Explaining the effects of non-convergent MCMC in the training of Energy-Based Models

Elisabeth Agoritsas [1] **Giovanni Catania** [2] **Aurélien Decelle** [2 3] **Beatriz Seoane** [2 3]

Abstract

In this paper, we quantify the impact of using non-convergent Markov chains to train Energy-Based models (EBMs). In particular, we show analytically that EBMs trained with non-persistent short runs to estimate the gradient can perfectly reproduce a set of empirical statistics of the data, not at the level of the equilibrium measure, but through a precise dynamical process. Our results provide a first-principles explanation for the observations of recent works proposing the strategy of using short runs starting from random initial conditions as an efficient way to generate high-quality samples in EBMs, and lay the groundwork for using EBMs as diffusion models. After explaining this effect in generic EBMs, we analyze two solvable models in which the effect of the non-convergent sampling in the trained parameters can be described in detail. Finally, we test these predictions numerically on a ConvNet EBM and a Boltzmann machine.

1. Introduction

Encoding the statistics of a given dataset and generating data points that match those statistics are crucial issues in machine learning. Many different strategies have been devised. Among the most popular solutions are Variational Autoencoders (VAEs) (Kingma & Dhariwal, 2018), Generative Adversarial Networks (GANs) (Goodfellow et al., 2020), Energy Based Models (EBMs) (Zhu et al., 1998; LeCun et al., 2006; Lu et al., 2016; Xie et al., 2016) or their close relatives, Diffusion Models (Sohl-Dickstein et al., 2015; Ho et al., 2020). The latter are becoming increasingly popular nowadays due to their amazing ability to generate photorealistic images (Dhariwal & Nichol, 2021).

EBMs offer several fundamental advantages over their competitors due to their simplicity: A single neural network is involved in training, which means that fewer parameters need to be learned and training is less costly. They are also particularly appealing for interpretive applications: Once trained, the energy function can be analyzed with statistical mechanics tools (Decelle & Furtlehner, 2021b), or shallow EBMs can serve as an effective model to "learn" something from the data. EBMs have been exploited for instance to infer the three dimensional structure (Morcos et al., 2011) or building blocks (Tubiana et al., 2019) of proteins, to generate artificial pieces of genome (Yelmen et al., 2021), for neuroimaging (Hjelm et al., 2014), simulation of complex wavefunctions in quantum many-body physics (Carleo & Troyer, 2017; Melko et al., 2019), or to impute missing input (Fissore et al., 2019). Furthermore, EBMs naturally implement compositionability, i.e. the possibility of combining information from different trained models to generate new composite data (Hinton, 1999; Du et al., 2020).

EBMs have been around for a very long time, starting with the well-known single-layer (Ackley et al., 1985), Restricted (Smolensky, 1987) or the Deep (Salakhutdinov & Hinton, 2009) Boltzmann machines, to modern applications where EBMs are encoded in Deep Convolutional Neural Networks (ConvNets) (Nijkamp et al., 2019; 2020; Du & Mordatch, 2019; Du et al., 2020), which show very similar performance in image generation as state-of-the-art GAN methods (Du & Mordatch, 2019). Despite their many advantages, the direct training of EBMs is hard and unstable in practice, a problem directly related to the approximations made to estimate the negative term of the gradient (Decelle et al., 2021). As it is intractable, it must be approximated by sampling methods, typically Markov Chain Monte Carlo (MCMC) or Langevin dynamics, and such sampling is required each time parameters are updated. However, practical applications use a number of iterations far shorter than the mixing time of the algorithm, required to ensure an ergodic exploration of the model phase space and a reliable estimation of the log-likelihood gradient. To circumvent this "equilibration" problem, several strategies have been proposed, mainly relying either on good initializations of the MCMC chains to

[*]Equal contribution [1]Department of Quantum Matter Physics, University of Geneva, 1211 Geneva, Switzerland [2]Departamento de Física Teórica, Universidad Complutense de Madrid, 28040 Madrid, Spain [3]Université Paris-Saclay, CNRS, INRIA Tau team, LISN, 91190 Gif-sur-Yvette, France. Correspondence to: Aurélien Decelle <adecelle@ucm.es>, Beatriz Seoane <beseoane@ucm.es>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

reduce the equilibration time, such as the (persistent) contrastive divergence method (Hinton, 2002; Tieleman, 2008) and its refinements (Du & Mordatch, 2019), or on perturbative methods to approximate the gradient (Gabrié et al., 2015; Kim & Bengio, 2016; Kumar et al., 2019).
A very successful alternative to avoid altogether worrying about thermalization of the sampling processes during training has been to give up encoding the empirical data statistics in the Boltzmann weight of a given energy model, as it is the normal EBMs' modus operandi. The idea is instead to adjust the parameters of that model to reproduce a dynamical path able to generate new data points (Sohl-Dickstein et al., 2015), using an idea inherited from out-of-equilibrium thermodynamics. That is, sacrificing the usual equilibrium-like interpretability of the EBMs for the sake of a fast and high quality generation performance. This idea is at the core of the so-called diffusion models (Ho et al., 2020).

Furthermore, recent experimental studies in several EBM families have shown that such a dynamic path is also printed in EBMs when non-convergent samplings are used to train the models, and that this can be easily exploited to generate high-quality samples in short times both in sampling and training (Nijkamp et al., 2019; 2020; Decelle et al., 2021; Muntoni et al., 2021; Liao et al., 2022; Xie et al., 2022). It was shown that this optimal regime for sample generation of EBMs is achieved by forcing out-of-equilibrium sampling of the model during training, specifically by always initializing the Markov chains with random conditions and iterating the MCMC process for only a few steps k. The key observation was that particularly high-quality samples are generated when sampling the trained model for exactly the same time and initial distribution that was used to train the model (Nijkamp et al., 2019; Decelle et al., 2021). Nicely, these observations can be rationalized by moment-matching arguments at the fixed point of the training dynamics, as described in Ref. (Nijkamp et al., 2019). Furthermore, the apparition of this strong non-monotonic performance in the sample generation can be directly related to the MCMC mixing time, which typically increases with learning progress (Decelle et al., 2021). In fact, the same work showed that most of the instabilities and difficulties typically encountered in training RBMs are directly related to this out-of-equilibrium regime, since all these effects disappear when the number of Gibbs steps used to compute the gradient exceeds the thermalization time, e.g. when an incredibly long number of steps per iteration is considered (Decelle et al., 2021).

In this paper, we provide a first-principle analytical explanation of these previous experimental observations. In our theoretical framework, we show that the Gibbs-Boltzmann distribution of models trained in the out-of-equilibrium regime poorly reproduce the target statistics of the dataset. We also show that non-convergent samples can be used to accurately reproduce certain statistics of the dataset, but only through a very specific dynamic path fixed by the training choice. We first prove these results for generic EBMs. Then, we analyze two simple EBMs, the Gaussian model and the Bernoulli Boltzmann machine in the early stages of training, where the effects introduced into the model by the non-convergent samplings can be analytically described along the entire training dynamics. Finally, we illustrate this phenomenon by performing real learnings in a ConvNet EBM and the Boltzmann machine.

2. Preliminaries: Maximum-likelihood training of Energy-Based Models

Suppose $\boldsymbol{x}$ is the value taken by a data point. We consider an arbitrary energy-based model (EBM) in order to model its probability in terms of a Gibbs-Boltzmann distribution:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{e^{-E_{\boldsymbol{\theta}}(\boldsymbol{x})}}{Z(\boldsymbol{\theta})} \quad \text{with} \quad Z(\boldsymbol{\theta}) = \int d\boldsymbol{x}\, e^{-E_{\boldsymbol{\theta}}(\boldsymbol{x})}, \quad (1)$$

where $E_{\boldsymbol{\theta}}(\boldsymbol{x})$ is a dimensionless energy function that depends on a set of parameters $\boldsymbol{\theta} = \{\theta_i\}$. We let $E_{\boldsymbol{\theta}}(\boldsymbol{x})$ be an arbitrarily complex function (e.g. parameterized by a ConvNet or a single-layer Boltzmann machine), the only requirement being that $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ must exist together with its moments as functions of their conjugate variables $\boldsymbol{\theta}$.

Suppose now that we want to find a good set of parameters $\boldsymbol{\theta}$ that minimizes the distance between $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ and the empirical distribution $p_{\mathcal{D}}(\boldsymbol{x})$ of a given set of training examples. This is usually done by maximizing the log-likelihood:

$$\mathcal{L}(\boldsymbol{\theta}) = \langle \log p_{\boldsymbol{\theta}}(\boldsymbol{x}) \rangle_{p_{\mathcal{D}}} = -\langle E_{\boldsymbol{\theta}}(\boldsymbol{x}) \rangle_{p_{\mathcal{D}}} - \log Z(\boldsymbol{\theta}),$$

where $\langle \cdot \rangle_p$ denotes the average over a given distribution $p(\boldsymbol{x})$. This function is intractable in general because so is $Z(\boldsymbol{\theta})$: as a consequence, $\mathcal{L}(\boldsymbol{\theta})$ is typically maximized through a (possibly stochastic) gradient ascent process. Its derivative with respect to parameter θ_i is

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \theta_i} = \langle f_{i,\boldsymbol{\theta}}(\boldsymbol{x}) \rangle_{p_{\boldsymbol{\theta}}} - \langle f_{i,\boldsymbol{\theta}}(\boldsymbol{x}) \rangle_{p_{\mathcal{D}}}, \quad (2)$$

with the local 'forces' in the phase space of parameters

$$f_{i,\boldsymbol{\theta}}(\boldsymbol{x}) \equiv \frac{\partial E_{\boldsymbol{\theta}}(\boldsymbol{x})}{\partial \theta_i}. \quad (3)$$

Note that $\langle f_{i,\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{\theta}) \rangle_p$ are referred as the generalized 'moments' of the distribution $p(\boldsymbol{x})$. Similarly, the Hessian matrix is given by the components

$$H_{ij}(\boldsymbol{\theta}) \equiv \frac{\partial^2 \mathcal{L}(\boldsymbol{\theta})}{\partial \theta_i \partial \theta_j} = \left\langle \frac{\partial f_{j,\boldsymbol{\theta}}(\boldsymbol{x})}{\partial \theta_i} \right\rangle_{p_{\boldsymbol{\theta}}} - \left\langle \frac{\partial f_{j,\boldsymbol{\theta}}(\boldsymbol{x})}{\partial \theta_i} \right\rangle_{p_{\mathcal{D}}}$$
$$- \langle f_{i,\boldsymbol{\theta}}(\boldsymbol{x}) f_{j,\boldsymbol{\theta}}(\boldsymbol{x}) \rangle_{p_{\boldsymbol{\theta}}} + \langle f_{i,\boldsymbol{\theta}}(\boldsymbol{x}) \rangle_{p_{\boldsymbol{\theta}}} \langle f_{j,\boldsymbol{\theta}}(\boldsymbol{x}) \rangle_{p_{\boldsymbol{\theta}}}.$$

This means that in models where all $f_{j,\boldsymbol{\theta}}(\boldsymbol{x})$ are a function only of $\boldsymbol{x}$ and not of the $\boldsymbol{\theta}$ (e.g. $E_{\boldsymbol{\theta}}$ is a linear function of the

parameters), the optimization problem is always convex (i.e. the Hessian is negative semi-definite). This occurs because the covariance matrix is positive semi-definite. We will later analyze various models that satisfy this condition.

