Towards Local Learning and MCMC Inference in Biologically Plausible Deep Generative Networks

Jost Tobias Springenberg[1] **Katharina Anna Wilmes**[2,3] **Joschka Boedecker**[1]
[1] Department of Computer Science, University of Freiburg
[2] Institute for Theoretical Biology, Humbold Universitaet zu Berlin
[3] Bernstein Center for Computational Neuroscience

Abstract

We propose an ansatz for approximate Markov Chain Monte Carlo (MCMC) inference and learning in multi-layered (deep) generative models that exhibit biologically plausible dynamics. In particular, our approach builds on a recently observed connection between the classical integrate-and-fire models of biological neurons and Bayesian inference based on Langevin dynamics. Utilizing this connection, we propose that a recurrent neural network model can be derived that learns to perform inference for a feed-forward generative model from local learning signals only. When combined with such a generative model – which can also be trained from information locally available at each neuron – our model provides new perspectives on how biologically plausible learning algorithms could be reconciled with statistical learning theory. We experimentally validate our model using synthetic images as well as the MNIST dataset and show that it approaches the level of performance of standard machine learning algorithms for training generative models on these simple tasks.

1 Introduction

How does the brain learn a model of the world; how does it perceive and reason? One very old hypothesis to this long-standing question is that the brain obtains a model of the world by performing statistical inference on given sensory information [Helmholtz, 1867, Hinton and Sejnowski, 1983, Dayan et al., 1995]. Many behavioral neuroscience studies support the idea that human performance under uncertainty is close to Bayes' optimal (see e.g. Knill and Pouget [2004], Körding and Wolpert [2004]). To date, however, the question remains how inference can be implemented in biological neural circuits. For inference, biological circuits need a representation of the relation between sensory inputs and unobserved underlying causes in the world.

This problem of inference and learning descriptive representations about the world can be framed as that of learning a generative model (generating observed stimuli from unobserved causes) with an accompanying inference procedure (inferring unobserved causes from observed stimuli). Within the field of machine learning, training such generative models has a long tradition and, more recently, neural network based generative models have been proven successful in learning complex tasks, e.g. generating images [Kingma and Welling, 2014, Rezende et al., 2014]. Machine learning algorithms however often lack biological plausibility as they require the backpropagation of an exact error term to all neurons in all layers. In the field of computational neuroscience, biologically plausible learning rules have been derived for single-layer neural networks [Brea et al., 2013, Rezende and Gerstner, 2014]. However, the biologically plausible ideas do not yet generalize to networks with many layers.

The goal of combining these seemingly separate perspectives has recently gained increased attention from both fields [Bengio and Fischer, 2016, Scellier and Bengio, 2016, Bengio et al., 2016, Marblestone et al., 2016]. Perhaps most notably – following an idea first outlined in Hinton [2007] – a series

of papers were recently dedicated to the aim of implementing the classical backpropagation training procedure for multi-layer networks in a biologically plausible way [Bengio and Fischer, 2016, Scellier and Bengio, 2016]. Among these Bengio and Fischer [2016] first noticed an interesting connection between the classical integrate-and-fire models of biological neurons and Bayesian inference based on Langevin dynamics Duane et al. [1987], Neal [2010].

Building on this idea we propose an ansatz for training a biologically plausible recurrent network that *learns to perform approximate Bayesian inference* for a multi-layered generative model using only layer-local learning signals, which are available in biological circuits. When paired with such a generative model – that under some assumptions is also implementable in a biologically plausible manner – our approach provides a consistent perspective on how biological circuits could perform statistical learning.

2 Background

We consider neural networks of M observed (or visible) neurons characterized through binary spikes $\mathbf{v}(t) = \mathbf{x}_0(t) = [X_{0,0}(t), \dots, X_{0,M}(t)]^T$, and a series of L layers of latent (or hidden) neurons – with variable sizes $\{|1|, \dots, |L|\}$. We refer to to the spikes of layer $l \in \{1, \dots, L\}$ with $\mathbf{x}_l(t) = [X_{l,0}(t), \dots, X_{l,|l|}(t)]^T$. The connections between neurons are assumed to form a directed acyclic graph, except for a self-connection (which exists for each neuron). For brevity of notation, and without loss of generality, we will restrict ourselves to fully connected networks that instantiate a generative model of the visible spikes. In such networks a unit i in layer l is connected to all neurons in layer $l+1$ (and thus receives input $\mathbf{x}_{l+1}(t)$). The neurons in the first layer will then be trained to generate the activity of the fixed (or "clamped") visible neurons.

2.1 Integrate-and-fire neural dynamics

We model the spiking behavior of individual neurons as a generalized linear model. Specifically, following a leaky integrate-and-fire neuron model (a special case of the *spike-response model* [Gerstner and Kistler, 2002a,b, Gerstner, 2008]), the change in membrane potential $u_{l,i}(t)$ of a neuron i in layer l at time t is given as

$$\tau \dot{u}_{l,i}(t) = -u_{l,i}(t) + W^l_{i,i} X_{l,i}(t) + b^l_i + \sum_{j=1}^{|l+1|} W^l_{i,j} X_{l+1,j}(t), \tag{1}$$

where τ is the membrane time constant, b^l_i denotes a bias term, which corresponds to constant external input. $X_{l,i}(t) = \sum_{f=1}^{N} \delta(t - t^f_i)$ is the spike train of neuron i with spike times t^f_i. $W^l_{i,j}$ are connection weights from neuron j to neuron i. Instead of modeling a time-resolved membrane dynamics during an action potential, we realize the reset of the membrane potential after a spike by a negative self-connection with weight $W^l_{i,i}$. Hence, the self-connection $W^l_{i,i}$ is not learned but set to a pre-determined value in contrast to bias and inter-layer connections.

