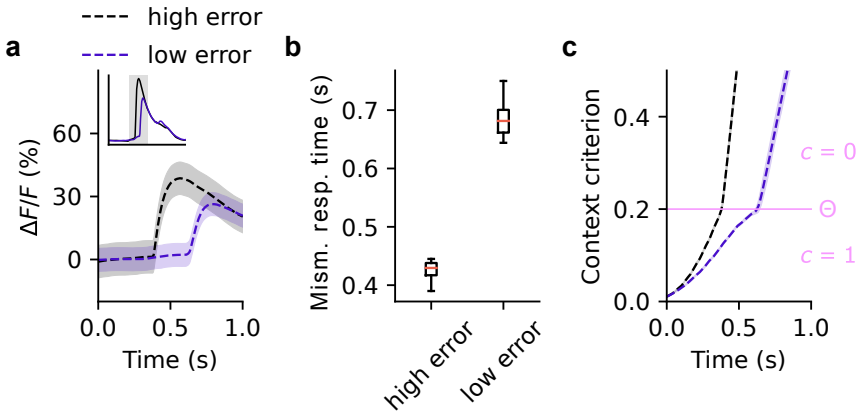

Fig 4. The mismatch detection difficulty influences latency of model PMRs. To show this the network was presented with a pattern that is a mix of the predicted (B, 30%) and the unpredicted pattern (A, 70%), which means the network observes only a partial mismatch between prediction and stimulus. **(a)** If the mismatch of prediction and observation is smaller (low error), a longer latency of PMRs can be observed in the response of mismatch neurons (i.e., -B neurons). Inset shows the same responses over a longer time-window. **(b)** The median delay of the mismatch response (here simply measured as the time of $\Delta F/F$ crossing 20%) is several hundred milliseconds longer. Note, that this constitutes only a qualitative prediction of our model, and smaller or larger delays could be obtained by choosing different parameters (e.g., Θ). **(c)** The difference in latency originates from the dynamics of the criterion for switching the context variable c (time averaged prediction error), which increases with a slower rate in case of a partial mismatch. This results in the network reaching the threshold for initiating a correction of the encoding (Θ) later. All panels show results for 50 trials in each condition. Outliers are not shown in panel **b**.

Experimental predictions Our model makes three key predictions:

i) When neurons in a population show dedicated mismatch responses resulting from the proposed mechanism, the same population will also contain stimulus predictive neurons (or, at least, stimulus selective neurons that are strongly biased towards the prediction). In our model these neurons cooperate to implement an efficient and responsive spiking code [38]. A possibly similar co-location of mismatch- and stimulus-selective neurons has been found in several experiments [9, 13, 39].

ii) Our model predicts that mismatch responses emerge with a longer delay when detecting the mismatch becomes more difficult (e.g., for a smaller mismatch), since more evidence has to be gathered before committing to a correction (Fig 4). This could be tested by measuring the time to the onset of the mismatch response depending on the stimulus noise level, or mismatch size. There have been several experiments showing the predicted (or a similar) effect in EEG recordings of MMN [5, 40, 41, 42, 43, 44, 45]. We expect that this effect can also be measured in single neuron recordings of PMRs, where especially mismatch-selective neurons should be affected.

iii) Our model predicts distinct origins for the driving connections in positive and negative PMRs (Supplementary Fig S1). Positive PMRs (i.e., the stimulus is bigger than expected) result from excess drive through bottom-up connections, which is not cancelled by lateral inhibition in the population. Negative PMRs, in turn, result from drive within the population, which is not cancelled by top-down inhibition (these neurons are the 'negative' coding neurons that appear as

dedicated error neurons; Fig 3). In practice, however, it might be difficult to achieve situations where these two types of PMRs can be observed in isolation, and in our simulation both cases appear. Note, that in both cases the PMRs are nevertheless a result of mispredictions, typically from top-down inputs. Top-down inputs in our model, however, do not directly drive the delayed PMRs, but only indirectly cause them when the wrong predictions they have provided are finally ignored.

How does this model differ from other explanations of mismatch responses? Previously, prediction mismatch responses have been explained with the dedicated error neurons that are proposed by classical hPC and similar models [18, 46]. While some neurons in our model show responses that appear as dedicated error responses (Fig 3), the interpretation of the purpose of PMRs in our model and in classical hPC is very different. In classical hPC, the responses of error neurons can only be interpreted in conjunction with associated activity in prediction neurons. That means that any higher-level area that wants to make sense of these error responses has to maintain predictions in its activity, which is updated on errors and sent back down to compute the new prediction error. In our model, mismatch responses simply constitute a part of a predictive population code. In both interpretations, mismatch responses can be understood to signal that the prediction that was conveyed up until that point was wrong, and that the target area might have to adapt to this (e.g., by changing the planned course of action in motor ares). In contrast to classical hPC, however, our model does not require feedback-signals from every target area that cancel the prediction error.

Experimentally, classical hPC and our model might be distinguished in two ways. First, by looking at the inputs that drive mismatch responses. In error neurons, PMRs are typically assumed to arise when bottom-up drive and top-down inhibition do not match (Supplementary Fig S1), as opposed to the different origins in our model we have discussed before. Second, by looking at the temporal dynamics of PMRs. As we have discussed, our model predicts that the delay of PMRs depend on the difficulty of detecting a mismatch, e.g., through noise in the stimulus or the size of mismatch (Fig 4). In contrast, in error neurons the mismatch would be expected to be signalled as early as possible [19, 20, 21]. Therefore, this idea seems to predict that noise in the signal, or the size of mismatch, would only influence the magnitude of PMRs but not their latency. This means that the same experiment we have proposed to test our model in the last section might also be used to distinguish between theories with or without error neurons.

Another set of work has proposed that prediction mismatch responses could be a signature of an efficient adaptive code [33, 34]. In this idea the neural code is continually adapted to most efficiently encode a signal, and a temporary mis-adaptation results in an inefficient encoding, that is, increased activity. This means that in this theory, similar to error neurons, the mismatch response is immediate, and ceases with the adaptation. We have employed the same idea of an adaptive code, but showed that in a population code the correction *after* an adaptation causes a surge of activity. Therefore, our model constitutes an extension of these adaptive coding models and is compatible with their ideas. Future models might look more in detail at the possible interaction of these two proposed components of mismatch responses that arise early and late after stimulus onset.

In previous work we have also proposed a model for the emergence of multimodal mismatch responses [47]. In this model, mismatch responses arise when different areas simultaneously encode sensory information, as these areas compete to encode the signal and thereby cancel activity in their respective partner area. Here the proposed purpose of mismatch responses (i.e., encoding the signal that is not explained by the other area) is very different from the presented model, but there is no reason why not both of these computations could exist alongside each other. Based on our models we argue that different types of mismatch responses in cortex can have strictly different meanings in cortical computations, and it might not be appropriate to describe all these observations with a single computational principle.

Limitations In a similar vein, the assumptions of our model might not be appropriate for all model corrections an animal can perform. We here assumed that the top-down predictive signal (i.e., the location) is perfectly certain and fixed, which makes sense in the modeled experiment where the mouse can orient itself with landmarks [13]. However, if the animal is not certain of the content of higher-level representations, in light of new sensory evidence these higher-level representations might be corrected (instead of the 'coupling' between levels, as in the presented model). This type of correction likely has very different temporal dynamics than the one presented here, and future work could investigate these dynamics in a more complex multi-level model.

Furthermore, to show the essential elements of the proposed mechanism for the generation of PMRs we made use of a highly simplified model network. Most centrally, the network consisted only of 6 neurons, neurons inhibited each other directly and not via interneurons, and we employed very low dimensional orthogonal stimuli as model inputs. Past work with similar networks indicates that such simplifications are not necessary [35, 48], and future work should be able to relax them for more biologically plausible models.

Implementation in cortical circuits So far we have operated with an abstract model, but in the context of a theory of dendritic predictive coding in cortex [24] we can speculate about possible biological implementations of the proposed mechanism. Consistent with the proposed model, this existing theory proposes that layer 2/3 pyramidal cells find a predictive encoding of sensory stimuli, where bottom-up sensory information arrives at basal dendrites, and top-down predictions from other areas at apical dendrites. Thus, the only component in our model that so far has no biological interpretation is the adaptive switch that allows/disallows top-down predictions to influence the population code (i.e., variable c; Fig 1). Since top-down predictions arrive at apical dendrites, one possibility would be that the coupling of the apical dendrite to soma is adapted [49], which would imply that apical dendrites decouple from the soma during a mismatch event. Similar mechanism have been proposed to be responsible for adaptive associations in cortex [50] or the conscious processing of sensory stimuli [51]. Another, less intrusive mechanism would be that apical inhibition precisely controls the impact of specific predictions on neural activity, by cancelling apical inputs in case of a prediction mismatch. Indeed, somatostatin-expressing (SST) interneurons, mostly targeting the apical dendrite, seem to play a central role in the generation of PMRs [4, 52]. This proposal leads to the seemingly paradoxical prediction that apical inhibition of pyramidal cells through SST interneurons would be decreased when top-down inputs are predictive, but increased when they are mis-predictive and pyramidal cells show strong mismatch responses, for which there are some indications [52].

While these biological connections are speculative, they provide testable predictions for how exactly cortex can adapt its internal model in cases where internally generated predictions do not match sensory observations. Testing for the biological mechanism behind these PMRs might be important in order to understand the causes of mental disorders where PMRs are altered, such as schizophrenia or certain learning disorders [53].

Acknowledgements

We want to thank Fabian Sinz, Suhas Shrinivasan and the Priesemann Lab, especially Andreas Schneider, for helpful discussions. F.A.M. and L.R. were funded by the German Research Foundation (DFG), SFB1286. V.P. received support from the SFB1528, Cognition of Interaction.

Author contributions

KvD: Investigation, Software, Writing - Original Draft. **LR**: Investigation, Writing - Review & Editing. **VP**: Supervision, Writing - Review & Editing. **FAM**: Conceptualization, Investigation, Supervision, Writing - Original Draft.

Methods

Generation of input signals

We model the experimental setup of Fiser and Keller [13] by defining the input signals model V1 receives. Visual inputs $\mathbf{x}^*(t)$ were 2 different patterns (A and B) and an inter-stimulus signal, which were presented with a one-hot encoding. The location in the tunnel $\mathbf{x}^*_{\text{loc}}(t)$ was assumed to be perfectly known and similarly given by a one hot encoding akin to the representation in place cells, which each corresponded to the location of a pattern (or inter-stimulus signal) in the tunnel.

Patterns were presented for 1.5 s before switching to the next pattern. Input signals were low-pass filtered to simulate the integration in visual cortex $\tau\dot{\mathbf{x}}(t) = \mathbf{x}^*(t) - \mathbf{x}(t)$. The location signal was filtered similarly, but presented shortly (100 ms) before the stimulus to achieve anticipatory spiking. Intuitively, we argue that the mouse reaches a location in the tunnel where it predicts the pattern shortly before it actually observes it. Note, that for the effect of mismatch responses this is not strictly necessary, but a choice we made to model this specific experiment.

Generative model of sensory data

To simulate the perception process of the mouse, we first defined a generative model we assume the mouse has of sensory data, and then found a network that sampled from the inverted model (i.e., the posterior for the variables of interest).

The model was defined via a hierarchy of Gaussian distributions, where stimuli $\mathbf{x}$ were generated by activity $V1$ corresponding to the hidden causes of sensory data (here the identity of the pattern)

$$p_\theta(\mathbf{x}|V1) = \mathcal{N}_{\mathbf{x}}(D_{V1}V1, \sigma_{V1}), \tag{1}$$

and hidden causes $V1$ were generated by the location of the mouse $\mathbf{x}_{\text{loc}}$

$$p_\theta(V1|\mathbf{x}_{\text{loc}}) = \mathcal{N}_{V1}(D_{\text{loc}}\mathbf{x}_{\text{loc}}, \sigma_{\text{loc}}(c)). \tag{2}$$

For simplicity, we set all parameters $\theta = \{D_{V1}, \sigma_{V1}, D_{\text{loc}}, \sigma_{\text{loc}}(c)\}$ by hand. The decoder weights D_{V1} and D_{loc} might in principle be learned using voltage-based plasticity rules [24], but were here chosen as the identity matrix.

To model the fact that objects (i.e., patterns) can be absent from a certain location, the variance $\sigma^2_{\text{loc}}(c)$ of the location prediction of $V1$ is adaptive. Specifically, $\sigma^2_{\text{loc}}(c)$ depends on a binary context variable $c \in \{0, 1\}$ which indicates if the location is predictive ($c = 1$) or not ($c = 0$)

$$\sigma^2_{\text{loc}}(c) = \begin{cases} \sigma^2_{\text{large}}, & \text{if } c = 0 \\ \sigma^2_{\text{small}}, & \text{if } c = 1. \end{cases} \tag{3}$$

The (implicit) prior distribution $p_\theta(c)$ we use for c will be discussed later.

Neural dynamics performing inference through sampling

The goal of neural inference is to find an approximation to the posterior $p_\theta(V1 \mid \mathbf{x}, \mathbf{x}_{\text{loc}})$ using a neural network. To this end, we replace the continuous variable $V1$ by a spike based representation, similar to previous work on spike-based representations [35, 54]. We define $V1 = D\mathbf{r}(t)$ as a transformation of neural responses $\mathbf{r}(t)$, generated by 6 neurons. The transformation matrix D was defined as

$$D = \alpha \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}. \tag{4}$$

Neural responses are defined as a readout of neural spiking $s_i(t) \in \{0, 1\}$ convolved with exponential spike traces $r_i(t) = \sum_{t' \leq t} \kappa(t - t')s_i(t')$, where $\kappa(\Delta t) = \exp(-\Delta t/\tau)$. Neural spikes are thus read out in the future, and can be considered to constitute a prediction of the upcoming signal.

For spike based inference, we introduce an additional prior on neural spiking

$$p_\theta(s_i(t) = 1) = \nu \ \delta t, \tag{5}$$

where δt denotes the physical time elapsed between successive timesteps. This prior can also be thought of as a metabolic cost on neural activity.

Representing $V1$ in this manner, combined with the additional prior allows us to heuristically derive a network of stochastic spiking neurons performing inference. The derivation relies on two key observations. First, we can write the spiking probability for neuron i in terms of log-probabilities as

$$p_\theta(s_i(t) \mid \mathbf{x}, \mathbf{x}_{\text{loc}}, \mathbf{r}, c) \propto \exp\Big(s_i(t)\Big(\ln p_\theta(s_i(t) = 1, \mathbf{x}, \mathbf{x}_{\text{loc}}, \mathbf{r}, c) - \ln p_\theta(s_i(t) = 0, \mathbf{x}, \mathbf{x}_{\text{loc}}, \mathbf{r}, c)\Big)\Big), \tag{6}$$

which is similar to previous approaches for spike-based neural sampling [55], and can be considered a stochastic generalization of the spike-by-spike framework [35]. We note that in general the spiking probability for neuron i is not independent from the rest of the network. Our second observation is that if we suppose δt to be small our metabolic prior in Eq. (5) forces the probability of simultaneous spiking to approach zero, and therefore introduces an independence between between spiking.

By using these observations and writing out the difference in logarithms in Eq. (6) for our choice of model and representation we find that the spiking probability of neuron i can be expressed as

$$p_\theta(s_i(t) = 1 \mid \mathbf{x}, \mathbf{x}_{\text{loc}}, \mathbf{r}, c) \propto \delta t \exp\left(u_i - T_i\right), \tag{7}$$

for a membrane potential u_i and threshold T_i given by

$$\mathbf{u} = F\mathbf{x} + F_{\text{loc}}\mathbf{x}_{\text{loc}} + W\mathbf{r} \tag{8}$$

$$\mathbf{T} = \frac{1}{2}\text{diag}(W) - \ln \nu \tag{9}$$

with

$$F = \frac{1}{\sigma_{V1}^2} D^T D_{\text{V1}}^T \tag{10}$$

$$F_{\text{loc}} = \frac{1}{\sigma_{\text{loc}}^2(c)^2} D^T D_{\text{loc}} \tag{11}$$

$$W = -\frac{1}{\sigma_{V1}^2} D^T D_{\text{V1}}^T D_{\text{V1}} D - \frac{1}{\sigma_{\text{loc}}^2(c)} D^T D \tag{12}$$

where diag(W) denotes the vector containing the diagonal elements of W and the superscript on D^T to denotes the matrix transpose. This form of the spiking probability can be seen as a special case of the spike response model with exponential escape noise [56]. Alternatively, similar results could be obtained by simulating neural spiking as performing maximum a-posteriori inference in the generative model, which can be implemented with similar but deterministic neural dynamics [35]. To obtain fluctuation based results (e.g., Fig 3c), however, additional noise on neural inputs would be required in this case.

Context switching algorithm

The context variable c that indicates the validity of the location prediction also has to be inferred. We performed this inference using an approach that has been proposed in previous research [33, 34]. At time t we selected c as the maximum a-posteriori estimate for the recent past time-window T: $c \leftarrow \max_c \sum_{t-T<t'<t} \log p_\theta(c|V1(t'), \mathbf{x}(t'), \mathbf{x}_{\text{loc}}(t'))$. Since c is binary we selected $c = 0$ iff

$$\frac{1}{T} \sum_{t-T<t'<t} \log p_\theta(c=0|V1(t'), \mathbf{x}(t'), \mathbf{x}_{\text{loc}}(t')) > \frac{1}{T} \sum_{t-T<t'<t} \log p_\theta(c=1|V1(t'), \mathbf{x}(t'), \mathbf{x}_{\text{loc}}(t')) \quad (13)$$

$$\Leftrightarrow \frac{1}{T} \sum_{t-T<t'<t} ||V1(t') - D_{\text{loc}}\mathbf{x}_{\text{loc}}(t')||^2 > \Theta, \quad (14)$$

where Θ is a switching threshold that combines the prior on c and the variance of the model $p_\theta(V1|\mathbf{x}_{\text{loc}})$. For simplicity, instead of explicitly defining a prior on c we directly chose the switching threshold Θ.

Simulation of calcium fluorescence signals

To simulate fluorescence signals $\Delta F/F$ as measured in experiment we first created calcium traces for each neuron by convolving the spike train with a realistic fluorescence kernel

$$f(t) = b + \sum_{t'<t} \kappa_f(t-t')s(t'). \quad (15)$$

Here, $b = 4.0$ is a baseline activity in the signal which models average measurement noise and other activity, and re-scales the normalized signal $\Delta F/F$. The kernel of the fluorescence elicited by a spike was defined by $\kappa_f(\Delta t) = \exp(-\Delta t/\tau_{\text{decay}})(1 - \exp(-\Delta t/\tau_{\text{rise}}))$, with $\tau_{\text{rise}} = 80$ms and $\tau_{\text{decay}} = 400$ms, as measured in experiment [57]. The normalized fluorescence signal was then computed via

$$\Delta F/F = \frac{f(t) - \langle f(t)\rangle_t}{\langle f(t)\rangle_t}. \quad (16)$$

Parameters

Parameter	Value
δt	1 ms
τ	200 ms
α	0.15
σ^2_{V1}	1/300
σ^2_{small}	1/300
σ^2_{large}	1
ν	12.18
(context criterion window) T	500 ms
(context criterion threshold) Θ	0.2

References

[1] Risto Näätänen. "The mismatch negativity: a powerful tool for cognitive neuroscience". In: *Ear and hearing* 16.1 (1995), pp. 6–18.

[2] Renée M Symonds et al. "Distinguishing neural adaptation and predictive coding hypotheses in auditory change detection". In: *Brain topography* 30 (2017), pp. 136–148.

[3] Patrick JC May. "The adaptation model offers a challenge for the predictive coding account of mismatch negativity". In: *Frontiers in Human Neuroscience* 15 (2021), p. 721574.

[4] Jordan P Hamm and Rafael Yuste. "Somatostatin interneurons control a key component of mismatch negativity in mouse visual cortex". In: *Cell reports* 16.3 (2016), pp. 597–604.

[5] Kaitlin Fitzgerald and Juanita Todd. "Making sense of mismatch negativity". In: *Frontiers in Psychiatry* 11 (2020), p. 468.

[6] Zenas C Chao et al. "Large-scale cortical networks for hierarchical prediction and prediction error in the primate brain". In: *Neuron* 100.5 (2018), pp. 1252–1266.

[7] Marie E Bellet et al. "Prefrontal neural ensembles encode an internal model of visual sequences and their violations". In: *bioRxiv* (2021).

[8] Amit Yaron, Itai Hershenhoren, and Israel Nelken. "Sensitivity to complex statistical regularities in rat auditory cortex". In: *Neuron* 76.3 (2012), pp. 603–615.