In practice the gradient (2) is approximated by a stochastic sampling process (MCMC or Langevin dynamics). Thus, it is correctly estimated as long as this stochastic process converges to equilibrium. However, such a sampling process must be repeated each time a gradient ascent step is performed. As a consequence, ensuring convergence at each training step is not only tedious, but also infeasible in terms of the time required to train an adequate EBM. A shortcut was proposed by Hinton: a few Gibbs sampling steps are sufficient if the sampling process is initialized wisely. This is the idea underlying the recipe of contrastive divergence (CD), where the chains are always initialized at the training samples in the minibatch, or of its refinement, persistent CD (PCD), where the configurations reached after a sampling process are kept as initial configurations for the subsequent sampling process after an update of the parameters.

The effects introduced into EBMs by the lack of convergence of these Markov chains at each update of the training, have gone largely unnoticed for decades, and only recently has it become clear that they introduce memory effects on the training process into models. Even more interesting is the observation that this issue can be exploited to our advantage: non-persistent short MCMC runs are the most efficient strategy to train good generative EBMs, admittedly at the cost of forgetting the link between the equilibrium properties of the family of models parametrized by $\boldsymbol{\theta} = \{\theta_i\}$, and the statistics of the dataset encoded in $p_{\mathcal{D}}$. In the next section, we provide an analytical explanation of these results.

3. Finite-time sampling: The master equation

The implications of using short, nonconvergent MCMC runs to estimate the gradient of Eq. (2) in the EBM training can be formalized and proved generically as follows. For simplicity, we assume here that the number of possible states of our model is finite and that time is continuous. The description for discrete time can be found in the Appendix A, and the generalization to continuous variables in the Appendix B. The latter case will also be considered later in an example.

We consider a MCMC sampling for a given model $E_{\boldsymbol{\theta}}$, and denote $p_a(k)$ as the probability of being at the a-th possible configuration at time k. From now on, we denote the *sampling* time by k to distinguish it from t, the *training* time. Firstly, the *master equation* describes the time evolution of the probability of the system to occupy each of the states a

$$\frac{\mathrm{d}p_a(k)}{\mathrm{d}k} = \sum_{b(\neq a)} w_{b\to a}\, p_b(k) - \sum_{b(\neq a)} w_{a\to b}\, p_a(k), \qquad (4)$$

where $w_{a\to b}$ is the transition probability of going from state a to b for that model. These transition probabilities are independent of k as the process is by definition Markovian. We introduce the vector notation $\boldsymbol{p}(k)$, whose a-th component is $p_a(k)$. The evolution of the probabilities can then be described by a *stochastic evolution* matrix $\boldsymbol{\mathcal{U}}$ ($\mathcal{U}_{ab} = w_{b\to a}$ for $a \neq b$ and $\mathcal{U}_{aa} = -\sum_{b(\neq a)} w_{a\to b}$), as

$$\dot{\boldsymbol{p}}(k) = \boldsymbol{\mathcal{U}}\boldsymbol{p}(k) \quad \Rightarrow \quad \boldsymbol{p}(k) = e^{k\boldsymbol{\mathcal{U}}}\,\boldsymbol{p}_0, \qquad (5)$$

where $\boldsymbol{p}_0$ represents the initial distribution (i.e. $\boldsymbol{p}(0)$).

Secondly, if the Markov chain is reversible, or equivalently satisfies detailed balance (which is the typical situation in EBMs), then the matrix $\boldsymbol{\mathcal{U}}$ is guaranteed to have a spectral decomposition. Therefore, $\boldsymbol{p}(k)$ can be explicitly obtained from (5) through a spectral expansion. Let $\{\lambda_\alpha\}$ be the eigenvalues of $\boldsymbol{\mathcal{U}}$, and $\boldsymbol{u}_\alpha$ the associated eigenvectors. Rewriting the initial distribution $\boldsymbol{p}_0$ in this basis, we have

$$\boldsymbol{p}_0 = \sum_\alpha c_\alpha \boldsymbol{u}_\alpha \quad \Rightarrow \quad \boldsymbol{p}(k) = \textstyle\sum_\alpha c_\alpha e^{k\lambda_\alpha}\boldsymbol{u}_\alpha, \qquad (6)$$

with the coefficient $\{c_\alpha\}$ encoding the memory of the initial condition of the sampling run. The probability $\boldsymbol{p}(k)$ must be normalized at all k (i.e. $\sum_a p_a(k) = 1$), hence at least one eigenvalue must be 0, and the rest must be necessarily negative for the long-time probabilities to remain bounded.

Thirdly, when $\boldsymbol{\mathcal{U}}$ is also irreducible (i.e. any configuration can be reached from any other in a finite number of steps), it follows from the Perron-Frobenius theorem that the 0 eigenvalue must be unique (namely, λ_0) (Keizer, 1972). We emphasize here that the theorem holds whether or not MCMC chains mix in a given simulation. It is a property that arises solely from the structure of the stochastic matrix. Furthermore, one can always choose $c_0 = 1$, which imposes from the long-time normalization that $\sum_a u_{0,a} = 1$, and consequently for the initial normalization that all the eigenvectors with negative eigenvalues must fulfill $\sum_a u_{\alpha,a} = 0$ with $\alpha \neq 0$. Under these constraints, we have then

$$\boldsymbol{p}(k) = \boldsymbol{u}_0 + \sum_{\alpha>0} c_\alpha\, e^{-k|\lambda_\alpha|}\boldsymbol{u}_\alpha. \qquad (7)$$

An analogous expression is obtained for discrete sampling steps in Appendix A and for continuous variables for the states in Appendix B. By construction, it is guaranteed that the stochastic dynamics converges to the distribution $\boldsymbol{u}_0$ after a long time. Moreover, the *mixing* time is $\kappa_{\mathrm{mix}} = 1/|\lambda_1|$, with λ_1 the second smallest absolute eigenvalue. Up to this point, the discussion was completely generic.

Fourthly, we can now explicitly focus on EBMs by forcing the long-time distribution $\boldsymbol{u}_0$ to coincide with the Gibbs distribution (1), i.e. $u_{0,a} = p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(a)}) \equiv \exp\left[-E_{\boldsymbol{\theta}}(\boldsymbol{x}^{(a)})\right]/Z(\boldsymbol{\theta})$, where $\boldsymbol{x}^{(a)}$ are the variables in

configuration a. To be consistent, $\boldsymbol{u}_0$ must be the solution of the steady-state condition $\boldsymbol{\mathcal{U}}(\boldsymbol{\theta})\,\boldsymbol{p}_{\boldsymbol{\theta}} = \mathbf{0}$ (cf. (5)). It is in particular directly satisfied if we impose the *detailed balance* condition $w_{a\to b}\, e^{-E_{\boldsymbol{\theta}}(\boldsymbol{x}^{(a)})} = w_{b\to a}\, e^{-E_{\boldsymbol{\theta}}(\boldsymbol{x}^{(b)})}$ to hold, which has previously been imposed for the spectral expansion to exist. Standard choices for the update rules, such as the Metropolis-Hastings or the Heat Bath dynamics satisfy detail balance.

Finally, under this choice of transition probabilities, and the absence of some pathological cycles in the dynamics, we have the guarantee that the sampled distribution will converge to the equilibrium one with increasing k as

$$\boldsymbol{p}(k) = \boldsymbol{p}_{\boldsymbol{\theta}} + \boldsymbol{\Lambda}_{\boldsymbol{\theta}}\left(k, \boldsymbol{p}_0\right), \tag{8}$$

where $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}$ is the term $\sum_{\alpha>0}$ in Eq. (7), a decay function of k/κ_{mix} at long times. One cannot compute this function $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}$ in general, but we know it depends only on the initial conditions $\boldsymbol{p}_0 \leftrightarrow \{c_\alpha\}$, the run duration k, and the evolution matrix $\boldsymbol{\mathcal{U}}(\boldsymbol{\theta})$. Then, the expected value for $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}) \equiv \boldsymbol{\nabla} E_{\boldsymbol{\theta}}(\boldsymbol{x})$ in the log-likelihood gradient (cf. Eqs. (2)-(3)), estimated after a sampling time k, is given by

$$\begin{aligned} \langle \boldsymbol{f}_{\boldsymbol{\theta}} \rangle_{k,\boldsymbol{p}_0} &\equiv \textstyle\sum_a \boldsymbol{f}_{\boldsymbol{\theta}}^{(a)}\, p_a(k) \\ &= \langle \boldsymbol{f}_{\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}} + \textstyle\sum_a \boldsymbol{f}_{\boldsymbol{\theta}}^{(a)} \Lambda_{\boldsymbol{\theta},a}\left(k, \boldsymbol{p}_0\right). \end{aligned} \tag{9}$$

with $f_{i,\boldsymbol{\theta}}^{(a)} \equiv f_{i,\boldsymbol{\theta}}(\boldsymbol{x}^{(a)})$. In other words, by approximating the gradient with a non-convergent MCMC sampling, one computes the real model average plus a correction depending on k, $\boldsymbol{p}_0$, and the model parameters $\boldsymbol{\theta}$. Thus, at each training update, the update rule of Eq. (2) is replaced by

$$\left.\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \theta_i}\right|_{k,\boldsymbol{p}_0} = \langle f_{i,\boldsymbol{\theta}} \rangle_{p_{\boldsymbol{\theta}}} - \langle f_{i,\boldsymbol{\theta}} \rangle_{p_{\mathcal{D}}} + \sum_a f_{i,\boldsymbol{\theta}}^{(a)} \Lambda_{\boldsymbol{\theta},a}(k, \boldsymbol{p}_0)\,.$$