2.2 Weak coupling approximation to neural dynamics

To simplify the derivation, and make the analogy to existing machine learning models clearer, we consider a weak coupling approximation [Toyoizumi et al., 2009, Rezende and Gerstner, 2014] to Equation (1) in which we replace the binary spike trains $X_{l,i}(t)$ with spiking probabilities $\rho_{l,i}(u_{l,i}(t))$ (or, alternatively, spiking rates). Let $\rho_{l,i}(t) = \rho_{l,i}(u_{l,i}(t)) = g(u_{l,i}(t))$, where g is a transfer function. For our purposes we consider either a sigmoidal transfer function $g(u) = 1/(1+e^{-u})$ or a rectified linear transfer function $g(u) = \max(0, u)$, but other common choices such as an exponential transfer function [Jolivet et al., 2006] could be accommodated. For the weak coupling approximation we then have

$$\begin{aligned} \tau \dot{u}_{l,i}(t) &= -u_{l,i}(t) + \xi_i(t) + W^l_{i,i} \rho_{l,i}(t) + b^l_i + \sum_{j=1}^{|l+1|} W^l_{i,j} \rho_{l+1,j}(t), \\ &\text{with } \xi_i(t) = \mathcal{N}\left(0, \sigma_i^2\right), \end{aligned} \tag{2}$$

where $\xi_i(t)$ captures the noise of the spiking process. The variance σ_i^2 of the noise process is given as $\sigma_i^2 = \sigma_0^2 + \frac{1}{\tau^2}(W_{i,i}^l)^2\rho_{l,i}(t) + \frac{1}{\tau^2}\sum_{j=1}^{|l+1|}(W_{i,j}^l)^2\rho_{l+1,j}(t)$ with σ_0^2 being the intrinsic system noise (which can be chosen arbitrarily).

Finally, to bring Equation (2) into a form that is convenient to simulate and reason about we first define the notion of a main driving force, i.e. the main contribution to a change in the membrane potential as $F_{l,i}(t) = \overleftarrow{F}_{l,i}(t) + \overline{F}_{l,i}(t)$. Where $\overleftarrow{F}_{l,i}(t) = W_{i,i}^l\rho_{l,i}(t)$ denotes the spike after-potential achieved by negative self-recurrence and $\overline{F}_{l,i}(t) = b_i^l + \sum_{j=1}^{|l+1|} W_{i,j}^l\rho_{l+1,j}(t)$ denotes the feed-forward input dynamics. We then simulate the differential equation in discretized steps yielding the update equation

$$u_{l,i}^{t+1} = u_{l,i}^t + \epsilon\left(\frac{F_{l,i}^t - u_{l,i}^t}{\tau} + \xi_i^t\right), \text{ with } \xi_i^t = \mathcal{N}\left(0, \sigma_i^2/\tau^2\right), \tag{3}$$

where we mark the dependency on time in the superscript for notational convenience (i.e $u_{l,i}^t$ is the value of the membrane potential at discrete time point t), and where ϵ is the integration step-size. We refer to the collection of all hidden layer membrane potentials with $\mathbf{u}^t = \{\mathbf{u}_l^t, \ldots, \mathbf{u}_L^t\}$, where $\mathbf{u}_l^t = [u_{l,0}^t, \ldots, u_{l,|l|}^t]^T$. Analogously, we refer to the collection of spiking probabilities with $\boldsymbol{\rho}^t = \{\boldsymbol{\rho}_l^t, \ldots, \boldsymbol{\rho}_L^t\}$, where $\boldsymbol{\rho}_l^t = [\rho_{l,0}^t, \ldots, \rho_{l,|l|}^t]^T$, and to the driving forces of neurons in layer l with $\mathbf{F}_l^t = \overleftarrow{\mathbf{F}}_l^t + \overline{\mathbf{F}}_l^t$, where $\overleftarrow{\mathbf{F}}_l^t = [\overleftarrow{F}_{l,0}^t, \ldots, \overleftarrow{F}_{l,|l|}^t]^T$ and $\overline{\mathbf{F}}_l^t = [\overline{F}_{l,0}^t, \ldots, \overline{F}_{l,|l|}^t]^T$.

3 Generative neural network model

With these definitions in place we can define the likelihood of a given hidden layer's membrane potential and spiking probability (assuming, for the moment, a sigmoidal activation function) as

$$p(\mathbf{u}_l^{t+1}|\mathbf{u}_{l+1}^t, \mathbf{u}_l^t) = \mathcal{N}\left(\mathbf{u}_l^t + \epsilon\frac{\mathbf{F}_l^t - \mathbf{u}_l^t}{\tau}, \epsilon^2\sigma_l^2/\tau^2\mathbf{I}\right), \tag{4}$$

$$p(\boldsymbol{\rho}_l^{t+1}|\mathbf{u}_l^{t+1}) = \frac{1}{1 + e^{\mathbf{u}_l^{t+1}}}, \tag{5}$$

and for the visible neurons we have $p(\mathbf{v}^{t+1}|\mathbf{u}_1^{t+1}) = p(\boldsymbol{\rho}_0^{t+1}|\mathbf{u}_1^{t+1})$.