[9] Jordan P Hamm et al. "Cortical ensembles selective for context". In: *Proceedings of the National Academy of Sciences* 118.14 (2021), e2026179118.

[10] Connor G Gallimore, David Ricci, and Jordan P Hamm. "Spatiotemporal dynamics across visual cortical laminae support a predictive coding framework for interpreting mismatch responses". In: *bioRxiv* (2023).

[11] Gloria G Parras et al. "Neurons along the auditory pathway exhibit a hierarchical organization of prediction error". In: *Nature communications* 8.1 (2017), p. 2148.

[12] Elyse Sussman and Istvan Winkler. "Dynamic sensory updating in the auditory system". In: *Cognitive Brain Research* 12.3 (2001), pp. 431–439.

[13] Aris Fiser et al. "Experience-dependent spatial expectations in mouse visual cortex". In: *Nature Neuroscience* 19.12 (2016), pp. 1658–1664.

[14] Matthew F Tang et al. "Expectation violations enhance neuronal encoding of sensory information in mouse primary visual cortex". In: *Nature Communications* 14.1 (2023), p. 1196.

[15] Colleen J Gillon et al. “Learning from unexpected events in the neocortical microcircuit”. In: *bioRxiv* (2021).

[16] Byron H Price et al. “Expectation violations produce error signals in mouse V1”. In: *bioRxiv* (2022).

[17] Rajesh P. N. Rao and Dana H. Ballard. “Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects”. In: *Nature Neuroscience* 2.1 (1999), pp. 79–87.

[18] Georg B. Keller and Thomas D. Mrsic-Flogel. “Predictive Processing: A Canonical Cortical Computation”. In: *Neuron* 100.2 (2018), pp. 424–435.

[19] Catherine Wacongne, Jean-Pierre Changeux, and Stanislas Dehaene. “A neuronal model of predictive coding accounting for the mismatch negativity”. In: *Journal of Neuroscience* 32.11 (2012), pp. 3665–3678.

[20] Auguste Schulz et al. “The generation of cortical novelty responses through inhibitory plasticity”. In: *Elife* 10 (2021), e65309.

[21] Loreen Hertäg and Claudia Clopath. “Prediction-error neurons in circuits with multiple neuron types: Formation, refinement, and functional implications”. In: *Proceedings of the National Academy of Sciences* 119.13 (2022), e2115699119.

[22] Claudio Ruben Mirasso et al. “Competition between bottom-up visual input and internal inhibition generates error neurons in a model of the mouse primary visual cortex”. In: *bioRxiv* (2023).

[23] Beren Millidge, Anil Seth, and Christopher L. Buckley. “Predictive Coding: a Theoretical and Experimental Review”. In: *arXiv* (2021).

[24] Fabian A Mikulasch et al. “Where is the error? Hierarchical predictive coding through dendritic error computation”. In: *Trends in Neurosciences* 46.1 (2023), pp. 45–59.

[25] Rajesh P Rao. “Hierarchical Bayesian inference in networks of spiking neurons”. In: *Advances in neural information processing systems* 17 (2004).

[26] Laurence Aitchison and Máté Lengyel. “With or without you: predictive coding and Bayesian inference in the brain”. In: *Current Opinion in Neurobiology* 46 (2017), pp. 219–227.

[27] David J Heeger. “Theory of cortical function”. In: *Proceedings of the National Academy of Sciences* 114.8 (2017), pp. 1773–1782.

[28] David Rotermund and Klaus R Pawelzik. “Biologically plausible learning in a deep recurrent spiking network”. In: *bioRxiv* (2019).

[29] Robert Urbanczik and Walter Senn. “Learning by the dendritic prediction of somatic spiking”. In: *Neuron* 81.3 (2014), pp. 521–528.

[30] Stephanie E Palmer et al. “Predictive information in a sensory population”. In: *Proceedings of the National Academy of Sciences* 112.22 (2015), pp. 6908–6913.

[31] Luca Mazzucato, Giancarlo La Camera, and Alfredo Fontanini. “Expectation-induced modulation of metastable activity underlies faster coding of sensory stimuli”. In: *Nature neuroscience* 22.5 (2019), pp. 787–796.

[32] Matthew Chalk, Olivier Marre, and Gašper Tkačik. “Toward a unified theory of efficient, predictive, and sparse coding”. In: *Proceedings of the National Academy of Sciences* 115.1 (2018), pp. 186–191.

[33] Wiktor F Młynarski and Ann M Hermundstad. “Adaptive coding for dynamic sensory inference”. In: *Elife* 7 (2018), e32055.

[34] Wiktor F Młynarski and Ann M Hermundstad. "Efficient and adaptive sensory codes". In: *Nature Neuroscience* 24.7 (2021), pp. 998–1009.

[35] Wieland Brendel et al. "Learning to represent signals spike by spike". In: *PLoS computational biology* 16.3 (2020), e1007692.

[36] Fabian A. Mikulasch, Lucas Rudelt, and Viola Priesemann. "Local Dendritic Balance Enables Learning of Efficient Representations in Networks of Spiking Neurons". In: *Proceedings of the National Academy of Sciences* 118.50 (2021).

[37] Sophie Denève and Christian K. Machens. "Efficient codes and balanced networks". In: *Nature Neuroscience* 19.3 (2016), pp. 375–382.

[38] Nuno Calaim et al. "The geometry of robustness in spiking neural networks". In: *Elife* 11 (2022), e73276.

[39] Muneshwar Mehra, Adarsh Mukesh, and Sharba Bandyopadhyay. "Separate functional subnetworks of excitatory neurons show preference to periodic and random sound structures". In: *Journal of Neuroscience* 42.15 (2022), pp. 3165–3183.

[40] Mikko Sams et al. "Auditory frequency discrimination and event-related potentials". In: *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section* 62.6 (1985), pp. 437–448.

[41] Tess K Koerner et al. "Neural indices of phonemic discrimination and sentence-level speech intelligibility in quiet and noise: A mismatch negativity study". In: *Hearing Research* 339 (2016), pp. 40–49.

[42] Alexandra Muller-Gass et al. "The intensity of masking noise affects the mismatch negativity to speech sounds in human subjects". In: *Neuroscience Letters* 299.3 (2001), pp. 197–200.

[43] Brett A Martin, Diane Kurtzberg, and David R Stapells. "The effects of decreased audibility produced by high-pass noise masking on N1 and the mismatch negativity to speech sounds/ba/and/da". In: *Journal of Speech, Language, and Hearing Research* 42.2 (1999), pp. 271–286.

[44] LH Van der Tweel, O Este, CR Cavonius, et al. "Invariance of the contrast evoked potential with changes in retinal illuminance". In: *Vision research* 19.11 (1979), pp. 1283–1287.

[45] Elisabeth Fonteneau and Jules Davidoff. "Neural correlates of colour categories". In: *Neuroreport* 18.13 (2007), pp. 1323–1327.

[46] Karl Friston. "Does predictive coding have a future?" In: *Nature neuroscience* 21.8 (2018), pp. 1019–1021.

[47] Fabian A Mikulasch, Lucas Rudelt, and Viola Priesemann. "Visuomotor mismatch responses as a hallmark of explaining away in causal inference". In: *Neural computation* 35.1 (2022), pp. 27–37.

[48] William F Podlaski and Christian K Machens. "Approximating nonlinear functions with latent boundaries in low-rank excitatory-inhibitory spiking networks". In: *arXiv preprint arXiv:2307.09334* (2023).

[49] Rodney J. Douglas and Kevan A.C. Martin. "Neuronal Circuits of the Neocortex". In: *Annual Review of Neuroscience* 27.1 (2004), pp. 419–451.

[50] Matthew Larkum. "A cellular mechanism for cortical associations: an organizing principle for the cerebral cortex". In: *Trends in neurosciences* 36.3 (2013), pp. 141–151.

[51] Jaan Aru, Mototaka Suzuki, and Matthew E Larkum. "Cellular mechanisms of conscious processing". In: *Trends in Cognitive Sciences* 24.10 (2020), pp. 814–825.

[52] Georgia Bastos et al. "A frontosensory circuit for visual context processing is synchronous in the theta/alpha band". In: *bioRxiv* (2023).

[53] Ulrich Schall. "Is it time to move mismatch negativity into the clinic?" In: *Biological psychology* 116 (2016), pp. 41–46.

[54] Ralph Bourdoukan et al. "Learning optimal spike-based representations". In: *Advances in neural information processing systems*. 2012, pp. 2285–2293.

[55] Lars Buesing et al. "Neural dynamics as sampling: a model for stochastic computation in recurrent networks of spiking neurons". In: *PLoS computational biology* 7.11 (2011), e1002211.

[56] Wulfram Gerstner et al. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.

[57] Tsai-Wen Chen et al. "Ultrasensitive fluorescent proteins for imaging neuronal activity". In: *Nature* 499.7458 (2013), pp. 295–300.

Supplementary figures

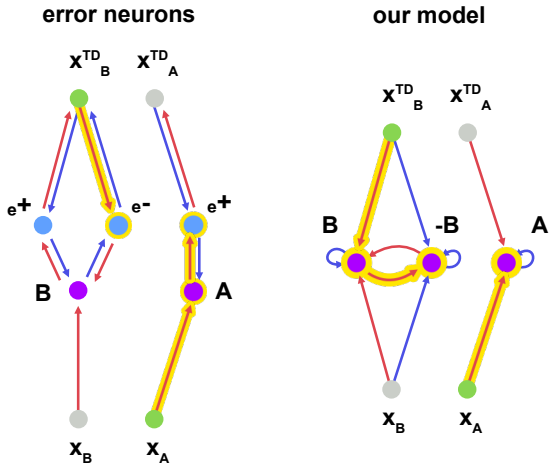

Fig S1. The driving connections of PMRs are slightly different in classical hPC and our model. Arrows denote excitatory (red) and inhibitory (blue) connections. Yellow arrows indicate the path of activity that leads to PMRs. Note, that in our model mismatch neurons (-B) only become active once top-down inhibition ceases after the correction is initiated.

6 Overall Discussion

In this thesis we have made several contributions towards a theory of inference and learning in cortex. A special focus has been put on i) finding biologically plausible learning rules that might allow neurons to learn a model of the world and ii) connecting the developed theory closely to cortical physiology and dynamics, in order to make it testable.

Towards the first aim (i) we first derived, from the goal of representing sensory stimuli, learning rules based on voltage-dependent plasticity and the local excitatory-inhibitory (**E-I**) balance in neural dendrites (chapter 2). This overcame a limitation of previous models of representation learning in spiking neurons, which had to rely on a unrealistically strong decorrelation of spiking in the network. Our theory also provides a possible role for the experimentally observed influence of the local dendritic voltage on synaptic plasticity. Building on this model, we then proposed a theory of hierarchical inference in cortex (chapter 3). Here we found different learning rules for synapses on apical dendrites, which we propose learn to incorporate hierarchical (top-down) priors for neural activity, and on basal dendrites, which learn a representation of sensory (bottom-up) information (as in chapter 2). We then connected this model to existing theories of representation learning (Brendel et al. 2020), learning of apical predictions (Urbanczik et al. 2014), and hierarchical predictive coding (Rao and Ballard 1999), showing up the close relation that exists between these models.

Towards the second aim (ii) we laid out a detailed account of how the theory of hierarchical inference can map to cortical physiology (chapter 3). Specifically, we proposed clear roles for some neuron types and connectivity patterns in neocortex. Even more specifically, we argued that pyramidal neurons in layer 2/3 learn a predictive model of sensory data, while fast-spiking basket cells are the main mediator of 'explaining-away' effects between pyramidal neurons. This yields a functional interpretation of some of the core distinguishing properties of these neurons (e.g., sparse activity in layer 2/3 pyramidal neurons, extremely fast inhibition by basket cells). It also enables to device concrete experiments that can test for predicted mechanism (e.g., the exact form of voltage-based plasticity in neural dendrites, or basal dendrites being balanced, apical dendrites being predictive). Finally, in chapters 4 and 5 we explained how two different forms of mismatch

responses can arise in models without dedicated 'error-neurons' (as proposed by classical hPC; Rao and Ballard 1999). As the mechanism required to explain the observed phenomena are different for models with or without error-neurons, this allowed us to suggest experiments that could enable distinguishing between these two competing possibilities.

6.1 Unifying and extending theories of cortical computation

A central contribution of the work in this thesis was to show up connections between formerly mostly parallel lines of research in the study of spiking networks. One the one hand, previous work introduced the theory of efficient coding in balanced spiking networks, which casts spiking as the discrete optimization of a quadratic coding goal function (Brendel et al. 2020; Denève et al. 2016; Kadmon et al. 2020). On the other hand, it has been proposed that neural activity could code for uncertainty by sampling from probability distributions (Knill et al. 2004), which has been realized formally in spiking networks before (Buesing et al. 2011; Nessler et al. 2013). We connected these two frameworks by pointing out that balance based coding is the deterministic equivalent to spike based sampling in a linear Gaussian model of sensory data (chapters 2 & 3)[1]. That sampling in a linear Gaussian model can imply network dynamics with E-I balance has been found before in rate-based models, and has been used to understand a range of features in cortical dynamics (Echeveste et al. 2020; Hennequin et al. 2014). The explicit connection between efficient spike coding and spike-based sampling in turn might prove fruitful, by enabling a transfer of insights between the two approaches to spike-based computation.

We furthermore drew a connection between spike-based sampling and theories of variance adaptation in the cortical model. In the general theory of world modeling, setting the variances of the model is important to differently weigh sensory and internal information according to their reliability[2], which for example is a central element of Kalman filtering (Grewal et al. 2020). Applied to the brain, variance adaptation has for example been used to explain multiple effects of uncertainty on cortical dynamics (Orbán et al. 2016), retinal coding (Młynarski et al. 2021), or sensorimotor estimation (Kwon et al. 2013), and some predictive coding models have proposed it as a principle that underlies attention (Feldman et al. 2010). Adding to these proposals, in chapter 5 we have employed

[1] As an interesting side-note, this connection is analogous to the connection between deterministic pattern storing networks, which minimize an energy function (Hopfield networks; Hopfield 1982), and their stochastic generalization (Boltzmann machines; Amit et al. 1985), that has been realized in the 80's.

[2] Coincidentally, this year a Japanese lunar lander crashed on account of erroneously readjusting a sensory variance in its internal model (ispace 2023).

the idea of adjusting the model variance to show how a switch in variance can result in the reorganization of the neural representation and, in turn, in a burst in spiking activity. We used this to explain why mis-predicted stimuli lead to strong neural activity (Fiser et al. 2016). Furthermore, the physiological interpretation of our theory allowed us to conjecture how changes in the model variance could be mediated by specific neural mechanisms, such as layer specific neuromodulation (chapter 3) or the action of somatostatin-positive interneurons (chapter 5). These proposals are admittedly speculative, but there are good reasons to believe that cortex makes use of adjustable variances in its internal model, and carefully designed experiments might rule out or strengthen these candidate mechanisms in the future.

Finally we have pointed out the partial equivalence of the balanced spiking network formalism (Brendel et al. 2020) and the formalism of classical hPC (Rao and Ballard 1999). Specifically, a balanced spiking network solves the same computational problem as a shallow, single layer predictive coding network (and, equivalently, a sparse coding network ; Boutin et al. 2021; Olshausen et al. 1996). This allowed us to extend the ideas of the balanced spiking framework to hierarchical networks, by relying on the theory previously developed in classical hPC (chapter 3). A direct result of this connection is that work that has been done on models of hPC now also becomes relevant for balanced spiking models, when extended to the hierarchical case. One example is the finding that hierarchical inference in a Gaussian model can explain the emergence of extra-classical receptive field effects (chapter 3; Boutin et al. 2021), which now is not anymore exclusive to classical hPC. Such relations between theories of cortical computation are therefore important to consider when evaluating which experimental observations are consistent with which theories.

6.2 Missing bits and open avenues

The unification of theories I have outlined in the last section comes with several questions attached that might be addressed in the future. Perhaps the most immediate open question is how hierarchical inference in our framework can be implemented by networks consisting of multiple layers of spiking neurons (but see, e.g., Rao 2004; Rotermund et al. 2019, for a different approach). We here so far have described the underlying theory (which in the rate based case is equivalent to that of classical hPC; chapter 3), and used it to implement inference in a single spiking layer within a hierarchical circuit (chapter 5). What problems might arise when performing inference with multiple spiking layers? So far, the balanced spiking framework has been applied only to continuous input signals (chapters 2, 4 & 5; Brendel et al. 2020), but with interconnected spiking layers it has to be understood how it can

operate with spiking input. There is no problem in principle for this, as the continuous inputs can be simply replaced by continuous spike traces. In practice, however, learning and inference could become difficult if input spikes are sparse and uncoordinated. In this case, most of the time only very few input dimensions have nonzero values, and thus the prediction errors for single dimensions would almost always be relatively large. This problem would arise in hierarchical networks if firing rates of individual neurons are low, as in cortex. We hypothesize that there are additional mechanisms required to achieve coordination between the sparse firing of input and coding neurons. One idea, which we have also touched upon in chapter 2, would be to understand the continuous inputs as a collection of synaptic inputs into the dendrites, but the implications of this idea on learning have yet to be explored (see also Appendix A.3). Another possible mechanism for coordinating neural spiking could be via oscillations, as oscillations can ensure that spikes arrive roughly simultaneously at the next layer, making it easier to detect the presence of specific patterns in a signal. The idea that oscillations facilitate communication between cortical areas has already been discussed in a more general context, for example in the communication through coherence hypothesis (Fries 2015).

Next to understanding the hierarchical generalization of the balanced spiking framework, it would be important to understand what role the extensive lateral excitatory connectivity in cortex could play in our model. In fact, in cat visual cortex, for example, 40-60% of excitatory connections in layer 2/3 originate from within the layer (Binzegger et al. 2004). In the spike sampling framework, these recurrent connections arise naturally if the generative model specifies dependencies in the signal over time (see Appendix A.3 for a derivation in a simple linear model; Kappel et al. 2014). These connections can thus be thought of as integrating information from the past representation into the spike decision, which in the control literature is known as Bayesian filtering (Särkkä et al. 2023). In this case, sampling a single spike trajectory, as we have implemented it here, becomes less meaningful. Two common alternative approaches are to find the maximum a-posteriori estimate, like in the Kalman filter (Grewal et al. 2020), or to sample multiple trajectories simultaneously to approximate the posterior, which is referred to as particle filtering (Kutschireiter et al. 2017). It is still a matter of ongoing research which of these algorithms is more closely related to what happens in cortex (although the sampling interpretation of neural inference found considerable support; Echeveste et al. 2020; Knill et al. 2004), and future work might focus more on biologically plausible implementations of Bayesian filtering in cortex and its relation to existing theories of recurrent computation in cortical networks (e.g., Bienenstock 1995; Boerlin et al. 2013). These and other open questions we also shortly touched upon in chapter 3 (Outstanding questions).

In relation to these questions, a more technical aspect of our models that should be explored in more detail are the spike based sampling algorithms that might be employed by cortical networks. Much previous work on spike-based sampling developed theories for networks sampling from binary distributions (Buesing et al. 2011; Deneve 2008) (with a few exceptions, e.g., Rotermund et al. 2019; Savin et al. 2014). We here derived approximate algorithms to sample from continuous distributions using probabilistic neural spiking. To this end we employed a heuristic approach, which prevented us to give strong guarantees on the convergence of our sampling algorithms (See also Limitations in Appendix A.3). Future work might tackle this problem with a more sophisticated mathematical formalism that enables to guarantee unbiased sampling from the posterior distribution of the target variables.

Taking a step back, in the light of our results it might also be worthwhile to reevaluate the basic model assumptions made in models of inference in cortex. Our work indicates that the generative model of cortex might not be well described by a simple hierarchy of Gaussians, as it has often been conceptualized before (Millidge et al. 2021; Rao and Ballard 1999). In chapter 4 we have introduced a model where multiple areas simultaneously explained sensory input, which we argued is the reason for the effect of locomotion on activity in V1. In chapter 5 we have introduced a model where a binary variable decides if top-down inputs are informative or not, which we used to explain the sudden (and delayed) burst of activity for mis-predictive top-down input. These details of the generative models can be understood as inductive biases that the brain incorporates to model the underlying states of sensory inputs, which often follow nontrivial distributions. Extrapolating from our results, I argue that much of the dynamics of inference in cortex hinges on such inductive biases, such as the forms of distributions, or the structural assumptions in the generative model. Thus, in order to make sense of these dynamics, it might be necessary to first understand what particular inductive biases, what form of generative model, the brain has to implement in order to effectively deal with the sensory data it acquires.

