A probabilistic model for learning in cortical microcircuit motifs with data-based divisive inhibition

Robert Legenstein*, Zeno Jonke*, Stefan Habenschuss, Wolfgang Maass

Institute for Theoretical Computer Science

Graz University of Technology

October 3, 2018

* These authors contributed equally to the work.

Abstract

Previous theoretical studies on the interaction of excitatory and inhibitory neurons proposed to model this cortical microcircuit motif as a so-called Winner-Take-All (WTA) circuit. A recent modeling study however found that the WTA model is not adequate for data-based softer forms of divisive inhibition as found in a microcircuit motif in cortical layer 2/3. We investigate here through theoretical analysis the role of such softer divisive inhibition for the emergence of computational operations and neural codes under spike-timing dependent plasticity (STDP). We show that in contrast to WTA models — where the network activity has been interpreted as probabilistic inference in a generative mixture distribution — this network dynamics approximates inference in a noisy-OR-like generative model that explains the network input based on multiple hidden causes. Furthermore, we show that STDP optimizes the parameters of this model by approximating online the expectation maximization (EM) algorithm. This theoretical analysis corroborates a preceding modelling study which suggested that the learning dynamics of this layer 2/3 microcircuit motif extracts a specific modular representation of the input and thus performs blind source separation on the input statistics.

1 Introduction

Winner-take-all-like (WTA-like) circuits constitute a ubiquitous motif of cortical microcircuits [Douglas and Martin, 2004]. Previous models and theories for competitive Hebbian learning in WTA-like circuit from [Rumelhart and Zipser, 1985] to [Nessler et al., 2013] were based on the assumption of strong WTA-like lateral inhibition. Several theoretical studies showed that spike-timing dependent plasticity (STDP) supports the emergence of Bayesian computation in such winner-take-all (WTA) circuits [Nessler et al., 2013, Habenschuss et al., 2013b, Klampfl and Maass, 2013]. These analyses were based on a probabilistic generative model approach. In particular, it was shown that the network implicitly represents the distribution of input patterns through a generative mixture distribution and that STDP optimizes the parameters of this mixture distribution. But this analysis assumed that the input to a WTA is explained at any point in time by a single neuron, and that strong lateral inhibition among pyramidal cells ensures a basically fixed total output rate of the WTA. These assumptions, however, may not be suitable in the context of more realistic activity dynamics in cortical networks.

In fact, recent modeling results [Avermann et al., 2012, Jonke et al., 2017] show that the WTA model is not adequate for a softer form of inhibition that has been reported for cortical layer 2/3. This softer form of inhibition is often referred to as feedback inhibition, or lateral inhibition, and has been termed more abstractly based on its influence on pyramidal cells as divisive inhibition [Wilson et al., 2012, Carandini and Heeger, 2012]. It stems from dense bidirectional interconnections between layer 2/3 pyramidal cells and nearby Parvalbumin-positive (PV^+) interneurons (often characterized as fast-spiking interneurons, in particular basket cells), see e.g. [Packer and Yuste, 2011, Fino et al., 2012, Avermann et al., 2012]. The simulations results in [Jonke et al., 2017] also indicate that blind source separation emerges as the computational function of this microcircuit motif when STDP is applied to the input synapses of the circuit.

The results of [Jonke et al., 2017] raise the question whether they can be understood from the perspective of a corresponding probabilistic generative model, that could replace the mixture model that underlies the analysis of emergent computational properties of microcircuit motivs with hard WTA-like inhibition. We propose here such a model that is based on a Gaussian prior over the number of active excitatory neurons in the network and a noisy-OR-like likelihood term. We develop a novel analysis technique based on the neural sampling theory [Buesing et al., 2011] to show that the microcircuit motif model approximates probabilistic inference in this probabilistic generative model. Further, we derive a plasticity rule that optimizes the parameters of this generative model through online expectation maximization (EM), the arguably most powerful tool from statistical learning theory for the optimization of generative models. We show that this plasticity rule can be approximated by an STDP-like learning rule.

This theoretical analysis strengthens the claim that blind source separation [Földiak, 1990] — also referred to as independent component analysis [Hyvärinen et al., 2004] — emerges as a fundamental computation on assembly codes through STDP in this microcircuit motif. This computational operation enables a network to disentangle and separately represent superimposed inputs that result from independent assembly activations in different upstream networks. Furthermore, our theoretical analysis reveals that the ability of this cortical microcircuit motif to perform blind source separation is facilitated either by the normalization of activity patterns in input populations, or by homeostatic mechanisms that normalize excitatory synaptic efficacies within each neuron.

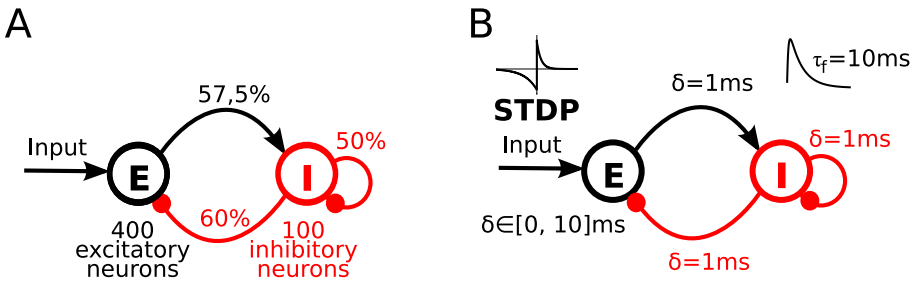

Figure 1: **Data-based network model $\mathcal{M}$ for a microcircuit motif.** **A**) Network anatomy. Circles denote excitatory (black) and inhibitory (red) pools of neurons. Black arrows indicate excitatory connections. Red lines with dots indicate inhibitory connections. Numbers above connections denote corresponding connection probabilities. **B**) Network physiology. Same as in (A), but connection delays δ are indicated. All synapses are modeled with the same PSP shape using a decay time constant of $\tau_f = 10$ ms as indicated on top right. Input synapses are subject to STDP.

2 Results

A data-based microcircuit motif model for the interaction of pyramidal cells with PV$^+$ inhibitory neurons in layer 2/3 has been introduced in [Avermann et al., 2012]. Based on this study, [Jonke et al., 2017] analyzed the computational properties that emerge in this microcircuit motif from synaptic plasticity. We first briefly introduce the microcircuit motif model analyzed in [Jonke et al., 2017] and discuss its properties. Subsequently, we present a theoretical analysis of this network motif based on a probabilistic generative model $\mathcal{P}$.

2.1 A data-based model for a network motif consisting of excitatory and inhibitory neurons

[Jonke et al., 2017] proposed a specific model for interacting populations of pyramidal cells with PV$^+$ inhibitory neurons in cortical layer 2/3 based on data from the Petersen Lab [Avermann et al., 2012], see Fig. 1A, B. We refer to this specific model as the microcircuit motif model $\mathcal{M}$.

The model $\mathcal{M}$ consists of two reciprocally connected pools of neurons, an excitatory pool and an inhibitory pool. M stochastic spiking neurons constitute the excitatory pool. Their dynamics is given by a stochastic version of the spike response model that has been fitted to experimental data in [Jolivet et al., 2006]. The instantaneous firing rate ρ_m of a neuron m depends exponentially on its current membrane potential u_m,

$$\rho_m(t) \ = \frac{1}{\tau} \exp(\gamma \cdot u_m(t)) \ , \tag{1}$$

where $\tau = 10$ ms and $\gamma = 2$ are scaling parameters that control the shape of the response function. After emitting a spike, the neuron enters a refractory period.

The excitatory neurons are reciprocally connected to a pool of recurrently connected inhibitory neurons. All connection probabilities in the model were taken from [Avermann et al., 2012]. Excitatory neurons receive excitatory synaptic inputs $\tilde{y}_1(t), .., \tilde{y}_N(t)$ with corresponding synaptic efficiencies w_{im} between the input neuron i and neuron m. These afferent connections are subject to a standard form of STDP. Thus, the membrane po-

tential of excitatory neuron m is given by the sum of external inputs, inhibition from inhibitory neurons, and its excitability α

$$u_m(t) = \sum_i w_{im}\tilde{y}_i(t) - \sum_{j\in\mathcal{I}_m} w^{\mathrm{IE}} I_j(t) + \alpha, \tag{2}$$

where $\mathcal{I}_m$ denotes the set of indices of inhibitory neurons that project to neuron m, and w^{IE} denotes the weight of these inhibitory synapses. $I_j(t)$ and $\tilde{y}_i(t)$ denote synaptic input from inhibitory neurons and input neurons respectively, see above.

Inhibitory contributions to the membrane potential of pyramidal cells have in this neuron model a divisive effect on the firing rate. This can be seen by by substituting eq. (2) in eq. (1), see also eq. (23) in *Methods*, thus implementing divisive inhibition (see [Carandini and Heeger, 2012] for a recent review). Divisive inhibition has been shown to be a ubiquitous computational primitive in many brain circuits (see [Carandini and Heeger, 2012] for a recent review). In mouse visual cortex, divisive inhibition is implemented through PV^+ inhibitory neurons [Wilson et al., 2012]. Although the inhibitory signal is common to all neurons in the pool of excitatory neurons, contrary to the inhibition modeled in [Nessler et al., 2013] it does not normalize the firing rates of neurons exactly and therefore the total firing rate in the excitatory pool is variable and depends on the input strength. Importantly, in contrast to [Nessler et al., 2013] where inhibition strictly enforced that only a single neuron in the excitatory pool is active at any given time, the data-based model $\mathcal{M}$ allows several neurons to be active concurrently.

2.2 Emergent properties of the data-based network model: From WTA to k-WTA

The computational properties of this data-based network model $\mathcal{M}$ were extensively studied through simulations in [Jonke et al., 2017]. In order to compare the properties of this data-based network model $\mathcal{M}$ to previously considered WTA models, they examined the emergence of orientation selectivity, which we briefly discuss here. For details, please see [Jonke et al., 2017]. Pixel-representations of noisy bars in random orientations were provided as external spike inputs (Fig. 2A). Input spike trains were generated from these pixel arrays by converting pixel values to Poisson firing rates of input neurons (black pixel: 75 Hz; white pixel: 1 Hz). Randomly oriented bars were presented to the network for 400 s where each bar was presented for 50 ms, see Fig. 2B (STDP was applied to synapses from input neurons to excitatory neurons). The resulting network response (Fig. 2D) shows the emergence of assembly codes for oriented bars. The resulting Gaussian-like tuning curves of excitatory neurons (Fig. 2E) densely cover all orientations, resembling experimental data from orientation pinwheels (see Fig. 2 d,e in [Ohki et al., 2006]). Also consistent with experimental data [Kerlin et al., 2010, Isaacson and Scanziani, 2011], inhibitory neurons did not exhibit orientation selectivity (not shown).