Moreover, this implies that the fixed point $\boldsymbol{\theta} = \boldsymbol{\theta}^*$ of this dynamics (i.e. $\nabla\mathcal{L} = 0$), if it can be reached, satisfies

$$\langle \boldsymbol{f}_{\boldsymbol{\theta}^*} \rangle_{p_{\mathcal{D}}} = \langle \boldsymbol{f}_{\boldsymbol{\theta}^*} \rangle_{p_{\boldsymbol{\theta}^*}} + \textstyle\sum_a \boldsymbol{f}_{\boldsymbol{\theta}^*}^{(a)} \Lambda_{\boldsymbol{\theta}^*,a}(k, \boldsymbol{p}_0)\,. \tag{10}$$

Theorem 3.1. *We consider a training of an EBM that follows the gradient in Eq. (2) estimated with nonconvergent, irreducible, and reversible Markov chains. If it reaches a dynamic fixed point where $\boldsymbol{\nabla}\mathcal{L} = \mathbf{0}$, then, the trained model generates samples consistent with the statistics of the training set, i.e. same generalized moments – given by $\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^*} \rangle_{k,\boldsymbol{p}_0} = \langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^*} \rangle_{p_{\mathcal{D}}}$ –, but only if the samples are drawn via the same identical stochastic process used for the training: i.e., the same update rules, the same initialization, and the same Markov chain length.*

The proof results from combining the Eqs. (9)-(10) using a model with parameters $\boldsymbol{\theta}^*$ trained with a given sampling dynamics and $(\boldsymbol{p}_0, k)$. Now, the samples generated with this model, with $\boldsymbol{\mathcal{U}}(\boldsymbol{\theta}^*)$, an arbitrary initial distribution $\boldsymbol{q}_0$ and after a time k', must satisfy the following statistics:

$$\langle \boldsymbol{f}_{\boldsymbol{\theta}^*} \rangle_{k',\boldsymbol{q}_0} = \langle \boldsymbol{f}_{\boldsymbol{\theta}^*} \rangle_{p_{\mathcal{D}}} + \boldsymbol{D}_{\boldsymbol{\theta}^*}(k', \boldsymbol{q}_0, k, \boldsymbol{p}_0), \tag{11}$$

with the mismatch given by Eq. (7):

$$\begin{aligned} \boldsymbol{D}_{\boldsymbol{\theta}^*}(k', \boldsymbol{q}_0, k, \boldsymbol{p}_0) &= \textstyle\sum_a \boldsymbol{f}_{\boldsymbol{\theta}^*}^{(a)} \left[\Lambda_{\boldsymbol{\theta}^*,a}(k', \boldsymbol{q}_0) - \Lambda_{\boldsymbol{\theta}^*,a}(k, \boldsymbol{p}_0)\right] \\ &= \textstyle\sum_{a,\alpha} \boldsymbol{f}_{\boldsymbol{\theta}^*}^{(a)} \left(c_\alpha(\boldsymbol{q}_0) e^{-k'|\lambda_\alpha|} - c_\alpha(\boldsymbol{p}_0) e^{-k|\lambda_\alpha|}\right) u_{\alpha,a}. \end{aligned}$$

and the model-dependent $\{\lambda_\alpha(\boldsymbol{\theta}^*), \boldsymbol{u}_\alpha(\boldsymbol{\theta}^*)\}$. This means that if we choose the same sampling time $k' = k$, the same initialization probabilities $\boldsymbol{q}_0 = \boldsymbol{p}_0$, and the same dynamic rules as used in learning, then we will match perfectly the empirical statistics. This statement holds for all derivatives of the energy function $E_{\boldsymbol{\theta}^*}$ with respect to the model parameters. Also, a long sampling ($k' \to \infty$) can deviate strongly from the empirical expectation if the training k is much shorter than the mixing time κ_{mix}. This discrepancy results from the remaining factors $\{-c_\alpha(\boldsymbol{p}_0)\, \exp^{-k|\lambda_\alpha|}\}$ in Eq. (11), which would vanish in the opposite limit $k \gg \kappa_{\text{mix}}$.

The results of Theorem 3.1 also hold before reaching the fixed point [i.e. for trained nonconvex EBMs or unfinished trainings of convex EBMs] but with several limitations.

Theorem 3.2. *We consider a non-convergent EBM trained to a point where $\boldsymbol{\nabla}\mathcal{L}(\boldsymbol{\theta}^\dagger) = \boldsymbol{\varepsilon} \neq \mathbf{0}$. Then, the samples generated by the model $\boldsymbol{\theta}^\dagger$ (with the same stochastic process used for training –the same k, the same initialization $\boldsymbol{p}_0$ and update rules) reproduce the statistics of $\langle \boldsymbol{\nabla} E_{\boldsymbol{\theta}^\dagger} \rangle_{p_{\mathcal{D}}}$ of the training set with an error $\boldsymbol{\varepsilon}$. Thus, the error committed after time k decreases (in average) as the training approaches its fixed point. Moreover, for a conveniently small $\boldsymbol{\varepsilon}$ (which depends on $\boldsymbol{p}_0$, k and $\boldsymbol{f}_{\boldsymbol{\theta}^\dagger}$), it is possible to find a sampling time $k^\dagger(\varepsilon_i) \neq 0$ at which the error for $\langle f_{i,\boldsymbol{\theta}^\dagger} \rangle_{p_{\mathcal{D}}}$ is zero. This $k^\dagger$ moves toward the length k of the Markov chains used for training as the training approaches its fixed point.*

This second theorem is straightforward from the definition of the gradient and Eq. (11). Then, upon sampling model $\boldsymbol{\theta}^\dagger$ with the same stochastic matrix $\boldsymbol{\mathcal{U}}(\boldsymbol{\theta}^\dagger)$, the absolute error of $f_{\boldsymbol{\theta}^\dagger}$ averaged on samples generated at a time k' from the initial distribution $\boldsymbol{q}_0$ is:

$$\left|\langle \boldsymbol{f}_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}) \rangle_{k',\boldsymbol{p}_0} - \langle \boldsymbol{f}_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}) \rangle_{p_{\mathcal{D}}}\right| = \left|\boldsymbol{\varepsilon} + \boldsymbol{D}_{\boldsymbol{\theta}^\dagger}(k', \boldsymbol{q}_0, k, \boldsymbol{p}_0)\right|,$$

and $\boldsymbol{D}_{\boldsymbol{\theta}^\dagger}(k', \boldsymbol{q}_0, k, \boldsymbol{p}_0)$ vanishes for $k' = k$ and $\boldsymbol{q}_0 = \boldsymbol{p}_0$. The second statement stands from the fact that $\boldsymbol{D}_{\boldsymbol{\theta}^\dagger}$ is a smooth function of k' that (for k short compared to κ_{mix}) changes sign around $k' = k$. This means that the error of $\langle f_{i,\boldsymbol{\theta}} \rangle_{p_{\mathcal{D}}}$ will be strictly zero at $k^\dagger(\varepsilon_i) \neq k$ for ε_i sufficiently small, with ε_i being the discrepancy between the statistics of the data set and the model associated to the i-th moment statistics). This means that $k^\dagger$ moves progressively toward

k as $\varepsilon_i \to 0$. In a typical MCMC scheme, time is always an integer variable, so this optimum might not be easy to access. However, in Langevin dynamics this is possible, as we show in a later example. This also means that as learning progresses ($\boldsymbol{\varepsilon} \to \mathbf{0}$), the best sampling time $k^\dagger$ progressively converges to k for all 'moments' simultaneously.

We assess sample quality based on the agreement of the energy moments, $\langle f_{i,\boldsymbol{\theta}} \rangle$, between the training and the generated sets. The key question here is whether the samples resulting from convergent MCMC training are more representative of the distribution of the dataset than those obtained using a non-persistent, non-convergent strategy. The answer is not straightforward. An immediate conclusion from previous results is that convergent MCMC training does not improve the agreement of moments because $\boldsymbol{p}_{\boldsymbol{\theta}}$ reproduces the same statistics than $\boldsymbol{p}(k, \boldsymbol{p}_0)$ in the nonconvergent protocol. The same is true for all observables that can always be expressed as functions of the distribution's 'moments' (3). However, no further generalizations can be made for the rest of observables because the expected committed error is not only related with the non-convergent sampling, but also with the limitations of the model chosen (which may not be powerful enough to capture all features of the dataset), and both contributions can have any sign. Consequently, these changes may either cancel or accumulate, depending on the observable or EBM under consideration.

4. Solvable models

In the next sections we will examine two simple classes of EBMs for which: *(i)* the correction $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(k, \boldsymbol{p}_0)$ introduced by the non-convergent Markov chains can be explicitly computed, and *(ii)* its effect on the inferred model parameters can be explicitly quantified along the entire learning process. Specifically, we focus on the inverse problem for families of exponentials with quadratic arguments: Given a dataset drawn from an empirical distribution $p_{\mathcal{D}}(\boldsymbol{x})$, we wish to infer the best parameters $\boldsymbol{\theta} \equiv \{\boldsymbol{J}, \boldsymbol{h}\}$ of the target distribution with an energy function of the type

$$E_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{1}{2} \sum_{ij} x_i J_{ij} x_j + \sum_i h_i x_i.$$

For simplicity we consider $h_i = 0$ in the following, but a generalization with external fields/biases is straightforward. As in the general case in Sec. 2, we estimate the parameters by maximizing the log-likelihood of the model evaluated on the dataset, which reads

$$\mathcal{L}(\boldsymbol{\theta}) = -\tfrac{1}{2} \sum_{ij} J_{ij} \langle x_i x_j \rangle_{p_{\mathcal{D}}} - \log Z(\boldsymbol{\theta}). \tag{12}$$

Again, the classical approach to maximizing log-likelihood is based on calculating its gradient

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial J_{ij}} = -\Big(\langle x_i x_j \rangle_{p_{\mathcal{D}}} - \langle x_i x_j \rangle_{p_{\boldsymbol{\theta}}} \Big), \tag{13}$$

The average over $\boldsymbol{p}_{\boldsymbol{\theta}}$ is usually calculated using the heat-bath MCMC algorithm. In the next subsections, we analytically investigate the effect of replacing the equilibrium averages with estimates obtained by a non-convergent MCMC process. We first consider the Gaussian model and then the high-temperature regime of the inverse Ising problem.