6.3 Practical applications for theories of cortical computation

Broadening our outlook on future research, we should also address a hitherto neglected question: the general contribution of implementation-level theories of neural computation in cortex, as developed here. Undoubtedly, understanding how animals and humans function is by itself a sufficient motivation for developing such models, but what might be their practical impact?

From a technological perspective, it has been argued that by understanding how neural systems produce intelligence, we will also improve our ability to construct artificial intelligence (AI) systems (Hassabis et al. 2017; Zador et al. 2023). A challenge for such a transfer of insights is that, despite their superficial similarity, current AI systems seem to operate on quite different basic principles than biological neural systems, and the direct impact of (implementation-level) theoretical neuroscience on AI models has been limited in the past (Hassabis et al. 2017). Nevertheless, even if it remains to be seen if the low-level biological principles of neural computation can be directly relevant for AI research, they are closely connected to the high-level algorithmic principles of sensory processing and cognition (we have touched upon such principles while justifying our models in chapters 4 and 5). These principles could help building competent AI systems that act and think in similar ways as humans, by informing the high-level architectural design of intelligent algorithms (Lake et al. 2017; Sinz et al. 2019).

Perhaps a more directly relevant field where implementation-level theories of neural computation in cortex can have significant impact is in health applications. One crucial area of potential application is brain-computer interfaces, where understanding how information is encoded in neural circuits can lead to improved interfaces with applications in treating various diseases (Shih et al. 2012). Another promising avenue lies in computational psychiatry, which seeks to understand mental disorders through high-level computational descriptions, and to link them to deficits in specific neural mechanisms (Huys et al. 2016). This approach opens the possibility of developing better treatments, such as drug interventions targeting these identified neural mechanisms to address the disorders effectively.

To illustrate how such a connection might look like in a particular example, we can look at schizophrenia, a mental disorder which is characterized by a broad range of symptoms (e.g., dilusions and hallucinations; Fletcher et al. 2009; Sterzer et al. 2018). It is long known that schizophrenia is connected to altered mismatch negativity (Umbricht et al. 2005), and subsequent research has related this effect to reduced activity in Somatostatin-expressing (SST) interneurons in schizophrenia patients (Van Derveer et al. 2021). From the theoretical side, there have been several attempts to describe schizophrenia in terms of predictive coding models, which suggest that schizophrenia results from an incorrectly adjusted precision weighting of top-down and bottom-up information (Sterzer et al. 2018). While there have been different accounts of how exactly the precisions in the internal model are maladjusted (Sterzer et al. 2018), in sensory processing the occurrence of hallucinations in schizophrenia might be explained by an abnormally high top-down precision, which would lead perception to follow internal predictions rather than sensory evidence.

We can build on the work in this thesis to connect these computational considerations to the experimental observations in SST interneurons. In chapter 5 we have proposed that SST interneurons could mediate the re-weighting (i.e., cancelling) of top-down inputs at apical dendrites if they are mispredictive. If SST neurons are affected by the disorder and show reduced activity, this would imply a continually increased precision of top-down inputs in the internal model, meaning that these inputs would be cancelled insufficiently and could thereby mediate the observed hallucinations and alterations in mismatch negativity. It has to be acknowledged that, for now, these are highly speculative connections of a high-level theory of schizophrenia to dysfunctions in specific neural mechanisms, and here mainly serve as an illustration of possible neuro-computational approaches to mental disorders. There still remain many uncertainties about how exactly cortical neurons implement the intended computations, and a strong theoretical basis for such continuative conjectures should first be established through further research.

6.4 Testing theories of cortical computation

It thus remains to reflect on the progress we made on the question we set out to answer: How does cortex implement a model of the world? Certainly, the theory we have formulated can only grant us insight insofar it can be tested in experiment. In the work of this thesis we have encountered two major difficulties that hinder this project: i) The ambiguity of empirical observations, i.e., multiple theories can explain an isolated empirical fact (e.g., the mismatch responses discussed in chapters 4 & 5), and ii) the incompleteness of the theory, i.e., due to the complexity of the cortical circuitry there unquestionably are computations and dynamics that are not described by the model. In the context of inference in cortex, to address the latter problem (ii), experiments have often looked at the *sine qua non* of certain theories—for example, a much researched prediction is that of mismatch responses as proposed by classical hPC (Walsh et al. 2020). However, this is certainly not enough to also tackle the first problem (i). In this thesis we have therefore tried to seek out predictions that are clearly (and ideally necessarily) different between theories. One major contribution of our work in this direction is the observation that delayed prediction mismatch responses are not straightforwardly explained by error neurons in classical hPC, but by the implementation of inference in our model (chapter 5). We have also discussed the differences of predictions on the circuit level between the theories (chapter 3), and differences in how visuomotor mismatch responses might evolve over time in error neurons or prediction neurons (chapter 4). We expect that conducting

experiments that test these and other predictions we have generated will help to constrain in a systematic way how exactly cortex implements a model of the world.

Given that these experiments have been conducted and they produced a result that rules out a certain model, how should be proceeded from there? This is not trivial to answer, since the models in question, such as the one presented here, are often flexible enough to incorporate these new observations through additional mechanisms or assumptions (see also Lakatos 1976). There is no problem to salvage a falsified model in this way, because one otherwise might, as they say, throw out the baby with the bathwater. But what about the Bayesian framework, shouldn't it prevent such arbitrariness by introducing a sense of optimality into the models? Not really. First, the framework also encompasses 'non-optimal' models, which can justify any computation as Bayesian through a specific choice of prior and likelihood distributions (Jones et al. 2011). Second, the mapping of the algorithm to the neural substrate is fundamentally unconstrained (Sprevak 2021). Thus, models relying on the Bayesian framework are 'regular' models of neural dynamics, that is, models based on observations and assumptions, and the framework brings the benefit of allowing to systematically generate functional interpretations of specific neural mechanisms (Chater et al. 2011). In consequence, models based on classical hPC, or the theory of hPC presented in this thesis, have to be treated the same as models without any principles attached: They have to be questioned in their complexity and biological plausibility, and their predictions have to be carefully tested in a systematic way.

6.5 Conclusion

Despite these considerations about the challenges in testing the developed theory we can conclude on a cautiously optimistic note. In this thesis we have started with two assumptions: i) (sensory) cortex performs spiking inference in a hierarchical model of sensory data, and ii) errors of the model are computed in neural dendrites. It is remarkable how many properties of cortical dynamics and plasticity are more or less directly implied by these very few assumptions (e.g., local E-I balance, voltage-dependent plasticity, asynchronous irregular activity, Gamma oscillations, Gabor-wavelet receptive fields in V1, extra-classical receptive field effects and their top-down modulation, ...). We have also demonstrated that some effects of expectation violations on cortical responses can be qualitatively reproduced in this model, given rather straightforward additional assumptions about the precise generative model that cortex implements. Future work might test the predictions we have generated in experiment, and expand upon our ideas to form a more comprehensive picture of learning, inference and the generation of behavior in the brain.

References

Amit, D. J., H. Gutfreund, and H. Sompolinsky (1985). "Spin-glass models of neural networks". *Physical Review A* 32, p. 1007.

Bienenstock, E. (1995). "A model of neocortex". *Network: Computation in neural systems* 6, pp. 179–224.

Binzegger, T., R. J. Douglas, and K. A. Martin (2004). "A quantitative map of the circuit of cat primary visual cortex". *Journal of Neuroscience* 24, pp. 8441–8453.

Boerlin, M., C. K. Machens, and S. Denève (2013). "Predictive Coding of Dynamical Variables in Balanced Spiking Networks". *PLOS Computational Biology* 9, e1003258.

Boutin, V., A. Franciosini, F. Chavane, F. Ruffier, and L. Perrinet (2021). "Sparse deep predictive coding captures contour integration capabilities of the early visual system". *PLOS Computational Biology* 17, e1008629.

Brendel, W., R. Bourdoukan, P. Vertechi, C. K. Machens, and S. Denéve (2020). "Learning to represent signals spike by spike". *PLoS computational biology* 16, e1007692.

Buesing, L., J. Bill, B. Nessler, and W. Maass (2011). "Neural dynamics as sampling: a model for stochastic computation in recurrent networks of spiking neurons". *PLoS computational biology* 7, e1002211.

Chater, N., N. Goodman, T. L. Griffiths, C. Kemp, M. Oaksford, and J. B. Tenenbaum (2011). "The imaginary fundamentalists: The unshocking truth about Bayesian cognitive science". *Behavioral and Brain Sciences* 34, p. 194.

Deneve, S. (2008). "Bayesian spiking neurons I: inference". *Neural computation* 20, pp. 91–117.

Denève, S. and C. K. Machens (2016). "Efficient codes and balanced networks". *Nature Neuroscience* 19, pp. 375–382.

Echeveste, R., L. Aitchison, G. Hennequin, and M. Lengyel (2020). "Cortical-like dynamics in recurrent circuits optimized for sampling-based probabilistic inference". *Nature neuroscience* 23, pp. 1138–1149.

Feldman, H. and K. J. Friston (2010). "Attention, uncertainty, and free-energy". *Frontiers in human neuroscience* 4, p. 215.

Fiser, A., D. Mahringer, H. K. Oyibo, A. V. Petersen, M. Leinweber, and G. B. Keller (2016). "Experience-dependent spatial expectations in mouse visual cortex". *Nature Neuroscience* 19, pp. 1658–1664.

Fletcher, P. C. and C. D. Frith (2009). "Perceiving is believing: a Bayesian approach to explaining the positive symptoms of schizophrenia". *Nature Reviews Neuroscience* 10, pp. 48–58.

Fries, P. (2015). "Rhythms for cognition: communication through coherence". *Neuron* 88, pp. 220–235.

Grewal, M. S., A. P. Andrews, and C. G. Bartone (2020). *Kalman filtering*. Wiley Telecom.

Hassabis, D., D. Kumaran, C. Summerfield, and M. Botvinick (2017). "Neuroscience-inspired artificial intelligence". *Neuron* 95, pp. 245–258.

Hennequin, G., L. Aitchison, and M. Lengyel (2014). "Fast sampling-based inference in balanced neuronal networks". *Advances in neural information processing systems* 27.

Hopfield, J. J. (1982). "Neural networks and physical systems with emergent collective computational abilities." *Proceedings of the national academy of sciences* 79, pp. 2554–2558.

Huys, Q. J., T. V. Maia, and M. J. Frank (2016). "Computational psychiatry as a bridge from neuroscience to clinical applications". *Nature neuroscience* 19, pp. 404–413.

ispace (2023). *Results of the "HAKUTO-R" Mission 1 Lunar Landing*. URL: https://web.archive.org/web/20230602162544/https://ispace-inc.com/news-en/?p=4691.

Jones, M. and B. C. Love (2011). "Bayesian fundamentalism or enlightenment? On the explanatory status and theoretical contributions of Bayesian models of cognition". *Behavioral and brain sciences* 34, pp. 169–188.

Kadmon, J., J. Timcheck, and S. Ganguli (2020). "Predictive coding in balanced neural networks with noise, chaos and delays". *Advances in Neural Information Processing Systems* 33.

Kappel, D., B. Nessler, and W. Maass (2014). "STDP installs in winner-take-all circuits an online approximation to hidden Markov model learning". *PLoS computational biology* 10, e1003511.

Knill, D. C. and A. Pouget (2004). "The Bayesian brain: the role of uncertainty in neural coding and computation". *Trends in Neurosciences* 27, pp. 712–719.

Kutschireiter, A., S. C. Surace, H. Sprekeler, and J.-P. Pfister (2017). "Nonlinear Bayesian filtering and learning: a neuronal dynamics for perception". *Scientific reports* 7, p. 8722.

Kwon, O.-S. and D. C. Knill (2013). "The brain uses adaptive internal models of scene statistics for sensorimotor estimation and planning". *Proceedings of the National Academy of Sciences* 110, E1064–E1073.

Lakatos, I. (1976). *Falsification and the methodology of scientific research programmes*. Springer.

Lake, B. M., T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman (2017). "Building machines that learn and think like people". *Behavioral and brain sciences* 40, e253.

Millidge, B., A. Seth, and C. L. Buckley (2021). "Predictive Coding: a Theoretical and Experimental Review". *arXiv:2107.12979*. arXiv: 2107.12979.

Młynarski, W. F. and A. M. Hermundstad (2021). "Efficient and adaptive sensory codes". *Nature Neuroscience* 24, pp. 998–1009.

Nessler, B., M. Pfeiffer, L. Buesing, and W. Maass (2013). "Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity". *PLoS computational biology* 9, e1003037.

Olshausen, B. A. and D. J. Field (1996). "Emergence of simple-cell receptive field properties by learning a sparse code for natural images". *Nature* 381, pp. 607–609.

Orbán, G., P. Berkes, J. Fiser, and M. Lengyel (2016). "Neural variability and sampling-based probabilistic representations in the visual cortex". *Neuron* 92, pp. 530–543.

Rao, R. P. (2004). "Hierarchical Bayesian inference in networks of spiking neurons". *Advances in neural information processing systems* 17.

Rao, R. P. and D. H. Ballard (1999). "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects". *Nature neuroscience* 2, pp. 79–87.

Rotermund, D. and K. R. Pawelzik (2019). "Biologically plausible learning in a deep recurrent spiking network". *bioRxiv*.

Särkkä, S. and L. Svensson (2023). *Bayesian filtering and smoothing*. Vol. 17. Cambridge university press.

Savin, C. and S. Deneve (2014). "Spatio-temporal representations of uncertainty in spiking neural networks". *Advances in Neural Information Processing Systems* 27, pp. 2024–2032.

Shih, J. J., D. J. Krusienski, and J. R. Wolpaw (2012). "Brain-computer interfaces in medicine". In: *Mayo clinic proceedings*. Vol. 87. Elsevier, pp. 268–279.

Sinz, F. H., X. Pitkow, J. Reimer, M. Bethge, and A. S. Tolias (2019). "Engineering a less artificial intelligence". *Neuron* 103, pp. 967–979.

Sprevak, M. (2021). "Predictive coding IV: The implementation level". *[Preprint]*.

Sterzer, P., R. A. Adams, P. Fletcher, C. Frith, S. M. Lawrie, L. Muckli, P. Petrovic, P. Uhlhaas, M. Voss, and P. R. Corlett (2018). "The predictive coding account of psychosis". *Biological psychiatry* 84, pp. 634–643.

Umbricht, D. and S. Krljes (2005). "Mismatch negativity in schizophrenia: a meta-analysis". *Schizophrenia research* 76, pp. 1–23.

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

Van Derveer, A. B., G. Bastos, A. D. Ferrell, C. G. Gallimore, M. L. Greene, J. T. Holmes, V. Kubricka, J. M. Ross, and J. P. Hamm (2021). "A role for somatostatin-positive interneurons in neuro-oscillatory and information processing deficits in schizophrenia". *Schizophrenia Bulletin* 47, pp. 1385–1398.

Walsh, K. S., D. P. McGovern, A. Clark, and R. G. O'Connell (2020). "Evaluating the neurophysiological evidence for predictive processing as a model of perception". *Annals of the new York Academy of Sciences* 1464, p. 242.

Zador, A., S. Escola, B. Richards, B. Ölveczky, Y. Bengio, K. Boahen, M. Botvinick, D. Chklovskii, A. Churchland, C. Clopath, et al. (2023). "Catalyzing next-generation artificial intelligence through neuroai". *Nature communications* 14, p. 1597.

A Appendix

PNAS
www.pnas.org

Supplementary Information for

Local dendritic balance enables learning of efficient representations in networks of spiking neurons

Fabian A. Mikulasch, Lucas Rudelt, Viola Priesemann

Viola Priesemann.
E-mail: viola.priesemann@ds.mpg.de

This PDF file includes:

Supplementary text
Figs. S1 to S11 (not allowed for Brief Reports)
Table S1 (not allowed for Brief Reports)
SI References

Supporting Information Text

Symbols.

- $\mathbf{X}_{0,T} = \{\mathbf{x}(t)|t \in \{0, ..., T\}\}$: Input signal over time to be encoded
- $\mathbf{S}_{0,T} = \{\mathbf{s}(t)|t \in \{0, ..., T\}\}$: Spikes of coding neurons
- $\mathbf{z}(t)$: 'Outputs' of coding neurons, proportional to evoked post-synaptic potentials
- $\hat{\mathbf{x}}(t) = D\mathbf{z}(t)$: Input signal reconstructed from network activity
- D: Decoder matrix of the decoder model
- σ: Variance of the decoder model
- $\mathbf{b}$: Spiking probability prior of decoder model
- θ: Decoder model parameters $\{D, \sigma, \mathbf{b}\}$
- F: Feedforward weights (mostly excitatory)
- W: Recurrent weights connecting to the soma (mostly inhibitory)
- W^i: Recurrent weights, connecting to the dendrites to input i (mostly inhibitory)
- T_j: Soft threshold of neuron j
- Δu: Stochasticity of neural spiking
- τ: Membrane time constant of leak
- $\eta_{(\cdot)}$: learning rate of parameter $(\cdot)$
- ρ: Target firing rate of neurons
- $1/Z(\cdot)$: Normalization of probability function

A. Stochastic neural dynamics

We simulated stochastic leaky integrate and fire neurons in discrete timesteps. The model can be seen as a special case of the spike response model with escape noise (1). In timestep $t \in \{0, 1, ..., T\}$ with length δ neuron j spikes with a probability

$$p_{\text{dyn}}\left(s_j(t) = 1|\mathbf{x}(t), \mathbf{z}(t)\right) = p_{spike}(u_j(t)) = \text{sig}\left(\frac{u_j(t) - T_j}{\Delta u}\right), \quad [1]$$

where $\text{sig}(x) = [1 + \exp(-x)]^{-1}$, $u_j(t)$ is the membrane potential of the neuron, T_j the firing threshold, Δu defines how stochastic the spiking is and $s_j(t)$ is a spike indicator, which is 1 if neuron j spiked in time step t, otherwise $s_j = 0$. Emitted spikes are then transmitted to other neurons and elicit post synaptic potentials (PSPs) $\mathbf{z}(t)$ with

$$z_j(t) = \sum_{t_s^j < t} \exp\left(-\frac{t - 1 - t_s^j}{\tau}\right),$$

which account for the leaky integration at the membrane. Here, t_s^j are the spike times of neuron j and τ the membrane time constant, which was chosen the same for all neurons. Please note that, in order to ease the upcoming derivations, we changed notation such that t is the index of the discrete timestep and τ has the unit of timesteps. The time delay of PSP arrival of the length of one time step δ is interpreted as a finite traveling time of neural impulses over axons. The PSPs together with input signal $\mathbf{x}(t)$ are then summed up linearly at the soma to give the membrane potential

$$u_j(t) = \sum_i F_{ji} x_i(t) + \sum_k W_{jk} z_k(t).$$

In order to model neurons that make use of dendritic balance we subdivided the somatic potentials such that they are sums of dendritic potentials: $u_j(t) = \sum_i u_i^j(t)$, where the dendritic potentials $u_i^j(t) = F_{ji} x_i(t) + \sum_k W_{jk}^i z_k(t)$. To summarize, stochastic neural dynamics are modeled through the spike probability $p_{\text{dyn}}\left(\mathbf{s}(t)|\mathbf{x}(t), \mathbf{z}(t)\right)$ with neural parameters $\{F, W, \mathbf{T}, \Delta u\}$.

B. Learning an efficient code with expectation maximization

With the following derivations we provide a link between learned balanced state inhibition (2) and neural sampling in graphical models (3). Hence we provide new derivations for the network dynamics and learning rules used in (2), showing how they implement unsupervised learning in a graphical model. Furthermore, with the dendritic balance learning scheme we will address the linear case of the quite general problem that arises through explaining away effects, i.e. converging arrows in graphical models: Converging arrows imply that neurons should cooperate to encode the input and lead to non-localities in update rules when the neural dynamics are based on point neurons. In related studies this problem so far has been avoided in various ways, which all prevent the network from explaining the input through possibly correlated neurons simultaneously and thus limit coding versatility (3–7).