In contrast, previously considered models with idealized strong inhibition in WTA-circuits [Nessler et al., 2013] show a clearly distinct behavior, see Fig. 2F. For this model, at most a single neuron could fire at any moment of time, and as a result at most two neurons responded after a corresponding learning protocol with an increased firing rate to a given orientation (see Fig. 2F and Fig. 5 in [Nessler et al., 2013]). In the simulations of the data-based model $\mathcal{M}$, on average $k = 17$ neurons responded to each orientation with an increased firing rate. This suggests that the emergent computational operation of the layer 2/3 microcircuit motif with divisive inhibition is better described as k-WTA computation, where k winners may emerge simultaneously from the competition. This

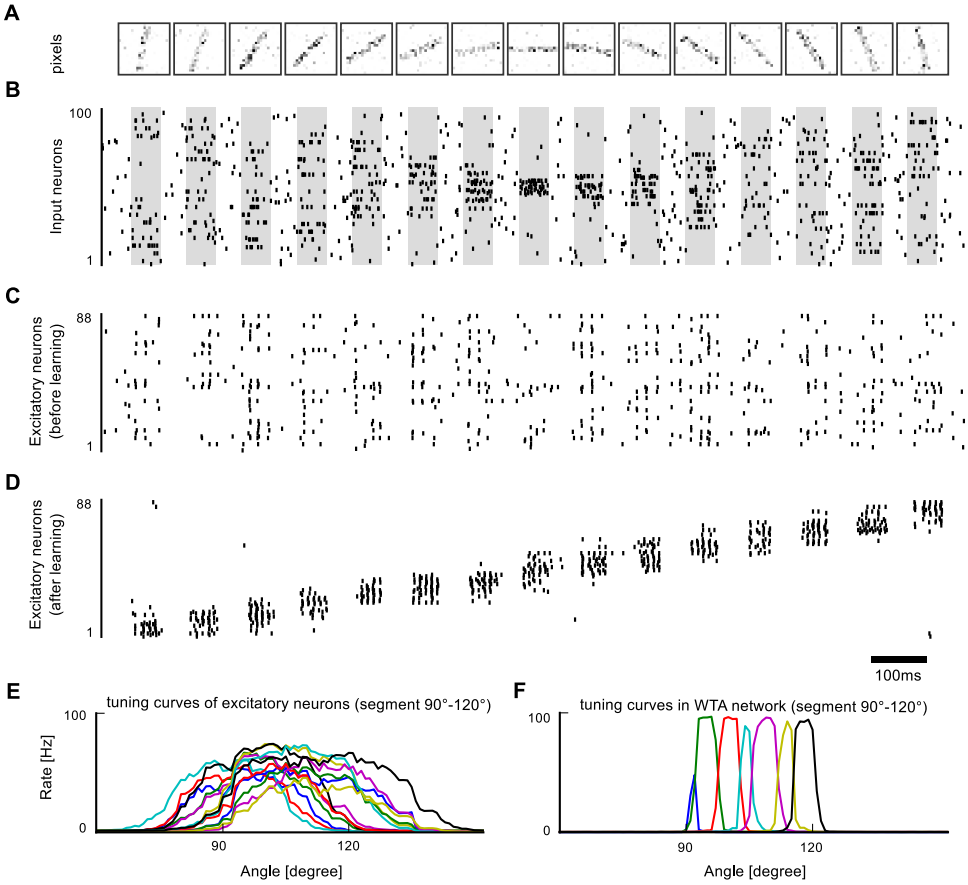

Figure 2: **Emergent computational properties of the data-based network model $\mathcal{M}$. A**) Network inputs are given by images of randomly oriented bars (inputs arranged in 2D for clarity; pixel gray-level indicates effective network input $\tilde{y}_i(t)$, see eq. (22)). **B**) Input neuron spike patterns (every 4^{th} neuron shown). Presence of a bar in the input with orientation indicated in panel A indicated by gray shading. **C, D**) Spike responses of a subset of excitatory neurons in $\mathcal{M}$ to the input in (B) before (C) and after (D) learning (neurons sorted by preferred orientation). **E**) Tuning curves of excitatory neurons in with preferred orientations between 90 and 120 degrees. **F**) Orientation tuning curves in a WTA model [Nessler et al., 2013, Habenschuss et al., 2013b]. Figure modified from Fig. 2 in [Jonke et al., 2017].

number k is however not a strict constraint in the data-based model $\mathcal{M}$. The actual number of winners depends on synaptic weights and the external input. Its computation is thus better describe as an adaptive k-WTA operation. The k-WTA characterisitcs of the layer 2/3 microcircuit motif is quite attractive, since it is known from computational complexity theory that the k-WTA computation is more powerful than the simple WTA computation (for $k > 1$) [Maass, 2000].

2.3 Theoretical framework for understanding emergent computational properties the layer 2/3 microcircuit motif

Fig. 2 demonstrates that significantly different computational properties emerge in the data-based model $\mathcal{M}$ through STDP as compared to previously considered WTA models [Nessler et al., 2013, Habenschuss et al., 2013b]. The main aim of this article is to understand this different emergent computational capability theoretically, in particular since the analysis from [Nessler et al., 2013] and [Habenschuss et al., 2013b] in terms of mixture distributions is only applicable to WTA circuits. The novel analysis technique that we will use is summarized as follows. First, using some simplifications on the network dynamics, we formulate the network dynamics in the neural sampling framework [Buesing et al., 2011]. This allows us to deduce the distribution $p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W})$ of activities $\boldsymbol{z}$ of excitatory neurons in the network for a given input $\boldsymbol{y}$ and for the given network weights $\boldsymbol{W}$. We then show that this distribution approximates the posterior distribution of a generative probabilistic model $\mathcal{P}$. This generative model is not a mixture distribution as in the WTA case [Nessler et al., 2013], but a more complex distribution that is based on a noisy-OR-like likelihood. We make the nature of the approximation explicit and evaluate its severity through simulations. Finally, we derive a plasticity rule that implement online EM in this generative model, thus implementing blind source separation. We find that this plasticity rule can be approximated by an STDP-like learning rule.

2.3.1 Formulation of the network dynamics of $\mathcal{M}$ in the neural sampling framework

The neural sampling framework [Buesing et al., 2011] provides us with the ability to determine the stationary distribution (defined in the following) of network states for the given network parameters and a given network input. In order to be able to describe the probabilistic relationships between input and network activity, we describe network inputs by binary vectors $\boldsymbol{y}(t)$ and responses of excitatory neurons in $\mathcal{M}$ by binary vectors $\boldsymbol{z}(t)$. The vectors $\boldsymbol{y}(t)$ and $\boldsymbol{z}(t)$ capture the spiking activity of ensembles of spiking neurons in continuous time according to the common convention introduced in [Berkes et al., 2011] and [Buesing et al., 2011]: A spike of the i^{th} neuron in the ensemble at time t sets the corresponding component $y_i(t)$ ($z_i(t)$) of the bit vector $\boldsymbol{y}(t)$ ($\boldsymbol{z}(t)$) from its default value 0 to 1 for some duration τ (that can be chosen for example to reflect the typical time constant of an EPSP), see *Methods*. Note the difference between the vectors $\boldsymbol{y}(t)$, $\boldsymbol{z}(t)$ and the output traces $\tilde{\boldsymbol{y}}(t)$, $\tilde{\boldsymbol{z}}(t)$ used in eq. (2) (and defined in eq. (22) in *Methods*). The former constitute an abstract convention to describe the momentary state of the network based on its current firing activity, while the latter describe the impact that the neurons have on their postsynaptic targets in terms of real-valued double-exponential EPSPs.

We want to describe the distribution of network states $\boldsymbol{z}(t)$ for given inputs $\boldsymbol{y}(t)$ and network parameters $\boldsymbol{W}$ in terms of a probability distribution $p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W})$. In this distribution, the activities of network inputs and excitatory neurons in $\mathcal{M}$ are represented by two vectors of binary random variables: $\boldsymbol{y} = (y_1, \ldots, y_N)$ (termed input variables in the

following) and $\boldsymbol{z} = (z_1, \ldots, z_M)$ (termed hidden variables or hidden causes). The network state $\boldsymbol{y}(t), \boldsymbol{z}(t)$ at time t is interpreted as one specific realization of these random variables.

In order to make this mapping between network activity in $\mathcal{M}$ and the distribution of network states feasible, one has to make three simplifying assumptions about the dynamics of the neural network models similar as in [Buesing et al., 2011]. First, PSPs of inputs and network neurons are rectangular with length τ (chosen here to be $\tau = 10$ ms) and network neurons are refractory for the same time span τ after each spike. Second, synaptic connections are idealized in the sense that the synaptic delay is 0 (i.e., a presynaptic spike leads instantaneously to a PSP in the postsynaptic neuron). And finally, the weights of recurrent synaptic connections are symmetric (i.e., the weight from neuron i to neuron j is identical to the weight from neuron j to neuron i). This necessitates that lateral inhibition is not implemented through a pool of inhibitory neurons. Instead, the network dynamics is defined by only one pool of M network neurons (the same number as the number of excitatory network neurons in $\mathcal{M}$). Since the inhibitory neurons in $\mathcal{M}$ show linear response properties, the inhibition in the network depends linearly on the activity of excitatory neurons in the network. One can therefore model the inhibition in the network by direct inhibitory connections between excitatory neurons (where synaptic delays are neglected) with weight β. For clarity, we provide the full description of the approximate dynamics in the following.