4.1. Exact results for the Gaussian model

In the Gaussian case, we consider data points $\boldsymbol{x} \in \mathbb{R}^D$. To be well-defined, the coupling matrix $\boldsymbol{J}$ must be positively definite, namely all its eigenvalues must be strictly positive. Under this condition, and given the convexity of this likelihood function, we can explicitly compute the correlation function, and the stationary solution of Eq. (13) reads:

$$\left\langle \boldsymbol{x}\boldsymbol{x}^{\mathrm{T}} \right\rangle_{p_{\boldsymbol{\theta}}} = \boldsymbol{J}^{-1} \underset{k\to\infty}{\overset{[\nabla\mathcal{L}=0]}{\Longrightarrow}} \boldsymbol{J}^* = \left\langle \boldsymbol{x}\boldsymbol{x}^{\mathrm{T}} \right\rangle_{p_{\mathcal{D}}}^{-1}.$$

Despite the maximum likelihood point can be found analytically, we now compute it through a gradient ascent method, with a suitable learning rate. Moreover, the gradient can be projected on the eigenbasis of $\boldsymbol{J}$. Denoting $\{\boldsymbol{u}^\alpha\}$ the eigenvectors, we have that $J_{ij} = \sum_\alpha u_i^\alpha J_\alpha u_j^\alpha$. Following (Decelle et al., 2017; 2018; Decelle & Furtlehner, 2021b), we can write a set of evolution equations for the eigenvalues $\{J_\alpha\}$ and for the rotation $\Omega_{\alpha\beta}$ of the eigenvectors

$$\frac{\mathrm{d}J_\alpha}{\mathrm{d}t} = -\hat{c}_{\alpha\alpha} + 1/J_\alpha, \tag{14}$$

$$\Omega_{\alpha\beta} \equiv \boldsymbol{u}^\alpha \frac{\mathrm{d}\boldsymbol{u}^\beta}{\mathrm{d}t} = \frac{\langle x_\alpha x_\beta \rangle_{p_{\boldsymbol{\theta}}} - \hat{c}_{\alpha\beta}}{J_\alpha - J_\beta}, \tag{15}$$

where we defined $\hat{c}_{\alpha\beta} = \sum_{i,j} u_i^\alpha u_j^\beta \langle x_i x_j \rangle_{p_{\mathcal{D}}}$ and $x_\alpha = \sum_i x_i u_i^\alpha$, the projection on the eigenmodes of $\boldsymbol{J}$ of the correlation matrix and $\boldsymbol{x}$, respectively. In this formulation, we see that the eigenvectors have to diagonalize the correlation matrix to cancel the rotation, i.e. each vector $\boldsymbol{u}^\alpha$ will align itself with the principal directions of the dataset, and the eigenvalues will eventually adjust to $J_\alpha = 1/\hat{c}_{\alpha\alpha} = J_\alpha^*$.

However, we are interested here in the case where non-convergent MCMC chains are used to approximate the correlation function $\left\langle \boldsymbol{x}\boldsymbol{x}^{\mathrm{T}} \right\rangle_{p_{\boldsymbol{\theta}}}$, and what it changes for Eqs. (13)-(14)-(15). We assume for simplicity that the dataset is centered with zero mean, $\langle x_i \rangle_{p_{\mathcal{D}}} = 0$. The evolution of the chain is described by the following Langevin equation

$$\frac{\mathrm{d}x_i(k)}{\mathrm{d}k} = -\sum_j J_{ij} x_j(k) + \eta_i(k), \tag{16}$$

with $\boldsymbol{\eta}$ a Gaussian white noise of zero mean and variance $\overline{\eta_i(k)\eta_j(k')} = 2\delta_{ij}\delta(k-k')$, where $\overline{(\cdot)}$ denotes the average with respect to different independent trajectories. These equations can be decoupled by projecting them on the eigenmodes of $\boldsymbol{J}$, leading upon sampling to

$$x_\alpha(k) = x_\alpha(0) e^{-J_\alpha k} + \int_0^k \mathrm{d}\tilde{k}\, e^{J_\alpha(\tilde{k}-k)} \eta_\alpha(\tilde{k}).$$

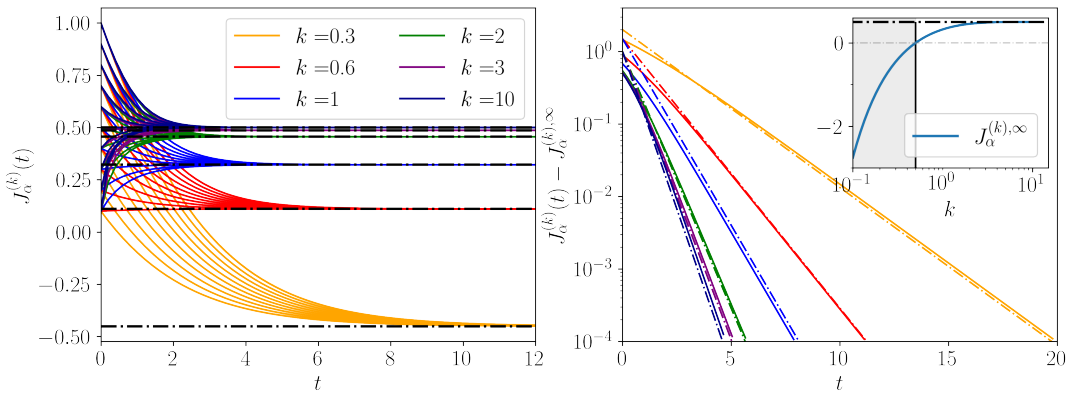

Figure 1. **Left:** Evolution of $J_\alpha^{(k)}(t)$ for different values of k (different colors) and initial conditions for $J_\alpha^{(k)}(0)$. **Right:** Convergence for a large training time t, where the higher k, the faster the convergence; dash-dotted lines are the numerical integration of Eq. (18), full lines show the exponential fit $J_\alpha^{(k)}(t) - J_\alpha^{(k),\infty} \sim A\exp(-t/\tau)$ with τ given by Eq. (20). **Inset:** the asymptotic values $J_\alpha^{(k)}(t\to\infty)$ as a function of k. The gray shaded region corresponds to the non-physical solutions and the vertical line to the limit $k = (\hat{c}_{\alpha\alpha} - \langle x_\alpha^2\rangle_0)/2$.

We see that the convergence properties are dominated by the eigenvalues J_α. Hence, for $k \lesssim J_\alpha^{-1}$, the MCMC chain will be far away from its equilibrium value. We can obtain a similar result for the correlation function; let us directly look at the average over realizations at time k

$$\langle x_\alpha x_\beta\rangle_{k,\boldsymbol{p}_0} = \langle x_\alpha x_\beta\rangle_0 \, e^{-(J_\alpha+J_\beta)k} + \frac{\delta_{\alpha\beta}}{J_\alpha}\left(1-e^{-2J_\alpha k}\right), \tag{17}$$

where $\langle\cdot\rangle_0$ is the average over the initialization distribution. Thus, the asymptotic value is given by $\langle x_\alpha x_\beta\rangle_\infty = \delta_{\alpha\beta}/J_\alpha$. We now analyze how the learning dynamics are affected when the model average $\langle x_\alpha x_\beta\rangle_{p_{\boldsymbol{\theta}}}$ is replaced in the gradient by $\langle x_\alpha x_\beta\rangle_k$, i.e. the averages computed at fixed time k when the chains are initialized at a fixed random initial distribution $p_0(\boldsymbol{x})$. We first consider only the evolution of the eigenmodes given by Eq. (14), and keep the eigenvectors $\boldsymbol{u}^\alpha$ fixed. Under these conditions, the gradient calculated with non-convergent chains is quite different to the real one

$$\left(\frac{\partial\mathcal{L}}{\partial\boldsymbol{J}}\right)_{\alpha\beta}\Bigg|_{k,\boldsymbol{p}_0} = -\left[\langle x_\alpha x_\beta\rangle_{p_\mathcal{D}} - \langle x_\alpha x_\beta\rangle_{k,\boldsymbol{p}_0}\right]$$
$$= \left(\frac{\partial\mathcal{L}}{\partial\boldsymbol{J}}\right)_{\alpha\beta} + \left(\langle x_\alpha x_\beta\rangle_0 - \frac{\delta_{\alpha\beta}}{J_\alpha}\right)e^{-(J_\alpha+J_\beta)k}.$$

Note that the gradient now depends on both the sampling time k and the initialization $\boldsymbol{p}_0$ of the chains. Then the evolution for the eigenvalues of $\boldsymbol{J}$ in Eq. (14) in the finite k case transforms into

$$\frac{\mathrm{d}J_\alpha^{(k)}}{\mathrm{d}t} = -\hat{c}_{\alpha\alpha} + \frac{1}{J_\alpha^{(k)}} + \left(\langle x_\alpha^2\rangle_0 - \frac{1}{J_\alpha^{(k)}}\right)e^{-2J_\alpha^{(k)}k}, \tag{18}$$

whose stationary solution at $t\to\infty$, $J_\alpha^{(k),\infty}$, satisfies

$$\hat{c}_{\alpha\alpha} - \langle x_\alpha^2\rangle_0 \, e^{-2J_\alpha^{(k),\infty}k} = \frac{1}{J_\alpha^{(k),\infty}}\left(1-e^{-2J_\alpha^{(k),\infty}k}\right). \tag{19}$$

Theorem 4.1. *If $k > (\hat{c}_{\alpha\alpha} - \langle x_\alpha^2\rangle_0)/2$ holds, then Eq. (19) admits a strictly positive solution. Otherwise, $J_\alpha^{(k),\infty}$ is negative and the non-convergent learning procedure diverges.*

This theorem is proved by analyzing the function $g(J) = J\hat{c}_{\alpha\alpha} - 1 + e^{-2Jk}(1 - J\langle x_\alpha^2\rangle_0)$, see the appendix C for more details. Therefore, we can define a lower bound for the sampling time k below which the learning dynamics leads to an unphysical solution (unbounded Gaussian integrals). From Eq. (19) we recover straightforwardly the correct solution $J_\alpha^* = 1/\hat{c}_{\alpha\alpha}$ in the limit $k\to\infty$.

Equation (18) can be integrated numerically, and the results for various k and initial conditions are shown in Fig. 1 (left). The smaller k, the slower the convergence to the fixed point $J_\alpha^{(k),\infty}$. Indeed, one can show that the typical training time τ required to reach the stationary point decreases with k as

$$\frac{1}{\tau} \sim k\left[\langle x_\alpha^2\rangle_0 - \frac{e^{-2kJ_\alpha^{(k),\infty}}}{J_\alpha^{(k),\infty}}\right] + \frac{\left(1-e^{-2kJ_\alpha^{(k),\infty}}\right)}{2\left(J_\alpha^{(k),\infty}\right)^2}, \tag{20}$$

by simply assuming an exponential relaxation for a large training time t, i.e., $J_\alpha^{(k)}(t) \sim J_\alpha^{(k),\infty} + A\exp(-t/\tau)$ in Eq. (18). Therefore, from a computational point of view, a smaller value of k means by definition faster sampling, but the time required for the learning process to converge increases as k decreases. Thus, the optimal value of k results from a trade-off between these two contributions.