The goal of neural spiking dynamics and plasticity throughout this paper is to find an efficient spike encoding, i.e. representing an input signal $\mathbf{X}_{0,T} = \{\mathbf{x}(t)|t \in \{0, ..., T\}\}$ through a collection of spikes $\mathbf{S}_{0,T} = \{\mathbf{s}(t)|t \in \{0, ..., T\}\}$. $\mathbf{X}_{0,T}$ can be seen here as an episode in an organisms life, which we will assume to be distributed according to $p^*(\mathbf{X}_{0,T})$. We say that $\mathbf{S}_{0,T}$ efficiently encodes $\mathbf{X}_{0,T}$ if the following two conditions are met:

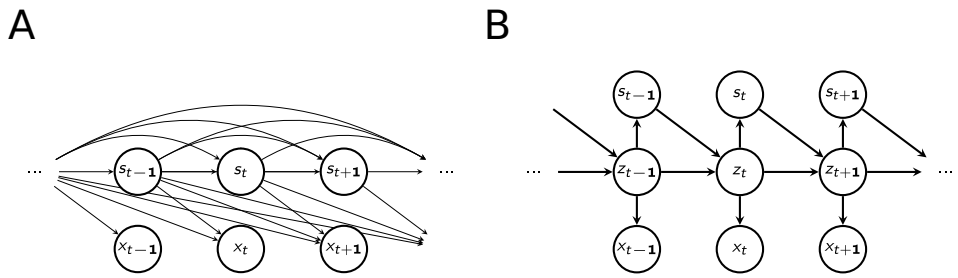

Fig. S1. Graphical representation of the decoder model. **A** We consider a decoding model where readouts of inputs $\mathbf{x}(t)$ (denoted here as $\mathbf{x}_t$) are conditioned on preceding spikes $\mathbf{s}(t)$ (denoted as $\mathbf{s}_t$). **B** By introducing the spike traces $\mathbf{z}(t)$ into the model, the model factorizes over timesteps, which is equivalent to viewing it as a hidden Markov model (HMM) with hidden states $\{\mathbf{z}(t), \mathbf{s}(t)\}$.

a) $\mathbf{X}_{0,T}$ can be accurately estimated from $\mathbf{S}_{0,T}$ via a decoding model $p_\theta(\mathbf{X}_{0,T}|\mathbf{S}_{0,T})$.

b) The number of spikes emitted is small.

Hence we want to maximize the likelihood $p_\theta(\mathbf{X}_{0,T}|\mathbf{S}_{0,T})$ over both the decoding model parameters θ and the latent variables $\mathbf{S}_{0,T}$ sampled by the (constrained) network dynamics $p_{\text{dyn}}(\mathbf{s}(t)|\mathbf{x}(t), \mathbf{z}(t))$.

To show how a stochastic spiking neural network can unsupervisedly learn such an encoding, we make use of the framework of online expectation-maximization (EM) learning (8). EM-learning can find maximum-likelihood estimates for parameters of latent variable models (here $p_\theta(\mathbf{X}_{0,T}, \mathbf{S}_{0,T})$) for observed data ($\mathbf{X}_{0,T}$). For these models it typically is intractable to marginalize out the latent variables ($\mathbf{S}_{0,T}$). In order to solve this problem one defines the log-likelihood lower bound

$$\begin{aligned}\mathcal{F}^*(\tilde{p}, \theta) &= \langle \log p_\theta(\mathbf{X}_{0,T}) - D_{KL}(\tilde{p}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})|p_\theta(\mathbf{S}_{0,T}|\mathbf{X}_{0,T}))\rangle_{p^*(\mathbf{X}_{0,T})} \\ &= \langle \log p_\theta(\mathbf{X}_{0,T}, \mathbf{S}_{0,T}) - \log \tilde{p}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})\rangle_{\tilde{p}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})p^*(\mathbf{X}_{0,T})}\end{aligned}, \quad [2]$$

where $\tilde{p}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})$ can be any (tractable) probability distribution, which in our case will be given through p_{dyn}. Finding maximum-likelihood parameters θ can then be done by iteratively maximizing $\mathcal{F}^*(\tilde{p}, \theta)$ with respect to $\tilde{p}$ (E-step) and θ (M-step). In the E-step p_θ is approximated by $\tilde{p}$ in order to estimate $\langle \log p_\theta(\mathbf{X}_{0,T}, \mathbf{S}_{0,T})\rangle_{\tilde{p}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})p^*(\mathbf{X}_{0,T})}$ and in the M-step this approximation is used to improve the model. This algorithm is guaranteed to converge to a local minimum, also if $\mathcal{F}^*$ is maximized only partially in every iteration, which makes it applicable to online learning.

Appealing to this theory in the following we show that: (i) Given a linear decoding model, a stochastic spiking neural network can be connected such that it can sample an efficient encoding online. This relates model- and network-parameters. (ii) The decoding model can be optimized online in respect to the sampled dynamics of the network. (iii) Combining (i) (the E-step) and (ii) (the M-step) yields update rules that can be applied by a stochastic spiking neural network to optimize its parameters in order to encode its inputs.

B.1. Online encoding by spiking neural network. Let us consider the following decoding model and prior on the spiking probability (Fig S1)

$$p_\theta(\mathbf{X}_{0,T}|\mathbf{S}_{0,T}) = \prod_t p_\theta(\mathbf{x}(t)|\mathbf{z}(t)) = \prod_t \mathcal{N}_{\mathbf{x}(t)}(D\mathbf{z}(t), \Sigma)$$

$$p_\theta(\mathbf{S}_{0,T}) = \prod_t p_\theta(\mathbf{s}(t)|\mathbf{z}(t)) = \prod_t \frac{1}{Z(\mathbf{b})} \exp\left(\mathbf{s}(t)^\top \mathbf{b}\right)$$

with $\Sigma = \sigma^2 I$ and parameters $\theta = \{D, \sigma, \mathbf{b}\}$. Notably this model asserts that at every time t, $\mathbf{x}(t)$ can be linearly decoded from spike traces $\mathbf{z}(t)$ with variance σ^2, where the spike traces are defined as before. Observe that the spike traces $\mathbf{z}(t)$ are deterministically defined given the preceding spikes $\mathbf{S}_{0,t-1}$. Also note that with the diagonal correlation matrix Σ, the decoder model assumes zero correlations between decoding errors. Input signals for which this assumption likely holds are for example signals with zero pairwise correlations between dimension, e.g. signals that have been whitened.

Since the model factorizes over time given the spike traces $\mathbf{z}(t)$, the log-likelihood lower bound (Eq 2) can be rewritten as

$$\begin{aligned}\mathcal{F}^*(p_{dyn}, \theta) &= \left\langle \sum_t \log p_\theta(\mathbf{x}(t), \mathbf{s}(t)|\mathbf{z}(t)) - \log p_{dyn}(\mathbf{s}(t)|\mathbf{x}(t), \mathbf{z}(t)) \right\rangle_{p_{dyn}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})p^*(\mathbf{X}_{0,T})} \\ &= \langle \log p_\theta(\mathbf{X}_{0,T})\rangle_{p^*(\mathbf{X}_{0,T})} - \\ &\quad \left\langle \sum_t \log p_{dyn}(\mathbf{s}(t)|\mathbf{x}(t), \mathbf{z}(t)) - \log p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{z}(t)) \right\rangle_{p_{dyn}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})p^*(\mathbf{X}_{0,T})}\end{aligned}$$

Here we substituted $\tilde{p}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T}) = \prod_t p_{dyn}(\mathbf{s}(t)|\mathbf{x}(t), \mathbf{z}(t))$ and made use of the facts that spikes alter only the future decoding and that they are independent of the past given $\mathbf{z}(t)$, i.e. $p_\theta(\mathbf{s}(t)|\mathbf{X}_{0,T}, \mathbf{S}_{0,t-1}) = p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{z}(t))$.

We now perform the E-step. $\mathcal{F}^*$ is approximately maximized over p_{dyn} if $p_{\text{dyn}}(\mathbf{s}(t)|\mathbf{x}(t), \mathbf{z}(t)) \approx p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{z}(t))$ at every time t. However, this poses two problems:

(i) p_{dyn} depends only on $\mathbf{x}(t)$ while the spike probability in the model is based on future values $\mathbf{X}_{t+1,T}$, which are not available to the network.

(ii) In order to compute $p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{z}(t)) = \sum_{\mathbf{S}_{t+1,T}} p_\theta(\mathbf{S}_{t,T}|\mathbf{X}_{t+1,T}, \mathbf{z}(t))$ future spikes have to be marginalized out, which is intractable.

For the purpose of this paper we introduced simple approximations that solve these problems and work well in practice for our inputs. Specifically we assumed input- and network activity to be approximately constant over time. Hence all future inputs $\mathbf{x}(t') \in \mathbf{X}_{t+1,T}$ were assumed to be known to be $\mathbf{x}(t') = \mathbf{x}(t)$. Future network activity (independent of the current spike $\mathbf{s}(t)$) was assumed to be well approximated by a single trajectory, where neural outputs $\mathbf{z}(t)$ were constant. With this we can compute

$$
\begin{aligned}
&\sum_{\mathbf{S}_{t+1,T}} p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{z}(t), \mathbf{S}_{t+1,T}) p_\theta(\mathbf{S}_{t+1,T}|\mathbf{X}_{t+1,T}, \mathbf{z}(t)) \\
&\quad \approx \prod_{t'=t}^{T} p_\theta\left(\mathbf{s}(t)|\mathbf{x}(t') = \mathbf{x}(t), \mathbf{z}(t') = \mathbf{z}(t) + \mathbf{s}(t)\exp\left(-\frac{t'-1-t}{\tau}\right)\right) \\
&\quad = \frac{1}{Z(\theta, \mathbf{x})} \exp\left(\mathbf{s}(t)^\top \mathbf{b}\right) \prod_{t'=t+1}^{T} \exp\left(\frac{\mathbf{z}(t')^\top}{\sigma^2}[D^\top \mathbf{x}(t') - \frac{1}{2} D^\top D \mathbf{z}(t')]\right) \\
&\quad = \frac{1}{Z(\theta, \mathbf{x})} \exp\left(\mathbf{s}(t)^\top \mathbf{b}\right) \prod_{t'=t+1}^{T} \exp\left(\frac{\left(\mathbf{z}(t) + \mathbf{s}(t)\exp\left(-\frac{t'-1-t}{\tau}\right)\right)^\top}{\sigma^2}\left[D^\top \mathbf{x}(t) - \right.\right. \\
&\qquad \left.\left. - \frac{1}{2} D^\top D\left(\mathbf{z}(t) + \mathbf{s}(t)\exp\left(-\frac{t'-1-t}{\tau}\right)\right)\right]\right) \\
&\quad = \frac{1}{Z(\theta, \mathbf{x}, \mathbf{z})} \exp\left(\mathbf{s}(t)^\top \left[\mathbf{b} + \frac{\tau}{\sigma^2} D^\top \mathbf{x}(t) - \frac{\tau}{\sigma^2} D^\top D\left(\mathbf{z}(t) + \frac{1}{4}\mathbf{s}(t)\right)\right]\right) \\
&\quad = \frac{1}{Z(\theta, \mathbf{x}, \mathbf{z})} \exp\left(\mathbf{s}(t)^\top \left[\mathbf{b} + \frac{\tau}{\sigma^2} D^\top \mathbf{x}(t) - \frac{\tau}{\sigma^2} D^\top D\mathbf{z}(t) - \frac{1}{4}\frac{\tau}{\sigma^2}\text{diag}(D^\top D)\right]\right)
\end{aligned}
$$

where we approximated $\sum_{t'=t+1}^{T} \exp(-\frac{t'-1-t}{\tau}) \approx \tau$ (that is τ and T large) and the last equality follows if timesteps are sufficiently small such that only one neuron spikes per timestep. Comparing with the network dynamics (Eq 1) from this we can conclude that a network that performs approximate online sampling from $p_\theta(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})$ has parameters

$$
\begin{aligned}
F &= D^\top \\
W &= -D^\top D \\
T_j &= \frac{1}{4} W_{jj} - \frac{\sigma^2}{\tau} b_j \\
\Delta u &= \frac{\sigma^2}{\tau}
\end{aligned}
\qquad [3]
$$

These results are similar to those yielded by a greedy spike encoding scheme (2). Please note that the sampling could be improved by using advanced sampling schemes, such as rejection sampling (6).

B.2. Online learning of an optimal decoder. As we have shown, the network dynamics implement an approximation of the E-step if the network parameters are chosen correctly. We will now use these samples produced by the network to incrementally improve the parameters of the decoding model in the M-step.

Recall that in the M-step we want to maximize under θ

$$
\left\langle \sum_t \log p_\theta(\mathbf{x}(t), \mathbf{s}(t)|\mathbf{z}(t)) \right\rangle_{p_{dyn}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T}) p^*(\mathbf{X}_{0,T})}.
$$

Updates of the decoder model parameters should thus follow the gradient

$$
\Delta\theta = \tilde{\eta}_\theta \frac{\partial \mathcal{F}^*}{\partial \theta} = \tilde{\eta}_\theta \left\langle \sum_t \frac{\partial}{\partial \theta} \log p_\theta(\mathbf{x}(t), \mathbf{s}(t)|\mathbf{z}(t)) \right\rangle_{p_{dyn}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T}) p^*(\mathbf{X}_{0,T})}
$$

In this paper we're only interested in the decoder weights D_{ij} from neuron j to input i, where the derivation yields

$$\Delta D_{ij} = \tilde{\eta}_D \left\langle \sum_t \sigma^{-2} z_j(t) \left(x_i(t) - \sum_k D_{ik} z_k(t) \right) \right\rangle_{p_{dyn}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T}) p^*(\mathbf{X}_{0,T})}$$

Here, $\tilde{\eta}_D$ is the update step size and σ^2 the variance of the decoder model. Note that in the following we will drop the dependence of the learning rate on σ^2, which has its motivation in covariant optimization (9). In covariant optimization, the gradient is multiplied by the inverse curvature of the loss function, because step size should be decreased when the curvature of the loss function is high. Since the curvature of the likelihood is proportional to the inverse variance, the variance drops out to yield a covariant gradient. This yields the update rule

$$\Delta D_{ij} = \tilde{\eta}_D \left\langle \sum_t z_j(t) \left(x_i(t) - \sum_k D_{ik} z_k(t) \right) \right\rangle_{p_{dyn}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T}) p^*(\mathbf{X}_{0,T})}$$

B.2.1. Online approximation. If many episodes $\mathbf{X}_{0,T}$ as sampled from $p^*(\mathbf{X}_{0,T})$ are presented in succession and spikes are sampled as outlined above, the average over samples from p_{dyn} can be replaced by an average over time

$$\left\langle \sum_t \cdot \right\rangle_{p_{dyn}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T}) p^*(\mathbf{X}_{0,T})} \approx \sum_t \langle \cdot \rangle_t \,.$$

If the update rules are performed every timestep this lets us rewrite them as

$$\delta D_{ij} = \eta_D z_j(t) \left(x_i(t) - \sum_k D_{ik} z_k(t) \right) \quad [4]$$

This requires, however, that the learning rate η_D is sufficiently small such that changes in D_{ij} are negligible in a sufficiently long time window T'. In that case, summing the equation over time window T' yields

$$\sum_{t=0}^{T'} \delta D_{ij} = \eta_D T' \left\langle z_j(t) \left(x_i(t) - \sum_k D_{ik} z_k(t) \right) \right\rangle_{t=0}^{T'} \overset{T' \to T}{\approx} \Delta D_{ij},$$

where the learning rates are related via $\tilde{\eta}_D = \eta_D T'$. A more refined statement can be made by rewriting the update equation as

$$\Delta D_{ij} = \tilde{\eta}_D \left(\langle z_j(t) x_i(t) \rangle_{p_{dyn}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T}) p^*(\mathbf{X}_{0,T})} - \sum_k D_{ik} \langle z_j(t) z_k(t) \rangle_{p_{dyn}(\mathbf{S}_{0,T}|\mathbf{X}_{0,T}) p^*(\mathbf{X}_{0,T})} \right)$$

This makes explicit that the only information required to compute the gradient of the decoder weights are the correlations between neural outputs and inputs and in between neural outputs over the input sequences. Thus in practice, the learning rate η_D is ideally chosen as large as possible to allow fast learning, but also sufficiently small such that weight updates are performed with respect to a time window that provides a good estimate of correlations under the whole sampled spike trains.

B.3. Online learning of network parameters. So far we showed that the parameters of a network that samples from a decoder model are directly connected to the parameters of the model. We also showed how the decoder weights have to be updated such that they maximize the model likelihood over the generated samples. We will now combine these two results to find update rules for neural parameters directly, that can be used by neurons to learn an efficient encoding without supervision online. To this end we will first show how an approximation to the previously derived gradients can be implemented by regular stochastic LIF neurons. In a second step we will show how a better approximation can be realized by neurons with dendritic compartments. The central insight for all derivations will be that learning an E-I balance on membrane potentials corresponds to the learning of a decoder to the excitatory inputs times a transformation matrix that brings them into the space of the membrane potentials.

B.3.1. Somatic balance approximation.

Feedforward weights From the equality $F = D^\top$ (Eq 3) derived earlier and the update rule for D (Eq 4) we directly arive at

$$\delta F_{ji} = \eta_F z_j(t) \left(x_i(t) - \sum_k F_{ki} z_k(t) \right)$$

We follow previous approaches (2) and omit contributions to the decoding $\sum_k F_{ki} z_k(t)$ that are not available for the neuron, which is equivalent to assuming that neural spiking in the population is uncorrelated $\forall j \neq k : \langle z_j(t) z_k(t) \rangle_t \approx 0$. This yields the regularized Hebbian rule

$$\delta F_{ji} = \eta_F z_j(t) \left(x_i(t) - F_{ji} z_j(t) \right) \quad [5]$$

Recurrent weights This rule will follow the optimal decoder gradient if spikes are indeed uncorrelated. However, if this is not the case the solution will be suboptimal and furthermore the previously derived recurrent weights $W = -D^\top D$ together with the suboptimal weights F does not enable a reasonable encoding anymore. Both problems can be addressed by observing that for the optimal membrane potential we derived

$$\mathbf{u}^{opt}(t) = D^\top \mathbf{x}(t) - D^\top D\mathbf{z}(t) = D^\top(\mathbf{x}(t) - D\mathbf{z}(t)),$$

i.e. the potentials are proportional to the decoding error. This can be approximately guaranteed even if the feedforward weights are suboptimal (but not zero) by setting $W = -FD$, since then

$$\mathbf{u}(t) = F\mathbf{x}(t) - FD\mathbf{z}(t) = F(\mathbf{x}(t) - D\mathbf{z}(t)) \overset{\approx}{\propto} \mathbf{u}^{opt}(t).$$

To make sure that neurons adapt their encoding for an optimal decoder, recurrent weights will adapt along the gradient of decoder weights. For fixed encoder weights F this yields

$$\begin{aligned}
\delta W_{jk} &= -\sum_i F_{ji}\delta D_{ik} \\
&= -\eta_W z_k(t)\left(\sum_i F_{ji}x_i(t) - \sum_{i,l} F_{ji}D_{il}z_l(t)\right) \\
&= -\eta_W z_k(t)\left(\sum_i F_{ji}x_i(t) + \sum_l W_{jl}z_l(t)\right) \\
&= -\eta_W z_k(t)u_j(t)
\end{aligned} \qquad [6]$$

This shows that through an E-I balance, this rule for W self-consistently finds the correct decoder 'inside' of the recurrent weights, and hence allows the projection of the right decoding error $x - Dz$. Thereby recurrent connections ensure a reasonable encoding even if feedforward weights are not learned optimally. Since in the equation above F_{ji} is assumed constant, we chose the learning rate η_W 2-4 times larger than η_F. In simulations we found that recurrent weights that evolve under Eq 6 indeed converged to $W = -FD$, where D is the optimal decoder weights obtained under the non-local update rule Eq 4 .

B.3.2. Learning encoder weights with dendritic balance . In the following we devise examples for local plasticity rules for feedforward inputs that follow the correct gradient of the likelihood lower bound. Locality requires that the decoding of other neurons is made available at the synapse, which can then be used to find the correct gradient. We argue that this can be mediated by dendritic recurrent connections W^i that target dendrites where the feedforward input i has formed a synapse. Due to strong attenuation between dendritic compartments, the membrane potential u_j^i in the vicinity of synapse i on that dendrite only integrates inputs that are present locally, i.e.

$$u_j^i(t) = \underbrace{F_{ji}x_i(t)}_{\text{feedforward input}} + \underbrace{\textstyle\sum_k W_{jk}^i z_k(t)}_{\text{recurrent input}}. \qquad [7]$$

We then assume a regime where currents from the dendrites are summed linearly, such that the total membrane potential at the soma is given by $u_j(t) = \sum_i u_j^i(t)$. Similar to recurrent somatic connections, we will show that recurrent dendritic connections can locally learn an optimal decoding of neural PSPs $\mathbf{z}$ by enforcing *dendritic balance* of feedforward and recurrent inputs. The central feature of this approach is that feedforward and recurrent connections both use the dendritic potential for learning, which requires their cooperation. We here show three possible mechanisms that realize this and yield very similar behaviour to the analytical solution (Fig S2, S3).