The approximate dynamics is described by M network neurons. Network neurons have instantaneous firing rates that depend exponentially on their membrane potential, as given in eq. (1). Whenever neuron m spikes, the output trace $\tilde{z}_m(t)$ of neuron m is set to 1 for a period of duration τ (this corresponds to a rectangular PSP; the same definition applies to output traces $\tilde{y}_m(t)$ of input neurons). After emitting a spike, the neuron enters a refractory period of duration $\tau = 10$ ms, during which its instantaneous spiking probability is zero. Note that for this definition of the output trace, the state vector $\boldsymbol{z}(t)$ is identical to the vector of output traces $\tilde{\boldsymbol{z}}(t)$. Lateral inhibition in the network is established by direct inhibitory connections between excitatory neurons, leading to membrane potentials

$$u_m(t) = \sum_{i}^{N} w_{im}\tilde{y}_i(t) - \sum_{j \neq m} \beta \tilde{z}_j(t) + c_m, \tag{3}$$

where c_m denotes some neuron-specific excitability of the neuron that is independent of the input and network activity. Each network neuron receives feedforward synaptic inputs $\tilde{y}_1(t), \ldots, \tilde{y}_N(t)$ whose contribution to the membrane potential of a neuron m at time t depends on the synaptic efficiency w_{im} between the input neuron i and the network neuron m. Network neurons are all-to-all recurrently connected. The second term in (3) specifies this recurrent input where β is the inhibitory recurrent weight of the connection between network neuron j and network neuron m. It has been shown in [Buesing et al., 2011] that for such membrane potentials, the distribution of network states is given by the Boltzmann distribution:

$$p_{\text{Network}}(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W}) = \frac{1}{Z} \exp\left\{ \gamma \cdot \left(\sum_{i,m} w_{im} y_i z_m + \frac{1}{2} \sum_{m \neq l} \beta z_m z_l + \sum_{m} c_m z_m \right) \right\}, \tag{4}$$

where Z is a normalizing constant.

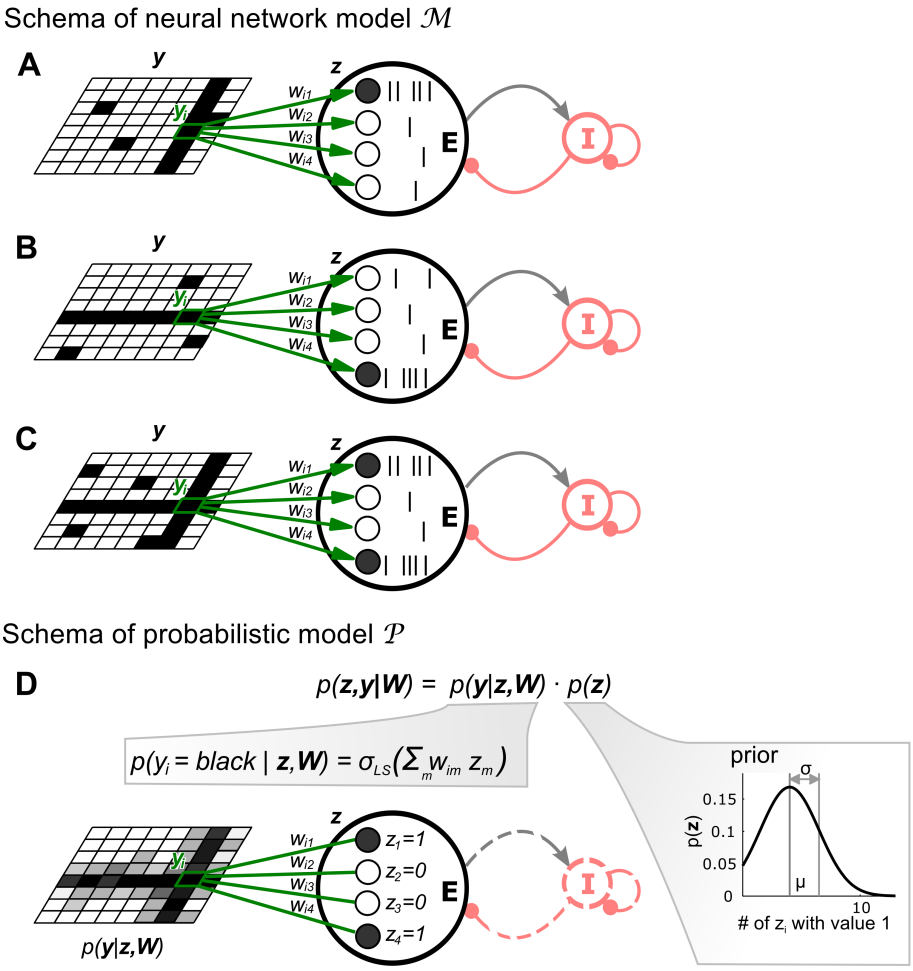

Figure 3: **Relationship between the data-based model $\mathcal{M}$ and the generative probabilistic model $\mathcal{P}$. A-C**) Schema of the response of model $\mathcal{M}$ to superimposed bars. Network inputs (left) are schematically arranged in a 2D array for clarity of the argument. Black indicates highly active inputs neurons. **A**) A vertical bar with added noise is presented to $\mathcal{M}$. This input activates an excitatory neuron (filled circle, spiking activity is indicated), similar as in hard WTA models. **B**) Another neuron is activated by a horizontal bar, also similar as in hard WTA models. **C**) A combination of these two basic input patterns activates both neurons in $\mathcal{M}$. This response is inconsistent with a WTA model, and with any generative model based on mixture distributions. But it can still be viewed as approximate inference of the posterior distribution $p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W})$ over hidden causes $\boldsymbol{z}$ for the given inputs $\boldsymbol{y}$ in the probabilistic model $\mathcal{P}$ shown in D. **D**) Schema of probabilistic model $\mathcal{P}$. The joint $p(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{W})$ is defined by the prior $p(\boldsymbol{z})$ and the likelihood $p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{W})$. Synaptic efficacies $\boldsymbol{W}$ implicitly define the likelihood over inputs $\boldsymbol{y}$ for given hidden causes $\boldsymbol{z}$ (probability values for inputs y_i indicated by shading of squares; σ_{LS} denotes the logistic sigmoid function). In the likelihood model, a given input y_i is 1 (corresponding to a black pixel in this example) with high probability if it has a large w_{im} to at least one active hidden cause z_m. In the depicted example, y_i belongs to two bars (see A, B) with corresponding active hidden causes. Due to the nonlinear behavior of the likelihood, its probability is comparable to one where only one of the hidden causes z_m is active. The inset on the right depicts the Gaussian prior $p(\boldsymbol{z})$ on hidden causes $\boldsymbol{z}$ with $\mu = 4$ and $\sigma = 2.5$. The prior implicitly incorporates in $\mathcal{P}$ the impact of the inhibitory feedback in the data-based model $\mathcal{M}$ (therefore indicated with dashed lines).

2.3.2 A probabilistic model $\mathcal{P}$ for the layer 2/3 microcircuit motif:

Fig. 3A-C illustrates the putative stochastic computation performed by the model $\mathcal{M}$, i.e., how network input leads to network activity in the model. Assume that we have a probabilistic model $\mathcal{P}$ for the inputs $\boldsymbol{y}$ defined by a prior $p(\boldsymbol{z})$ and a likelihood $p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{W})$. These distributions describe how one can generate input samples $\boldsymbol{y}$ by first drawing a *hidden state* vector $\boldsymbol{z}$ from $p(\boldsymbol{z})$ and then drawing an input vector $\boldsymbol{y}$ from $p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{W})$. Therefore, such a probabilistic model $\mathcal{P}$ is also called a generative model (for the inputs).

If the distribution of network states (4) is the posterior distribution given by

$$p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W}) = \frac{p(\boldsymbol{z})p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{W})}{\sum_{\boldsymbol{z}'} p(\boldsymbol{y}, \boldsymbol{z}'|\boldsymbol{W})}, \tag{5}$$

then the network performs probabilistic inference in this probabilistic model $\mathcal{P}$. The inference task described by eq. (5) assumes that $\boldsymbol{y}$ is given and the hidden causes $\boldsymbol{z}$ (such as the basic components of a visual scene) have to be inferred. This inference can intuitively also be described as providing an "explanation" $\boldsymbol{z}$ for the current observation $\boldsymbol{y}$ according to the generative model $\mathcal{P}$. In the following, we describe a probabilistic model $\mathcal{P}$ and show that eq. (4) approximates the posterior distribution of this model. This implies that the simplified dynamics of the data-based model $\mathcal{M}$ approximate probabilistic inference in the probabilistic model $\mathcal{P}$.

The probabilistic model $\mathcal{P}$ is defined by two distributions, the prior over hidden variables $p(\boldsymbol{z})$ (that captures constraints on network activity imposed for example by lateral inhibition) and the conditional likelihood distribution over input variables $p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{W})$ that describes the probability of input $\boldsymbol{y}$ for a given network state $\boldsymbol{z}$ in a network with parameters $\boldsymbol{W}$. These two distributions define the joint distribution over hidden and visible variables since $p(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{W}) = p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{W})p(\boldsymbol{z})$, see Fig. 3D. The specific forms of these two distributions in the probabilistic model $\mathcal{P}$ considered for $\mathcal{M}$ are discussed in the following and defined by eqs. (6)-(8) below.

In previously considered hard WTA models [Nessler et al., 2013, Habenschuss et al., 2013b], strong lateral inhibition was assumed. This corresponded to a prior where only a single component z_j of the hidden vector $\boldsymbol{z}$ can be active at any time. The biologically more realistic divisive inhibition in $\mathcal{M}$ allows several of them to fire simultaneously. This corresponds to a prior that induces sparse activity in a soft manner ("adaptive" k-WTA): It does not enforce a strict ceiling k on the number of z-neurons that can fire within a time interval of length τ, but only tries to keep this number within a desired range. Hence we use as prior in $\mathcal{P}$ a Gaussian distribution

$$p(\boldsymbol{z}) = \frac{1}{Z_{\text{prior}}} \exp\left(-\frac{1}{2\sigma^2}\left(\sum_{m=1}^{M} z_m - \mu\right)^2\right) \quad , \tag{6}$$

where Z_{prior} is a normalizing constant, $\mu \in \mathbb{R}$ is a parameter that shifts the mean of the distribution, and $\sigma^2 > 0$ defines the variance (see Fig. 3D). Note that the Gaussian is restricted to integers as the sum runs over binary random variables $z_1, \ldots, z_M$.

