ON THE INDUCTIVE BIAS OF STOCHASTIC GRADIENT DESCENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Stochastic gradient descent (SGD) is widely believed to perform implicit regularization when used to train deep neural networks, but the precise manner in which this occurs has thus far been elusive. We prove that SGD minimizes an average potential over the posterior distribution of weights along with an entropic regularization term. This potential, however, is in general, not the original loss function, and only depends on the architecture and the data. So SGD does perform variational inference, but for a loss different than the one used to compute the gradients. Even more surprisingly, SGD does not even converge in the classical sense: we show that the most likely trajectories of SGD for deep networks do not behave like Brownian motion around critical points. Instead, they resemble closed loops with deterministic components. We prove that such "out-of-equilibrium" behavior is a consequence of the fact that the gradient noise in SGD is highly non-isotropic; the covariance matrix of mini-batch gradients has a rank as small as 1% of its dimension. We provide extensive empirical validation of these claims, proven in the appendix.

1 INTRODUCTION

Our first result is to show precisely in what sense stochastic gradient descent (SGD) implicitly performs variational inference, as often claimed informally in the literature. For a loss function $f(x)$ with weights $x \in \mathbb{R}^d$, if ρ^{ss} is the steady-state distribution over the weights estimated by SGD,

$$\rho^{\text{ss}} = \arg\min_{\rho} \; \mathbb{E}_{x\sim\rho}\Big[\Phi(x)\Big] - \frac{\eta}{2b}\, H(\rho);$$

where $H(\rho)$ is the entropy of the distribution ρ and η and b are the learning rate and batch-size, respectively. The potential $\Phi(x)$, which we characterize explicitly, is related but not necessarily equal to $f(x)$. It is only a function of the architecture and the dataset. This implies that SGD implicitly performs variational inference with a uniform prior, albeit of a different loss than the one used to compute gradients.

We next prove that the implicit potential $\Phi(x)$ is equal to our chosen loss $f(x)$ if and only if the noise in mini-batch gradients is isotropic. This condition, however, is *not* satisfied for deep networks. Empirically, we find gradient noise to be highly non-isotropic with the rank of its covariance matrix being about 1% of its dimension. Thus, SGD on deep networks implicitly discovers locations where $\nabla\Phi(x) = 0$, these are not the locations where $\nabla f(x) = 0$. This is our second main result: the most likely locations of SGD are not the local minima, nor the saddle points, of the original loss. The deviation of these critical points, which we compute explicitly, can be quite large in practice, and scales linearly with η/b.

Indeed, when mini-batch noise is non-isotropic, SGD does not even converge in the classical sense. We prove that, instead of undergoing Brownian motion in the vicinity of a critical point, trajectories have a deterministic component that causes SGD to traverse closed loops in the weight space. These loops can be non-trivial if the deviation of $\Phi(x)$ from $f(x)$ is large. We detect such loops using a Fourier analysis of SGD trajectories. We also show through an example that SGD with non-isotropic noise can even converge to stable limit cycles around saddle points.

2 BACKGROUND ON CONTINUOUS-TIME SGD

Stochastic gradient descent performs the following updates while training a network

$$x_{k+1} = x_k - \eta \, \nabla f_{\mathscr{b}}(x_k); \tag{SGD}$$

where η is the learning rate and $\nabla f_{\mathscr{b}}(x_k)$ is the average back-propagation (back-prop) gradient over a mini-batch $\mathscr{b}$,

$$\nabla f_{\mathscr{b}}(x) = \frac{1}{\mathscr{b}} \sum_{k \in \mathscr{b}} \nabla f_k(x). \tag{1}$$

We overload notation $\mathscr{b}$ for both the set of examples in a mini-batch and its size. We assume that weights belong to a compact subset $\Omega \subset \mathbb{R}^d$, to ensure appropriate boundary conditions for the evolution of steady-state densities in SGD, although all our results hold without this assumption if the loss grows unbounded as $\|x\| \to \infty$, for instance, with weight decay as a regularizer.

Definition 1 (**Diffusion matrix** $D(x)$)**.** If examples in a mini-batch are sampled with replacement, we show in Appendix A.1 that the variance of mini-batch gradients is $\mathrm{var}\Big(\nabla f_{\mathscr{b}}(x)\Big) = \frac{D(x)}{\mathscr{b}}$ where

$$D(x) = \left(\frac{1}{N} \sum_{k=1}^{N} \nabla f_k(x) \, \nabla f_k(x)^\top \right) - \nabla f(x) \, \nabla f(x)^\top \succeq 0. \tag{2}$$

Note that $D(x)$ is independent of the learning rate η and the batch-size $\mathscr{b}$. It only depends on the weights x, architecture and loss defined by $f(x)$, and the dataset.