Slow feedforward adaptation One possibility to ensure the cooperation of feedforward and recurrent weights is to separate the timescales on which they are adapting. To that end we make the optimal ansatz for recurrent weights similar to before $W_{jk}^i = -F_{ji}D_{ik}$. Then, changing recurrent weights in the direction of the decoder gradient of Eq 4 yields

$$\begin{aligned}
\delta W_{jk}^i &= -F_{ji}\delta D_{ik} \\
&= -\eta_W z_k(t)(F_{ji}x_i(t) - \textstyle\sum_l F_{ji}D_{il}z_l(t)) \\
&= -\eta_W z_k(t)(F_{ji}x_i(t) + \textstyle\sum_l W_{jl}^i z_l(t)) \\
&= -\eta_W z_k(t)u_j^i(t).
\end{aligned}$$

where we again assumed that changes in feedforward weights are slow and can be neglected, and $\eta_W = \eta_D$. We conclude that enforcing dendritic balance by recurrent plasticity is equivalent to locally optimizing a decoder $D_{ik} = -W_{jk}^i/F_{ji}$.

The correct gradient of the decoder weights can also be calculated locally, but it can't be applied to the feedforward weights directly since this would contradict the previously made assumption of slow changes in feedforward weights. However, it is

possible to locally integrate the correct gradient and use this to adapt feedforward weights slowly, with a delay. To this end we introduce the local integration variable $I_{ji} = F_{ji}D_{ij}$, which adapts according to the decoder gradient times F_{ji}

$$\begin{aligned}\delta I_{ji} =&\eta_I F_{ji} z_j(t) \left(x_i(t) - \textstyle\sum_k D_{ik} z_k(t)\right) \\ =&\eta_I z_j(t) u^i_j(t),\end{aligned} \tag{8}$$

with $\eta_I = \eta_D$. F_{ji} then can slowly follow D_{ij} via

$$\delta F_{ji} =\eta_F (I_{ji}/F_{ji} - F_{ji}),$$

with $\eta_F \ll \eta_D$. Note that for $F_{ji} = 0$ the gradient for F_{ji} is not defined. In this case the learning process could be kickstarted via simple Hebbian learning on F_{ji}. Note also that the equation $W^i_{jk} = -F_{ji}D_{ik}$ has to hold at the start of learning, which can be guaranteed by simply choosing $W^i_{jk} = F_{ji} = 0$. To summarize, slow feedforward adaptation leads to neural parameters $W^i_{jk} = -F_{ji}D_{ik}$ and $F_{ji} = D_{ij}$. This shows that feedforward synapses slowly can evolve to minimize the decoder error along its gradient using only local information.

Simultaneous adaptation of feedforward and recurrent weights In principle it would also be possible to adapt feedforward and recurrent weights simultaneously without a separation of timescales. However, calculating the gradient for the derived recurrent weights is locally not feasible, since we find

$$\begin{aligned}\delta W^i_{jk} =& - D_{ji}\delta D_{ik} - \delta D_{ji} D_{ik} \\ =& - \eta_D (z_k(t) u^i_j(t) + z_j(t) u^i_k(t)).\end{aligned}$$

Empirically we found that the contralateral contributions $z_j(t)u^i_k(t)$ to the gradient can be approximated by the accessible contributions $z_k(t)u^i_j(t)$. We thus approximate $\langle z_j(t)u^i_k(t)\rangle_t \approx \langle z_k(t)u^i_j(t)\rangle_t$. While this equation does not hold for all i, j, k, we still find that the resulting learned contributions to the dendritic potentials have the correct magnitude, hence enabling feedforward learning. The gradient for the recurrent weights now are

$$\delta W^i_{jk} = - \eta_W z_k(t) u^i_j(t),$$

where $\eta_W = 2\eta_D$.

Assuming the correct recurrent weights $W^i_{jk} = -D_{ji}D_{ki}$ we can find the decoded population encoding locally at the dendrite. From the self-consistency $F_{ji} = D_{ij}$ and Eq 7 we have the relation

$$\sum_k D_{jk} z_k(t) = \frac{F_{ji}x_i(t) - u^i_j(t)}{F_{ji}}.$$

With this we can implement the learning of feedforward weights in way that highlights its similarity to previous approaches (Eq 5)

$$\delta F_{ji} = \eta_F z_j(t) \left(x_i(t) - \frac{F_{ji}x_i(t) - u^i_j(t)}{F_{ji}}\right),$$

i.e. the rule is a regularized Hebbian plasticity rule. Again, for very small F_{ji} the regularization becomes unstable, but can be left away (since it should go to zero) leaving a purely Hebbian rule. For the derivation we used $\eta_F = \eta_D$, which implies that we should choose $\eta_W \approx 2\eta_F$.

In simulations we verified that the approximations we made for this learning scheme are adequate and yield feed forward weights for which $F_{ji} = D_{ij}$ holds with high accuracy. Note that the network found by the presented learning scheme only corresponds to the decoding model if $\eta_W \approx 2\eta_F$. However, if the recurrent learning is faster this only results in a rescaling of feedforward weights by a factor of $2\eta_F/\eta_W$, since their adaptation is too slow by this factor. This means that in this case the "correct" dynamics of the network can be recovered via a rescaling of all weights, or equivalently, with firing rate adaptation a change in the stochasticity of spiking Δu by a factor of $2\eta_F/\eta_W$.

Learning of feedforward and recurrent weights via the weight decay trick For both learning schemes we have presented so far, the relation of feedforward and recurrent weights and their learning rates $\eta_{(\cdot)}$ are critical for learning, as changes in recurrent weights directly impact how feedforward weights are learned and vice versa. This can become problematic, if the recurrent weights are not initialized as $W^i_{jk} = -F_{ji}F_{ki}$ or if for some reason the match of feedforward and recurrent weights is disturbed during learning. This problematic co-dependence of the learning rules can be avoided via a simple trick, which we will call the weight decay trick. To this end we introduce a small weight decay with rate λ_j on the decoder weights

$$\delta D_{ij} = \eta_D z_j(t) \left(x_i(t) - \sum_k D_{ik} z_k(t)\right) - \lambda_j D_{ij}. \tag{9}$$

By doing so, we can readily derive an implicit equation for the fixed point of this update rule, which is

$$D^*_{ij} = \left\langle \lambda_j^{-1} z_j(t) \left(x_i(t) - \sum_k D^*_{ik} z_k(t)\right)\right\rangle_t = \left\langle \lambda_j^{-1} F_{ji}^{-1} z_j(t) u^i_j(t)\right\rangle_t.$$

This equation holds if recurrent weights were learned to approximately equal $W^i_{jk} = -F_{ji}D^*_{ik}$. This can be achieved by updating recurrent weights until convergence with

$$\begin{aligned}\delta W^i_{jk} &= -F_{ji}\delta D_{ik} \\ &= -\eta_W \left(z_k(t)u^i_j(t) - \lambda_k W^i_{jk}\right).\end{aligned}$$

Now the optimal feedforward weights can be learned by slowly tracking the fixed point D^*_{ij}, which can be computed locally

$$\begin{aligned}\delta F_{ji} &= \eta_F \lambda_j \left(D^*_{ij} - F_{ji}\right) \\ &\approx \eta_F \left(F^{-1}_{ji} z_j(t) u^i_j(t) - \lambda_j F_{ji}\right).\end{aligned}$$

Here the pre-factor λ_j normalizes the learning speed. Interestingly, this learning rule is simply the gradient for feedforward weights, as calculated before, with additional weight-decay similar to the recurrent learning rule. The difference of this learning scheme to the previous two learning schemes is that inhibition will not perfectly balance excitation, but the balance will be offset by a small amount. Feedforward learning then relies on this small mismatch between excitation and inhibition to find a good encoding and thereby avoids the problematic co-dependence of feedforward and recurrent learning.

How does this learning scheme, which evidently relies on a different decoder update, relate to the previously derived network dynamics corresponding to the optimal decoder? A valid concern would be that an offset in the E-I balance could lead to elevated or reduced spiking rates. The answer is that there exists a close relation between the weight decay λ_j and the spiking prior b_j, which helps to ensure optimal spiking. More technically, the weight decay of the decoder can be seen as a constraint on the L2-norm of decoder weights, to compensate for a fixed, sub-optimal threshold. To understand this, we start with the equation for the optimal threshold T_j (Eq 3). If the threshold T_j is fixed to an arbitrary value, this equation directly implies a length constraint on the decoder

$$\begin{aligned}T_j &= -\frac{1}{4}\sum_i D^2_{ij} - \frac{\sigma^2}{\tau} b_j \\ \Leftrightarrow \sum_i D^2_{ij} &= -4T_j - \frac{4\sigma^2}{\tau} b_j \stackrel{\text{def}}{=} a_j.\end{aligned}$$

This means that neurons can not only fire optimally for a given prior b_j by changing their thresholds in accordance with the strength of incoming connections, but also by changing the overall connection strength while keeping the threshold fixed. This constraint can be included into the optimization by augmenting the decoder loss (containing all relevant contributions of the likelihood, Eq 2) via Lagrangian optimization

$$\mathcal{L}(D) = \frac{1}{2}\left\langle ||\mathbf{x}(t) - D\mathbf{z}(t)||^2 \right\rangle_t + \sum_j \lambda_j \frac{1}{2}\left(||D_j||^2 - a_j\right).$$

From this loss the decoder update with weight decay (Eq 9) directly arises via gradient descent. Here, the Lagrangian multipliers λ_j correspond to specific firing priors of neurons b_j for some fixed threshold T_j. It is therefore possible to obtain correct network dynamics by either adapting λ_j according to $\delta\lambda_j \propto -\frac{\partial \mathcal{L}}{\partial \lambda_j}$, or by simply treating λ_j as a parameter of the model instead of b_j. It is therefore also evident that changes in network dynamics in comparison to the analytical network (Eq 3) are minimal as long as the λ_j are small. Especially with additional rapid firing rate adaptation, which we are using in our simulations, the difference to the analytical solution is negligible for small λ_j, as here the impact of λ_j on the firing rate is 'overwritten' by the rapidly adapting threshold.

B.3.3. Rapid firing rate adaptation. In the Bayesian framework Habenschuss and colleagues have shown that a rapid rate adaptation can be interpreted as a constraint on the variational approximation in the E-step (10). For the resulting constrained optimization formally a Lagrange multiplier is introduced which 'overwrites' the analytic threshold $T_j = \frac{1}{4}W_{jj} - \sigma^2\tau^{-1}b_j$. We will not make a notational difference between the two thresholds here. The fixed firing rate is enforced by adapting the threshold T_j such that neurons are firing with a target firing rate ρ.

$$\delta T_j = \eta_T(s_j - \rho\,\delta)$$

Here, $\rho\,\delta$ is the mean number of spikes in a time window of size δ if a neuron would spike with rate ρ and s_j is a spike indicator which is 1 if z_j spiked in the last time δ, otherwise $s_j = 0$. Since this is a constraint that is applied in the E-step, the learning rate η_T should be large.

B.3.4. Pruning recurrent weights. In the proposed learning schemes the number of recurrent connections grows very fast with network and input size (# of inh. conn. $= N_x \times N_z^2$). We here propose a principle by which recurrent connections that provide little contribution to neural learning can be pruned away (Please note that the following principle only considers *learning*; for correct *dynamics* it might be necessary to keep additional somatic weights that ensure efficient spiking). To identify these weights we again look at the learning rule of the proposed slow feedforward weight adaptation scheme (Eq 8)

$$\begin{aligned}\delta I_{ji} =& \eta_I z_j(t) u_j^i(t) \\ =& \eta_I z_j(t) \left(F_{ji} x_i(t) + \textstyle\sum_k W_{jk}^i z_k(t) \right).\end{aligned}$$

Here, recurrent connections that provide no systematic contribution to the gradient can be left away. In particular, those are connections W_{jk}^i for which $\langle z_j(t) W_{jk}^i z_k(t) \rangle_t \approx 0$. In other words, only large weights matter that connect neurons with correlated activities. Hence, the number of required weights for learning depends primarily on the sizes of neural receptive fields (as $W_{jk}^i \approx -F_{ji} F_{ki}$) and the number of correlated coding neurons and not the size of the network and input.

Based on this observation, one possibility to prune weights is for example to remove a certain fraction of the weights and leaving only the weights with the largest $|\langle z_j(t) W_{jk}^i z_k(t) \rangle_t|$. In biological neurons potential connections W_{jk}^i could continuously be probed and only be stably formed if their contribution is sizeable. For the bar task, we demonstrated that this allows us to prune a very large fraction of recurrent weights without compromising performance (Fig S9).

It is important to note that in no case feedforward weights should remain un-regularized, that is, the learning rule is purely Hebbian, as this would lead to unbounded growth of weights. The best solution to this problem is to always keep self-contributions to the gradient $W_{jj}^i z_j \approx -F_{ij} F_{ij} z_j$. This results in the same regularization as it is used in the somatic balance model and can arguably be always computed locally.

C. Relation to previous studies of representation learning

C.1. Comparison to other Hebbian-like learning rules. The Hebbian-like learning rule used in the somatic balance model is part of a group of Hebbian-like learning rules that have been used in the past to model representation learning in recurrent populations of neurons. We here present a (non-exhaustive) overview over such rules that learn feedforward weights F_{ji} (Table S1). All these rules can be seen as successors of the well known Oja's rule (11), which can be written in the form

$$\Delta F_{ji} \propto z_j(x_i - F_{ji}z_j),$$

where x_i is some input and $z_j = \sum_i F_{ji}x_i$ is the activity of a (linear) neuron. Specifically, all rules we will present can be written in the more general form

$$\Delta F_{ji} \propto \text{post} \times (\text{pre} - f(F_{ji}) \times \text{post}),$$

where "post" and "pre" denote aspects of post- and presynaptic activity, respectively, and $f(\cdot)$ is some function of the weight. We will write $s_j \in \{0, 1\}$ to denote a binary spike indicator and z_j to denote some form of analog postsynaptic activity.

Note that by itself Oja's rule always extracts the largest principal component of the input data x_i. This means that in order to learn non-redundant representations in a network, some form of recurrent inhibitory coupling is required. Importantly, as we have argued, in order to be generally applicable it requires inhibition that is nearly instantaneous and therefore biologically implausible. Consequently, most models we present here make use of some form of instantaneous (or implausible) inhibition. Some of the models get around this constraint by other means, e.g. by forcing zero correlations in an extremely slow-firing regime (12), or have only been tested for very simple scenarios (13).

Table S1. List of related papers modeling representation learning with Hebbian-like plasticity.

Reference	Rule	Comment
Foeldiak (1990) (14)	$\Delta F_{ji} \propto s_j(x_i - F_{ji}s_j)$	This paper uses binary neurons, where outputs s_j are determined by an optimization scheme.
Kung et al (1990) (15)	$\Delta F_{ji} \propto z_j(x_i - F_{ji}z_j)$	This paper uses linear neurons, where outputs z_j are determined in a strictly sequential manner.
Zylberg et al (2011) (12)	$\Delta F_{ji} \propto z_j(x_i - F_{ji}z_j)$	z_j is a spike-counter over a certain time window. This paper uses LIF neurons with recurrent inhibition in a slow-firing regime.
Kappel et al (2014) (6) (similarly (5))	$\Delta F_{ji} \propto s_j(x_i - e^{F_{ji}}s_j)$	To achieve "canonical" form we multiplied the rule with $e^{F_{ji}}$, changing the learning speed, but not the fixed point. This paper uses stochastic spiking neurons in a winner-take-all circuit.
Bill et al (2015) (4)	$\Delta F_{ji} \propto s_j(x_i - \text{sig}(F_{ji} + F_{0i})s_j)$	sig$(\cdot)$ is the logistic function and F_{0i} is a baseline constant. This paper uses stochastic spiking neurons in a winner-take-all circuit.
Bahroun et al (2017) (16)	$\Delta F_{ji} \propto [\sum_{t'}^{t} z_j(t')^2]^{-1} z_j(x_i - F_{ji}z_j)$	Learning speed is regulated with a pre-factor. This paper uses analog neurons, where outputs z_j are the results of an optimization scheme.
Pehlevan et al (2017) (17)	$\Delta F_{ji} \propto [\sum_{t'}^{t} z_j(t')^2]^{-1} z_j(x_i - F_{ji}z_j)$	This paper proposes a network with very similar behavior to (16), but performs non-negative source separation.
Jonke et al (2017) (13)	$\Delta F_{ji} \propto s_j(x_i - \text{sig}(\gamma F_{ji})s_j)$	sig$(\cdot)$ is the logistic function and γ is a scaling parameter. This paper uses stochastic spiking neurons in a k-winner-take-all circuit.
Tavanei et al (2018) (18)	$\Delta F_{ji} \propto s_j(x_i - (1-\lambda)F_{ji}s_j)$	λ is a sparsity factor. This paper uses spiking neurons in a winner-take-all circuit.
Brendel et al (2020) (2)	$\Delta F_{ji} \propto s_j(x_i - \alpha F_{ji}s_j)$	α is some regularization factor. This paper uses LIF neurons with recurrent inhibition and noisy potentials, resulting in a model that is practically identical to ours. Additionally, only one neuron is allowed to spike per time-bin.
This paper	$\Delta F_{ji} \propto z_j(x_i - F_{ji}z_j)$	This paper uses stochastic LIF neurons with recurrent inhibition and spike traces z_j.

C.2. Comparison to Brendel et al (2020). Our neural model and the model used by Brendel et al (2) are practically identical. Both models employ stochastic leaky integrate-and-fire neurons, which can be seen as instances of the spike response model with escape noise (19). Brendel et al employ a formulation with partial differential equations, while we use a formulation where the shape of PSP's is solved. Brendel et al add stochasticity to neural firing by adding Gaussian noise to the membrane potential, while we directly write down a probability function for spiking. Overall, this results in a stochastic neuron that is approximately equal to our probabilistic formulation (see e.g. (19), chapter 9.4).

The goal of coding is the same in both models. Hence, the feedforward learning rule of the somatic balance model is also, for all practical purposes in this paper, the same as it has been used by Brendel et al, which reads

$$\delta F_{ji} \propto s_j(t) \left(x_i(t) - \alpha F_{ji} s_j(t) \right), \qquad [10]$$

where α is some regularization factor. Notably, this rule only updates weights on spike-times, whereas our Hebbian-like rule (Eq 5) also incorporates non-spike-times into the update (the non-spike-times are an essential contribution in the dendritic balance learning scheme). For constant $x_i(t)$, which we use in our simulations, our rule can be integrated over time for a single spike $z_j(t) = s_j(t_s) \exp\left(-\frac{t-t_s}{\tau}\right)$ at time t_s:

$$\begin{aligned} \eta_F \int_{t=t_s}^{\infty} s_j(t_s) \exp\left(-\frac{t-t_s}{\tau}\right) \left(x_i(t) - F_{ji} s_j(t_s) \exp\left(-\frac{t-t_s}{\tau}\right) \right) dt \\ = \eta_F \tau s_j(t_s) \left(x_i(t) - 0.5 F_{ji} s_j(t_s) \right). \end{aligned} \qquad [11]$$

Hence, when spikes are rare events our rule is the same as the rule by Brendel et al, with $\alpha = 0.5$ and the learning rate η_F chosen appropriately. For fast spiking neurons the regularization is slightly different, since past spikes are taken into account when regularizing the weight. However, the overall learning outcome will be very similar since this only slightly changes the magnitude of the weight-vector. To verify this we adapted their implementation of the network (20), and found that the major effects we report in Fig 5F-H (SB) are preserved.

D. Datasets

MNIST The standard MNIST database of handwritten digits was used (21). Images were scaled down from 28×28 to 16×16 pixels. No further preprocessing was applied.

Correlated bars See description in Fig 4A. Pixels where bars are displayed (also in the case of overlap) were set to the value 1.0, pixels without bars were set to 0.0.

Natural scenes Images for natural scenes were taken from (22). A simple preprocessing was applied to ensure that they can be modeled by spiking neurons. Importantly we required that input is always positive. Every image χ in the database was whitened. This can be seen as an approximation of retinal on/off-cell preprocessing, where one on-cell and one off-cell with overlapping fields are lumped together in a single value χ_i, which can be positive or negative. We separated every value χ_i into two values $x'_{2i} = \chi_i$ and $x'_{2i+1} = -\chi_i$. We then applied a continuous nonlinear activation function to ensure that activations are positive and bimodally distributed (i.e. mostly close to either 0.0 or 1.0): $x_i = \mathrm{sig}(3.2(x'_i - 0.8))$, where $\mathrm{sig}(x) = 1/(1 + \exp(-x))$. For the display of learned weights we merge corresponding values again and display $x_{2i} - x_{2i+1}$.