As in other generative models we assume for the sake of theoretical tractability that the conditional likelihood $p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{W})$ factorizes, so that each input y_i is independently

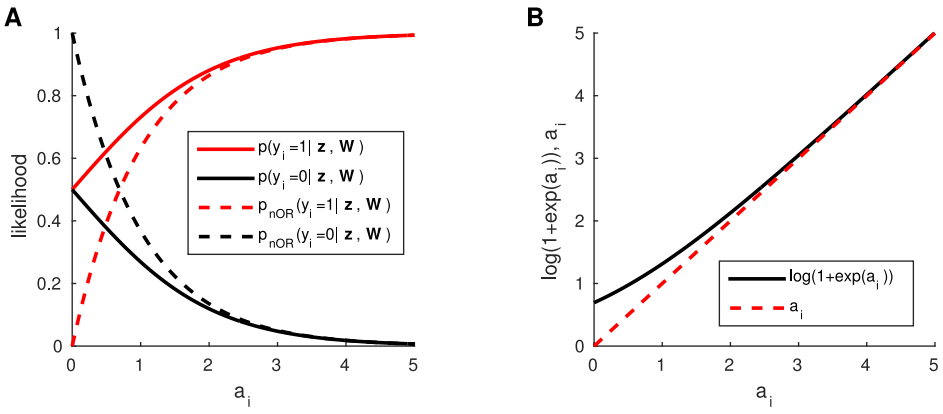

Figure 4: **Likelihood model of $\mathcal{P}$. (A)** Likelihood that an input y_i is 1 (red) or 0 (black) for the proposed likelihood model (8) (full lines) and for the noisy-OR likelihood (9) (broken lines). The likelihood of an input y_i depends on the hidden causes $\boldsymbol{z}$ through the weighted contribution $a_i = \gamma \bar{\boldsymbol{w}}_i^T \boldsymbol{z}$ of the hidden causes to this input. For large a_i, the likelihood of $y_i = 1$ approaches 1. While for $a_i = 0$, the likelihood of $y_i = 1$ is 0.5 in the proposed likelihood model, whereas it is 0 in the noisy-OR model. **B** Approximation of $\log(1 + \exp(a_i))$ (black full line) by a_i (red broken line) as used in Eq. (11).

explained by the current network state $\boldsymbol{z}$:

$$p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{W}) = \prod_{i=1}^{N} p(y_i|\boldsymbol{z}, \boldsymbol{W}). \tag{7}$$

A probabilistic model for hard WTA circuits can only explain each input variable y_i by a single hidden cause z_m. In contrast, in probabilistic models with soft inhibition and the prior (6), several hidden causes can be active simultaneously and explain together an input variable y_i. We define $\bar{\boldsymbol{w}}_i$ as the vector of weights $\bar{\boldsymbol{w}}_i = (w_{i1}, \ldots, w_{iM})^T$ that define the likelihood for variable y_i. We consider the following likelihood model

$$p(y_i|\boldsymbol{z}, \boldsymbol{W}) = \frac{\exp(\gamma \bar{\boldsymbol{w}}_i^T \boldsymbol{z})^{y_i}}{1 + \exp(\gamma \bar{\boldsymbol{w}}_i^T \boldsymbol{z})} = \frac{\exp(a_i)^{y_i}}{1 + \exp(a_i)} = \sigma_{\text{LS}}\left((-1)^{y_i} a_i\right), \tag{8}$$

where we have defined $a_i = \gamma \bar{\boldsymbol{w}}_i^T \boldsymbol{z}$, and σ_{LS} is the logistic sigmoid $\sigma_{\text{LS}}(u) = \frac{1}{1+\exp(-u)}$. This likelihood function is shown in Figure 4A together with the often used noisy-OR likelihood. Note that if none of the hidden causes z_m is active, i.e., $z_m = 0$ for all m, then $y_i = 1$ with probability 0.5. Each active hidden cause $z_m = 1$ with $w_{im} > 0$ increases the probability that input variable y_i assumes the value 1, see also Fig. 3D). This likelihood, allows the generative model to deal with situations where an input neuron can fire in the context of different hidden causes, for example with pixels in the network inputs that lie in the intersection of different patterns, see Fig. 3). The soft Gaussian prior (6) allows the internal model to develop modular representations for different components of complex input patterns. This likelihood is quite similar to the frequently used noisy-OR model (see e.g. [Neal, 1992, Saund, 1995]):

$$p_{\text{nOR}}(y_i = 0|\boldsymbol{z}, \boldsymbol{W}) = \exp(-a_i)^{1-y_i}(1 - \exp(-a_i))^{y_i} \tag{9}$$

One difference is that (for purely excitatory weights), the probability of an input y_i being zero is at most 0.5 in the proposed likelihood, while it can become 0 in the noisy-OR model. Such a model may reflect the situation that network inputs are noisy, so their firing rates are never zero.

We now analyze the relationship between the probabilistic model $\mathcal{P}$ and the description of the data-based model $\mathcal{M}$ in the neural sampling framework. We will see that $\mathcal{M}$ approximates probabilistic inference in $\mathcal{P}$. Finally, we show that adaptation of network parameters in $\mathcal{M}$ through STDP can be understood as an approximate stochastic expectation maximization (EM) process in the corresponding probabilistic model $\mathcal{P}$.

2.3.3 Interpretation of the dynamics of $\mathcal{M}$ in the light of $\mathcal{P}$:

Using the likelihood and prior of $\mathcal{P}$, the posterior of hidden states $\boldsymbol{z}$ for given inputs $\boldsymbol{y}$ and given parameters $\boldsymbol{W}$ is

$$p(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{W}) = \frac{1}{Z}\exp\left\{\gamma\cdot\left(\sum_{i,m} w_{im}y_i z_m - \frac{1}{2}\sum_{m\neq l}\beta z_m z_l + \sum_m \alpha z_m - \frac{1}{\gamma}\sum_i \log(1+\exp(a_i))\right)\right\}, \tag{10}$$

where Z is a normalizing constant, $\beta = \frac{1}{\gamma\sigma^2}$, and $\alpha = \frac{2\mu-1}{2\gamma\sigma^2}$. The terms including β and α stem from the prior, while the last term stems from the normalization of the likelihood. When we compare this posterior to the posterior of the network model eq. (4), we see that they are quite similar with β denoting the strength of inhibitory connections and α being the neural excitabilities.

The last term in eq. (10) is problematic since the a_i's depend on $\boldsymbol{z}$ and thus the whole posterior is not a Boltzmann distribution and can therefore not be computed by the model $\mathcal{M}$. It turns out however that this last term can be approximated quite well by a term that is linear in $\boldsymbol{z}$. Note that for zero a_i (i.e., for zero weights or for the zero-$\boldsymbol{z}$-vector), this last term evaluates to $\log(2)$. But as a_i increases, one can neglect the 1 in the logarithm and the expression quickly approaches a_i. We can thus write

$$\sum_i \log(1+\exp(a_i)) \approx \sum_i a_i = \gamma\sum_i\sum_m w_{im}z_m = \gamma\sum_m z_m\left(\sum_i w_{im}\right), \tag{11}$$

where the term in the brackets on the right is just the L1-norm of the weight vector of neuron j. Note that for a given weight matrix, an increased γ leads to a better approximation. The approximation of $\log(1+\exp(a_i))$ by a_i is illustrated in Fig. 4B. Hence, the first approximate posterior we consider is given by

$$p_{\mathrm{A1}}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{W}) = \frac{1}{Z}\exp\left\{\gamma\cdot\left(\sum_{i,m} w_{im}y_i z_m - \frac{1}{2}\sum_{m\neq l}\beta z_m z_l + \sum_m \alpha z_m - \sum_m z_m\sum_i w_{im}\right)\right\}. \tag{12}$$

This is a Boltzmann distribution of the form (4) and the last term accounts to a neuron-specific homeostatic bias that depends on the sum of incoming excitatory weights. Matching the terms of this equation to the terms in eq. (4) and performing the same match in the membrane potential (3), we see that the membrane potential of neurons in this

approximation is given by

$$u_m(t) = \sum_i^N w_{im}\tilde{y}_i(t) - \sum_{j\neq m} \beta\tilde{z}_j(t) + \alpha - \sum_i w_{im}. \tag{13}$$

If excitatory weight vectors are normalized to an L1-norm of $w_{\text{norm}} = \sum_i w_{im}$ for all m, this simplifies to

$$u_m(t) = \sum_i^N w_{im}\tilde{y}_i(t) - \sum_{j\neq m} \beta\tilde{z}_j(t) + \alpha - w_{\text{norm}}. \tag{14}$$

Note that w_{norm} can be incorporated into α. Such constant weight sum could be enforced in a biological network by a synaptic scaling mechanism [Turrigiano and Nelson, 2004, Savin et al., 2010] that normalizes the sum of incoming weights to a neuron. For the data-based model $\mathcal{M}$, [Jonke et al., 2017] used uniform excitabilities α for all excitatory neurons and no homeostasis for simplicity, see eq. (2). We will argue below under which conditions this approximation is justified. We consider the posterior distribution of such circuits as our second approximation of the exact posterior:

$$p_{\text{A2}}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{W}) = \frac{1}{Z}\exp\left\{\gamma\cdot\left(\sum_{i,m} w_{im}y_i z_m - \frac{1}{2}\sum_{m\neq l}\beta z_m z_l + \sum_m z_m(\alpha - w_{\text{norm}})\right)\right\}. \tag{15}$$

Note that in this case, w_{norm} effectively leads to a smaller mean of the Gaussian prior. A detailed discussion about how parameters of the generative probabilistic model $\mathcal{P}$ can be mapped to parameters of the data-based microcircuit motif model $\mathcal{M}$ is provided in *Network parameter interpretation* in *Methods*.

We evaluated the impact of these two approximations in Földiak's superposition-of-bars problem [Földiak, 1990]. This is a standard blind-source separation problem that has also been used in [Jonke et al., 2017] to evaluate the data-based network model $\mathcal{M}$. In this problem, input patterns $\boldsymbol{y}$ are two-dimensional pixel arrays on which horizontal and vertical bars (lines) are superimposed, see Fig. 5A. Input patterns were generated with a superposition of 1 to 3 bars from the distribution that was used in [Jonke et al., 2017]. We performed inference of hidden causes by sampling from the approximate posterior distribution $p_{\text{A1}}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{W})$ given by eq. (12) for a network of 20 hidden-cause neurons. We performed approximate stochastic online EM in order to optimize the parameters of the model. The synaptic update rule (20) used for parameter updates is discussed in detail below. We compared the approximated posterior to the exact one (10) by computing the Kullback-Leibler (KL) divergence $D_{\text{KL}}(p(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{W})||p_{\text{A1}}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{W}))$. The KL divergence was small throughout learning, with a slight decrease during the process, see Fig. 5B (mean KL divergence during the second half of training was 0.55). To evaluate what that KL-divergence means for the inference, we considered the hidden state vector $\boldsymbol{z}_{\max}$ with the maximum posterior probability after learning in the exact and approximate posterior and used this to reconstruct the input pattern by computing $\hat{y} = \sigma_{\text{LS}}(W^T\boldsymbol{z}_{\max})$. The reconstructed inputs for the 8 example inputs of Fig. 5A are shown in Fig. 5C for the exact posterior (top) and the approximate posterior (bottom). The approximate reconstructions resemble the exact ones in many cases, with occasional misses of a basic pattern. The final weights of the 20 neurons are shown in Fig. 5D in the two-dimensional layout of the input to facilitate interpretability. Note that all basic patterns were represented by individual neurons with additional neurons that specialized on combined patterns.