We will often discuss two cases:

(i) **isotropic diffusion** when $D(x)$ is a constant multiple of the identity, independent of x, and

(ii) **non-isotropic diffusion**, when $D(x)$ is a general function of the weights x.

We now construct a stochastic differential equation (SDE) for the discrete-time updates in (SGD).

Lemma 2 (**Continuous-time SGD**)**.** *The continuous-time limit of SGD is given by*

$$dx(t) = -\nabla f(x) \, dt + \sqrt{2\beta^{-1} \, D(x)} \, dW(t); \tag{3}$$

where $W(t)$ is a Brownian motion and the parameter β is the inverse temperature defined as

$$\beta^{-1} = \frac{\eta}{2 \, \mathscr{b}}. \tag{4}$$

The steady-state density of the weights $\rho(x,t) \propto \mathbb{P}\big(x(t) = x \,|\, x(0)\big)$, evolves according to the Fokker-Planck equation (Risken, 1996):

$$\frac{\partial \rho}{\partial t} = \nabla \cdot \big(\nabla f(x) \, \rho + \beta^{-1} D(x) \, \nabla \rho \big) \tag{FP}$$

where the notation $\nabla \cdot v$ denotes the divergence $\nabla \cdot v = \sum_i \partial_{x_i} \, v_i(x)$ for any vector $v(x) \in \mathbb{R}^d$.

We refer to Li et al. (2017b, Thm. 1) for the proof but note that (3) and (SGD) are equivalent in the sense that any function computed on trajectories of (3) differs by at most $\mathscr{O}(dt)$ from its value on trajectories of (SGD), the parameter η is identitified with the time discretization dt. Note that β^{-1} completely captures the magnitude of noise in SGD that depends only upon the learning rate η and the mini-batch size $\mathscr{b}$.

Assumption 3 (**Steady-state distribution exists**)**.** We assume that the steady-state distribution of the Fokker-Planck equation (FP) exists, this is denoted by $\rho^{\mathrm{ss}}(x)$ and satisfies,

$$\begin{aligned} 0 = \frac{\partial \rho^{\mathrm{ss}}}{\partial t} &= \nabla \cdot \big(\nabla f(x) \, \rho^{\mathrm{ss}} + \beta^{-1} \, D(x) \, \nabla \rho^{\mathrm{ss}} \big) \\ &\triangleq \nabla \cdot J^{\mathrm{ss}}. \end{aligned} \tag{5}$$

Note that the "probability current" J^{ss} needs not be zero at steady-state, only its divergence does. We will see that the case $J^{\mathrm{ss}} = 0$ corresponds to $D(x)$ being isotropic while the general case $\nabla \cdot J^{\mathrm{ss}} = 0$ will correspond to a non-isotropic diffusion matrix.

3 SGD PERFORMS VARIATIONAL INFERENCE

In this section, we implicitly define the potential $\Phi(x)$ using the steady-state distribution ρ^{ss} as

$$\Phi(x) = -\beta^{-1} \log \rho^{\text{ss}}(x), \tag{6}$$

up to a constant. The potential $\Phi(x)$ depends only on the full-gradient and the diffusion matrix, and will be made explicit in Section 5. For now, we express ρ^{ss} in terms of the potential using a normalizing constant $Z(\beta)$ as

$$\rho^{\text{ss}}(x) = \frac{1}{Z(\beta)}\, e^{-\beta\Phi(x)} \tag{7}$$

which is also the steady-state solution of

$$dx = -D(x)\, \nabla\Phi(x)\, dt + \sqrt{2\beta^{-1}D(x)}\, dW(t) \tag{8}$$

as can be verified by direct substitution in (FP).

The above observation is very useful because it suggests that, if $\nabla f(x)$ can be written in terms of the diffusion matrix and a gradient term $\nabla\Phi(x)$, the steady-state distribution of this SDE is easily obtained. We exploit this observation to rewrite $\nabla f(x)$ in terms of two quantities, a term $D\,\nabla\Phi$ that gives rise to the above steady-state and the remainder, a "conservative "force"

$$j(x) = -\nabla f(x) + D(x)\, \nabla\Phi(x), \tag{9}$$

interpreted as the part of $\nabla f(x)$ that cannot be written as $D\,\Phi'(x)$ for some Φ'. We now make an important assumption on $j(x)$ which has its origins in thermodynamics.