Speech The speech data-set is the same as used in (2) and was taken from (20). The speech signal was presented in a spectral decomposition with 25 frequency channels, sampled at 200Hz with linear interpolation between data-points. The signal was spatially whitened using Cholesky whitening. After whitening we applied the same splitting and rectification procedure as for the natural scenes input signals.

In contrast to our results in Fig 5F-H, the original task in (2) uses the unwhitened signal directly as input. For this unwhitened input, (2) show that the somatic balance model requires a learning rule that - additionally to the Hebbian-like learning - performs spatial whitening to remove pairwise correlations in the signal. To check that our pre-processing does not significantly alter the results, we also performed simulations without whitening and using this alternative learning rule as used by (2) (Fig S11), and observed similar behavior as in Fig 5F-H.

E. Parameters

For all tasks parameters were tuned to ensure that networks operate well. *DB* denotes networks where the analytic solution given the decoder was used for network dynamics. *DB slow* are networks with slow feedforward adaptation, *DB simultaneous* are networks with parallel adaptation of feedforward and recurrent weights. *SB* are networks learning with somatic balance. When the parameter $\eta_{\Delta u}$ is present the stochasticity of spiking was annealed starting from 1.0 with rate $\eta_{\Delta u}$. Learning rates η_θ are given in units of ms^{-1}. Networks in all simulations were initialized with zero initial weights, except for Figs 3 and 5C-E, where feedforward weights were initialized as $F_{ji} = \exp(\max(0, 0.3 \cdot r_{ji} - 0.2)) - 1$ with $r_{ji} \sim \mathcal{N}(0, 1)$.

MNIST (Fig 3, 5)

Parameter	DB decay	SB
δ	0.1ms / 0.3ms	0.1ms / 0.3ms
τ	10ms	10ms
Δu	0.1	0.1
ρ	$15\mathrm{s}^{-1}$	$15\mathrm{s}^{-1}$
λ	0.03	-
η_T	$7.0 \cdot 10^{-3}$	$7.0 \cdot 10^{-3}$
η_F	$1.5 \cdot 10^{-5}$	$4.0 \cdot 10^{-6}$
η_W	$3.0 \cdot 10^{-5}$	$3.0 \cdot 10^{-5}$
η_D	$1.0 \cdot 10^{-6}$	$1.0 \cdot 10^{-6}$

MNIST (Fig S2)

Parameter	DB	DB simultaneous	DB slow	DB decay	SB
δ	0.1ms	0.1ms	0.1ms	0.1ms	0.1ms
τ	10ms	10ms	10ms	10ms	10ms
Δu	0.1	0.1	0.1	0.1	0.1
ρ	$20\mathrm{s}^{-1}$	$20\mathrm{s}^{-1}$	$20\mathrm{s}^{-1}$	$20\mathrm{s}^{-1}$	$20\mathrm{s}^{-1}$
λ	-	-	-	0.005	-
η_T	$5.0 \cdot 10^{-3}$	$3.0 \cdot 10^{-3}$	$5.0 \cdot 10^{-4}$	$5.0 \cdot 10^{-4}$	$5.0 \cdot 10^{-3}$
η_F	$5.0 \cdot 10^{-6}$	$3.0 \cdot 10^{-6}$	$4.0 \cdot 10^{-7}$	$2.0 \cdot 10^{-6}$	$5.0 \cdot 10^{-6}$
η_I	-	-	$4.0 \cdot 10^{-5}$	-	
η_W	-	$6.0 \cdot 10^{-6}$	$4.0 \cdot 10^{-5}$	$6.0 \cdot 10^{-5}$	$1.0 \cdot 10^{-5}$
η_D	$5.0 \cdot 10^{-6}$	$3.0 \cdot 10^{-6}$	$5.0 \cdot 10^{-6}$	$5.0 \cdot 10^{-6}$	$5.0 \cdot 10^{-6}$

Correlated bars (Fig 4, S3, S9)

Parameter	DB	DB simultaneous	DB slow	DB decay	SB
δ	1.0ms	1.0ms	1.0ms	1.0ms	1.0ms
τ	10ms	10ms	10ms	10ms	10ms
Δu^*	0.1	0.1	0.1	0.1	0.1
$\eta_{\Delta u}$	$7.0 \cdot 10^{-8}$	$7.0 \cdot 10^{-8}$	$7.0 \cdot 10^{-8}$	$7.0 \cdot 10^{-8}$	$7.0 \cdot 10^{-8}$
ρ	15s^{-1}	15s^{-1}	15s^{-1}	15s^{-1}	15s^{-1}
λ	-	-	-	0.005	-
η_T	$1.0 \cdot 10^{-2}$	$1.0 \cdot 10^{-2}$	$5.0 \cdot 10^{-2}$	$5.0 \cdot 10^{-2}$	$1.0 \cdot 10^{-2}$
η_F	$5.0 \cdot 10^{-5}$	$5.0 \cdot 10^{-5}$	$1.0 \cdot 10^{-7}$	$2.0 \cdot 10^{-5}$	$5.0 \cdot 10^{-5}$
η_I	-	-	$5.0 \cdot 10^{-5}$	-	-
η_W	-	$1.0 \cdot 10^{-4}$	$5.0 \cdot 10^{-5}$	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$
η_D	$5.0 \cdot 10^{-5}$	$5.0 \cdot 10^{-5}$	$5.0 \cdot 10^{-5}$	$5.0 \cdot 10^{-5}$	$5.0 \cdot 10^{-5}$

Natural scenes (Fig 4, 5, S4, S5, S6, S7, S8, S10)

Parameter	DB	SB
δ	0.2ms	0.2ms
τ	10ms	10ms
Δu^*	0.13	0.13
$\eta_{\Delta u}$	$7.0 \cdot 10^{-8}$	$7.0 \cdot 10^{-8}$
$\rho \cdot$ # neurons	1000s^{-1}	1000s^{-1}
η_T until $t = 6000$s	$6.0 \cdot 10^{-3}$	$6.0 \cdot 10^{-3}$
η_T until $t = \infty$	$4.0 \cdot 10^{-3}$	$4.0 \cdot 10^{-3}$
η_F until $t = 6000$s	$4.0 \cdot 10^{-5}$	$4.0 \cdot 10^{-5}$
η_F until $t = \infty$	$4.0 \cdot 10^{-5}$	$3.0 \cdot 10^{-5}$
η_W until $t = 6000$s	-	$10.0 \cdot 10^{-5}$
η_W until $t = \infty$	-	$7.0 \cdot 10^{-5}$
η_D until $t = 6000$s	$4.0 \cdot 10^{-5}$	$4.0 \cdot 10^{-5}$
η_D until $t = \infty$	$3.0 \cdot 10^{-5}$	$3.0 \cdot 10^{-5}$

Speech (Fig 5, S11)

Parameter	DB	SB
δ	0.05ms	0.05ms
τ	10ms	10ms
Δu	0.05	0.05
ρ	5s^{-1}	5s^{-1}
η_T	$1.4 \cdot 10^{-2}$	$1.4 \cdot 10^{-2}$
η_F	$2.1 \cdot 10^{-4}$	$2.1 \cdot 10^{-4}$
η_W	-	$5.6 \cdot 10^{-4}$
η_D	$2.1 \cdot 10^{-4}$	$2.1 \cdot 10^{-4}$

References

1. W Gerstner, WM Kistler, *Spiking neuron models: Single neurons, populations, plasticity*. (Cambridge university press), (2002).
2. W Brendel, R Bourdoukan, P Vertechi, CK Machens, S Denéve, Learning to represent signals spike by spike. *PLoS computational biology* **16**, e1007692 (2020).
3. B Nessler, M Pfeiffer, L Buesing, W Maass, Bayesian Computation Emerges in Generic Cortical Microcircuits through Spike-Timing-Dependent Plasticity. *PLoS Comput. Biol.* **9**, e1003037 (2013).
4. J Bill, et al., Distributed Bayesian Computation and Self-Organized Learning in Sheets of Spiking Neurons with Local Lateral Inhibition. *PLOS ONE* **10**, e0134356 (2015).
5. B Nessler, M Pfeiffer, W Maass, Stdp enables spiking neurons to detect hidden causes of their inputs in *Advances in neural information processing systems*. pp. 1357–1365 (2009).
6. D Kappel, B Nessler, W Maass, STDP Installs in Winner-Take-All Circuits an Online Approximation to Hidden Markov Model Learning. *PLoS Comput. Biol.* **10**, e1003511 (2014).
7. S Deneve, Bayesian Spiking Neurons II: Learning. *Neural Comput.* **20**, 118–145 (2007).
8. RM Neal, GE Hinton, A view of the em algorithm that justifies incremental, sparse, and other variants in *Learning in graphical models*. (Springer), pp. 355–368 (1998).
9. DJ MacKay, *Information theory, inference and learning algorithms*. (Cambridge university press), (2003).
10. S Habenschuss, J Bill, B Nessler, Homeostatic plasticity in bayesian spiking networks as expectation maximization with posterior constraints in *Advances in neural information processing systems*. pp. 773–781 (2012).
11. E Oja, Simplified neuron model as a principal component analyzer. *J. mathematical biology* **15**, 267–273 (1982).
12. J Zylberberg, JT Murphy, MR DeWeese, A sparse coding model with synaptically local plasticity and spiking neurons can account for the diverse shapes of v1 simple cell receptive fields. *PLoS Comput. Biol* **7**, e1002250 (2011).
13. Z Jonke, R Legenstein, S Habenschuss, W Maass, Feedback inhibition shapes emergent computational properties of cortical microcircuit motifs. *J. Neurosci.* **37**, 8511–8523 (2017).
14. P Földiak, Forming sparse representations by local anti-hebbian learning. *Biol. cybernetics* **64**, 165–170 (1990).
15. SY Kung, KI Diamantaras, A neural network learning algorithm for adaptive principal component extraction (apex) in *International Conference on Acoustics, Speech, and Signal Processing*. (IEEE), pp. 861–864 (1990).
16. Y Bahroun, A Soltoggio, Online representation learning with single and multi-layer hebbian networks for image classification in *International Conference on Artificial Neural Networks*. (Springer), pp. 354–363 (2017).
17. C Pehlevan, S Mohan, DB Chklovskii, Blind nonnegative source separation using biological neural networks. *Neural computation* **29**, 2925–2954 (2017).
18. A Tavanaei, T Masquelier, A Maida, Representation learning using event-based stdp. *Neural Networks* **105**, 294–303 (2018).
19. W Gerstner, WM Kistler, R Naud, L Paninski, *Neuronal dynamics: From single neurons to networks and models of cognition*. (Cambridge University Press), (2014).
20. C Machens, Github - machenslab/spikes (https://github.com/machenslab/spikes) (2020) Online; accessed 2021-02-20.
21. Y LeCun, C Cortes, C Burges, Mnist handwritten digit database (http://yann.lecun.com/exdb/mnist) (2010) Online; accessed 2020-05-20.
22. B Olshausen, Sparse coding simulation software (http://www.rctn.org/bruno/sparsenet/) (1996) Online; accessed 2020-05-20.

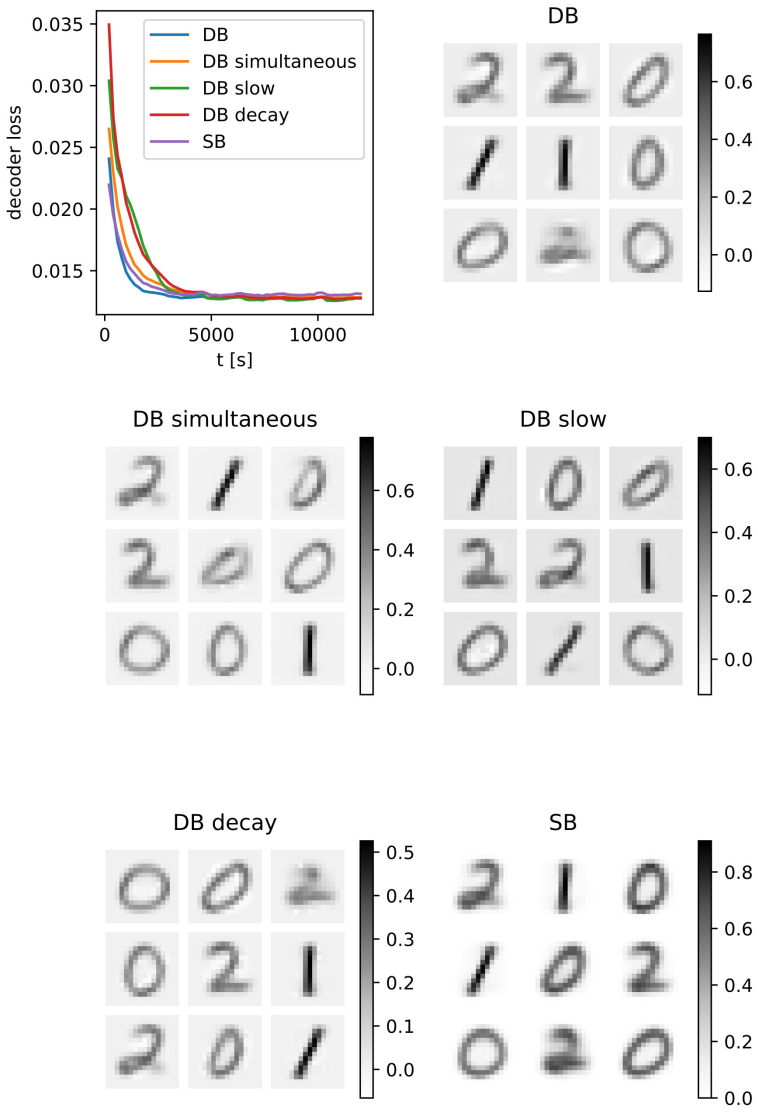

Fig. S2. Comparison of the different learning schemes on the MNIST task. All learning algorithms reach a very similar performance. The dendritic balance learning schemes with slow feedforward adaptation (DB slow) and the weight decay trick (DB decay) converge somewhat slower than dendritic balance with simultaneous feedforward and recurrent adaptation (DB simultaneous), dendritic balance with the analytical solution for recurrent weights (DB) and the somatic balance learning scheme (SB), as expected. DB decay finds smaller weights than other learning schemes, also as expected. As we derived, this can be compensated by a change in the firing threshold, which in our case is done via rapid firing rate adaptation. The learned feedforward weights are also very similar (bottom images).

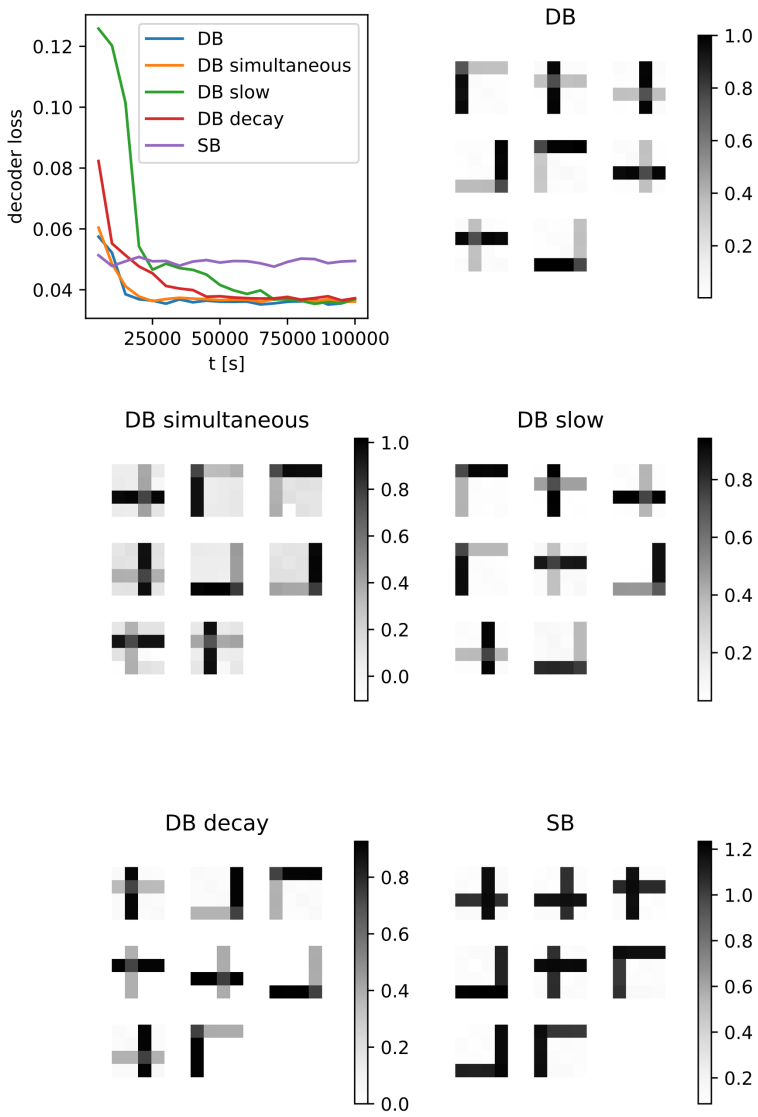

Fig. S3. Comparison of the different learning schemes on the bars task with $p = 0.7$. All dendritic balance algorithms reach a good performance, again DB slow and DB decay converge somewhat later. Learning in the SB network finds a sub-optimal solution. These results are reflected in the learned feedforward weights (bottom), where SB finds representations that do not contain single bars, as it would be optimal, but the collapsed corresponding bars instead.

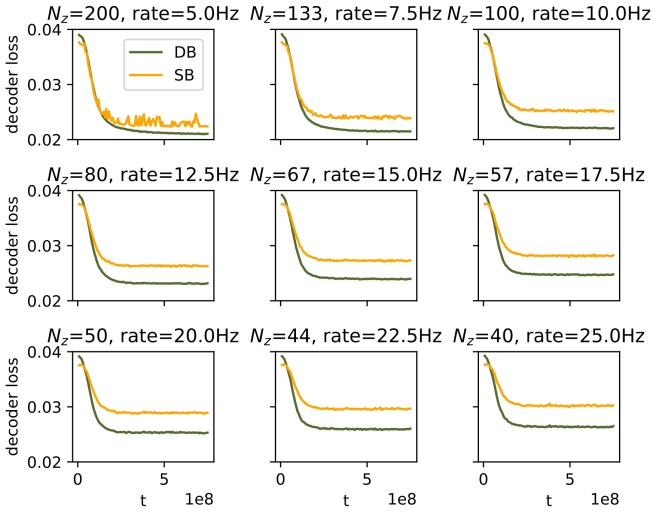

Fig. S4. All learning curves for the natural scenes task (Fig 4).