Two interesting observations can be made about Equation (4). First, if the weight of the self-connection is set to a large negative value such that ${}^{(W_{i,i}^l\boldsymbol{\rho}_l^{t_1}\epsilon)}/_{\tau} \approx -(\mathbf{u}_l^t - \epsilon\mathbf{u}_l^t)$ (or if the weighted sum of the inputs becomes large), then the dynamics of the activation is dominated by the instantaneous feed-forward input and thus can be described as purely feed-forward. That is, we would have

$$p(\mathbf{u}_l^{t+1}|\mathbf{u}_{l+1}^t, \mathbf{u}_l^t) \approx p(\mathbf{u}_l^{t+1}|\mathbf{u}_{l+1}^t) = \mathcal{N}\left(\frac{\epsilon}{\tau}\overline{\mathbf{F}}_l^t, \epsilon^2\sigma_l^2/\tau^2\right). \tag{6}$$

From a machine learning perspective, Equation (6) can be interpreted as the output of a layer in a fully connected artificial neural network (with additive Gaussian noise). Second, if the weight of the self-connection is small, constant, and negative – and thus does not completely cancel past spiking activity in each step – then Equation (4) can be seen as following a Langevin dynamics [Duane et al., 1987, Neal, 2010] whose change in position is proportional to ${}^{(\mathbf{F}_l^t - \mathbf{u}_l^t)}/_{\tau}$, i.e. the dynamics is driven by the output of a feed-forward neural network. This relationship was (to the best of our knowledge) first highlighted in Bengio and Fischer [2016] as a potential tool for biologically plausible inference in neural networks (albeit using energy-based models). We will make use of both observations in order to derive a local learning procedure – with accompanying inference mechanism – in the following section.

4 Local learning and approximate inference

Using the definitions from Section 3 we are ready to formulate a learning algorithm for the introduced network model. To give our learning algorithm a clear interpretation in terms of statistical learning principles we posit that the goal of the generative network is to generate a given, *fixed*, input $\mathbf{v}^T$ at time-point T from an inferred configuration of the hidden variables $\mathbf{u}^T$.[1] We will adopt the

[1] I.e. we note that we do not consider generating temporally changing signals in this manuscript.

assumption that for the generative network the membrane potential is dominated by the weighted sum of the inputs (and thus the influence of past activity is negligible), as described in Equation (6). We note that this assumption is not crucial and could potentially be lifted at the expense of a more tedious derivation.

Formally, we would like the generative network to maximize the marginal log likelihood of the observed visible pattern $\log p(\mathbf{v}^T)$. To accomplish this we first define the joint likelihood of visible pattern and latent membrane potentials as

$$p(\mathbf{v}^T, \mathbf{u}^T) = p(\mathbf{v}^T|\mathbf{u}_1^T)p(\mathbf{u}_L^T) \prod_{l=1}^{L-1} p(\mathbf{u}_l^T|\mathbf{u}_{l+1}^T), \tag{7}$$

where we chose to write $p(\mathbf{u}_l^T|\mathbf{u}_{l+1}^T)$ instead of $p(\boldsymbol{\rho}_l^T|\mathbf{u}_{l+1}^T)$; due to their deterministic relationship, $\boldsymbol{\rho}_l^T$ and $\mathbf{u}_l^T$ are coupled. The marginal log likelihood can then be written as

$$\log p(\mathbf{v}^T) = \log \mathbb{E}_{p(\mathbf{u}^T|\mathbf{v}^T)} \left[p(\mathbf{v}^T, \mathbf{u}^T) \right], \tag{8}$$

where we assume that $p(\mathbf{u}^T|\mathbf{v}^T)$ factorizes as $p(\mathbf{u}^T|\mathbf{v}^T) = p(\mathbf{u}_1^T|\mathbf{v}^T)p(\mathbf{u}_2^T|\mathbf{u}_1^T)\cdots p(\mathbf{u}_L^T|\mathbf{u}_{L-1}^T)$. Unfortunately, Equation (8) requires the computation of an intractable expectation (as we have no knowledge about the true posterior $p(\mathbf{u}^T|\mathbf{v}^T)$). Aside from this, the optimization of the marginal likelihood with respect to the generative network only requires layer-local gradient computations; it could thus be implemented in a biological circuit without the need for exact error propagation across neurons. Our major concern therefore is to circumvent the evaluation of the intractable integral by introducing an approximate inference mechanism. The task for this mechanism is to infer latent membrane potentials given observed network activity. While one can choose from many of such approximate inference techniques, we additionally require that the inference procedure is realizable by a biologically plausible mechanism. Concretely, we will consider to implement approximate inference with a second *inference network* that follows the biologically plausible recurrent dynamics described in Section 3 – and whose weights can be learned purely from observed local network activity.

4.1 Langevin MCMC solution

We start by noting that samples from $p(\mathbf{u}^T|\mathbf{v}^T)$ (as required in Equation (8)) can be obtained by Hamiltonian Markov Chain Monte Carlo (MCMC) sampling – and thus in particular through simulating Langevin Dynamics. Given a direct connection between Langevin dynamics and the neuronal dynamics in our network, it is appealing to reason about a biologically plausible inference procedure by first looking at an MCMC solution to the problem.

In particular, we can interpret the application of Langevin dynamics to Equation (8) as a form of approximate inference in which we replace sampling from the posterior $p(\mathbf{u}^T|\mathbf{v}^T)$ with samples $\mathbf{u}^T$ obtained from running Langevin dynamics for a sufficient amount of time. The Langevin dynamics follows the updates

$$\mathbf{u}_l^{t+1} = \mathbf{u}_l^t + \epsilon \frac{\partial \log p(\mathbf{v}^T, \mathbf{u}^t)}{\partial \mathbf{u}_l^t} + \xi_t, \text{ with } \xi_t \sim \mathcal{N}(0, 2\epsilon\mathbf{I}), \tag{9}$$

where we have overloaded notation to highlight the similarities to Equation (3). Each simulation step results in a sample from the proposal distribution $\hat{p}(\mathbf{u}_l^{t+1}|\mathbf{u}_l^t, \mathbf{v}^T) = \mathcal{N}\left(\mathbf{u}_l^t + \epsilon \frac{\partial \log p(\mathbf{v}^T, \mathbf{u}^t)}{\partial \mathbf{u}_l^t}, 2\epsilon\mathbf{I}\right)$. This sample can be used as the basis for the next proposal, [2] resulting in a chain of samples that converges to samples from the true posterior $p(\mathbf{u}^T \mid \mathbf{v}^T)$.