Assumption 4 (**Force $j(x)$ is conservative**). We assume that

$$\nabla \cdot j(x) = 0. \tag{10}$$

The Fokker-Planck equation (FP) typically models a physical system which exchanges energy with an external environment (Ottinger, 2005; Qian, 2014). In our case, this physical system is the gradient descent part $\nabla\cdot(\nabla f\,\rho)$ while the interaction with the environment is the stochastic part $\beta^{-1}\nabla\cdot(D\,\nabla\rho)$. The second law of thermodynamics states that the entropy of a system can never decrease; in Appendix B we show how the above assumption is sufficient to satisfy the second law. We also discuss some properties of $j(x)$ in Appendix D that are a consequence of this. The most important is that, out of the two forces $j(x)$ and $D\,\nabla\Phi$, in (9), only $D\,\Phi(x)$ makes progress towards convergence to the steady-state ρ^{ss}, whereas $j(x)$ is always orthogonal to $\nabla\rho^{\text{ss}}$.

This leads us to the main result of this section.

Theorem 5 (**SGD performs variational inference**). *The functional*

$$F(\rho) = \beta^{-1}\, \mathrm{KL}\big(\rho \,||\, \rho^{\text{ss}}\big) \tag{11}$$

decreases monotonically along the trajectories of the Fokker-Planck equation (FP) and converges to its minimum, which is zero, at steady-state. Moreover, we also have an energetic-entropic split

$$F(\rho) = \mathbb{E}_{x\in\rho}\left[\Phi(x)\right] - \beta^{-1}H(\rho) + \text{constant}. \tag{12}$$

Theorem 5, proven in Appendix E.1 shows that SGD implicitly minimizes a combination of two terms: an "energetic" term, and an "entropic" term. The first is the average potential over a distribution ρ. The steady-state of SGD in (7) is such that it places most of its probability mass in regions of the parameter space with small values of Φ. The second shows that SGD has an implicit bias towards solutions that maximize the entropy of the distribution ρ.

Note that the energetic term in (12) has potential $\Phi(x)$, instead of $f(x)$. This is an important fact and the crux of this paper.

Lemma 6 (**Potential equals original loss iff isotropic diffusion**). *If the diffusion matrix $D(x)$ is isotropic, i.e., a constant multiple of the identity, the implicit potential is the original loss itself*

$$D(x) = c\, I_{d\times d} \quad \Leftrightarrow \quad \Phi(x) = f(x). \tag{13}$$

This is proven in Appendix E.2. The definition in (9) shows that $j \neq 0$ when $D(x)$ is non-isotropic. This results in a deterministic component in the SGD dynamics which does not affect the functional $F(\rho)$, hence $j(x)$ is called a "conservative force." The following lemma is proven in Appendix E.3.

Lemma 7 (**Deterministic dynamics**). *The force $j(x)$ does not decrease $F(\rho)$ in (12) and introduces a deterministic component in SGD given by*

$$\dot{x} = j(x) \tag{14}$$

The condition $\nabla \cdot j(x) = 0$ in Assumption 4 implies that this dynamic makes SGD traverse closed trajectories in weight space.

3.1 Connection to Bayesian inference

Note the absence of any prior in (12). On the other hand, the evidence lower bound (Kingma and Welling, 2013) for the dataset Ξ is,

$$\begin{aligned} -\log p(\Xi) &\leq \mathbb{E}_{x\sim q}\big[f(x)\big] + \mathrm{KL}\Big(q(x\,|\,\Xi) \;||\; p(x\,|\,\Xi)\Big), \\ &\leq \mathbb{E}_{x\sim q}\big[f(x)\big] - H(q) + H(q,p); \end{aligned} \tag{ELBO}$$

where $H(q,p)$ is the cross-entropy of the estimated steady-state and the variational prior. The implicit loss function of SGD in (12) therefore corresponds to a uniform prior $p(x\,|\,\Xi)$. In other words, we have shown that SGD itself performs variational optimization with a uniform prior. Note that this prior is well-defined because $x \in \Omega$ and Ω is compact.

It is important to note that SGD implicitly minimizes a potential $\Phi(x)$ instead of the original loss $f(x)$ in (ELBO). We prove in Section 5 that this potential is quite different from $f(x)$, in particular with respect to critical points.

Remark 8 (**Information in representations**). Working with ELBO in practice involves one or multiple steps of SGD to minimize the energetic term along with an analytically computable KL-divergence term which is often achieved using a factored Gaussian prior (Kingma and Welling, 2013). As Theorem 5 shows, such an approach implicitly also enforces a uniform prior whose strength is determined by β^{-1} and conflicts with the externally imposed Gaussian prior in ELBO.

This conflict, which fundamentally arises from using SGD to minimize the energetic term, has resulted in researchers modulating the strength of the Gaussian prior (Mandt et al., 2016) in ELBO using a pre-factor in the KL-divergence term. This has also been shown to lead to invariant representations (Achille and Soatto, 2017) via the information bottleneck principle (Tishby et al., 1999).