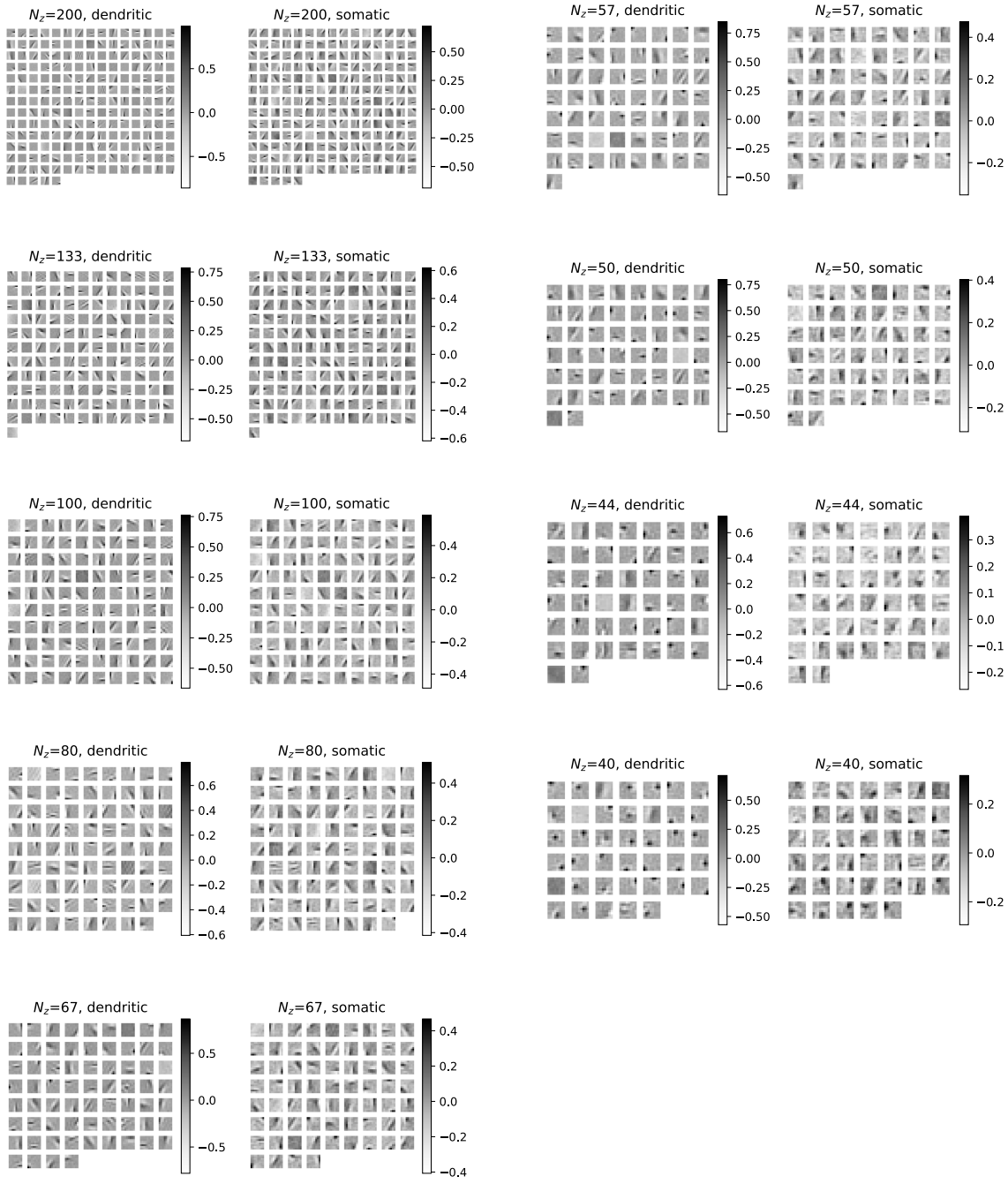

Fig. S5. All learned feedforward weights for the natural scenes task (Fig 4). For a large number of coding neurons neurons in both SB and DB learn weights with Gabor-wavelet like appearance. For smaller networks SB and DB learn qualitatively different weights: DB neurons become detectors for small blobs of activity in the images, similar to center-surround receptive fields. SB neurons also become detectors of blobs of activity but with much less coordination and larger diameter receptive fields.

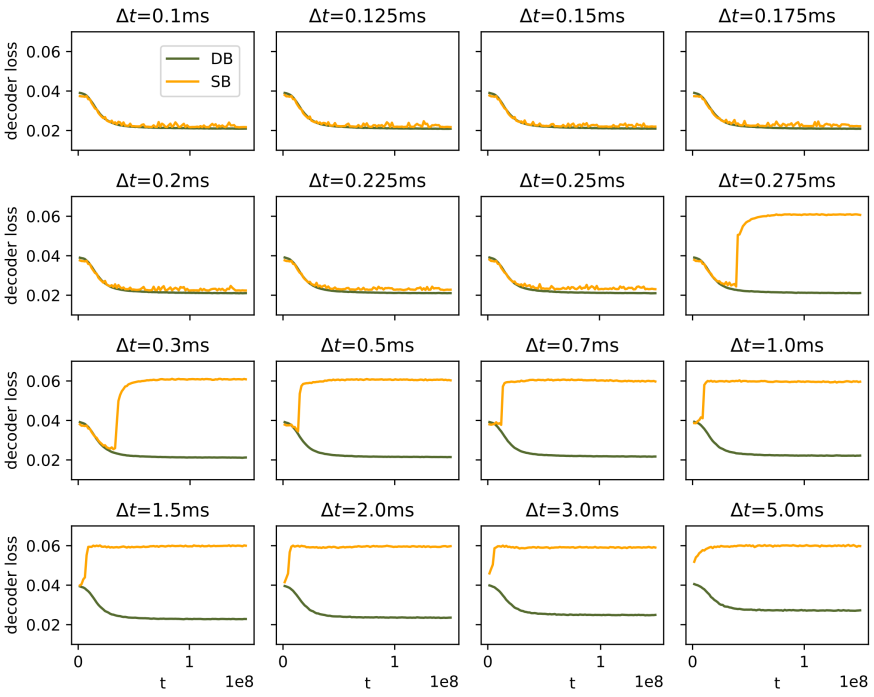

Fig. S6. All learning curves for the natural scenes task for different timesteps (Fig 5A). For long timesteps (i.e. transmission delays) SB learning fails.

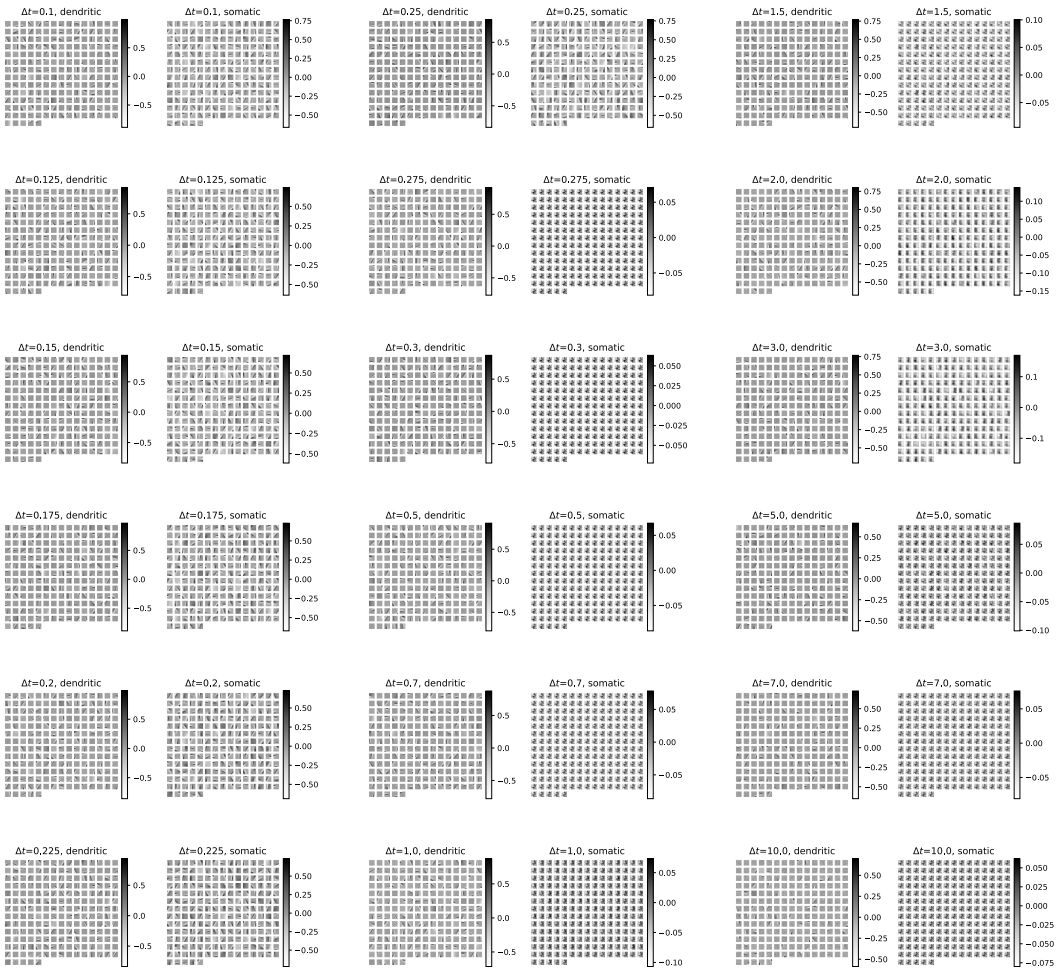

Fig. S7. All learned feedforward weights for the natural scenes task for different timesteps (Fig 5A). For long timesteps the representations learned by SB collapse, while DB continues to find good representations.

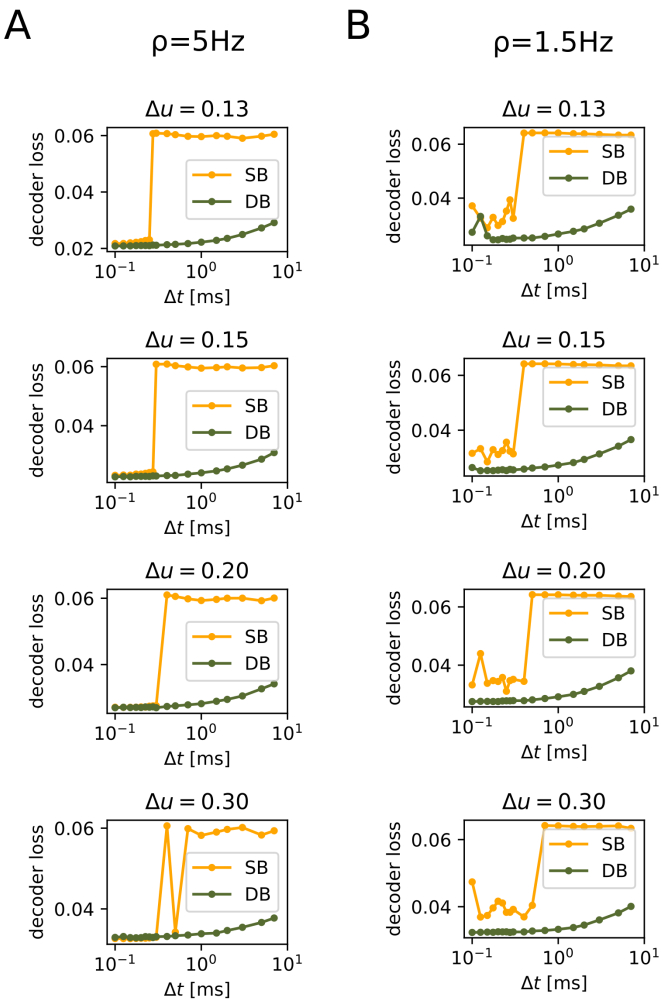

Fig. S8. The results in Fig 5A are robust in respect to the stochasticity of firing Δu and firing rate ρ. We tested firing rates of **A** $\rho = 5$Hz, where learning is mostly stable, and **B** $\rho = 1.5$Hz, where learning becomes slightly unstable. For higher stochasticity (larger Δu) neural firing becomes extremely random, for more deterministic neurons (smaller Δu) learning often does not converge.

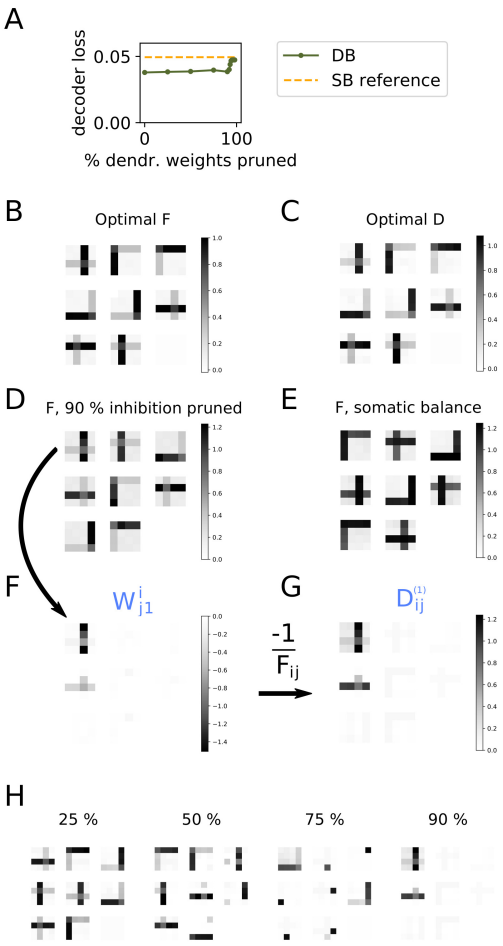

Fig. S9. In the dendritic balance learning scheme many of the dendritic recurrent weights can be pruned while retaining learning performance. We here demonstrate this in the correlated bars task with 4×4 pixel images and $p = 0.6$. The results presented here were obtained with DB with slow feedforward adaptation. Weights were pruned based on the principle presented above. Correlations between neurons were estimated in the first 1% of simulation-time, long before convergence, after which pruning commenced. Several pruned dendritic connections were replaced by a somatic connection to ensure a somatic balance if necessary. **A** 90% of the dendritic recurrent weights can be pruned without losing performance. If more dendritic weights are pruned performance approaches that of the SB learning scheme. **B** Feedforward weights learned with all dendritic connections in place. **C** Decoder corresponding to the network in B. **D** Feedforward weights learned when 90% of dendritic connections are pruned, which are remarkably similar to the optimal solution. **E** In comparison, weights learned by SB find collapsed representations containing two bars, which is a suboptimal solution. **F** Dendritic recurrent weights of neuron 1 in D, after pruning. Weights for a single neuron can be displayed in image space, showing which dendrites they connect to. After pruning with the proposed principle only weights that are important for learning remain. Self-contributions (top left) are always kept and can be computed locally. Other weights only connect to dendrites of the neuron, which codes for the "corresponding" correlated bar (center left). Only this neuron needs the information provided by the recurrent weights for learning, in order to prevent the collapse as we see it in E. **G** Since recurrent plasticity finds a "decoding" by balancing excitation with inhibition, we can find the corresponding "decoder" to the recurrent weights of neuron 1. This decoder only contains the two relevant bars that correspond to one another, demonstrating that our pruning principle can find exactly the relevant contributions and discard all the others. **H** The "decoder" as in G for progressive pruning fractions. It is clearly visible how for larger pruning fractions only the relevant dendritic connections remain.

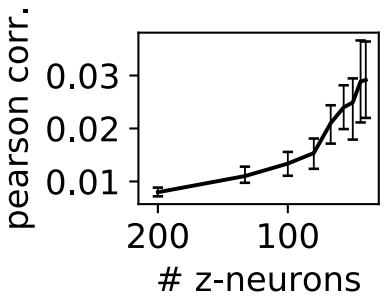

Fig. S10. Average pearson correlation coefficient between outputs $z_i(t)$ of pairs of neurons in DB networks coding for natural images (Fig 4D). The correlation of neural activity increases when the number of neurons decreases. This indicates that the input patterns the neurons learn to represent are more strongly correlated for small networks. Error-bars denote 95% bootstrapping confidence intervals.

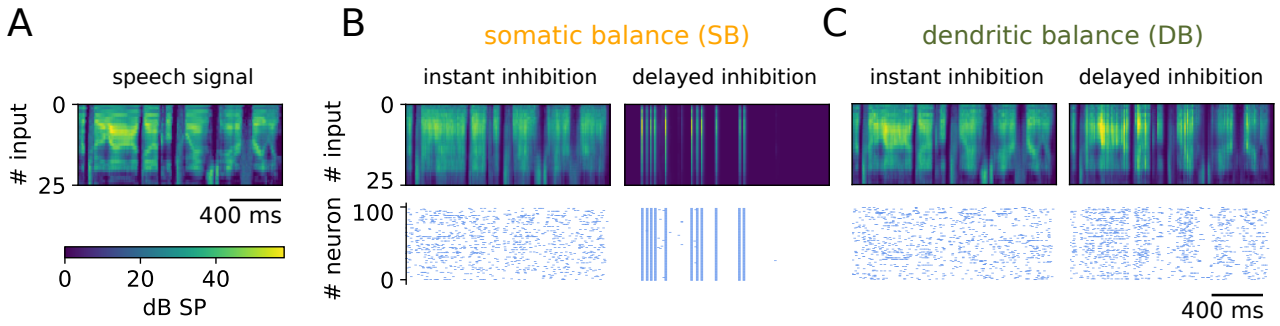

Fig. S11. This figure corresponds to Fig 5F-H, but uses unwhitened input signals. We observed similar results as in Fig 5F-H, here with transmission delays of $\delta = 0.5$ ms. For smaller delays we did not observe a collapse of the population code, likely because whitening leads to stimulus dimensions that have faster temporal dynamics than the original signal, making it harder for inhibition to decorrelate neural responses. For the somatic balance model (SB) the presence of pairwise correlations between inputs requires a different learning rule, where weights are updated according to $\Delta F_{ji} \propto z_j(x_i - \alpha x_i(F\mathbf{x})_j z_j)$ (see (2) for details). Although the dendritic balance model (DB) is also based on a decoder model that assumes inputs with zero pairwise correlations, it still manages to find a very good encoding. **A** Spectrogram of the signal presented in 25 frequency channels. **B** As can be seen in the reconstructed signal (top), SB finds a good encoding for instant inhibition (loss=0.08), but for small delays of 0.5 ms the learned representations collapse, leading to pathological network behavior and bad encoding performance (loss=1.25). **C** In contrast, DB finds a very good encoding for instant inhibition (loss=0.04) and a reasonable encoding with inhibitory delays of 0.5 ms (loss=0.2). Note that whitening changes the scale of the signal, hence decoder losses are not directly comparable between Fig 5 (where the loss is computed on the whitened signal) and Fig S11.

A.2 A comment about scaling time-steps

This section is an addendum to clarify some of the derivations in the supplementary material to Chapter 2 (Appendix A.1). Concretely, there are some intricacies that have to be considered regarding how the variance of the model should be re-scaled when time-steps are scaled. The following insights have been used implicitly, and we here make them explicit for clarity.

Consider the relation of time-step length δt, the kernel timescale τ, the variance of the model σ^2 and the stochasticity of spiking Δu that have been derived. τ was defined with units of time-steps, meaning that we can write it as

$$\tau = \tau^*/\delta t, \tag{A.1}$$

with τ^* some given timescale in units of time. We here see that for $\delta t \to 0$ we get $\tau \to \infty$, which in itself is not problematic. It seems to become problematic when considering the derived value for the stochasticity of spiking

$$\Delta u = \sigma^2/\tau = \delta t \sigma^2/\tau^*, \tag{A.2}$$

which for $\delta t \to 0$ seems to go to zero as well. This would mean that the model neurons become more deterministic when we simulate in smaller time-steps.

A simple solution would be to also re-scale the variance of the model as $\sigma^2 = \sigma^{2**}/\delta t$ for fixed σ^{2**}. This would result in a constant Δu independent of the time-step as we have used it in Chapter 2. There is a straightforward argument why this is the correct approach. In short, like for Gaussian white noise, the variance of the model has to go to infinity as δt goes to zero in order to have finite variance for any finite integral of the signal, or equivalently, to keep the information contained in the signal fixed. An intuitive explanation follows.

Consider that we have defined the model for some given (very short) time-step length δt^* with variance σ^{2*}. For illustration, say we have sampled a trajectory of random values using this model and we want to re-scale the time-step length to another length δt which is a multiple of $\delta t^* = \delta t/n$. We now collect the sampled trajectory into the new re-scaled time-bins by averaging over n bins, which creates a new set of samples. By the law of large numbers, these new samples have a variance of $\sigma^{2*}/n = \sigma^{2*}\delta t^*/\delta t$. This illustrates how the variance of samples changes when we look at a signal using different time-steps. Conversely this means that, in order to keep sampling in respect to the same underlying model, we have to change the variance as $\sigma^2 = \sigma^{2**}/\delta t$ with $\sigma^{2**} = \sigma^{2*}\delta t^*$ fixed, as suggested before.

A.3 Generalization to arbitrary spike kernels and hierarchical and recurrent predictions

Here we generalize the derivations of Appendix A.1 which required exponential spike kernels and only considered feed-forward driven networks. We propose an alternative approach that can handle arbitrary kernels and apply it to networks that are modulated by top-down inputs or recurrent inputs. In the end we will also discuss the limitations of these derivations and how they might be overcome.