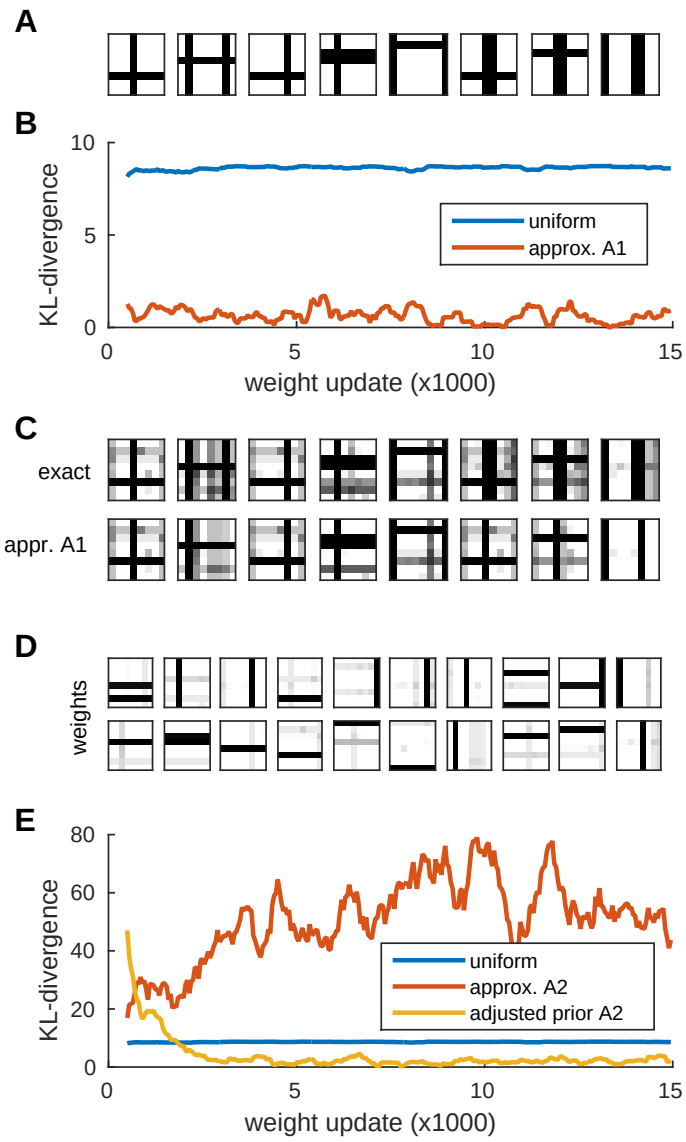

Figure 5: **Empirical evaluation of approximations in a superposition-of-bars task. A**) Sample input patterns, depicted on the 8×8 grid. Patterns consist of a varying number of superimposed horizontal or vertical bars. **B**) Evolution of the Kullback-Leibler divergence between the exact posterior and the posterior of approximation A1 during learning (red). As a comparison, the KL-divergence to a uniform distribution is indicated in blue. **C**) Example reconstruction of inputs from panel A from the posterior according to the hidden states with maximum probability in the exact posterior (top row) and the posterior of approximation A1 after learning (bottom). Scale between 0 (white) and 1 (black). **D**) Weights vectors of network neurons depicted on the 8×8 grid as the input in panel A. Scale between 0 (white) and 6 (black). **E**) Evolution of the Kullback-Leibler divergence between the exact posterior, the posterior of approximation A2 during learning (red), and the posterior of approximation A2 with an adjusted sparsity prior d (yellow) during learning. As a comparison, the KL-divergence to a uniform distribution is indicated in blue.

The approximate posterior p_{A2} is equivalent to the approximate posterior p_{A1} if the synaptic weights of each neuron are normalized to a common $L1$ norm. It turned out in [Jonke et al., 2017] that such a normalization is not strictly necessary. In this work, the data-based model $\mathcal{M}$ managed to perform blind source separation with a posterior that can be best described by p_{A2} without normalization of synaptic efficacies. We found that for the superposition-of-bars problem, the posterior distribution p_{A2} differs significantly from the exact posterior if we set $w_{\mathrm{norm}} = 0$ in eq. (15), see red line in Fig. 5E. This difference is mostly induced by the tendency of p_{A2} to prefer many hidden causes due to the missing last term in eq. (15). If the prior was corrected to reduce the number of hidden causes, we found that the approximation was significantly improved, in particular as the network weight vectors approached their final norm values, see yellow line in Fig. 5E. A closer inspection of eq. (15) in comparison with eq. (12) shows that this approximation is effective if basic patterns consist of a similar number of active units, because otherwise patterns with strong activity are preferred over weakly active ones (this is exactly what the last term in eq. (12) compensates for).

We conclude from this analysis that the microcircuit motif model $\mathcal{M}$ approximates probabilistic inference in a noisy-OR-like probabilistic model of its inputs. The prior on network activity favors sparse network activity, but does not strictly enforce a predefined activity level. Such a more flexible regulation of network activity is obviously important when the network input is composed of a varying number of basic component patterns. We show below that this network behavior in combination with STDP allows the microcircuit motif model $\mathcal{M}$ to perform blind source separation of mixed input sources. Our analysis above has shown that the computation of blind source separation in $\mathcal{M}$ can be facilitated either by the normalization of activity in input populations, or by homeostatic mechanisms that normalize excitatory synaptic efficacies within each neuron.

2.3.4 STDP in the microcircuit motif model $\mathcal{M}$ creates an internal model of network inputs:

After we have established a link between a well-defined probabilistic model $\mathcal{P}$ and the spiking dynamics of microcircuit motif model $\mathcal{M}$, we can now analyze plasticity in the network. The probabilistic model $\mathcal{P}$ defines a likelihood distribution over inputs $\boldsymbol{y}$ that depends on the parameters $\boldsymbol{W}$ through

$$p(\boldsymbol{y}|\boldsymbol{W}) = \sum_{\boldsymbol{z}} p(\boldsymbol{z}) p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{W}), \tag{16}$$

where the sum runs over all possible hidden states $\boldsymbol{z}$.

We propose that STDP in $\mathcal{M}$ can be viewed as an adaptation of the parameters $\boldsymbol{W}$ so that $p(\boldsymbol{y}|\boldsymbol{W})$ as defined by the probabilistic model $\mathcal{P}$ with parameters $\boldsymbol{W}$ approximates the actually encountered distribution of spike inputs $p^*(\boldsymbol{y})$ within the constraints of the prior $p(\boldsymbol{z})$. Since the prior typically is defined to favor sparse representations, this tends to extract the hidden sources of these patterns, an operation called blind source separation [Földiak, 1990].

More precisely, we show that STDP in $\mathcal{M}$ approximates stochastic online EM [Sato, 1999, Bishop, 2006] in $\mathcal{P}$. Given some external distribution $p^*(\boldsymbol{y})$ of synaptic inputs, EM adapts the model parameters $\boldsymbol{W}$ such that the model likelihood distribution $p(\boldsymbol{y}|\boldsymbol{W})$ approximates the given distribution $p^*(\boldsymbol{y})$. More formally, the Kullback-Leibler divergence between the likelihood of inputs in the internal model $p(\boldsymbol{y}|\boldsymbol{W})$ and the empirical data distribution $p^*(\boldsymbol{y})$ is brought to a local minimum. The theoretically optimal learning

rule for EM contains non-local terms which are hard to interpret from a biological point of view. In the following, we derive a local approximation to yield a simple STDP-like learning rule.

The goal of the EM algorithm is to find parameters that (locally) minimize the Kullback-Leibler divergence between the likelihood $p(\boldsymbol{y}|\boldsymbol{W})$ of inputs in the probabilistic model $\mathcal{P}$ and the empirical data distribution $p^*(\boldsymbol{y})$, that is, the distribution of inputs experienced by the network $\mathcal{M}$. This is equivalent to the maximization of the average data log-likelihood $E_{p^*}[\log p(\boldsymbol{y}|\boldsymbol{W})]$. For a given set of training data $\boldsymbol{Y}$ and corresponding unobserved hidden variables $\boldsymbol{Z}$, this corresponds to maximizing $\log p(\boldsymbol{Y}|\boldsymbol{W})$. The optimization is done by iteratively performing two steps. For given parameters $\boldsymbol{W}^{\text{old}}$, the posterior distribution over hidden variables $p(\boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{W}^{\text{old}})$ is determined (the E-step). Using this distribution, one then performs the M-step where $E_{p(\boldsymbol{Z}|\boldsymbol{Y},\boldsymbol{W}^{\text{old}})}[\log p(\boldsymbol{Y}, \boldsymbol{Z}|\boldsymbol{W})]$ is maximized with respect to $\boldsymbol{W}$ to obtain better parameters for the model. These steps are guaranteed to increase (if not already at a local optimum) a lower bound $\mathcal{L} = E_{p(\boldsymbol{Z}|\boldsymbol{Y},\boldsymbol{W}^{\text{old}})}\left[\log \frac{p(\boldsymbol{Y},\boldsymbol{Z}|\boldsymbol{W})}{p(\boldsymbol{Y},\boldsymbol{Z}|\boldsymbol{W}^{\text{old}})}\right]$ on the data log likelihood, that is, $\mathcal{L} \leq \log p(\boldsymbol{Y}|\boldsymbol{W})$. These steps are iterated until convergence of parameters to a local optimum [Bishop, 2006]. Computation of the M-step in the probabilistic model $\mathcal{P}$ is hard. In the generalized EM algorithm, the M-step is replaced by a procedure that just improves the parameters, without necessarily obtaining the optimal ones for a single M-step. This can be done for example by changing the parameters in the direction of the gradient

$$\Delta w_{im} \propto \frac{\partial}{\partial w_{im}} E_{p(\boldsymbol{Z}|\boldsymbol{Y},\boldsymbol{W}^{\text{old}})}[\log p(\boldsymbol{Y}, \boldsymbol{Z}|\boldsymbol{W})]. \tag{17}$$