Aside from this description of the sampling procedure as a dynamical process, we can also use the definition of Langevin MCMC to rewrite the marginal from Equation (8) as

$$\begin{aligned} &\log p(\mathbf{v}^T) \approx \log \mathbb{E}_{p(\mathbf{u}^T, \mathbf{u}^0, \ldots, \mathbf{u}^{T-1}|\mathbf{v}^T)} \left[p(\mathbf{v}^T, \mathbf{u}^T) \right], \\ &\text{with} \qquad p(\mathbf{u}^T, \mathbf{u}^0, \ldots, \mathbf{u}^{T-1}|\mathbf{v}^T) = p(\mathbf{u}^0) \prod_{t=0}^{T-1} \prod_{l=1}^{L} \hat{p}(\mathbf{u}_l^{t+1}|\mathbf{u}_l^t, \mathbf{v}^T), \end{aligned} \tag{10}$$

where we chose the prior $p(\mathbf{u}^0)$ to be isotropic Gaussian; $p(\mathbf{u}^0) = \mathcal{N}(\mathbf{0}, \mathbf{I})$.

[2] We here omit the Metropolis-Hastings step that is typically used in Monte Carlo techniques. This potentially biases our inference procedure, although MH acceptance rates for HMC procedures are typically close to 1.

4.2 Approximate inference through biologically plausible dynamics

While the Langevin dynamics concisely describes the estimation of the marginal – as calculated through MCMC – it has not (yet) brought us closer to a biologically plausible inference procedure. The key idea to perform this final step is as follows: inspired by the similarities between Equation (4) and Equation (9) we will introduce auxiliary inference neurons which are trained such that their dynamics (according to the spike response model) follows the Langevin dynamics from (9). Hence they can be used to replace the inference model $p(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T)$ in (10) with an approximate inference model $q(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T)$ that follows the biologically plausible dynamics. Inspecting the two equations we can already see that equality will be achieved if $\frac{\mathbf{F}_l^t - \mathbf{u}_l^t}{\tau} = \frac{\partial \log p(\mathbf{v}^T, \mathbf{u}^t)}{\partial \mathbf{u}_l^T}$ and, additionally, the noise variables are scaled appropriately. That is, we need $(\mathbf{F}_l^t - \mathbf{u}_l^t)/\tau$ to be a predictor of the generative model gradient.

This suggests the following setup for deriving $q(\cdot)$. Let

$$q(\mathbf{u}_l^{t+1}|\mathbf{u}_{l-1}^t, \mathbf{u}_l^t) = \mathcal{N}\left(\mathbf{u}_l^t + \epsilon \frac{\hat{\mathbf{F}}_l^t - \mathbf{u}_l^t}{\tau}, \epsilon^2 \sigma_l^2 / \tau^2 \mathbf{I}\right) \tag{11}$$

be the likelihood of the membrane potential for a given layers' inference neurons, which are connected to the neurons in layer $l-1$, and therefore $\hat{\mathbf{F}}_l^t = [\hat{F}_{l,0}^t, \dots, \hat{F}_{l,|l|}^t]^T$ with $F_{l,i}(t) = \hat{W}_{i,i}^l \rho_{l,i}(t) + \hat{b}_i + \sum_{j=1}^{|l+1|} \hat{W}_{i,j}^l \rho_{l-1,j}(t)$ (and where $\hat{W}_{i,j}^l$ and $\hat{b}_i^l$ are the parameters of the inference network). This likelihood follows the same form as Equation (4) (with its arguments reversed) and thus the same neuronal dynamics from Equation (3). The only difference being that the inference neurons in layer l are connected to neurons from layer $l-1$ instead of $l+1$, i.e. they form a bottom-up rather than a top-down hierarchy. Using these approximate inference neurons we can write the likelihood of a chain of samples (a forward simulation of the membrane potentials) as

$$q(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T) = p(\mathbf{u}^0) \prod_{t=0}^{T-1} \prod_{l=1}^{L} q(\mathbf{u}_l^{t+1}|\mathbf{u}_{l-1}^t, \mathbf{u}_l^t). \tag{12}$$

Using this definition we can measure the distance between the approximate inference distribution $q(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T)$ and the MCMC inference distribution $p(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T)$ through the KL divergence

$$\begin{aligned} \mathcal{L}_{\mathbf{u}} &= \mathbb{E}_{q(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T)} \left[\log \frac{q(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T)}{p(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T)}\right] \\ &= \sum_{l=1}^{L} \sum_{t=0}^{T-1} \mathrm{KL}\left(q(\mathbf{u}_l^{t+1}|\mathbf{u}_{l-1}^t, \mathbf{u}_l^t) \| \hat{p}(\mathbf{u}_l^t|\mathbf{u}_l^{t+1}, \mathbf{v}^T)\right), \end{aligned} \tag{13}$$

where, if we assume that τ and σ_l are chosen such that $\epsilon^2 \sigma_l^2 / \tau^2 = 2\epsilon$, the KL terms simplify to

$$\mathrm{KL}\left(q(\mathbf{u}_l^{t+1}|\mathbf{u}_{l-1}^t, \mathbf{u}_l^t) \| \hat{p}(\mathbf{u}_l^t|\mathbf{u}_l^{t+1}, \mathbf{v}^T)\right) \propto \left(\epsilon \frac{\partial \log p(\mathbf{v}^T, \mathbf{u}^t)}{\partial \mathbf{u}_l^t} - \epsilon \frac{\hat{\mathbf{F}}_l^t - \mathbf{u}_l^t}{\tau}\right)^2 \frac{1}{2\sigma_l^2}. \tag{14}$$