3.2 Practical implications

We will show in Section 5 that the potential $\Phi(x)$ does not depend upon the optimization process, it is only a function of the dataset and the architecture. Therefore, the effect of two important parameters, the learning rate η and the mini-batch size $\mathscr{b}$ completely determines the strength of the entropic regularization term. If $\beta^{-1} \to 0$, the implicit regularization of SGD goes to zero. This implies that

$$\beta^{-1} = \frac{\eta}{2\mathscr{b}} \text{ should not be small}$$

is a good tenet for regularization of SGD.

Remark 9 (**Learning rate should scale linearly with batch-size to generalize well**). In order to maintain the same level of entropic regularization, the learning rate η needs to scale linearly with the batch-size $\mathscr{b}$. This theoretical prediction fits very well with empirical evidence wherein one obtains good generalization performance only with small mini-batches in deep networks (Keskar et al., 2016), or via such linear scaling (Goyal et al., 2017).

Remark 10 (**Sampling with replacement is better regularized than without replacement**). The diffusion matrix for the case when mini-batches are sampled with replacement is almost the same as that of (2), see Appendix A.2. However, the expression for the inverse temperature shows that

$$\beta'^{-1} = \frac{\eta}{2\mathscr{b}}\left(1 - \frac{\mathscr{b}}{N}\right) \text{ should not be small.}$$

The extra factor of $\left(1-\frac{b}{N}\right)$ in β'^{-1} reduces the entropic regularization in (12). In fact, as $b \to N$, the inverse temperature $\beta' \to \infty$. As a consequence, for the same learning rate η and batch-size b, Theorem 5 predicts that sampling with replacement has better regularization than sampling without replacement. This effect is particularly pronounced at large batch-sizes.

4 EMPIRICAL CHARACTERIZATION OF SGD DYNAMICS

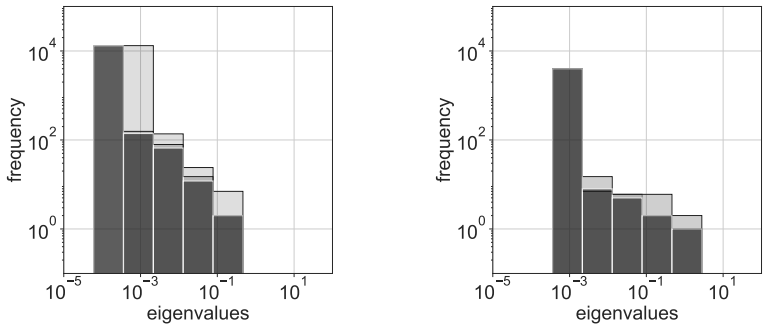
Section 4.1 shows that the diffusion matrix $D(x)$ for modern deep networks is highly non-isotropic with very low rank. We also analyze trajectories of SGD and detect periodic components using a frequency analysis in Section 4.2.

We consider three networks for these experiments: a convolutional network called small-lenet and a two-layer, fully-connected network named small-fc on MNIST (LeCun et al., 1998) and a smaller version of the All-CNN-C architecture of Springenberg et al. (2014) on the CIFAR-10 and CIFAR-100 datasets (Krizhevsky, 2009), see Appendix C for more details.

We use networks with about $20,000$ weights to keep the eigen-decomposition of $D(x) \in \mathbb{R}^{d\times d}$ tractable. These networks however possess all the architectural intricacies such as convolutions, dropout, batch-normalization etc. We evaluate $D(x)$ using (2) with the network in evaluation mode.

4.1 HIGHLY NON-ISOTROPIC $D(x)$ FOR DEEP NETWORKS

Figs. 1 and 2 show the eigenspectrum[1] of the diffusion matrix. In all cases, it has a large fraction of almost-zero eigenvalues with a very small rank that ranges between 0.3% - 2%. Moreover, non-zero eigenvalues are spread across a vast range with a large variance.



(a) MNIST: small-lenet
$\lambda(D) = (0.3 \pm 2.11) \times 10^{-3}$
$\text{rank}(D) = 1.8\%$

(b) MNIST: small-fc
$\lambda(D) = (0.9 \pm 18.5) \times 10^{-3}$
$\text{rank}(D) = 0.6\%$

Figure 1: Eigenspectrum of $D(x)$ at three instants during training (20%, 40% and 100% completion, darker is later). The eigenspectrum in Fig. 1b for the fully-connected network has a much smaller rank and much larger variance than the one in Fig. 1a which also performs better on MNIST. This indicates that convolutional networks are better conditioned than fully-connected networks in terms of $D(x)$.