Since in our model, we assume that synaptic efficacy changes are instantaneous for each pre-post spike pair, we need to consider an online-version of the generalized EM algorithm. In stochastic online EM, for each data example $\boldsymbol{y}^{(k)}$, a sample $\boldsymbol{z}^{(k)}$ from the posterior is drawn (the stochastic E-step) and parameters are changed according to this sample-pair. As shown above, the $\mathcal{M}$ network implements an approximation of the stochastic E-step. In the M-step, each parameter w_{im} is then updated in the direction of the gradient $\Delta w_{im} \propto \frac{\partial}{\partial w_{im}} \log p(\boldsymbol{y}^{(k)}, \boldsymbol{z}^{(k)}|\boldsymbol{W})$. As the prior $p(\boldsymbol{z})$ in $\mathcal{P}$ does not depend on $\boldsymbol{W}$, this is equivalent to $\Delta w_{im} \propto \frac{\partial}{\partial w_{im}} \log p(\boldsymbol{y}^{(k)}|\boldsymbol{z}^{(k)}, \boldsymbol{W})$. For the likelihood given by eq. (8), this derivative is given by

$$\frac{\partial}{\partial w_{im}} \log p(\boldsymbol{y}^{(k)}|\boldsymbol{z}^{(k)}, \boldsymbol{W}) = \gamma z_m \left(y_i - \frac{\exp(a_i)}{1 + \exp(a_i)}\right) = \gamma z_m \left(y_i - \sigma_{LS}(a_i)\right), \tag{18}$$

where σ_{LS} denotes the logistic sigmoid function. Hence, the synaptic update rule for weight w_{im} is given by

$$\Delta w_{im} = \eta z_m \left(y_i - \sigma_{LS}(a_i)\right), \tag{19}$$

where $\eta > 0$ is a learning rate. This learning rule is not local as it requires information about the activation of all output neurons as well as values of all synaptic weights originating from input neuron i. In order to make this biologically plausible we approximate rule (19) by

$$\Delta w_{im} = \eta z_m \left(y_i - \sigma_{LS}(\gamma w_{im})\right), \tag{20}$$

This rule uses only locally available information at the synapse. What are the consequences of this approximation during learning? If only a single neuron in the network is active, then the approximation is exact. Otherwise, the approximation ignores what other neurons contribute to the explanation of input component y_i. This means that for $y_i = 1$, the

weight will further be increased even if $y_i = 1$ is already fully explained by the network activity. For $y_i = 0$, the decrease will in general be smaller than in the exact rule (since weights are non-negative). Note however that only the magnitudes of weigh changes are affected, but not which weights change and the sign of the change. Hence, we can conclude that the angle between the approximate parameter change vector and the exact parameter change vector is between 0 and 90 degrees. In other words, the inner product of these two vectors is always non-negative and the updates are performed in the correct direction. This was confirmed in simulations. In the learning experiment described in Fig. 5, we compared the approximate update (that was used to optimize the model) with the update that was proposed by the exact rule (18) at every 50^{th} update step. The angle between the exact and approximate update vector was between $0°$ and $84°$ with a mean of $57°$.

In the simplified dynamics, a value of $z_m^{(k)} = 1$ is indicated by a spike in network neuron m when pattern $\boldsymbol{y}^{(k)}$ is presented as input. We therefore map this update to the following synaptic plasticity rule: For each postsynaptic spike, update weight w_{im} according to

$$\Delta w_{im} = \eta \left(y_i(t) - \sigma_{LS}(\gamma w_{im}) \right) \ . \tag{21}$$

According to this learning rule, when the presynaptic neuron i spikes shortly before the postsynaptic neuron this results in long-term potentiation (LTP) which is weight dependent according to the term $\sigma_{LS}(w_{im})$. Due to the weight dependence, large weights lead to small weight changes, with vanishing changes for very large weights. When a post-synaptic spike by neuron m is not preceded by a presynaptic spike by neuron i (e.g. when the presynaptic spike comes after the post-synaptic spike), this results in long term depression (LTD). LTD is also weight dependent, but to a much lesser extent as the weight-dependent factor varies only between 0.5 and 1. This behavior is mimicked by the standard STDP rule implemented in the data-based model $\mathcal{M}$ that a standard weight dependence where updates exponentially decreased with w_{im} for LTP and did not depend on w_{im} for LTD.

Hence, the dynamics and synaptic plasticity of the data-based model $\mathcal{M}$ can be understood as an approximation of EM in the probabilistic model $\mathcal{P}$, that creates an internal model for the distribution $p^*(\boldsymbol{y})$ of network inputs. This internal probabilistic model is defined by a noisy-OR-like likelihood term and a sparse prior on the hidden causes of the current input pattern. Hence, STDP can be understood as optimizing model parameters such that the observed distribution of input patterns can be explained through a set of basic patterns (hidden causes). It is assumed that the input at each time point can be described by a combination of a sparse subset of these patterns. In other words, STDP in the microcircuit motif model $\mathcal{M}$ performs blind source separation of input patterns.

3 Discussion

We have provided a novel theoretical framework for analyzing and understanding computational properties that emerge from STDP in a prominent cortical microcircuit motif: interconnected populations of pyramidal cells and PV^+ interneurons in layer 2/3. The computer simulations in [Jonke et al., 2017], that were based on the data from [Avermann et al., 2012], indicate that the computational operation of this network motif cannot be captured adequately by a WTA model. Instead, this work suggests a k-WTA model, where a varying number of the most excited neurons become active. Since the WTA circuit model turns out to be inadequate for capturing the dynamics of interacting pyramidal cells and PV^+ interneurons, one needs to replace the probabilistic model that one had previously used to analyze the impact of STDP on the computational function of the network motif.

Mixture models such as those proposed by [Nessler et al., 2013] and [Habenschuss et al., 2013b] are inseparably tied to WTA dynamics: For drawing a sample from a mixture model one first decides stochastically from which component of the mixture model this sample should be drawn (and only a single component can be selected for that). We have shown here that a quite different generative model, similar to the noisy-OR model, captures the impact of soft lateral inhibition on emergent network codes and computations much better (Fig. 3). The noisy-OR model is well-known in machine learning [Neal, 1993, Saund, 1995], but has apparently not previously been considered in computational neuroscience. Our probabilistic model $\mathcal{P}$ further suggests that the varying number of active neurons in the circuit may depend both on a prior that is encoded by the network parameters and the familiarity of the network input.

We have shown that the evolution of the dynamics and computational function of the network motif under STDP can be understood from the theoretical perspective as an approximation of expectation maximization (EM) for fitting a noisy-OR based generative model to the statistics of the high dimensional spike input stream. This link to EM is very helpful from a theoretical perspective, since EM is one of the most useful theoretical principles that are known for understanding self-organization processes. In particular, this theoretical framework allows us to elucidate emergent computational properties of the network motif for spike input streams that contain superimposed firing patterns from upstream networks. It disentangles these patterns and represents the occurrence of each pattern component by a separate sparse assembly of neurons, as already postulated in [Földiak, 1990].

The established relationship between the network $\mathcal{M}$ and the probabilistic model $\mathcal{P}$ allows us to relate the network parameters α and w^{IE} (eq. (2)) of $\mathcal{M}$ to the parameters of the generative model $\mathcal{P}$. Briefly (for a detailed discussion, see *Network parameter interpretation* in *Methods*), the excitability α of pyramidal cells is proportional to $\frac{2\mu-1}{2\sigma^2}$, see eq. (26). The strength of inhibitory connections w^{IE} to the pool of pyramidal cells is proportional to $\frac{1}{\sigma^2}$, see eq. (30). Hence, a large μ in combination with a small σ^2 (i.e., a sharp activity prior), leads to a large spontaneous activity that is tightly regulated by strong inhibitory feedback. On the other hand, a broad prior (larger σ^2) leads to weaker inhibitory feedback, thus allowing the network to attain a broader range of activities.

Related work

A related theoretical study for WTA circuits was performed in [Nessler et al., 2013, Habenschuss et al., 2013a, Kappel et al., 2014] and extended to sheets of WTA circuits in [Bill et al., 2015]. It was assumed in these models that inhibition normalizes network activity exactly, leading to a strict WTA behavior. The analysis in the present work is much more complex and necessarily has to include a number of approximations. Out analysis reveals that the softer type of inhibition that we studied provides the network with additional computational functionality. There exists also a structural similarity of the proposed learning rule (21) to those reported in [Nessler et al., 2013, Habenschuss et al., 2013a, Kappel et al., 2014]. This is insofar significant as it raises the question why the application of almost the same learning rule in one motif leads to learning and extraction of a single hidden cause and in another to the extraction of multiple causes. The answer most likely lies in the interplay between "prior knowledge" in the model (e.g. in the form of the intrinsic excitability of neurons), the learning rule and inhibition strength: As there are multiple neurons in the proposed microcircuit motif model which can spike in response to the same input, each one of them can adapt its synaptic weights to increase the likelihood of spiking

again whenever the same or a similar input pattern is presented in the future, possibly in conjunction with other different input components. This is manifested through increased total input strength to those neurons when the pattern is seen again. But this results also in increased total inhibition to all other neurons, thereby effectively limiting the number of winners. As there is no fixed normalization of firing rates (probabilities), as soon as the input strength caused by a single feature component is strong enough to trigger the spike in some neuron, the neuron will respond to each pattern which consists of that particular feature. On average this will force neurons to specialize on a single feature component. Therefore, after learning, each spike can be interpreted as indication of a particular feature component.

The noisy-OR model eq. (9) is tightly related to the likelihood model used in this article. It is one of the most basic likelihood models that allows to combine basic patterns. Noisy-OR and related models have previously been used in the machine learning literature as models for nonlinear component extraction [Saund, 1995, Lücke and Sahani, 2008], or as basic elements in belief networks [Neal, 1992], but they have so far not been linked to cortical processing.