If this KL divergence is minimized, we can replace samples from $\hat{p}(\mathbf{u}_l^t|\mathbf{u}_l^{t+1}, \mathbf{v}^T)$ in Equation (10) with samples from $q(\mathbf{u}_l^{t+1}|\mathbf{u}_{l-1}^t, \mathbf{u}_l^t)$ yielding the following (biased) estimator to the marginal likelihood

$$\mathcal{L}_{\mathbf{v}} = \mathbb{E}_{q(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T)} \left[\log p(\mathbf{v}^T, \mathbf{u}^T)\right] \approx \mathbb{E}_{p(\mathbf{u}^T, \mathbf{u}^0, \dots, \mathbf{u}^{T-1}|\mathbf{v}^T)} \left[\log p(\mathbf{v}^T, \mathbf{u}^T)\right] \leq \log p(\mathbf{v}^T). \tag{15}$$

These derivations suggest a Monte Carlo Expectation Maximization like procedure for training both the generative and the inference network in which we alternate training of the generative network with updates to the inference neurons (which should learn to imitate the Langevin dynamics steps and thus, as described above, to predict the gradient of the generative model). More precisely, let $\Phi = \{\mathbf{W}^1, b^1, \dots, \mathbf{W}^L, \mathbf{b}^L\}$ be the parameters of the generative network and $\Psi = \{\hat{\mathbf{W}}^1, \hat{\mathbf{b}}^1 \dots, \hat{\mathbf{W}}^L, \hat{\mathbf{b}}^L\}$ be the parameters of the inference network. We follow a three step procedure:

(1) Inference. With Φ fixed we sample a trajectory of membrane potentials $\mathbf{u}^0, \dots \mathbf{u}^T \sim q(\mathbf{u}^T \mathbf{u}^1 \dots \mathbf{u}^{T-1} | \mathbf{v}^T)$ from the inference network by simulating the neural dynamics of inference and (fixed) generative network for T steps.

(2) Minimize $\mathcal{L}_u$. In each step we take a gradient descent step to minimize $\mathcal{L}_u$ wrt. Ψ.

(3) Maximize $\mathcal{L}_v$. We take a gradient ascent step to maximize $\mathcal{L}_v$ wrt. Φ using the sample $\mathbf{u}^T$.

Importantly, step (1) only requires simulating the neuronal dynamics and – as mentioned before – step (3) only requires the computation of local gradients. Step (2) however requires computation of Equation (14), which involves a calculation of the gradient of $p(\mathbf{v}^T, \mathbf{u}^t)$ with respect to $\mathbf{u}^t$ in each step. This involves (biologically implausible) backpropagation through the whole generative network. However, this gradient can be replaced by a likelihood ratio estimator (often also referred to as a REINFORCE gradient [Williams, 1992]) that only requires access to local quantities (including local reconstruction errors).

To derive this estimator we first recall that the KL divergence is exactly minimized if $(\hat{\mathbf{F}}_l^t - \mathbf{u}_l^t)/\tau$ corresponds to the gradient of the log likelihood (at u_l^t). Additionally the gradient of Equation (14) with respect to the parameters Φ of the inference network will be propotional to $\frac{\partial p(\mathbf{v}^T, \mathbf{u}^t)}{\partial \mathbf{u}_l^t}$. This suggests that we can learn q simply by ensuring that $(\hat{\mathbf{F}}_l^t - \mathbf{u}_l^t)/\tau$ "points" towards the minimum of the joint likelihood. We can achieve this through maximum likelihood optimization, using the gradient of $\hat{\mathcal{L}}_{\mathbf{u}} = \mathbb{E}_{q(\mathbf{u}_l^{t+1} | \mathbf{u}_{l-1}^t, \mathbf{u}_l^t)}[\log p(\mathbf{v}^T, \mathbf{u}^t)]$ for $\epsilon = 1$, instead of the gradient of $\mathcal{L}_u$. A likelihood ratio estimator can then be trivially obtained for this form as

$$\frac{\partial \hat{\mathcal{L}}_{\mathbf{u}}}{\partial \Phi} = l(\mathbf{v}^T, \mathbf{u}_l^t) \frac{\partial q(\mathbf{u}_l^{t+1} | \mathbf{u}_{l-1}^t, \mathbf{u}_l^t)}{\partial \Phi}, \tag{16}$$

where $l(\mathbf{v}^T, \mathbf{u}_l^t) = \log p(\mathbf{u}^L) \sum_{i=1}^{l-1} \log p(\mathbf{u}_i^t | \mathbf{u}_{i+1}^t)$ is the sum of the local likelihood from layers below l (i.e. a fraction of the total joint log likelihood $\log p(\mathbf{v}^T, \mathbf{u}^t)$). This gradient now only depends on the availability of one global signal per layer: a reconstruction error signal corresponding to the likelihood of reconstructions from subsequent layers. We note that, although this choice for the gradient of the inference network is somewhat ad-hoc, it results in updates that resemble the wake-phase updates employed in the wake-sleep algorithm [Hinton et al., 1995]. Unfortunately, using this changed objective can potentially also bias the inference network, weakening its connection to true Langevin MCMC inference. To simplify the task for the inference network we can additionally assume that the input to each layer of the inference neuron is given as the difference between the previous layer activations $u_l^t - u_l^{t-1}$. While our experiments suggest that a learned inference network based on Equation (16) performs according to our expectations for the considered benchmarks, future work should focus on deriving more direct, unbiased estimators of the gradient of Equation (14).

5 Relation to existing work

This work aims to present first steps towards bridging the gap between neuroscience and machine learning, re-using existing terminology and connecting currently emerging ideas. As such, it is related to and builds upon a large body of work from both the computational neuroscience and the machine learning community, that preceded this manuscript. Since a complete overview over these two vast fields would exceed the space constraints of this paper we aim to highlight and concisely explain the most obvious connections between them and our work. For a more general perspective on how ideas from machine learning connect to computational neuroscience we refer to Marblestone et al. [2016].