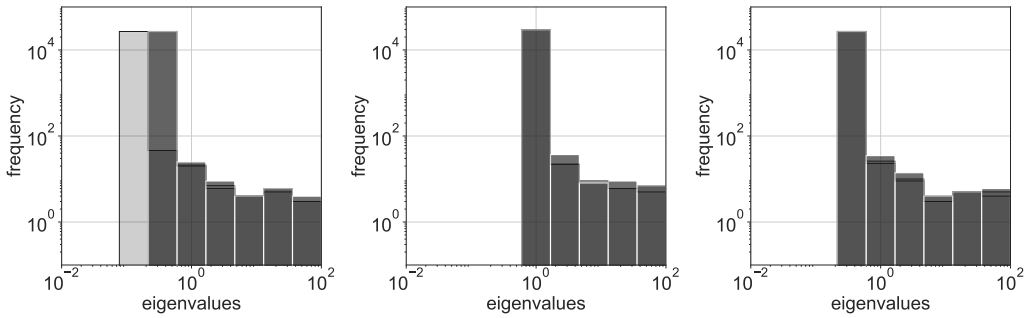
Remark 11 (**Noise in SGD is largely independent of the weights**). The variance of noise in (3) is

$$\frac{\eta\, D(x_k)}{b} = 2\,\beta^{-1} D(x_k).$$

We have plotted the eigenspectra of the diffusion matrix in Fig. 1 and Fig. 2 at three different instants, 20%, 40% and 100% training completion; they are almost indistinguishable. This implies that the variance of the mini-batch gradients in deep networks can be considered a constant, highly non-isotropic matrix.

Remark 12 (**More non-isotropic diffusion if data is diverse**). The eigenspectra in Fig. 2 for CIFAR-10 and CIFAR-100 have much larger eigenvalues and standard-deviation than those in Fig. 1,

[1] thresholded at $\lambda_{\max} \times d \times$ machine-precision. This formula is widely used, for instance, in numpy.



(a) CIFAR-10
$\lambda(D) = 0.27 \pm 0.84$
$\text{rank}(D) = 0.34\%$

(b) CIFAR-100
$\lambda(D) = 0.98 \pm 2.16$
$\text{rank}(D) = 0.47\%$

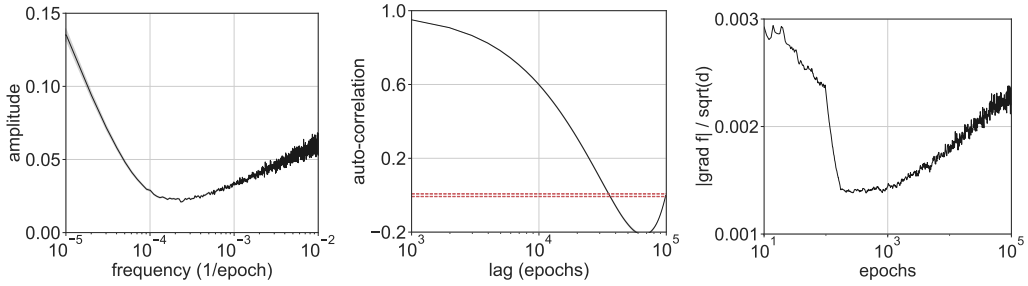
(c) CIFAR-10: data augmentation
$\lambda(D) = 0.43 \pm 1.32$
$\text{rank}(D) = 0.32\%$

Figure 2: Eigenspectrum of $D(x)$ at three instants during training (20%, 40% and 100% completion, darker is later). The eigenvalues are much larger in magnitude here than those of MNIST in Fig. 1, this suggests a larger gradient diversity for CIFAR-10 and CIFAR-100. The diffusion matrix for CIFAR-100 in Fig. 2b has larger eigenvalues and is more non-isotropic and has a much larger rank than that of Fig. 2a; this suggests that gradient diversity increases with the number of classes. As Fig. 2a and Fig. 2c show, augmenting input data increases both the mean and the variance of the eigenvalues while keeping the rank almost constant.

this is expected because the images in the CIFAR datasets have more variety than those in MNIST. Similarly, while CIFAR-100 has qualitatively similar images as CIFAR-10, it has $10\times$ more classes and as a result, it is a much harder dataset. This correlates well with the fact that both the mean and standard-deviation of the eigenvalues in Fig. 2b are much higher than those in Fig. 2a. Input augmentation increases the diversity of mini-batch gradients. This is seen in Fig. 2c where the standard-deviation of the eigenvalues is much higher as compared to Fig. 2a.

4.2 Analysis of long-term trajectories

We train a smaller version of small-fc on 7×7 down-sampled MNIST images (cf. Appendix C) for 10^5 epochs and store snapshots of the weights after each epoch to get a long trajectory in the weight space. We discard the first 10^3 epochs of training ("burnin") to ensure that SGD has reached the steady-state. The learning rate is fixed to 10^{-3} for the remaining 10^5 epochs.