The extraction of reoccurring components of input patterns is closely related to blind source separation and independent component analysis (ICA) [Hyvärinen et al., 2004]. Previous work in this direction includes implementations of ICA in artificial neural networks [Hyvärinen, 1999], see also [Lücke and Eggert, 2010]. These abstract models are only loosely connected to computation in cortical network motifs. [Savin et al., 2010] investigated ICA in the context of spiking neurons. Theoretical rules for intrinsic plasticity were derived which enable neurons in combination with input normalization, weight scaling, and STDP, to extract independent components of inputs. An interesting difference is that the inhibition in [Savin et al., 2010] acts to decorrelate neuronal activity. Intrinsic plasticity on the other hand enforces sparse activity (this sparsening has to happen on the time scale of input presentations). In our probabilistic model $\mathcal{P}$, sparse network activity is enforced by a prior over network activities, implemented in $\mathcal{M}$ through the inhibitory feedback that models experimentally found network connectivity [Avermann et al., 2012]. This inhibition naturally acts on a fast time scale [Okun and Lampl, 2008], while the time scale for intrinsic plasticity is unclear [Turrigiano and Nelson, 2004].

4 Methods

4.1 Definition of $\mathcal{M}$: Data-based network model for a layer 2/3 microcircuit motif

The layer 2/3 microcircuit motif was modeled in [Jonke et al., 2017] by the data-based model $\mathcal{M}$. The model is described here briefly for completeness. See [Jonke et al., 2017] for a thorough definition. The model $\mathcal{M}$ consists of two reciprocally connected pools of neurons, an excitatory pool and an inhibitory pool. Inhibitory network neurons are recurrently connected. Excitatory network neurons receive additional excitatory synaptic input from a pool of N input neurons. Fig. 1A summarizes the connectivity structure of the data-based model $\mathcal{M}$ together with connection probabilities. Connection probabilities have been chosen according to the experimental data described in [Avermann et al., 2012].

Let $t_i^{(1)}, t_i^{(2)}, \ldots$ denote the spike times of input neuron i. The *output trace* $\tilde{y}_i(t)$ of input neuron i is given by the temporal sum of unweighted postsynaptic potentials (PSPs)

arising from input neuron i:

$$\tilde{y}_i(t) = \sum_f \epsilon(t - t_i^{(f)}), \tag{22}$$

where ϵ is the synaptic response kernel, i.e., the shape of the PSP. It is given by a double-exponential function with a rise time constant $\tau_r = 1$ ms and a fall time constant $\tau_f = 10$ ms. For given spike times, output traces of excitatory network neurons and inhibitory network neurons are defined analogously and denoted by $\tilde{z}_m(t)$ and $I_j(t)$ respectively.

The network consists of $M = 400$ excitatory neurons, modeled as stochastic spike response model neurons [Jolivet et al., 2006], see eqs. (1) and (2). See Sec. 4.2 for a motivation of network parameter values from a theoretical perspective.

The instantaneous firing rate ρ_m of neuron m can be re-written (by substituting eq. (2) in eq. (1)) as:

$$\rho_m(t) = \frac{1}{\tau} \frac{\exp\left(\gamma \sum_i w_{im} \tilde{y}_i(t) + \gamma\alpha\right)}{\exp\left(\gamma \sum_{j \in \mathcal{I}_m} w^{\mathrm{IE}} I_j(t)\right)} \quad . \tag{23}$$

Here, the numerator includes all excitatory contributions to the firing rate $\rho_m(t)$. The denominator in this term for the firing rate describes inhibitory contributions, thereby reflecting divisive inhibition [Carandini and Heeger, 2012].

Apart from excitatory neurons there are $M_{\mathrm{inh}} = 100$ inhibitory neurons in the network. Inhibitory neurons are also modeled as stochastic spike response neurons with an instantaneous firing rate given by

$$\rho_m^{\mathrm{inh}}(t) \ = \sigma_{\mathrm{rect}}(u_m^{\mathrm{inh}}(t)), \tag{24}$$

where σ_{rect} denotes the linear rectifying function $\sigma_{\mathrm{rect}}(u) = u$ for $u \geq 0$ and 0 otherwise. The membrane potentials of inhibitory neurons are given by

$$u_m^{\mathrm{inh}}(t) = \sum_{i \in \mathcal{E}_m} w^{\mathrm{EI}} \tilde{z}_i(t) - \sum_{j \in \mathcal{II}_m} w^{\mathrm{II}} I_j(t), \tag{25}$$

where $\tilde{z}_i(t)$ denotes synaptic input (output trace) from excitatory network neuron i, $\mathcal{E}_m$ ($\mathcal{II}_m$) denotes the set of indices of excitatory (inhibitory) neurons that project to inhibitory neuron m, and w^{EI} (w^{II}) denotes the excitatory (inhibitory) weight to inhibitory neurons.

Synaptic connections from input neurons to excitatory network neurons are subject to STDP. A standard version of STDP is employed with an exponential weight dependency for potentiation [Habenschuss et al., 2013b], see [Jonke et al., 2017].

The simulations for Fig. 2 are described in detail in [Jonke et al., 2017].

4.2 Network parameter interpretation:

In the section *Interpretation of the dynamics of* $\mathfrak{M}$ *in the light of* $\mathcal{P}$, we have established a relationship between the parameters of the probabilistic model $\mathcal{P}$ and network parameters. This relationship was however derived based on a simplified network model that included for example rectangular EPSPs and direct inhibitory connections without explicit inhibitory neurons. Nevertheless, one can also determine reasonable parameter settings for the data-based model $\mathcal{M}$ based on a prior on network activity that is defined in the probabilistic model $\mathcal{P}$. These parameters are the excitability α and the synaptic weights

between and within excitatory and inhibitory network neurons.

In this section we start by assuming such a prior eq. (6) with parameters $\mu = -3.4$ (this includes already a correction of the prior for the missing w_{norm}) and $\sigma^2 = 0.35$ as well as a fitting parameter $\gamma = 2$ (eq. 1) and deduce the parameters used in the simulations. As shown above, the neural excitability α is then given by $\alpha = \frac{1}{\gamma}\frac{2\mu-1}{2\sigma^2}$. We obtain for the chosen $\gamma = 2$:

$$\alpha = \frac{1}{\gamma}\left(\frac{2\mu - 1}{2\sigma^2}\right) = -5.57. \tag{26}$$

The inhibition strength β of the approximate dynamics eq. (3) is replaced by the weight w^{IE} from inhibitory neurons to excitatory neurons in $\mathcal{M}$. From the probabilistic model, we determined β as $\beta = \frac{1}{\gamma\sigma^2}$ under the assumption of rectangular inhibitory PSPs. In $\mathcal{M}$ we use double-exponential PSPs instead of rectangular ones. One therefore has to correct for differences in the PSP integrals. Using this correction, one obtains

$$\beta' = c_{\text{PSP}}\beta = \frac{c_{\text{PSP}}}{\gamma\sigma^2} = 1.114, \tag{27}$$

where c_{PSP} is the ratio between the integrals over the rectangular PSPs used in the approximate dynamics and the double-exponential PSPs used in $\mathcal{M}$ ($c_{\text{PSP}} = 0.78$ for the shapes used for $\mathcal{M}$).

The weights w^{IE} can be determined by comparing eq. (2) with eq. (3) with corrected inhibition strength

$$\sum_{j\in\mathcal{I}_m} w^{\text{IE}} I_j(t) = \sum_{j=1}^{M} \beta' z_j(t). \tag{28}$$

Under the assumption that the number of spikes in the pool of inhibitory neurons is at any time (with a slight delay) approximately equal to the number of spikes in the pool of excitatory neurons, we obtain

$$p^{\text{IE}} w^{\text{IE}} = \beta', \tag{29}$$

where p^{IE} denotes the connection probability from inhibitory to excitatory network neurons. This yields

$$w^{\text{IE}} = \beta' \frac{1}{p^{\text{IE}}} = \frac{c_{\text{PSP}}}{\gamma\sigma^2 p^{\text{IE}}} = 1.86. \tag{30}$$

We now first consider the weights w^{EI} from excitatory to inhibitory neurons under the assumption of no I-to-I connections. In this case, in order to obtain the same number of spikes in the inhibitory neurons as in the excitatory neurons, each spike from an excitatory neuron should induce on average one spike within all inhibitory neurons, that is,

$$w^{\text{EI,no II}} \bar{\epsilon} M_{\text{inh}} p^{\text{EI}} = 1, \tag{31}$$

where $\bar{\epsilon} = 0.01/c_{\text{PSP}}$ is the integral over the alpha-PSPs, p^{EI} is the connection probability from excitatory to inhibitory neurons, and $M_{\text{inh}} = 100$ is the number of inhibitory neurons. We obtain

$$w^{\text{EI,no II}} = c_{\text{PSP}}/p^{\text{EI}} = 1.357. \tag{32}$$

Without I-to-I connections, this guarantees that excitation is balanced by inhibition. However, the single spike (on average) will occur on average with a delay of 5 ms. Interestingly, the I-to-I connections can help to decrease this delay. In particular, if one demands that the inhibitory spike is elicited with a delay of less than one ms on average, then one can

simply increase the weights w^{EI} by some factor $c^{\mathrm{EI}} = 10$, leading to

$$w^{\mathrm{EI}} = w^{\mathrm{EI,no\ II}} c^{\mathrm{EI}} = c_{\mathrm{PSP}} c^{\mathrm{EI}} / p^{\mathrm{EI}} = 13.57. \tag{33}$$

Now, each spike in the excitatory population induces in the inhibitory population for approximately 10 ms a total rate of 1000 Hz, leading to an average delay of 1 ms. Without I-to-I connections this would however lead to too many successive spikes within these 10 ms. The I-to-I connections can compensate this too large excitation. For an approximately correct compensation, the first inhibitory spike has to balance out this excitation, which is approximately achieved by providing exactly the same amount of inhibition to inhibitory neurons, leading to

$$w^{\mathrm{II}} = c_{\mathrm{PSP}} c^{\mathrm{EI}} / p^{\mathrm{II}}. \tag{34}$$

Since $p^{\mathrm{II}} \approx p^{\mathrm{EI}}$, we used $w^{\mathrm{II}} = w^{\mathrm{EI}}$ for simplicity. These are the parameter values used for the $\mathcal{M}$ model in [Jonke et al., 2017].

4.3 Details to simulations for Figure 5

Creation of basic patterns: Basic patterns were 64-dimensional vectors, each representing a horizontal or vertical bar on an 8×8 two-dimensional pixel array. We defined 16 basic patterns $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(16)}$ in total, corresponding to all possible horizontal and vertical bars of width 1 in this pixel array. For a horizontal (vertical) bar, all pixels of a row (column) in the array attained the value 1 while all other pixels were set to 0. The entries of the basic pattern vectors were then defined by the values of the corresponding pixels in the array.