First, the idea of interpreting the temporal dynamics of neuronal circuits as performing Bayesian inference and learning has appeared several times already in the literature. It can be traced back to early work by Hinton and Sejnowski [1983] and also appears in more recent work on implementing inference using STDP rules (see e.g Rezende and Gerstner [2014], Brea et al. [2013]). An interesting perspective on this idea, which has recently reignited interest in biologically plausible models among researchers from the machine learning community, was given by Hinton [2007] who suggested that the classical backpropagation learning algorithm could be implemented by using temporal changes to represent error gradients. Among these new studies, Bengio and Fischer [2016] further developed the idea of having neuronal dynamics approximately perform Langevin MCMC, which they employed to train an energy-based model. This approach – and related subsequent work [Scellier and Bengio, 2016,

Bengio et al., 2016] – is undoubtedly closely related to our work (and served as the main inspiration). Our approaches differ in two main respects: (1) they consider energy-based models and their method requires undirected connections, i.e. symmetric weights, which are biologically implausible, (2) their overarching aim is to exactly recover backpropagation from the neural dynamics whereas our method uses a reinforcement-like (neuromodulatory) signal. Whether the second difference is an advantage or a hindrance of our method is an open question subject to future work. A further difference is that the authors derive an explicit STDP-like learning rule from their model formulation, a step that is missing in our approach (we use local gradient-based optimization to minimize the cost functions). Given these close connections we believe it worthwhile to unify the two perspectives into one framework in future research.

Second, our model can be likened to a variational autoencoder [Rezende et al., 2014, Kingma and Welling, 2014] in which the inference network follows recurrent dynamics and learns to perform MCMC inference (instead of minimizing the evidence lower bound). On a more general level, it is thus similar to recent work on combining MCMC with variational inference by Salimans et al. [2015], and could potentially be re-derived within their framework. A combination of variational inference and a likelihood ratio estimator similar to the one we use has previously been considered e.g. in Mnih and Gregor [2014]. It would be interesting to reconcile these different perspectives.

Third, our model bears some similarity to the weight sleep algorithm [Hinton et al., 1995] and the reweighted wake sleep extension [Bornschein and Bengio, 2015]. Both use an importance sampling estimator to train the inference network that is similar to the one used here, and an EM-like procedure in which inference and generative network are trained in alternating steps. Both however uses fixed form inference and generative models, whereas we use a biologically motivated dynamical system formulation.

Finally, the idea of using a likelihood ratio estimate to provide a learning signal based on the reconstruction error has been explored for biologically plausible single-layer models in Brea et al. [2013] and Rezende et al. [2014]. From these Brea et al. [2013] is closely related to our work and employs a variational inference formulation using a biologically plausible inference network but does not easily generalize to multiple layers. It served as the second main inspiration for this work.

6 Experiments

We perform experiments using the model described in this paper in two settings. As a first step, and to visualize the inference process, we apply our model to modeling simple input stimuli consisting of small images of horizontal bars. Then we perform experiments for modeling a binarized version of the MNIST digits [Murray and Salakhutdinov, 2009]. This second experiment validates that the derived model not only consists of a biological plausible network but also performs well on a classical benchmark for generative models in machine learning. We again emphasize that we only consider modeling static input stimuli in this paper (but employ a temporal dynamics to perform the inference process).

6.1 Model setup

We consider training both single-layer and two-layer variants of our model. For the experiment on the simple bar task we use a single layer inference model with 10 hidden neurons and a corresponding generative network mapping from the 25 hidden neurons to the 10×10 dimensional pixel images. For the MNIST task we used inference and generator networks with 100 neurons in each hidden layer. For both experiments we chose to use a sigmoidal transfer function (as described in the Methods) in all layers, which naturally leads to a Bernoulli distribution for the visible variables. The time constant was arbitrarily chosen as $\tau = 0.1$ for all experiments and we used a step-size of $\epsilon = 1 \times 10^{-3}$. The number of inference steps was set to $T = 50$ for the MNIST experiment and $T = 30$ for the simple bar images.

To train both the generative and the inference network we used standard stochastic gradient descent on mini-batches of data. That is, we always performed inference for 100 input stimuli $\mathbf{v}$ at a time, computed the average of the local gradients of $\mathcal{L}_v$ and the likelihood ratio gradient of $\mathcal{L}_u$ over them and update the respective networks. To update Φ a learning rate of 1×10^{-3} was used whereas a small learning rate of 1×10^{-5} was used to update the parameters Ψ of the inference network – we

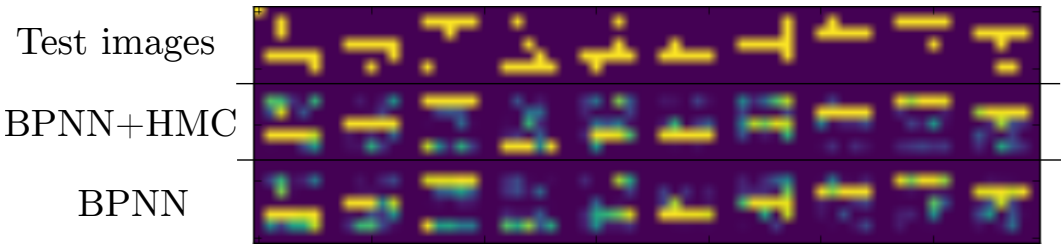

Figure 1: Visualization of 10 generated samples for the bars data using Langevin HMC inference as well as the approximate inference network. In the visualization each pixel is color coded according to its probability of being on (yellow = high).

Figure 2: Visualization of generated samples for a two layer network trained on the MNIST digits. In the visualization each pixel is color coded according to its probability of being on (white = high).

found it important to have a small learning rate in the inference network to avoid oscillations in the training – in conjunction with the Adam optimizer [Kingma and Ba, 2015] (all other parameters were set to their default values). We again note that using gradient descent to adapt the network parameters (albeit using only local quantities to estimate said gradients) is likely not exactly implementable in a biological circuit but could be approximated by STDP updates in a spiking dynamics. We refer to Section 7 for a more in-depth discussion.