(a) FFT of $x^i_{k+1} - x^i_k$

(b) Auto-correlation (AC) of x^i_k

(c) Normalized gradient $\frac{\|\nabla f(x_k)\|}{\sqrt{d}}$

Figure 3: Fig. 3a shows the Fast Fourier Transform (FFT) of $x^i_{k+1} - x^i_k$ where k is the number of epochs and i denotes the index of the weight. Fig. 3b shows the auto-correlation of x^i_k with 99% confidence bands denoted by the dotted red lines. Both Figs. 3a and 3b show the mean and one standard-deviation over the weight index i; the standard deviation is very small which indicates that all the weights have a very similar frequency spectrum. Figs. 3a and 3b should be compared with the FFT of white noise which should be flat and the auto-correlation of Brownian motion which quickly decays to zero, respectively. Figs. 3 and 3a therefore show that trajectories of SGD are not simply Brownian motion. Moreover the gradient at these locations is quite large (Fig. 3c).

Iterates of SGD after it reaches a neighborhood of a critical point $\|\nabla f(x_k)\| \leq \varepsilon$ are expected to perform Brownian motion with variance $\text{var}(\nabla f_{\theta}(x))$, the FFT in Fig. 3a would be flat if this were so.

Instead, we see low-frequency modes in the trajectory that are indicators of a periodic dynamics of the force $j(x)$. These modes are not sharp peaks in the FFT because $j(x)$ can be a non-linear function of the weights thereby causing the modes to spread into all dimensions of x. The FFT is dominated by jittery high-frequency modes on the right with a slight increasing trend; this suggests the presence of colored noise in SGD at high-frequencies.

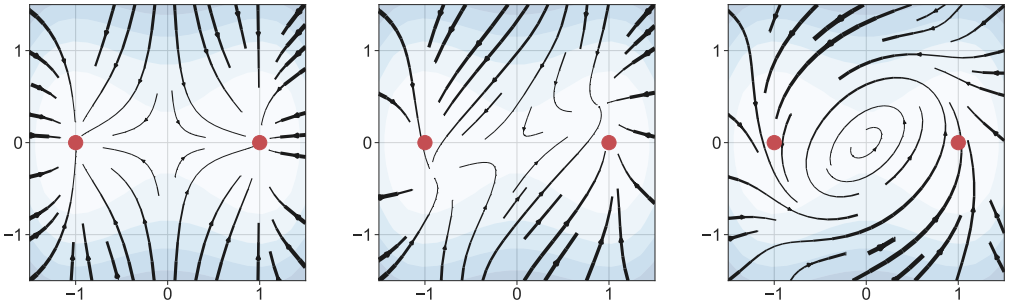
The auto-correlation (AC) in Fig. 3b should be compared with the AC for Brownian motion which decays to zero very quickly and stays within the red confidence bands (99%). Our iterates are significantly correlated with each other even at very large lags. This further indicates that trajectories of SGD do not perform Brownian motion.

Fig. 3c shows that the full-gradient computed over the entire dataset (without burnin) does not decrease much with respect to the number of epochs. While it is expected to have a non-zero gradient norm because SGD only converges to a neighborhood of a critical point for non-zero learning rates, the magnitude of this gradient norm is quite large. This magnitude drops only by about a factor of 3 over the next 10^5 epochs. The presence of a non-zero $j(x)$ also explains this, it causes SGD to be away from critical points, this phenomenon is made precise in Theorem 17. Let us note that a similar plot is also seen in Shwartz-Ziv and Tishby (2017) for the per-layer gradient magnitude.

5 SGD FOR DEEP NETWORKS IS OUT-OF-EQUILIBRIUM

This section now gives an explicit formula for the implicit potential $\Phi(x)$ that SGD creates. We also discuss implications of this for generalization in Section 5.3.

The fundamental difficulty in obtaining an explicit expression for Φ is that even if the diffusion matrix $D(x)$ is full-rank, there need not exist a function $\Phi(x)$ such that $\nabla\Phi(x) = D^{-1}(x)\,\nabla f(x)$ at all $x \in \Omega$. This precludes us from exploiting (8). We therefore split the analysis into two cases: (i) a local analysis near any critical point $\nabla f(x) = 0$ where we linearize $\nabla f(x) = Fx$ and $\nabla\Phi(x) = Ux$ to compute $U = G^{-1}\,F$, and (ii) the general case which pertains to Fig. 4c where $\nabla\Phi(x)$ cannot be written as a local rotation and scaling of $\nabla f(x)$. We introduce these cases with an example from Noh and Lee (2015).