One can also study the effect of the non-convergent chains on the alignment of the eigenvectors of $\boldsymbol{J}$, which we kept fixed until now. For this, we rely on the simplified situation proposed in (Decelle & Furtlehner, 2021b), where a reduced 2D subspace is considered in which the 2 first eigenvectors of $\boldsymbol{J}$ form an angle ϕ with the 2 principal directions of the dataset. In this case, one obtains a set of 3 dynamical

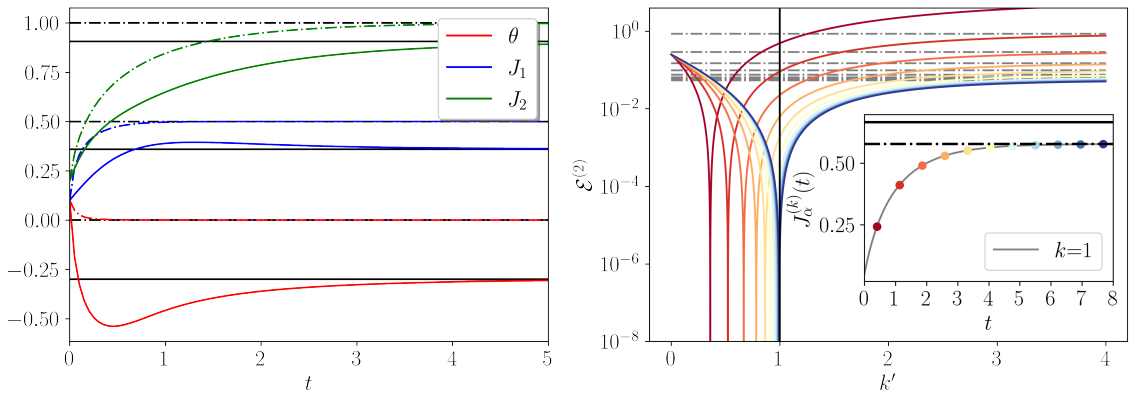

Figure 2. **Left:** Numerical resolution of the learning dynamics in presence of two modes. The dash-lines represent the resolution using a convergent dynamics $k \to \infty$, while the plain ones correspond to $k=1$. **Right: (inset)** evolution of $J_\alpha^{(k)}(t)$ for $k=1$ at various stages of the learning. **(Main)** The error on the correlation function, $\mathcal{E}^{(2)} = (\langle x_\alpha^2 \rangle_{k',\boldsymbol{p}_0} - \hat{c}_{\alpha\alpha})^2$, of the machine at different stages of the learning. We can clearly see the peak of best performance (of several orders of magnitude) at a $k^\dagger$ that moves to k as the learning converges. The color lines match the circles on the inset. The horizontal dashed-line represent the $k' \to \infty$ asymptotic value of the error.

equations for $\phi(t)$, and the 2 eigenvalues $J_1(t)$ and $J_2(t)$

$$\frac{\mathrm{d}\phi}{\mathrm{d}t} = \left[\sin(2\phi)(\hat{c}_1 - \hat{c}_2)/2 + \langle x_1 x_2 \rangle_0 \, e^{-(J_1+J_2)k}\right]/(J_2 - J_1),$$
$$\frac{\mathrm{d}J_1}{\mathrm{d}t} = J_1^{-1} - \hat{c}_1 \cos^2(\phi) - \hat{c}_2 \sin^2(\phi) - \left(J_1^{-1} - \langle x_1^2 \rangle_0\right) e^{-2J_1 k},$$
$$\frac{\mathrm{d}J_2}{\mathrm{d}t} = J_2^{-1} - \hat{c}_1 \sin^2(\phi) - \hat{c}_2 \cos^2(\phi) - \left(J_2^{-1} - \langle x_2^2 \rangle_0\right) e^{-2J_2 k}.$$

(see Appendix D for details) whose numerical solution is given in Fig. 2–left. In the limiting case $k \to \infty$, we see as expected the eigenvectors align (or anti-align) with the eigenvectors of the covariance matrix of the dataset and $\phi \to 0$ or π. Yet, for finite k this is not true, and an asymptotic finite ϕ is reached which embodies the sampling mismatch.

In Sec. 3, we showed that generic EBMs trained with non-convergent MCMC samplings can be used to generate samples that perfectly match the statistics of the dataset even before the learning has converged. For that purpose, we only need to mimic the same dynamic process used for training in the generation phase. We can then quantify this prediction in this solvable case, given the parameters of the model $J_\alpha^{(k)}(t)$ after a learning time t, by computing the average correlations through Eq. (17).

Theorem 4.2. *We consider a Gaussian model trained with non-convergent Markov chains of length k along the modes α (i.e. keeping fixed the eigenvectors $\{\boldsymbol{u}^\alpha\}$), and parameterized by $\{J_\alpha^{(k),\infty}\}$. It exactly reproduces the correlations of the dataset when sampled with the same dynamic process used in the training: same transition rules, initialization distribution, and chain length.*

The proof is straightforward by inserting the fixed point equation (19) in (17). In particular, let us generate new data $\boldsymbol{y}$ with the trained model. When $k' = k$, the averaged correlation along each mode α

$$\langle y_\alpha^2 \rangle_{k,\boldsymbol{p}_0} = \hat{c}_{\alpha\alpha} + \left(\langle y_\alpha^2 \rangle_0 - \langle x_\alpha^2 \rangle_0\right) e^{-2J_\alpha^{(k),\infty} k},$$

match those of the dataset but only if the chains are initialized from the same distribution used during the training, i.e, same $\langle x_\alpha^2 \rangle_0$. In Fig. 2–right, we show the mean squared error $\mathcal{E}^{(2)}$ in the correlations committed at different stages of learning, i.e., using the numerically integrated $J_\alpha^{(k)}(t)$. As anticipated in Sect. 3, one can also reproduce the statistics of the dataset before convergence, but at an effective sampling time $k^\dagger$ that moves to k as learning progresses.

As a final remark, from Eq. (17) we see that the cross-correlation terms $\langle x_\alpha x_\beta \rangle_k$ for $\alpha \neq \beta$ are zero only when the initial conditions are de-correlated along both modes, and thus the eigenvectors $\boldsymbol{u}^\alpha$ easily align with the principal directions of the dataset. In purely random initializations and to some extent in the PCD recipe, the cross terms are zero, but in the CD scheme this is not the case in the initial stages of learning, which should affect the rotational dynamics of the eigenvectors of the coupling matrix.

4.2. Exact results on the Boltzmann machine

The previous results can be extended to the case of binary variables, such as the classical Ising inverse problem (Nguyen et al., 2017), or the Boltzmann machine (BM). In this context, we have $\boldsymbol{x} = \{\pm 1\}^N$, for which the distribution is defined for all $\boldsymbol{J}$. In general, the learning of the BM cannot be solved exactly, since there is no general closed-form expression for the correlation function. However, we can deal with the so-called mean-field case (MF), where the coupling matrix scales with $1/N$. In this regime, BM can be solved analytically using MF theory (Kappen & Rodríguez, 1998; Nguyen & Berg, 2012; Ricci-Tersenghi, 2012). It is then possible to derive the evolution of the magnetization and the correlation function for a given MCMC dynamics. Following (Suzuki & Kubo, 1968), we first define the heat-

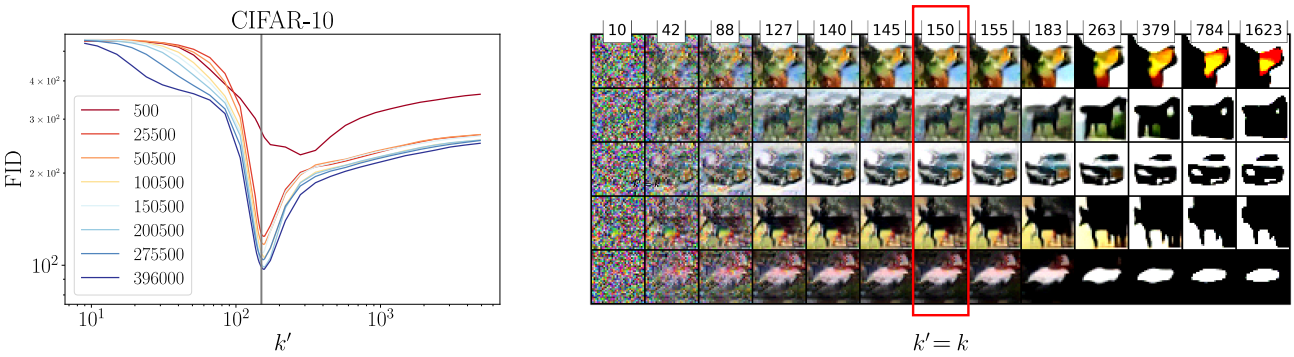

Figure 3. Generation results obtained with a ConvNet EBM trained on CIFAR-10 using $k = 150$ Langevin MCMC sampling steps from random initial conditions. **Left:** We use the Frechet Inception Distance Score (FID) to evaluate the generation quality as a function of the sampling time k'. The different colors correspond to different training epochs. We see again that the best score is achieved at $k' = k$ (corresponding to the gray line). **Right:** Example of 5 generated images as a function of k' for the most trained machine.

bath dynamics transition matrix as the probability of a given spin s_j when all others are fixed: $W_j(s_j^{t+1}|\{s_i^t\}_{i\neq j})$. Then, using the master equation as Eq. (4), it is possible to write the evolution equation of the moments of the spins, which in the case of the correlations gives

$$\frac{\mathrm{d}\langle s_i s_j\rangle_k}{\mathrm{d}k} = -2\,\langle s_i s_j\rangle_k + \langle s_i \tanh(\boldsymbol{J}\boldsymbol{s})_j\rangle_k + \langle s_j \tanh\,(\boldsymbol{J}\boldsymbol{s})_i\rangle_k\,.$$

The above equation is exact for any coupling matrix $\boldsymbol{J}$, but cannot be solved in general. In the MF regime, we can expand the $\tanh$ at first order and solve the set of linear differential equations. Since it is very similar to the Gaussian case, we only provide the results. At leading order, the evolution upon sampling of the two-body correlations, in the eigenbasis of the couplings $\boldsymbol{J}$, can be written

$$\frac{\mathrm{d}\,\langle x_\alpha x_\beta\rangle_k}{\mathrm{d}k} = -\langle x_\alpha x_\beta\rangle_k\,(2 - J_\alpha - J_\beta) + 2\delta_{\alpha\beta},$$

whose solution is

$$\langle x_\alpha x_\beta\rangle_k = \frac{\delta_{\alpha\beta}}{1 - J_\alpha} + \left[\langle x_\alpha x_\beta\rangle_0 - \frac{\delta_{\alpha\beta}}{1 - J_\alpha}\right] e^{-(2-J_\alpha-J_\beta)\frac{k}{\tau}}.$$

This analytical description is valid only in the high-temperature regime, where $\max\{J_\alpha\} < 1$. Nevertheless, in this regime we can reproduce the same analysis of the Gaussian model and reach the same conclusion: BMs trained with non-convergent MCMC to compute the gradient fit models that reproduce the training set statistics, but only when sampled following the dynamics of the training: same initial conditions, transition rules, and number of steps k.