6.2 Modeling simple input stimuli

For the synthetic data experiment we generated 1000 images. Each image contains one horizontal bar plus additional pixel noise; exemplary images are depicted in Figure 1 (top). We used these artificial stimuli to compare our proposed approximate inference network to the Langevin MCMC solution outlined in Section 4.1.

In particular, we tested three different generative model / inference model pairs for this task: 1) a generative model trained using Langevin dynamics as inference procedure during training and during testing; 2) a generative model trained with the approximate inference procedure and tested using the same approximate inference mechanism for generating samples; 3) the same generative model as for option 2) but at test time using Langevin dynamics for inference. As a first result, we observed that all models reached a similar log likelihood during training. We then visually compared samples for all pairs on a set of noisy bar images. The results are visualized in Figure 1, which shows that the generated samples for both inference methods combined with the generator network give qualitatively similar results, indicating that the approximate inference network can learn to predict the gradient of the joint log likelihood and thus performs well at imitating the transition operation used in the Langevin dynamics. A video of intermediate generated images when generating for $T = 50$ time steps using both Langevin dynamics and the learned inference network, which can be found at `http://goo.gl/6cZ1A2`, further supports this claim.

6.3 Modeling MNIST images

We next tested our model on the MNIST dataset, a slightly more complicated dataset containing 60000 training and 10000 test images depicting handwritten digits. We again trained a generative model in combination with Langevin MCMC inference and one using our approximate inference model. Additionally we considered both single layer and two layer networks for this task. Exemplary

Table 1: Comparison of our biologically plausible neural networks (BPNN) with state-of-the art generative models for MNIST. We show results for our model with a true Monte Carlo EM procedure (BPNN+ MCEM) – assuming biologically implausible access to $p(\mathbf{u}^T, \mathbf{u}^0, \dots \mathbf{u}^{T-1}|\mathbf{v}^T)$ during training – and the completely local learning procedure (using the likelihood ratio estimator to approximately minimize Equation (14)). As a baseline we also include our method combined with a random inference network replacing the gradient in the Langevin MCMC steps. Results are from [1] Murray and Salakhutdinov [2009], [2] Murray and Larochelle [2014], [3] Gregor et al. [2014], [4] Gregor et al. [2015], [5] Mnih and Gregor [2014], [6] Salimans et al. [2015], [7] Bornschein and Bengio [2015].

MNIST	$-\log p(\mathbf{v}) \leq$	$-\log p(\mathbf{v}) \approx$
DBN (2 layers) [1]	86.22	84.55
EoNADE [2]	-	85.10
DARN (1 layer) [3]	88.30	84.13
DRAW [4]	80.97	
NVIL (SBN) [5]	-	96.7
NVIL (fDARN) [5]	-	90.7
HVI + VAE [6]	88.30	85.51
RWS (SBN) [7]	-	85.48
BPNN+ random (2 layer)	-	181.45
BPNN+ MCEM (1 layer)	-	103.64
BPNN+ MCEM (2 layers)	-	101.79
BPNN(1 layer)	-	105.72
BPNN(2 layers)	-	102.50

generated images by two layer networks trained with either inference model are visualized in Figure 2. The quality of the samples again indicates that our approximate inference model can be successfully learned for this task.

As a quantitative analysis we compared the quality of the generative model learned with our approach to state of the art results from the literature. For this purpose we performed a simple monte carlo estimate of the marginal likelihood $p(\mathbf{v})$ by drawing 50 samples from $p(\mathbf{u}^T)$ via a 100 step Langevin MCMC chain for each test-set sample which are then passed through the learned generator to estimate the marginal likelihood. We note that this procedure *only evaluates the quality of the generator* and does not involve the inference model. We obtain results that approach the state of the art from the literature as shown in Table 1.

To evaluate the quality of the inference network we additionally measure the KL divergence between the proposal distribution of the approximate inference network and the MCMC proposal distribution (as given in (14)) in each step of a 100 step MCMC chain for the 1000 test examples. At the end of training the average divergence on these examples was 0.284, indicating a good fit.

7 Discussion and Conclusion

In this paper, we presented an algorithm for training multi-layered neural networks that employ a biologically plausible dynamics. We derived the algorithm from a machine learning perspective that includes: (1) a hierarchical generative model aiming to model (static) incoming input stimuli, and (2) an approximate inference network that learns to adapt its neuronal dynamics to approximately perform MCMC sampling of the latent membrane potentials - the neuronal activities.

It is important to understand that our approach is merely a first step towards developing a biologically plausible MCMC inference algorithm for generative networks and relies on a number of crucial assumptions. First, we assume different integration periods for the generative and the inference network: a small integration period for the generative network, i.e. approximately feed-forward dynamics, and a large integration period for the inference network, i.e. slow and recurrent dynamics. Second, our three learning steps implicitly require a local mechanism (on the neuronal level) to switch between inference and learning in the generative model – although this is a strong assumption such a switching behavior is not implausible and could be realized by a local inhibitory circuit that

controls intracellular learning signals such as backpropagating action potentials [Paulsen and Moser, 1998, Häusser and Mel, 2003, Wilmes et al., 2016] in a biological circuit. Third, we require a scalar local feedback signal corresponding to the layer local log-likelihoods, e.g. a reconstruction matching error, to compute the likelihood ratio estimate – this could correspond to neuromodulatory signals in biological circuits [Frémaux and Gerstner, 2015]. Fourth, our implementation at present relies on gradient based optimization of the model parameters and does not implement an STDP learning rule (although the optimization is based on local information only). Aside from biological considerations we want to point out that among these assumptions the third is potentially the most limiting one since the reliance of our model on a likelihood ratio (REINFORCE) estimator could potentially be a severe obstacle for scaling up the presented model: It is well known that such estimates have a high variance which, in our case, grows linearly with the number of neurons.