(a) $\lambda = 0$ (b) $\lambda = 0.5$ (c) $\lambda = 1.5$

Figure 4: Gradient field for the dynamics in Example 13: line-width is proportional to the magnitude of the gradient $\|\nabla f(x)\|$, red dots denote the most likely locations of the steady-state $e^{-\Phi}$ while the potential Φ is plotted as a contour map. The critical points of $f(x)$ and $\Phi(x)$ are the same in Fig. 4a, namely $(\pm 1, 0)$, because the force $j(x) = 0$. For $\lambda = 0.5$ in Fig. 4b, locations where $\nabla f(x) = 0$ have shifted slightly as predicted by Theorem 17. The force field also has a distinctive rotation component, see Remark 15. In Fig. 4c with a large $\|j(x)\|$, SGD converges to limit cycles around the saddle point at the origin. This is highly surprising and demonstrates that the solutions obtained by SGD may be very different from local minima.

Example 13 (Double-well potential with limit cycles). Define $\Phi(x) = \frac{(x_1^2-1)^2}{4} + \frac{x_2^2}{2}$. Instead of constructing a diffusion matrix $D(x)$, we will directly construct different gradients $\nabla f(x)$ that lead to the same potential Φ; these are equivalent but the later is much easier. The dynamics is given by $dx = -\nabla f(x)\,dt + \sqrt{2}\,dW(t)$ where $\nabla f(x) = -j(x) + \nabla\Phi(x)$. We pick $j = \lambda e^{\Phi}\,J^{\text{ss}}(x)$ for some parameter $\lambda > 0$ where $J^{\text{ss}}(x) = e^{-\frac{(x_1^2+x_2^2)^2}{4}}\,(-x_2,\, x_1)$. Note that this satisfies (7) and does not change $\rho^{\text{ss}} = e^{-\Phi}$. Fig. 4 shows the gradient field $f(x)$ along with a discussion.

5.1 LINEARIZATION AROUND A CRITICAL POINT

Without loss of generality, let $x = 0$ be a critical point of $f(x)$. This critical point can be a local minimum, maximum, or even a saddle point. We linearize the gradient around the origin and define a fixed matrix $F \in \mathbb{R}^{d\times d}$ (the Hessian) to be $\nabla f(x) = Fx$. Let $D = D(0)$ be the constant diffusion matrix matrix. The dynamics in (3) can now be written as

$$dx = -Fx\,dt + \sqrt{2\beta^{-1}\,D}\,dW(t). \tag{15}$$

Lemma 14 (**Linearization**). *The matrix F in (15) can be uniquely decomposed into*

$$F = (D+Q)\,U; \tag{16}$$

D and Q are the symmetric and anti-symmetric parts of a matrix G with $GF^\top - FG^\top = 0$, to get

$$\Phi(x) = \frac{1}{2}x^\top Ux. \tag{17}$$

The above lemma is a classical result if the critical point is a local minimum, i.e., if the loss is locally convex near $x = 0$; this case has also been explored in machine learning before (Mandt et al., 2016). We refer to Kwon et al. (2005) for the proof that linearizes around any critical point.

Remark 15 (**Rotation of gradients**). We see from (17) that locally near a critical point,

$$\nabla\Phi = (D+Q)^{-1}\,\nabla f \quad \text{and} \quad j(x) = -Q\,\nabla\Phi(x). \tag{18}$$

This suggests that the effect of $j(x)$ is to rotate the local gradient field, this is also seen in Fig. 4b.

5.2 GENERAL CASE

The key idea in the general case is that all the expressions in Lemma 14 and (18) are kept the same but with the matrices now dependent on the weights x

$$\nabla f(x) = \Big(D(x) + Q(x)\Big)\,\nabla\Phi(x) - j_{\text{off}}(x). \tag{19}$$

We have again decomposed the gradient $\nabla f(x)$ into the part that comes from a potential $\nabla\Phi(x)$ and the extra term $j_{\text{off}}(x)$. It is already clear from (19) that the critical points of $f(x)$ are different from the most likely locations of ρ^{ss} if $j_{\text{off}}(x) \neq 0$.

Assumption 16 (**Steady-state does not change**). As done in Ao et al. (2007), we will assume that the force $j_{\text{off}}(x)$ does not change the steady-state for the SDE in (3), i.e., we still have $\rho^{\text{ss}} \propto e^{-\beta\Phi}$. This effectively amounts to assuming that the symmetric matrix $D(x)$ and the anti-symmetric matrix $Q(x)$ captures most of $\nabla f(x)$ that results from a potential $\Phi(x)$. The assumption above is often used in non-equilibrium studies because it explains experimental data in physics and biology (Wang et al., 2008) and its implications have been supported by perturbation-based calculations (Kwon and Ao, 2011) along with more theoretical analyses (Tel et al., 1989; Qian, 2015; Kaiser et al., 2017). It also explains our experimental data from Section 4.2.

The main result of this section now follows.