5. Numerical experiments

This nonmonotonic effect in the quality of the generated samples and its relation to the mixing times of the algorithm was intensively studied numerically in the case of RBMs (Decelle et al., 2021) and yielded a picture fully compatible with the results obtained here. For example, the evolution of their "quality estimators" with sampling time is qualitatively very similar to those shown in Fig. 2 for discrete sampling times. In Fig. 3, we reproduce essentially the same results using the ConvNet from Ref. (Nijkamp et al., 2020) (same architecture and hyperparameters) trained on the CIFAR-10 (Krizhevsky et al., 2009). Here, we use the Frechet Inception Distance Score (Heusel et al., 2017) to evaluate the generation quality at different training times and as a function of sampling time.

Now we test our analytic predictions on the BM for the *inverse Ising problem*, where $\boldsymbol{x} \in \{-1, 1\}^N$, and the energy function is again quadratic in the variables $\boldsymbol{x}$: $E_{\boldsymbol{\theta}}(\boldsymbol{x}) = -\sum_{i<j} J_{ij}x_ix_j - \sum_i h_ix_i$. In the following experiment, the dataset is made of configurations drawn from a 2D ferromagnetic Ising model, with $N = 7^2$ spins, $h_i = 0$, and $J_{ij} = \beta$ when (i, j) are nearest neighbors on the 2D lattice, zero otherwise. We consider $\beta = 0.44$, i.e a temperature slightly above the critical point. Learning is done with k-steps of nonconvergent heat-bath Markov chains initialized with random conditions. For $k = 5$ and for machines obtained at different stages of the learning process, we show in Fig. 4–left the evolution of the error in the 2-point correlations for generated samples as a function of the number of MCMC steps used to synthesize them. We clearly observe the same behavior as in the Gaussian case, with a peak of best quality developing at $k' = k$. We can also observe that the quality of the best generated samples does not depend on k, as shown in the inset of Fig. 4, where the error of $\mathcal{E}^{(2)}$ at $k' = k$ (at the end of the training) is plotted as a function of the k used for training. Remarkably, we obtain the same result for two other observables that are not directly related to the generalized moments in the gradient, for example, the 4-th order correlations between spins or the relative entropy (computed with gzip (Baronchelli et al., 2005)). Surprisingly, trainings using $k = 1$ generate samples as good as those with $k = 50$. Concerning the behavior during training,

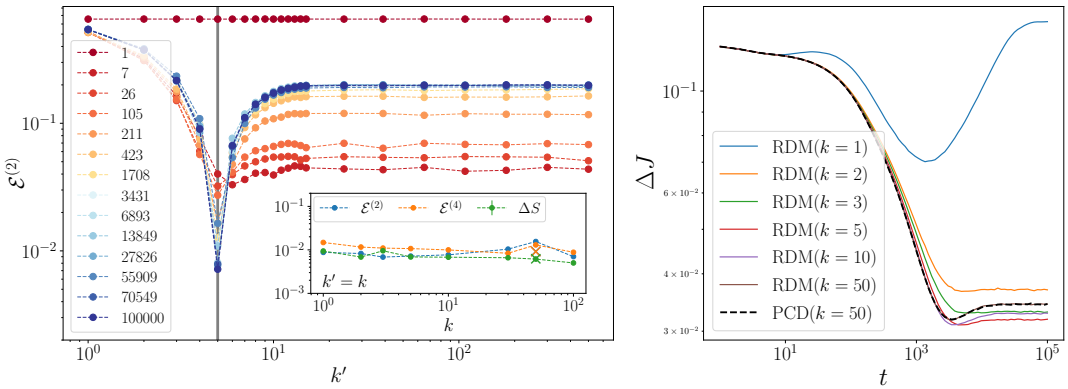

Figure 4. **Left:** Error over the covariance matrices $\mathcal{E}^{(2)} = \sum_{i<j} (\langle x_i x_j \rangle_{k',\boldsymbol{p}_0} - \langle x_i x_j \rangle_{p_\mathcal{D}})^2 / \binom{N}{2}$ between the training set and the data generated at different learning ages vs the generation time k', for a BM trained with data sampled from a 2D ferromagnetic Ising model with $N = 7^2$ at $k = 5$. Similarly to Fig. 2 (right), the error shows a minimum at $k' = k$. **Inset:** Comparison of generated data's quality vs. the k used for the learning. We show the error over covariance matrix, $\mathcal{E}^{(2)}$, the error over the $4-$th moments on $2-$sided plaquettes in the 2D lattice, $\mathcal{E}^{(4)}$, and the relative entropy ΔS computed using gzip. Each of them is computed using samples generated in $k' = k$ steps by machines trained for $t = 10^5$ updates using different k. Cross-marks refer to the same result obtained by using persistent chains (PCD) with $k = 50$. **Right:** Error over the learned couplings w.r.t. the true ones as a function of the training time t for different values of k. A comparison with a BM trained using persistent chains (PCD) is also shown. The learning rate is set to $\gamma = 10^{-2}$.

we show in Fig. 5 in the supplementary material the error in the first ten eigenvalues of the covariance matrix computed either with the real or the synthetic samples, just as shown for the Gaussian model in Fig. 2–right. Also in this case, the evolution of the error in the different eigenvalues in the early stages of learning has a minimum at $k' \neq k$, as predicted, but the location of this peak does not necessarily coincide for the different eigenvalues. The early peaks are not as pronounced as in the Gaussian case because we do not have access to the best possible sampling time with discrete times, and the number of chains is relatively small. As training progresses, these best quality peaks gradually collapse towards $k' = k$ for all eigenvalues.

Finally, in Fig. 4–right, we show the error in the inferred couplings $\boldsymbol{J}$ for different values of k as a function of training time. These curves illustrate how very short times k (e.g. $k = 1$) can lead to very poor inferred models, while models learned with short ($k \geq 2$) nonconvergent chains can still catch important information about the data that is useful for interpretive applications.

6. Conclusions

Our results show that EBMs trained with non-persistent and non-convergent MCMCs to estimate the log-likelihood gradient are models with memory of their training scheme. We also show analytically how to exploit such memory effects to generate high-quality samples in a very short time, through a very precise sampling process that must perfectly match that of the training: i.e., same update rules, initializations and sampling time. EBMs trained with the contrastive divergence recipe fall into this category. However, they have the disadvantage that they cannot be used to generate new data because they provide minimal variation in the sample used for initialization. In this sense, we show that EBMs trained with short *randomly* initialized MCMC runs can be effectively used as diffusion models for sample generation. On the other hand, we also prove that the equilibrium measure of an EBM trained in this way cannot correctly describe the statistics of the data and should be used with caution for interpretive applications. The lack of convergence of MCMC runs is particularly critical for datasets with an effectively low dimension, i.e., datasets that follow multimodal distributions, as mixing times become excessively long (Béreux et al., 2022; Decelle & Furtlehner, 2021a; Nijkamp et al., 2022) to be thermalized by brute force.

Acknowledgements

G.C, A.D. and B.S. acknowledge financial support by the Comunidad de Madrid and the Complutense University of Madrid (UCM) through the Atracción de Talento programs (Refs. 2019-T1/TIC-13298 and 2019-T1/TIC-12776), the Banco Santander and the UCM (grant PR44/21-29937), and Ministerio de Economía y Competitividad, Agencia Estatal de Investigación and Fondo Europeo de Desarrollo Regional (Ref. PID2021-125506NA-I00). E.A. acknowledges support from the Swiss National Science Foundation by the SNSF Ambizione Grant No. PZ00P2_173962, and from the Simons Foundation (Grant No. 348126 to Sidney Nagel). B.S. acknowledges the support of the Agence Nationale de la Recherche (Ref. ANR CPJ-ESE 2022).

References

Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. A learning algorithm for boltzmann machines. *Cognitive science*, 9 (1):147–169, 1985.

Baronchelli, A., Caglioti, E., and Loreto, V. Measuring complexity with zippers. *European Journal of Physics*, 26(5):S69, jul 2005. doi: 10.1088/0143-0807/26/5/S08. URL https://dx.doi.org/10.1088/0143-0807/26/5/S08.

Béreux, N., Decelle, A., Furtlehner, C., and Seoane, B. Learning a restricted boltzmann machine using biased monte carlo sampling. *arXiv preprint arXiv:2206.01310*, 2022.

Carleo, G. and Troyer, M. Solving the quantum many-body problem with artificial neural networks. *Science*, 355(6325):602–606, 2017. doi: 10.1126/science.aag2302. URL https://www.science.org/doi/abs/10.1126/science.aag2302.

Decelle, A. and Furtlehner, C. Exact training of restricted boltzmann machines on intrinsically low dimensional data. *Phys. Rev. Lett.*, 127:158303, Oct 2021a. doi: 10.1103/PhysRevLett.127.158303. URL https://link.aps.org/doi/10.1103/PhysRevLett.127.158303.

Decelle, A. and Furtlehner, C. Restricted boltzmann machine: Recent advances and mean-field theory. *Chinese Physics B*, 30(4):040202, 2021b.

Decelle, A., Fissore, G., and Furtlehner, C. Spectral dynamics of learning in restricted boltzmann machines. *Europhysics Letters*, 119(6):60001, nov 2017. doi: 10.1209/0295-5075/119/60001. URL https://dx.doi.org/10.1209/0295-5075/119/60001.

Decelle, A., Fissore, G., and Furtlehner, C. Thermodynamics of restricted boltzmann machines and related learning dynamics. *Journal of Statistical Physics*, 172 (6):1576–1608, 2018. doi: https://doi.org/10.1007/s10955-018-2105-y.

Decelle, A., Furtlehner, C., and Seoane, B. Equilibrium and non-equilibrium regimes in the learning of restricted boltzmann machines. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. doi: 10.48550/ARXIV.2105.13889. URL https://openreview.net/forum?id=Bq_RoftLEeN.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

Du, Y. and Mordatch, I. Implicit generation and modeling with energy based models. *Advances in Neural Information Processing Systems*, 32, 2019.