The presented approach contributes to current research in biologically-plausible learning algorithms, as synapse-local information (pre- and postsynaptic activity, synaptic weight) and layer-local information (scalar neuromodulatory signal, local inhibitory circuits) are sufficient for the algorithm to work. In contrast to previous work in this area, our algorithm does not require symmetric connections (a biologically implausible assumption). Overall, our work thus contributes to the larger aim of bridging the gap between machine learning and biological learning algorithms, which might yield interesting results for both fields.

References

Hermann von Helmholtz. *Handbuch der Physiologischen Optik*. Leipzig: Voss, 1867.

Geoffrey E Hinton and Terrence J Sejnowski. Optimal perceptual inference. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 448–453, 1983.

Peter Dayan, Geoffrey E Hinton, Radford M Neal, and Richard S Zemel. The helmholtz machine. *Neural computation*, 7(5):889–904, 1995.

David C Knill and Alexandre Pouget. The bayesian brain: the role of uncertainty in neural coding and computation. *TRENDS in Neurosciences*, 27(12):712–719, 2004.

Konrad P Körding and Daniel M Wolpert. Bayesian integration in sensorimotor learning. *Nature*, 427(6971): 244–247, 2004.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *The International Conference on Learning Representations (ICLR)*, 2014. arXiv preprint arXiv:1312.6114.

Danilo J. Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In Tony Jebara and Eric P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1278–1286. JMLR Workshop and Conference Proceedings, 2014.

Johanni Brea, Walter Senn, and Jean-Pascal Pfister. Matching recall and storage in sequence learning with spiking neural networks. *The Journal of Neuroscience*, 33(23):9565–9575, 2013. doi: 10.1523/JNEUROSCI.4098-12.2013.

Danilo Jimenez Rezende and Wulfram Gerstner. Stochastic variational learning in recurrent spiking networks. *Frontiers in Computational Neuroscience*, 8(38.), 2014.

Yoshua Bengio and Asja Fischer. Early inference in energy-based models approximates back-propagation. In *arXiv:1510.02777*, 2016.

B. Scellier and Y. Bengio. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *ArXiv e-prints*, 2016.

Y. Bengio, B. Scellier, O. Bilaniuk, J. Sacramento, and W. Senn. Feedforward initialization for fast inference of deep generative networks is biologically plausible. *ArXiv e-prints*, 2016.

Adam Henry Marblestone, Greg Wayne, and Konrad P Körding. Towards an integration of deep learning and neuroscience. *bioRxiv*, 2016. doi: 10.1101/058545. URL `http://biorxiv.org/content/early/2016/06/13/058545`.

Geoffrey Hinton. How to do backpropagation in a brain. In *Invited talk at the NIPS'2007 Deep Learning Workshop*, 2007. URL `http://www.cs.utoronto.ca/{~}hinton/backpropincortex2014.pdf`.

Simon Duane, A. D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics Letters B*, 195(2):216 – 222, 1987. ISSN 0370-2693.

Radford M. Neal. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 54:113–162, 2010.

Wulfram Gerstner and Werner M. Kistler. Mathematical formulations of hebbian learning. *Biological Cybernetics*, 87(5-6):404–415, 2002a.

Wulfram Gerstner and Werner M. Kistler. Spiking neuron models. *Cambridge University Press*, 2002b.

Wulfram Gerstner. Spike-response model. *Scholarpedia*, 3((12):1343.), 2008.

Taro Toyoizumi, Kamiar Rahnama Rad, and Liam Paninski. Mean-field approximations for coupled populations of generalized linear model spiking neurons with markov refractoriness. *Neural Computation*, 21(5):1203–1243, 2009.

Renaud Jolivet, Alexander Rauch, Hans-Rudolf Lüscher, and Wulfram Gerstner. Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *Journal of Computational Neuroscience*, 21(1): 35–49, 2006.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. In *Machine Learning*, pages 229–256, 1992.

Geoffrey E. Hinton, Peter Dayan, Brendan J. Frey, and Radford M. Neal. The wake-sleep algorithm for unsupervised neural networks. *Science*, 268:1158–1161, 1995.

Tim Salimans, Diederik P. Kingma, and Max Welling. Markov chain monte carlo and variational inference: Bridging the gap. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 2015.

Andriy Mnih and Karol Gregor. Neural variational inference and learning in belief networks. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014.

Jörg Bornschein and Yoshua Bengio. Reweighted wake-sleep. In *The International Conference on Learning Representations (ICLR)*, 2015.

Iain Murray and Ruslan R Salakhutdinov. Evaluating probabilities under high-dimensional latent variable models. In *Advances in Neural Information Processing Systems 21 (NIPS)*, 2009.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

Iain Murray and Hugo Larochelle. A deep and tractable density estimator. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014.

Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. Deep autoregressive networks. In *Proceedings of the 31th International Conference on Machine Learning (ICML)*, 2014.

Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. DRAW: A recurrent neural network for image generation. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 2015.

Ole Paulsen and Edvard Moser. A model of hippocampal memory encoding and retrieval: GABAergic control of synaptic plasticity. *Trends in Neurosciences*, 21(7):273–278, 1998. doi: 10.1016/S0166-2236(97)01205-8.

Michael Häusser and Bartlett Mel. Dendrites: bug of feature? *Current Opinion in Neurobiology*, 13(3):372–383, 2003.

Katharina A. Wilmes, Henning Sprekeler, and Susanne Schreiber. Inhibition as a binary switch for excitatory plasticity in pyramidal neurons. *PLoS Computational Biology*, 12(3):1–27, 03 2016. doi: 10.1371/journal.pcbi.1004768. URL `http://dx.doi.org/10.1371%2Fjournal.pcbi.1004768`.

Nicolas Frémaux and Wulfram Gerstner. Neuromodulated spike-timing-dependent plasticity, and theory of three-factor learning rules. *Frontiers in Neural Circuits*, 9:85, 2015. doi: 10.3389/fncir.2015.00085.