Superposition of basic patterns: To generate an input pattern, basic rate patterns were superimposed as follows. The number of superimposed basic patters n_{sup} was chosen between 1 and 3 drawn from the distribution $p(n_{sup} = k) = \frac{0.9^k 0.1^{3-k}}{\sum_{l=1}^{3} 0.9^l 0.1^{3-l}}$. Then each basic pattern to be superimposed was drawn uniformly from the set of basic patterns without replacement. This corresponds to the distribution used in [Jonke et al., 2017]. The input vector $\boldsymbol{y}$ was then given by $\boldsymbol{y} = \max\{1, \sum_{i=1}^{n_{sup}} \boldsymbol{x}^{(bp(i))}\}$, where $bp(i)$ denotes the i^{th} basic pattern to be superimposed and the max operation is performed element-wise.

Optimization of the generative model: The generative model (6)–(8) with 20 hidden causes $\boldsymbol{z}$ was fitted to this data in an iterative manner. One iteration of the fitting algorithm was performed as follows:

1. draw an input vector $\boldsymbol{y}$ as described above;
2. draw a sample from the approximate posterior $p_{\mathrm{A1}}(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W})$, eq. (12);
3. update the parameters $\boldsymbol{W}$ of the model according to eq. (20).

Since the posterior in step (2) is intractable, it was approximated by assuming that a maximum of 4 hidden causes are active in the posterior distribution (state vectors with more active hidden causes usually had negligible probabilities). This allowed us to compute the partition function and therefore to sample hidden state vectors in a straight-forward manner. Further, we did not consider hidden state vectors with no active hidden state since those would not lead to any parameter changes.

Parameters of the model and learning rule: A prior distribution $p(\boldsymbol{z})$ with parameters $\mu = 6$ and $\sigma^2 = 0.35$ was used. The scaling parameter γ was set to 1. Weights w_{ij}

were initialized with values drawn from a uniform distribution in $[0, 0.1]$. Weights were clipped between a minimal value of 0 and a maximal value of 6. A constant learning rate of $\eta = 0.1$ was used. Training was performed for 15000 updates. Network characteristics (such as KL divergences) were computed every 50^{th} update.

Figure 5B: Every 50^{th} update, we computed the KL-divergence between the true posterior and the approximate posterior $D_{\text{KL}}(p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W})||p_{\text{A1}}(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W}))$. In addition we also computed the KL-divergence between $p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W})$ and a uniform distribution over state vectors. In all these divergences, we only considered the distribution over vectors with at most 4 active hidden causes for tractability (see above). For Fig. 5B, the data was smoothed using a box-car filter of size 10.

Figure 5C: We considered the input patters given in panel A and computed the hidden state $\boldsymbol{z}_{\text{max}}$ with the maximum posterior probability (computed as described above) after learning in the exact and approximate posterior. This hidden state vector was then used to reconstruct the input pattern by computing $\hat{y} = \sigma_{\text{LS}}(W^T \boldsymbol{z}_{\text{max}})$. Note that this is not a sample of $\boldsymbol{y}$ but it defines the probability of each individual pixel to be 1.

Figure 5E: Every 50^{th} update of the simulation described above, we computed the KL-divergence $D_{\text{KL}}(p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W})||p_{\text{A2}}(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{W}))$ between the true posterior and the posterior according to approximation $A2$. For the red curve we used p_{A2} with $w_{\text{norm}} = 0$ and the same $\mu = 6$ as given for the original model. For the yellow curve, we corrected the prior of the model to have $\mu = -12$. The KL-divergence to the uniform distribution was computed as described above.

Acknowledgements: Written under partial support by the Human Brain Project of the European Union #604102 and #720270, and the Austrian Science Fund (FWF): I 3251-N33.

References

[Avermann et al., 2012] Avermann, M., Tomm, C., Mateo, C., Gerstner, W., and Petersen, C. (2012). Microcircuits of excitatory and inhibitory neurons in layer 2/3 of mouse barrel cortex. *Journal of neurophysiology*, 107(11):3116–3134.

[Berkes et al., 2011] Berkes, P., Orban, G., Lengyel, M., and Fiser, J. (2011). Spontaneous cortical activity reveals hallmarks of an optimal internal model of the environment. *Science*, 331:83–87.

[Bill et al., 2015] Bill, J., Buesing, L., Habenschuss, S., Nessler, B., Maass, W., and Legenstein, R. (2015). Distributed bayesian computation and self-organized learning in sheets of spiking neurons with local lateral inhibition. *PloS one*, 10(8):e0134356.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer, New York.

[Buesing et al., 2011] Buesing, L., Bill, J., B., and Maass, W. (2011). Neural dynamics as sampling: A model for stochastic computation in recurrent networks of spiking neurons. *PLoS Computational Biology*, 7(11):e1002211.

[Carandini and Heeger, 2012] Carandini, M. and Heeger, D. J. (2012). Normalization as a canonical neural computation. *Nature Reviews Neuroscience*, 13(1):51–62.

[Douglas and Martin, 2004] Douglas, R. J. and Martin, K. A. (2004). Neuronal circuits of the neocortex. *Annual Reviews of Neuroscience*, 27:419–451.

[Fino et al., 2012] Fino, E., Packer, A. M., and Yuste, R. (2012). The logic of inhibitory connectivity in the neocortex. *The Neuroscientist*, 19(3):228–237.

[Földiak, 1990] Földiak, P. (1990). Forming sparse representations by local anti-hebbian learning. *Biological cybernetics*, 64(2):165–170.

[Habenschuss et al., 2013a] Habenschuss, S., Jonke, Z., and Maass, W. (2013a). Stochastic computations in cortical microcircuit models. *PLoS Computational Biology*, 9(11):e1003311.

[Habenschuss et al., 2013b] Habenschuss, S., Puhr, H., and Maass, W. (2013b). Emergence of optimal decoding of population codes through STDP. *Neural computation*, 25(6):1371–1407.

[Hyvärinen, 1999] Hyvärinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *Neural Networks, IEEE Transactions on*, 10(3):626–634.

[Hyvärinen et al., 2004] Hyvärinen, A., Karhunen, J., and Oja, E. (2004). *Independent Component Analysis*. John Wiley & Sons.

[Isaacson and Scanziani, 2011] Isaacson, J. S. and Scanziani, M. (2011). How inhibition shapes cortical activity. *Neuron*, 72(2):231–243.

[Jolivet et al., 2006] Jolivet, R., Rauch, A., Lüscher, H., and Gerstner, W. (2006). Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *Journal of Computational Neuroscience*, 21:35–49.

[Jonke et al., 2017] Jonke, Z., Legenstein, R., Habenschuss, S., and Maass, W. (2017). Feedback inhibition shapes emergent computational properties of cortical microcircuit motifs. *arXiv preprint arXiv:1705.07614*.

[Kappel et al., 2014] Kappel, D., Nessler, B., and Maass, W. (2014). STDP installs in winner-take-all circuits an online approximation to hidden markov model learning. *PLoS Comput. Biol*, 10:e1003511.

[Kerlin et al., 2010] Kerlin, A. M., Andermann, M. L., Berezovskii, V. K., and Reid, R. C. (2010). Broadly tuned response properties of diverse inhibitory neuron subtypes in mouse visual cortex. *Neuron*, 67(5):858–871.

[Klampfl and Maass, 2013] Klampfl, S. and Maass, W. (2013). Emergence of dynamic memory traces in cortical microcircuit models through stdp. *The Journal of Neuroscience*, 33(28):11515–29.

[Lücke and Eggert, 2010] Lücke, J. and Eggert, J. (2010). Expectation truncation and the benefits of preselection in training generative models. *The Journal of Machine Learning Research*, 9999:2855–2900.

[Lücke and Sahani, 2008] Lücke, J. and Sahani, M. (2008). Maximal causes for non-linear component extraction. *The Journal of Machine Learning Research*, 9:1227–1267.

[Maass, 2000] Maass, W. (2000). On the computational power of winner-take-all. *Neural Computation*, 12(11):2519–2535.

[Neal, 1992] Neal, R. M. (1992). Connectionist learning of belief networks. *Artificial intelligence*, 56(1):71–113.

[Neal, 1993] Neal, R. M. (1993). Probabilistic inference using markov chain monte carlo methods. Technical report, University of Toronto Department of Computer Science.

[Nessler et al., 2013] Nessler, B., Pfeiffer, M., Buesing, L., and Maass, W. (2013). Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity. *PLoS Computational Biology*, 9(4):e1003037.

[Ohki et al., 2006] Ohki, K., Chung, S., Kara, P., Hübener, M., Bonhoeffer, T., and Reid, R. C. (2006). Highly ordered arrangement of single neurons in orientation pinwheels. *Nature*, 442(7105):925–928.

[Okun and Lampl, 2008] Okun, M. and Lampl, I. (2008). Instantaneous correlation of excitation and inhibition during ongoing and sensory-evoked activities. *Nature Neuroscience*, 11(5):535–537.

[Packer and Yuste, 2011] Packer, A. M. and Yuste, R. (2011). Dense, unspecific connectivity of neocortical parvalbumin-positive interneurons: a canonical microcircuit for inhibition? *The Journal of Neuroscience*, 31(37):13260–13271.

[Rumelhart and Zipser, 1985] Rumelhart, D. E. and Zipser, D. (1985). Feature Discovery by Competitive Learning. *Cognitive Science*, 9(1):75–112.

[Sato, 1999] Sato, M. (1999). Fast learning of on-line EM algorithm. Technical report, ATR Human Information Processing Research Laboratories, Kyoto, Japan.

[Saund, 1995] Saund, E. (1995). A multiple cause mixture model for unsupervised learning. *Neural Computation*, 7(1):51–71.

[Savin et al., 2010] Savin, C., Joshi, P., and Triesch, J. (2010). Independent component analysis in spiking neurons. *PLoS computational biology*, 6(4):e1000757.

[Turrigiano and Nelson, 2004] Turrigiano, G. G. and Nelson, S. B. (2004). Homeostatic plasticity in the developing nervous system. *Nature Reviews Neuroscience*, 5(2):97–107.

[Wilson et al., 2012] Wilson, N. R., Runyan, C. A., Wang, F. L., and Sur, M. (2012). Division and subtraction by distinct cortical inhibitory networks in vivo. *Nature*, 488(7411):343–348.