Du, Y., Li, S., and Mordatch, I. Compositional visual generation with energy based models. *Advances in Neural Information Processing Systems*, 33:6637–6647, 2020.

Dyer, M., Greenhill, C., and Ullrich, M. Structure and eigenvalues of heat-bath markov chains. *Linear Algebra and its Applications*, 454:57–71, 2014.

Fissore, G., Decelle, A., Furtlehner, C., and Han, Y. Robust multi-output learning with highly incomplete data via restricted boltzmann machines. In *Proceedings of the 9th European Starting AI Researchers' Symposium 2020*. arXiv, 2019. doi: 10.48550/ARXIV.1912.09382. URL https://arxiv.org/abs/1912.09382.

Gabrié, M., Tramel, E. W., and Krzakala, F. Training restricted Boltzmann machine via the Thouless-Anderson-Palmer free energy. In *Advances in neural information processing systems*, pp. 640–648, 2015.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Guo, H., Kara, K., and Zhang, C. Layerwise systematic scan: Deep boltzmann machines and beyond. In *International Conference on Artificial Intelligence and Statistics*, pp. 178–187. PMLR, 2018.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf.

Hinton, G. Products of experts. In *1999 Ninth International Conference on Artificial Neural Networks ICANN 99. (Conf. Publ. No. 470)*, volume 1, pp. 1–6 vol.1, 1999. doi: 10.1049/cp:19991075.

Hinton, G. E. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002. doi: 10.1162/089976602760128018.

Hjelm, R. D., Calhoun, V. D., Salakhutdinov, R., Allen, E. A., Adali, T., and Plis, S. M. Restricted

boltzmann machines for neuroimaging: An application in identifying intrinsic networks. *NeuroImage*, 96:245–260, 2014. ISSN 1053-8119. doi: https://doi.org/10.1016/j.neuroimage.2014.03.048. URL https://www.sciencedirect.com/science/article/pii/S1053811914002080.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Kappen, H. J. and Rodríguez, F. B. Efficient Learning in Boltzmann Machines Using Linear Response Theory. *Neural Computation*, 10(5):1137–1156, 07 1998. ISSN 0899-7667. doi: 10.1162/089976698300017386. URL https://doi.org/10.1162/089976698300017386.

Keizer, J. On the solutions and the steady states of a master equation. *Journal of Statistical Physics*, 6(2):67–72, 1972.

Kim, T. and Bengio, Y. Deep directed generative models with energy-based probability estimation. *arXiv preprint arXiv:1606.03439*, 2016.

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Kumar, R., Ozair, S., Goyal, A., Courville, A., and Bengio, Y. Maximum entropy generators for energy-based models. *arXiv preprint arXiv:1901.08508*, 2019.

LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., and Huang, F. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Liao, R., Kornblith, S., Ren, M., Fleet, D. J., and Hinton, G. Gaussian-bernoulli rbms without tears. *arXiv preprint arXiv:2210.10318*, 2022.

Lu, Y., Zhu, S.-C., and Wu, Y. Learning frame models using cnn filters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

Melko, R. G., Carleo, G., Carrasquilla, J., and Cirac, J. I. Restricted boltzmann machines in quantum physics. *Nature Physics*, 15(9):887–892, 2019.

Morcos, F., Pagnani, A., Lunt, B., Bertolino, A., Marks, D. S., Sander, C., Zecchina, R., Onuchic, J. N., Hwa, T., and Weigt, M. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–E1301, 2011.

Muntoni, A. P., Pagnani, A., Weigt, M., and Zamponi, F. adabmdca: adaptive boltzmann machine learning for biological sequences. *BMC bioinformatics*, 22(1):1–19, 2021.

Nguyen, H. C. and Berg, J. Bethe–peierls approximation and the inverse ising problem. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(03): P03004, mar 2012. doi: 10.1088/1742-5468/2012/03/P03004. URL https://dx.doi.org/10.1088/1742-5468/2012/03/P03004.

Nguyen, H. C., Zecchina, R., and Berg, J. Inverse statistical problems: from the inverse ising problem to data science. *Advances in Physics*, 66(3):197–261, 2017. doi: 10.1080/00018732.2017.1341604. URL https://doi.org/10.1080/00018732.2017.1341604.

Nijkamp, E., Hill, M., Zhu, S.-C., and Wu, Y. N. Learning non-convergent non-persistent short-run mcmc toward energy-based model. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/2bc8ae25856bc2a6a1333d1331a3b7a6-Paper.pdf.

Nijkamp, E., Hill, M., Han, T., Zhu, S.-C., and Wu, Y. N. On the anatomy of mcmc-based maximum likelihood learning of energy-based models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):5272–5280, Apr. 2020. doi: 10.1609/aaai.v34i04.5973. URL https://ojs.aaai.org/index.php/AAAI/article/view/5973.

Nijkamp, E., Gao, R., Sountsov, P., Vasudevan, S., Pang, B., Zhu, S.-C., and Wu, Y. N. Mcmc should mix: learning energy-based model with neural transport latent space mcmc. In *International Conference on Learning Representations (ICLR 2022)*., 2022.

Ricci-Tersenghi, F. The bethe approximation for solving the inverse ising problem: a comparison with other inference methods. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(08):P08015, aug 2012. doi: 10.1088/1742-5468/2012/08/P08015. URL https://dx.doi.org/10.1088/1742-5468/2012/08/P08015.

Salakhutdinov, R. and Hinton, G. Deep boltzmann machines. In van Dyk, D. and Welling, M. (eds.), *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pp. 448–455, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18

Apr 2009. PMLR. URL https://proceedings.mlr.press/v5/salakhutdinov09a.html.

Smolensky, P. *Information Processing in Dynamical Systems: Foundations of Harmony Theory*, volume 6. 1987. ISBN 9780262291408.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

Suzuki, M. and Kubo, R. Dynamics of the ising model near the critical point. i. *Journal of the Physical Society of Japan*, 24(1):51–60, 1968. doi: 10.1143/JPSJ.24.51.

Tieleman, T. Training restricted Boltzmann machines using approximations to the likelihood gradient. In *Proceedings of the 25th international conference on Machine learning*, pp. 1064–1071, 2008. doi: 10.1145/1390156.1390290.

Tubiana, J., Cocco, S., and Monasson, R. Learning protein constitutive motifs from sequence data. *Elife*, 8:e39397, 2019.

Xie, J., Lu, Y., Zhu, S.-C., and Wu, Y. A theory of generative convnet. In *International Conference on Machine Learning*, pp. 2635–2644. PMLR, 2016.

Xie, J., Zhu, Y., Li, J., and Li, P. A tale of two flows: cooperative learning of langevin flow and normalizing flow toward energy-based model. *arXiv preprint arXiv:2205.06924*, 2022.

Yelmen, B., Decelle, A., Ongaro, L., Marnetto, D., Tallec, C., Montinaro, F., Furtlehner, C., Pagani, L., and Jay, F. Creating artificial human genomes using generative neural networks. *PLOS Genetics*, 17(2):1–22, 02 2021. doi: 10.1371/journal.pgen.1009303. URL https://doi.org/10.1371/journal.pgen.1009303.

Zhu, S. C., Wu, Y., and Mumford, D. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27:107–126, 1998.

A. Finite and discrete time sampling

In Sec. 3 we discussed the non-convergent sampling of generic EBMs assuming time to be continuous. Here, for completeness, we provide the analogous derivation for discrete time k, i.e., *number of Gibbs steps* during sampling.

For simplicity, we assume that the number of possible states of our model is finite, and that they are indexed by a. For finite and discrete time, the *master equation* states that the probability of being at a at a time $k+1$ is as follows

$$p_a^{(k+1)} = p_a^{(k)} + \sum_{b(\neq a)} w_{b\to a}\, p_b^{(k)} - \sum_{b(\neq a)} w_{a\to b}\, p_a^{(k)} \tag{21}$$

where $w_{a\to b}$ is the transition probability of going from state a to b, and is model-dependent. These transition probabilities are independent of k because the process is Markovian. We introduce the vector notation $\boldsymbol{p}^{(k)}$, whose a-th component is $p_a^{(k)}$. The evolution of the probabilities can be described by a *stochastic evolution* matrix $\mathcal{U}(\boldsymbol{\theta})$ with the following elements

$$\mathcal{U}_{ab} = w_{b\to a}\ (a \neq b), \quad \mathcal{U}_{aa} = 1 - \textstyle\sum_{b\neq a} w_{a\to b}, \tag{22}$$

as

$$\boldsymbol{p}^{(k+1)} = \boldsymbol{\mathcal{U}}\, \boldsymbol{p}^{(k)}. \tag{23}$$

The probability distribution of a MCMC process after k sampling steps can then be easily described in terms of the initial distribution $\boldsymbol{p}^{(0)} = \boldsymbol{p}_0$ as

$$\boldsymbol{p}^{(k)} = \boldsymbol{\mathcal{U}}^k\, \boldsymbol{p}_0 = \sum_\alpha c_\alpha \lambda_\alpha^k \boldsymbol{u}_\alpha, = \boldsymbol{u}_0 + \sum_{\alpha>0} c_\alpha \lambda_\alpha^k \boldsymbol{u}_\alpha \tag{24}$$

where $\{\lambda_\alpha\}$ and $\{\boldsymbol{u}_\alpha\}$ are the eigenvalues and eigenvectors of the $\boldsymbol{\mathcal{U}}$ matrix, and c_α is the α-th component of $\boldsymbol{p}^{(0)}$ in the same basis. Since the probability $\boldsymbol{p}^{(k)}$ must be normalized at all k (i.e. $\sum_a p_a^{(k)} = 1$), it is straightforward to check that the largest eigenvalue (namely, λ_0) must be 1, $\sum_a u_{0,a} = 1$ and $c_0 = 1$, and that the rest must be $|\lambda_\alpha| < 1$ for $\alpha \neq 0$ as long as $\mathcal{U}$ is irreducible (i.e. any configuration can be reached from any other in a finite number of steps)[1]. Then

$$\boldsymbol{p}^{(k)} = \boldsymbol{u}_0 + \sum_{\alpha>0} \mathrm{sign}\lambda_\alpha c_\alpha\, e^{-k|\log\lambda_\alpha|} \boldsymbol{u}_\alpha, \tag{25}$$

which means that it is guaranteed that the stochastic dynamics converge to the distribution $\boldsymbol{u}_0$ after a long time. Also, $\kappa_{\mathrm{mix}} = 1/|\log\lambda_1|$ is now the *mixing* time, with $|\lambda_1|$ being the second largest eigenvalue. For the standard choice for the Gibbs sampling, the Heat Bath update, all eigenvalues are nonnegative (Dyer et al., 2014; Guo et al., 2018). As in the continuous case, we can impose $\boldsymbol{u}_0$ to be the model Gibbs distribution $\boldsymbol{p}_{\boldsymbol{\theta}}$ by imposing detailed balance. This means that we can again write the time dependence of the probability in the form

$$\boldsymbol{p}^{(k)} = \boldsymbol{p}_{\boldsymbol{\theta}} + \boldsymbol{\Lambda}_{\boldsymbol{\theta}}^{(k)}(\boldsymbol{p}_0), \tag{26}$$

where $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}$ is now the term $\sum_{\alpha>0}$ in Eq. (25). From there, we can come back to Eq. (8) in the main text.