We will use a slightly different notation for kernels κ and encodings r. As a simple mnemonic, $\kappa(t'|t)$ is the kernel at t' given there was a spike at t. For the exponential kernel (which could be replaced by any desired kernel) the reads as

$$\kappa(t'|t) = \exp\left(-\frac{t'-1-t}{\tau}\right).$$

Encodings can then be computed via

$$r(t') = \sum_{t=0}^{t'} \kappa(t'|t)s(t).$$

Top-down modulated network

Here we assume model distributions where the encoding is also modeled by a higher level population (similar to the model in Rao et al. 1999)

$$\begin{aligned} p_\theta(\mathbf{x}(t)|\mathbf{r}(t)) &= \mathcal{N}_\mathbf{x}(t)(D\mathbf{r}(t), \sigma) \\ p_\theta(\mathbf{r}(t)|\mathbf{r}_{\text{TD}}(t)) &= \mathcal{N}_\mathbf{r}(t)(D_{\text{TD}}\mathbf{r}_{\text{TD}}(t), \sigma_{\text{TD}}) \\ p_\theta(\mathbf{s}(t)) &= \frac{1}{Z}\exp(\mathbf{s}(t)^\top \mathbf{c}) \end{aligned}$$

We employ the same approach as before, by assuming a single future trajectory of inputs $\mathbf{x}$, representations $\mathbf{r}$ and top-down inputs $\mathbf{r}_{\text{TD}}$. Inputs $\mathbf{x}$ are simply assumed constant, as before. However, to allow for arbitrary kernels, we have to consider how $\mathbf{r}$ will develop depending on the kernel and past spikes. Ignoring future spikes, this future evolution is given by $\mathbf{r}(t') = \sum_{t''=0}^{t} \mathbf{s}(t'')\kappa(t'|t'') + \mathbf{s}(t)\kappa(t'|t)$. As a simple first guess (which might be improved upon in the future, but this generalizes the assumption in Appendix A.1) we assume that future spikes

are a mirror image of past spikes, leading to $\mathbf{r}(t') = 2\sum_{t''=0}^{t} \mathbf{s}(t'')\kappa(t'|t'') + \mathbf{s}(t)\kappa(t'|t)$. This allows us to derive

$$
\begin{aligned}
& p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{R}^{\text{TD}}_{t+1,T}, \mathbf{S}_{0,t}) \\
=& \sum_{\mathbf{S}_{t+1,T}} p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{R}^{\text{TD}}_{t+1,T}, \mathbf{S}_{0,t}, \mathbf{S}_{t+1,T}) p_\theta(\mathbf{S}_{t+1,T}|\mathbf{X}_{t+1,T}, \mathbf{R}^{\text{TD}}_{t+1,T}, \mathbf{S}_{0,t}) \\
\approx& p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{R}^{\text{TD}}_{t+1,T}, \mathbf{S}_{0,t}, \mathbf{S}_{t+1,T} = \text{reverse}(\mathbf{S}_{0,t})) \\
=& \prod_{t'=t}^{T} p_\theta\left(\mathbf{s}(t)|\mathbf{x}(t') = \mathbf{x}(t), \mathbf{r}_{\text{TD}}(t'), \mathbf{r}(t') = 2\sum_{t''=0}^{t} \mathbf{s}(t'')\kappa(t'|t'') + \mathbf{s}(t)\kappa(t'|t)\right) \\
=& \frac{1}{Z(\theta, \mathbf{x})} \exp\Bigg(\mathbf{s}(t)^\top \mathbf{c} - \underbrace{\sum_{t'=t}^{T} \frac{1}{2\sigma^2}\left[\mathbf{x}(t') - D\mathbf{r}(t')\right]^\top \left[\mathbf{x}(t') - D\mathbf{r}(t')\right]}_{\mathbf{b}(\mathbf{s}(t))} - \underbrace{\sum_{t'=t}^{T} \frac{1}{2\sigma_p^2}\left[\mathbf{r}_{\text{TD}}(t') - D_{\text{TD}}\mathbf{x}_{\text{TD}}(t')\right]^\top \left[\mathbf{r}_{\text{TD}}(t') - D_{\text{TD}}\mathbf{x}_{\text{TD}}(t')\right]}_{\mathbf{a}(\mathbf{s}(t))}\Bigg)
\end{aligned}
$$

Here, **b** are bottom-up (basal) contributions and **a** top-down (apical) contributions. For bottom-up contributions we find

$$
\begin{aligned}
\mathbf{b}(\mathbf{s}(t)) \propto& \frac{1}{\sigma^2}\sum_{t'=t}^{T} -\mathbf{r}(t')^\top D^\top \mathbf{x}(t') + \frac{1}{2} r(t')^\top D^\top D\mathbf{r}(t') \\
\propto& \frac{1}{\sigma^2}\left(-\mathbf{s}(t)^\top D^\top \mathbf{x}(t)\left[\sum_{t'=t}^{T} \kappa(t'|t)\right] + \frac{1}{2}\sum_{t'=t}^{T}\left(\mathbf{s}(t)\kappa(t'|t) + 2\sum_{t''=0}^{t} \mathbf{s}(t'')\kappa(t'|t'')\right)^\top D^\top D\left(\mathbf{s}(t)\kappa(t'|t) + 2\sum_{t''=0}^{t} \mathbf{s}(t'')\kappa(t'|t'')\right)\right) \\
\propto& \frac{\mathbf{s}(t)^\top}{\sigma^2}\left(-D^\top \mathbf{x}(t)\left[\sum_{t'=t}^{T} \kappa(t'|t)\right] + D^\top D\mathbf{s}(t)\left[\frac{1}{2}\sum_{t'=t}^{T} \kappa^2(t'|t)\right] + D^\top D\sum_{t''=0}^{t} \mathbf{s}(t'')\left[2\sum_{t'=t}^{T} \kappa(t'|t'')\kappa(t'|t)\right]\right)
\end{aligned}
$$

As before we can identify the inputs

$$
\begin{aligned}
\text{excitation: } & D^\top \mathbf{x}(t)\left[\sum_{t'=t}^{T} \kappa(t'|t)\right] \\
\text{inhibition: } & -D^\top D\sum_{t''=0}^{t} \mathbf{s}(t'')\left[2\sum_{t'=t}^{T} \kappa(t'|t'')\kappa(t'|t)\right] \\
\text{threshold: } & -\text{diag}(D^\top D)\left[\frac{1}{2}\sum_{t'=t}^{T} \kappa^2(t'|t)\right]
\end{aligned}
$$

For top-down contributions let $\mathbf{r}_{\text{TD}}(t') = \mathbf{s}(t)\kappa_{\text{TD}}(t'|t) + 2\sum_{t''=0}^{t}\mathbf{s}(t'')\kappa_{\text{TD}}(t'|t'') = \mathbf{p}(t') + 2\mathbf{q}(t')$. We then find

$$
\begin{aligned}
\mathbf{a}(\mathbf{s}(t)) &\propto \frac{1}{2\sigma_p^2}\sum_{t'=t}^{T}\left[(\mathbf{p}(t')+2\mathbf{q}(t')) - D_{\text{TD}}\mathbf{x}_{\text{TD}}(t')\right]^\top\left[(\mathbf{p}(t')+2\mathbf{q}(t')) - D_{\text{TD}}\mathbf{x}_{\text{TD}}(t')\right] \\
&\propto \frac{1}{2\sigma_p^2}\sum_{t'=t}^{T}\mathbf{p}(t')^\top\mathbf{p}(t') + 4\mathbf{p}(t')^\top\mathbf{q}(t') - 2\mathbf{p}(t')^\top D_{\text{TD}}\mathbf{x}_{\text{TD}}(t') \\
&= \frac{\mathbf{s}(t)^\top}{\sigma_p^2}\left(\mathbf{s}(t)\left[\frac{1}{2}\sum_{t'=t}^{T}\kappa_{\text{TD}}^2(t'|t)\right] + \sum_{t''=0}^{t}\mathbf{s}(t'')\left[2\sum_{t'=t}^{T}\kappa_{\text{TD}}(t'|t)\kappa_{\text{TD}}(t'|t'')\right] - D_{\text{TD}}\mathbf{x}_{\text{TD}}(t')\left[\sum_{t'=t}^{T}\kappa_{\text{TD}}(t'|t)\right]\right)
\end{aligned}
$$

$$
\begin{aligned}
\text{excitation: } & D_{\text{TD}}\mathbf{x}_{\text{TD}}(t')\left[\sum_{t'=t}^{T}\kappa_{\text{TD}}(t'|t)\right] \\
\text{self-inhibition: } & -\sum_{t''=0}^{t}\mathbf{s}(t'')\left[2\sum_{t'=t}^{T}\kappa_{\text{TD}}(t'|t)\kappa_{\text{TD}}(t'|t'')\right] \\
\text{threshold: } & -\left[\frac{1}{2}\sum_{t'=t}^{T}\kappa_{\text{TD}}^2(t'|t)\right]
\end{aligned}
$$

Recurrent network

Here we assume recurrent contributions in the prior from slower spike traces $\mathbf{r}_p(t') = \sum_{t<t'}\kappa_p(t'|t)\mathbf{s}(t)$

$$
\begin{aligned}
p_\theta(\mathbf{x}(t)|\mathbf{r}(t)) &= \mathcal{N}_{\mathbf{x}}(t)(D\mathbf{r}(t), \sigma) \\
p_\theta(\mathbf{s}(t)|\mathbf{r}_p(t)) &= \frac{1}{Z}\exp(\mathbf{s}(t)^\top\mathbf{c})\mathcal{N}_{\mathbf{r}}(t)(D_p\mathbf{r}_p(t), \sigma_p)
\end{aligned}
$$

$$
\begin{aligned}
& p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{S}_{0,t}) \\
=& \sum_{\mathbf{S}_{t+1,T}} p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{S}_{0,t}, \mathbf{S}_{t+1,T})p_\theta(\mathbf{S}_{t+1,T}|\mathbf{X}_{t+1,T}, \mathbf{S}_{0,t}) \\
\approx& p_\theta(\mathbf{s}(t)|\mathbf{X}_{t+1,T}, \mathbf{S}_{0,t}, \mathbf{S}_{t+1,T} = \text{reverse}(\mathbf{S}_{0,t})) \\
=& \prod_{t'=t}^{T} p_\theta\left(\mathbf{s}(t)|\mathbf{x}(t') = \mathbf{x}(t), \mathbf{r}(t') = 2\sum_{t''=0}^{t}\mathbf{s}(t'')\kappa(t'|t'') + \mathbf{s}(t)\kappa(t'|t)\right) \\
=& \frac{1}{Z(\theta, \mathbf{x})}\exp\Bigg(\mathbf{s}(t)^\top\mathbf{c} - \underbrace{\sum_{t'=t}^{T}\frac{1}{2\sigma^2}\left[\mathbf{x}(t') - D\mathbf{r}(t')\right]^\top\left[\mathbf{x}(t') - D\mathbf{r}(t')\right]}_{\mathbf{b}(\mathbf{s}(t))} - \underbrace{\sum_{t'=t}^{T}\frac{1}{2\sigma_p^2}\left[\mathbf{r}(t') - D_p\mathbf{r}_p(t')\right]^\top\left[\mathbf{r}(t') - D_p\mathbf{r}_p(t')\right]}_{\mathbf{a}(\mathbf{s}(t))}\Bigg)
\end{aligned}
$$

Here we assume $\mathbf{r}(t') = \mathbf{s}(t)\kappa(t'|t) + \sum_{t''=0}^{t}\mathbf{s}(t'')\kappa(t'|t'') = \mathbf{p}(t') + \mathbf{q}(t')$ and $\mathbf{r}_p(t')$ similarly. Assuming future spikes might be detrimental here, as this could lead to overconfidence in

spiking. It has to be noted that the outcome of these assumptions have not been tested and might not be appropriate in general.

$$
\begin{aligned}
\mathbf{a}(\mathbf{s}(t)) \propto & \frac{1}{2\sigma_p^2} \sum_{t'=t}^{T} \Big[(\mathbf{p}(t') + \mathbf{q}(t')) - D_p(\mathbf{p}_p(t') + \mathbf{q}_p(t'))\Big]^\top \Big[(\mathbf{p}(t') + \mathbf{q}(t')) - D_p(\mathbf{p}_p(t') + \mathbf{q}_p(t'))\Big] \\
\propto & \frac{1}{2\sigma_p^2} \sum_{t'=t}^{T} \mathbf{p}(t')^\top \mathbf{p}(t') + 2\mathbf{p}(t')^\top \mathbf{q}(t') - 2\mathbf{p}(t')^\top D_p \mathbf{p}_p(t') - 2\mathbf{p}(t')^\top D_p \mathbf{q}_p(t') \\
& - 2\mathbf{p}_p(t')^\top D_p^\top \mathbf{q}(t') + \mathbf{p}_p(t')^\top D_p^\top D_p \mathbf{p}_p(t') + 2\mathbf{p}_p(t')^\top D_p^\top D_p \mathbf{q}_p(t') \\
= & \frac{s(t)^\top}{\sigma_p^2} \Bigg(\mathbf{s}(t) \left[\frac{1}{2} \sum_{t'=t}^{T} \kappa^2(t'|t) \right] + \sum_{t''=0}^{t} \mathbf{s}(t'') \left[\sum_{t'=t}^{T} \kappa(t'|t'')\kappa(t'|t) \right] - D_p \mathbf{s}(t) \left[\sum_{t'=t}^{T} \kappa(t'|t)\kappa_p(t'|t) \right] \\
& - D_p \sum_{t''=0}^{t} \mathbf{s}(t'') \left[\sum_{t'=t}^{T} \kappa(t'|t)\kappa_p(t'|t'') \right] - D_p^\top \sum_{t''=0}^{t} \mathbf{s}(t'') \left[\sum_{t'=t}^{T} \kappa_p(t'|t)\kappa(t'|t'') \right] + D_p^\top D_p \mathbf{s}(t) \left[\frac{1}{2} \sum_{t'=t}^{T} \kappa_p^2(t'|t) \right] \\
& + D_p^\top D_p \sum_{t''=0}^{t} \mathbf{s}(t'') \left[\sum_{t'=t}^{T} \kappa_p(t'|t)\kappa_p(t'|t'') \right] \Bigg)
\end{aligned}
$$

$$
\begin{aligned}
\text{excitation: } & D_p \sum_{t''=0}^{t} \mathbf{s}(t'') \left[\sum_{t'=t}^{T} \kappa(t'|t)\kappa_p(t'|t'') \right] \\
\text{acausal exc. (ignore): } & D_p^\top \sum_{t''=0}^{t} \mathbf{s}(t'') \left[\sum_{t'=t}^{T} \kappa_p(t'|t)\kappa(t'|t'') \right] \\
\text{inhibition: } & - D_p^\top D_p \sum_{t''=0}^{t} \mathbf{s}(t'') \left[\sum_{t'=t}^{T} \kappa_p(t'|t)\kappa_p(t'|t'') \right] \\
\text{self-inhibition (refractory): } & - \sum_{t''=0}^{t} \mathbf{s}(t'') \left[\sum_{t'=t}^{T} \kappa(t'|t'')\kappa(t'|t) \right] \\
\text{threshold: } & - \left[\frac{1}{2} \sum_{t'=t}^{T} \kappa^2(t'|t) \right] - \text{diag}(D_p^\top D_p) \left[\frac{1}{2} \sum_{t'=t}^{T} \kappa_p^2(t'|t) \right] \\
& + \text{diag}(D_p) \left[\sum_{t'=t}^{T} \kappa(t'|t)\kappa_p(t'|t) \right]
\end{aligned}
$$

Here, the acausal excitation is the overlap between the past encoding and future predictions, which is small for reasonable kernels and can therefore be ignored.

Limitations

The assumptions and approximations made in the previous derivations are necessary to compute an analytical approximation of the posterior, when information about the future is not available, as for biological networks. In preliminary simulations we found that these approximations are decent and result in functional networks, but the sampled posterior might be biased. While this bias seems to be small for the scenarios we have tested it, in

conjunction with the EM algorithm it can in some cases prevent the convergence of learning (i.e., the bias in the encoding means the model does not capture the relation between encoding and data, which is also not solved by adapting it, effectively 'pushing' both encoding and model parameters around indefinitely).

We here propose two ways to address this convergence problem. One is to explicitly use a constrained posterior distribution, which we have done in chapter A.1 already. This can cancel possibly wrongly defined parameters of the posterior and thereby stabilize learning (see also Habenschuss et al. 2012). Another idea would be to use a hybrid approach in the E-step. The intuition here is that it would be desirable to explicitly minimize the log-likelihood lower bound in the E-step, e.g., via gradient descent, but this would effectively prevent us from deriving biologically plausible update rules as we did before. Our proposal is to keep the analytical approximation, but to reduce the bias in sampling by explicitly optimizing some of the parameters of this approximate distribution. A parameter that affords itself to this is the threshold of neural spiking, which could be adapted by some algorithm that directly minimizes the loss, e.g., via the REINFORCE algorithm (Appendix A.4). Future work might investigate the viability of these approaches.

Another problem for hierarchical and recurrent models is that neural activity typically is sparse. This means that most of the time the representations $\mathbf{r}$ are zero, and the Gaussian prior distributions become inappropriate. A solution to this problem would be not to assume the Gaussian priors on the single neuron traces, but on functional assemblies of neurons. This is equivalent to performing the modeling in a lower dimensional space of continuous variables, rather than the high-dimensional single neuron representation space (a related model has demonstrated a similar approach before; Boerlin et al. 2013). This solution, however, results in learning rules that are not easily localizeable anymore, as single synapses now require information about the entire functional assembly. Future work might look into how this problem can be circumvented, by extending our results in chapter A.1 where we have proposed a similar idea (clustering of excitatory synapses), or how different forms of prior distributions could facilitate recurrent spiking networks (taking inspiration for example from Kappel et al. 2014).

A.4 Reinforced spiking algorithm

We start with the goal function as the log-likelihood lower bound as before. Using the fact that the model factorizes over time given the spike traces, we write the log-likelihood lower bound as an average of an instantaneous reward

$$\mathcal{F}^*(\psi,\theta) = \langle \log p_\theta(\mathbf{X}_{0,T}, \mathbf{S}_{0,T}) - \log p_\psi(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})\rangle_{p_\psi(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})p^*(\mathbf{X}_{0,T})} \tag{A.3}$$

$$= \left\langle \sum_t R(\mathbf{x}_t, \mathbf{r}_t, \mathbf{s}_t, \psi, \theta) \right\rangle_{p_\psi(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})p^*(\mathbf{X}_{0,T})} \tag{A.4}$$

with reward $R(\mathbf{x}_t, \mathbf{r}_t, \mathbf{s}_t, \psi, \theta) = \log p_\theta(\mathbf{x}_t, \mathbf{r}_t) - \log p_\psi(\mathbf{s}_t|\mathbf{x}_t, \mathbf{r}_t)$.

To update the parameters using gradient decent we have to compute $\frac{\partial}{\partial\psi}\mathcal{F}^*(\psi,\theta)$, which involves taking the derivative 'through' the expectation. One approach to do that is the score function trick

$$\frac{\partial}{\partial\theta}\langle f(z,\theta)\rangle_{p_\psi(z)} = \left\langle \left[\frac{\partial}{\partial\psi}\log p_\psi(z)\right] f(z,\theta) + \frac{\partial}{\partial\psi} f(z,\theta) \right\rangle_{p_\psi(z)}.$$

Now the derivative can be written as

$$\frac{\partial}{\partial\psi}\mathcal{F}^*(\psi,\theta) \propto \left\langle \left[\sum_t \frac{\partial}{\partial\psi}\log p_\psi(\mathbf{s}_t|\mathbf{x}_t,\mathbf{r}_t)\right]\left[\sum_t R(\mathbf{x}_t,\mathbf{r}_t,\mathbf{s}_t,\psi,\theta)\right]\right\rangle_{p_\psi(\mathbf{S}_{0,T}|\mathbf{X}_{0,T})p^*(\mathbf{X}_{0,T})}. \tag{A.5}$$

The remaining problem is that these two sums over time cannot be evaluated in an online fashion, but only after the whole training sequence, which is biologically implausible. To address this, the REINFORCE algorithm (Williams 1992) replaces the left-hand sum through an eligibility trace

$$\dot{e} = -\frac{1}{\tau}e + \frac{\partial}{\partial\psi}\log p_\psi(\boldsymbol{s}(t) \mid \boldsymbol{x}(t), \mathbf{r}(t)). \tag{A.6}$$

The derivative $\frac{\partial}{\partial\psi}\log p_\psi(\boldsymbol{s}(t) \mid \boldsymbol{x}(t), \mathbf{r}(t))$ typically takes a convenient (local) form (as the dynamics p_ψ are local), such as the product of pre and postsynaptic activity. Now the update rule can be implemented online:

$$\frac{\partial}{\partial\psi}\mathcal{F}^*(\psi,\theta) \approx \sum_t e_t R(\mathbf{x}_t, \mathbf{r}_t, \mathbf{s}_t, \psi, \theta). \tag{A.7}$$

Although this algorithm is implementable with local dynamics (assuming the reward is broadcasted to all neurons), the updates have a high variance, which makes learning very slow. Intuitively, we correlate *all* actions of the network with the entire reward, which means that we will have to sample a large number of 'action trajectories' in order to understand which of the actions were beneficial (or not). It is possible to reduce this variance to some extent, e.g., by redefining the reward (Williams 1992), but ultimately this algorithm is unsuitable for training recurrent networks of spiking neurons.