B. Extension to continuous variables

It is possible to extend the previous results to the case of continuous variables, in particular to analyze the evolution of the distribution of MCMC chains. For simplicity, we will consider the discrete time case.

We consider continuous variables $\boldsymbol{x}$ instead of the discrete set of variables denoted a in the previous section. Since the state space is continuous, the previous state probabilities $\{p^{(k)}(a)\} = \boldsymbol{p}^{(k)}$ are replaced by the probability density $\boldsymbol{p}^{(k)}(\boldsymbol{x})$ at time k. The master equation can now be written as follows

$$p^{(k+1)}(\boldsymbol{x}) = \int d\boldsymbol{x}' w(\boldsymbol{x}|\boldsymbol{x}') p^{(k)}(\boldsymbol{x}')$$

where $w(\boldsymbol{x}|\boldsymbol{x}')$ is the kernel matrix for the transition from state $\boldsymbol{x}'$ to state $\boldsymbol{x}$. We now assume that the dynamics is irreducible and that the transition kernel satisfies the detailed balance conditions

$$p_{\mathrm{eq}}(\boldsymbol{x}') w(\boldsymbol{x}|\boldsymbol{x}') = p_{\mathrm{eq}}(\boldsymbol{x}) w(\boldsymbol{x}'|\boldsymbol{x}),$$

[1] All elements of the $\mathcal{U}$ matrix are positive semi-definite, then, the Perron-Frobenius theorem guarantees that $\lambda_0 = 1$ is non degenerate if $\mathcal{U}$ is irreducible.

where $p_{\rm eq}(\boldsymbol{x})$ will be the Boltzmann distribution of the model, i.e, what we call $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ in the main-text. In such a case, and this is where the subtleties of working with continuous versus discrete variables matters, we can show that the operator w is bounded and self-adjoint with respect to the inner product defined by the equilibrium distribution. These properties imply that the operator can be diagonalized and that the spectrum is real and bounded in $[-1, 1]$, and the highest eigenvalue is $\lambda_{\max} = 1$. We can show that the equilibrium distribution is the eigenvector associated to the highest eigenvalue $\lambda_{\max} = 1$ (as for the case of discrete states in the previous section):

$$\int d\boldsymbol{x}' w(\boldsymbol{x}|\boldsymbol{x}') p_{\boldsymbol{\theta}}(\boldsymbol{x}') = \int d\boldsymbol{x}' w(\boldsymbol{x}'|\boldsymbol{x}) p_{\boldsymbol{\theta}}(\boldsymbol{x}) = p_{\boldsymbol{\theta}}(\boldsymbol{x}).$$

Now, following the same line as in the article, we can decompose any initial probability distribution in the complete basis of continuous eigenfunctions $\{\phi_\alpha(\boldsymbol{x})\}_\alpha$ as

$$p_0(\boldsymbol{x}) = \sum_{\alpha=0}^{\infty} c_\alpha \phi_\alpha(\boldsymbol{x}) = p_{\boldsymbol{\theta}}(\boldsymbol{x}) + \sum_{\alpha=1}^{\infty} c_\alpha \phi_\alpha(\boldsymbol{x}).$$

From this point, by using the repeated application of the master equation to relate the distribution at time k to the initial distribution, we obtain as in the previous section at Eq. (24):

$$p^{(k)}(\boldsymbol{x}) = p_{\boldsymbol{\theta}}(\boldsymbol{x}) + \sum_{\alpha=1}^{\infty} c_\alpha \lambda_\alpha^k \phi_\alpha(\boldsymbol{x}).$$

From this point on, the same derivations apply as in the main text, but as in Eq. (26) the definition of $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}$ is now

$$\boldsymbol{\Lambda}_{\boldsymbol{\theta}}^{(k)}(\boldsymbol{x}; \boldsymbol{p}_0) = \sum_{\alpha=1}^{\infty} c_\alpha \lambda_\alpha^k \phi_\alpha(\boldsymbol{x}). \tag{27}$$

C. Proof of theorem 4.1

We want to analyze the solution of Eq. (19). In order to do that, we define the following function

$$g(J) = J\hat{c}_{\alpha\alpha} - 1 + e^{-2Jk}(1 - J\left\langle x_\alpha^2 \right\rangle_0). \tag{28}$$

We can compute the first and second order derivative

$$g'(J) = \hat{c}_{\alpha\alpha} + e^{-2Jk}\left(2Jk\left\langle x_\alpha^2 \right\rangle_0 - 2k - \left\langle x_\alpha^2 \right\rangle_0\right), \tag{29}$$

$$g''(J) = e^{-2Jk}\left(4k\left\langle x_\alpha^2 \right\rangle_0 + 4k^2 - 4Jk^2\left\langle x_\alpha^2 \right\rangle_0\right). \tag{30}$$

The second order derivative cancels at

$$J_0 = \frac{k\left\langle x_\alpha^2 \right\rangle_0 + 1}{k\left\langle x_\alpha^2 \right\rangle_0}. \tag{31}$$

Therefore for large negative value of J, g'' is positive and decreasing. It becomes negative for $J > J_0$ and then converge towards zero, remaining negative. Thus, g' is an increasing function up to J_0, and then it decreases towards $\hat{c}_{\alpha\alpha}$. We are interested in knowing where the first order derivative crosses $J = 0$. It can be for negative values of J, in that case the dynamics converges to a negative value J^*, or for positive values of J, in that case the dynamics converges to a positive value J^*. Given the variation of the function, it is enough to look at the sign of the first derivative at $J = 0$.

$$g'(0) = 0 \Longleftrightarrow k^* = \frac{\hat{c}_{\alpha\alpha} - \left\langle x_\alpha^2 \right\rangle_0}{2}. \tag{32}$$

D. Dynamical equations with rotations

We consider a simplified case in two dimensions. We consider the eigenspace of the covariance matrix given by two orthogonal vectors $\hat{\boldsymbol{u}}_{1,2}$, and the ones of the weight matrix $\boldsymbol{u}_{1,2}$, and we denote by $\phi(t)$, the angle between the vectors $\hat{\boldsymbol{u}}_1$, $\boldsymbol{u}_1$ ($\hat{\boldsymbol{u}}_2$, $\boldsymbol{u}_2$.respectively)

$$\boldsymbol{u}_1 = \hat{\boldsymbol{u}}_1 \cos(\phi) + \hat{\boldsymbol{u}}_2 \sin(\phi) \tag{33}$$

$$\boldsymbol{u}_2 = -\hat{\boldsymbol{u}}_1 \sin(\phi) + \hat{\boldsymbol{u}}_2 \cos(\phi) \tag{34}$$

When the vectors are not aligned, and using the eigendecomposition of the covariance matrix: $\langle x_i x_j \rangle_{p_\mathcal{D}} = \sum_\gamma \hat{u}_i^\gamma \hat{c}_\gamma \hat{u}_j^\gamma$, we have that

$$\hat{c}_{\alpha\beta} = \sum_{i,j} u_i^\alpha u_j^\beta \langle x_i x_j \rangle_{p_\mathcal{D}} = \sum_{i,j,\gamma} u_i^\alpha u_j^\beta \hat{u}_i^\gamma \hat{c}_\gamma \hat{u}_j^\gamma \tag{35}$$

Therefore we have

$$\hat{c}_{11} = \hat{c}_1 \cos(\phi)^2 + \hat{c}_2 \sin(\phi)^2 \tag{36}$$

$$\hat{c}_{22} = \hat{c}_1 \sin(\phi)^2 + \hat{c}_2 \cos(\phi)^2 \tag{37}$$

$$\hat{c}_{12} = -\hat{c}_1 \cos(\phi)\sin(\phi) + \hat{c}_2 \cos(\phi)\sin(\phi) = \cos(\phi)\sin(\phi)(\hat{c}_2 - \hat{c}_1) \tag{38}$$

Now, given the parametrization of $\boldsymbol{u}_{1,2}$, we can see that

$$\boldsymbol{u}_1 \frac{d\boldsymbol{u}_2}{dt} = -\frac{d\phi}{dt} \tag{39}$$

Using Eq. (14),(15),(17), we obtain the dynamical equations of this simplified system.

E. Supplementary Figures

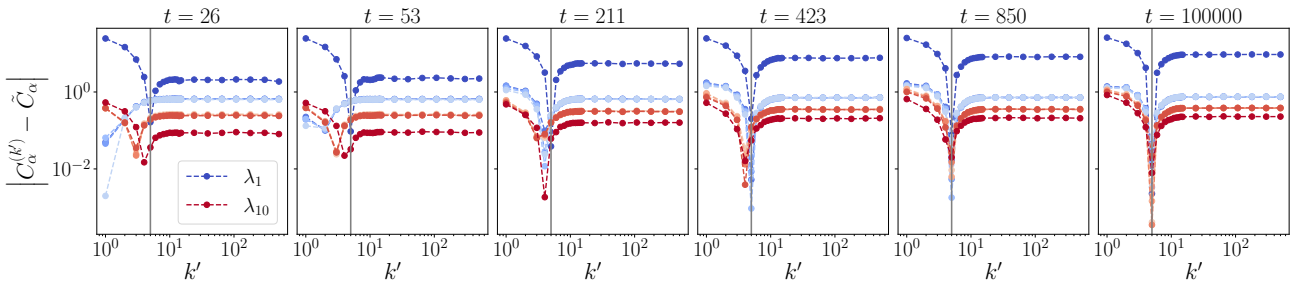

Figure 5. Error over the eigenvalues of the covariance matrix generated after k' steps of MCMC vs the dataset covariance matrix. The setting is the same of Figure 4 in the main text. Each panel shows the element-wise error between the eigenvalues of $\boldsymbol{C}^{(k')}$ and $\tilde{\boldsymbol{C}}$ (at different stages of the learning t shown at the top of each panel), computed for the first 10 largest (in absolute value) eigenvalues. Some eigenvalues are degenerated and superposed.