Theorem 17 (**Most likely locations are not the critical points of the loss**). *The most likely locations of SGD where $\nabla\Phi(x) = 0$ are different from the critical points of the original loss $\nabla f(x) = 0$, More precisely, for j_{off} defined in (19),*

$$j_{\text{off}}(x) = \beta^{-1}\,\Big(\nabla\cdot Q(x)\Big)^\top, \tag{20}$$

where the divergence operator is applied to the matrix $Q(x)$ column-wise. The matrix $Q(x)$ and the potential $\Phi(x)$ can be explicitly computed in terms of the gradient $\nabla f(x)$ and the diffusion matrix $D(x)$. In particular, $\Phi(x)$ does not depend on β.

See Appendix E.4 for the proof. The time spent by a Markov chain at a state x is proportional to its steady-state distribution $\rho^{\text{ss}}(x)$. While it is easily seen that SGD does not converge in the Cauchy sense due to the stochasticity, it is very surprising that it may spend a significant amount of time away from the critical points of the original loss. If $Q(x)$ has a large divergence, the set of states with

$\nabla\Phi(x) = 0$ might be drastically different than those with $\nabla f(x) = 0$. The example in Fig. 4c also demonstrates this effect. If the diffusion matrix $D(x)$ is such that the magnitude of the conservative force $j(x)$ is large, the trajectories can be highly non-trivial; as Fig. 4 demonstrates, SGD may even converge around a saddle point.

This also closes the logical loop we began in Section 3 where we assumed the existence of ρ^{ss} and defined the potential Φ using it. Lemma 14 and Theorem 17 show that both can be defined uniquely in terms of the original quantities, *i.e.*, the gradient term $\nabla f(x)$ and the diffusion matrix $D(x)$. There is no ambiguity as to whether the potential $\Phi(x)$ results in the steady-state $\rho^{\text{ss}}(x)$ or vice-versa.

Remark 18 (Consistent with the linear case). Theorem 17 presents a picture that is completely consistent with Lemma 14. If $j(x) = 0$ and $Q(x) = 0$, or if Q is a constant like the linear case in Lemma 14, the divergence of $Q(x)$ in (20) is zero. This lends credence to Assumption 16.

Remark 19 (Out-of-equilibrium effect can be large even if D is constant). The presence of a $Q(x)$ with non-zero divergence is the consequence of a non-isotropic $D(x)$ and it persists even if D is constant and independent of weights x. So long as D is not isotropic, as we discussed in the beginning of Section 5, there need not exist a function $\Phi(x)$ such that $\nabla\Phi(x) = D^{-1}\,\nabla f(x)$ at all x. This is also seen in our experiments, the diffusion matrix is almost constant with respect to weights for deep networks, but consequences of out-of-equilibrium behavior are still seen in Section 4.2.

Remark 20 (Out-of-equilibrium effect increases with β^{-1}). The effect predicted by (20) becomes more pronounced if $\beta^{-1} = \frac{\eta}{2b}$ is large. In other words, small batch-sizes or high learning rates cause SGD to be drastically out-of-equilibrium. Theorem 5 also shows that as $\beta^{-1} \to 0$, the implicit entropic regularization in SGD vanishes. Observe that these are exactly the conditions under which we typically obtain good generalization performance for deep networks (Keskar et al., 2016; Goyal et al., 2017). This suggests that non-equilibrium behavior in SGD is crucial to obtain good generalization performance, especially for high-dimensional models such as deep networks where such effects are expected to be more pronounced.

5.3 Generalization

It was found both analytically and empirically that solutions of discrete learning problems like perceptrons that generalize well belong to dense clusters in the weight space (Baldassi et al., 2015; 2016). Such dense clusters are exponentially fewer as compared to isolated solutions. To exploit these observations, the authors proposed a loss called "local entropy" that is out-of-equilibrium by construction and can in turn be used to find these well-generalizable solutions more easily. This same idea has also been highly successful in deep learning. Keskar et al. (2016); Chaudhari et al. (2016) found that weights with good generalization error have very low local curvature, i.e., they belong to "wide minima". The latter authors modified SGD to seek such solutions to obtain improvements in generalization performance as well as convergence rate (Chaudhari et al., 2017a).

Local entropy is a smoothed version of the original loss $f(x)$,

$$f_\gamma(x) = -\log\left(G_\gamma * e^{-f(x)}\right); \tag{21}$$

where G_γ is a Gaussian kernel of variance $\gamma > 0$. Even with an isotropic diffusion matrix $D(x)$, the steady-state distribution with $f_\gamma(x)$ as the loss function is $\rho^{\text{ss}}_\gamma(x) \propto e^{-\beta f_\gamma(x)}$. For large values of γ, the new loss $f_\gamma(x)$ is significantly different from the original loss which makes the original local minima exponentially (in β) less likely. In other words, local entropy does not rely on non-isotropic gradient noise to get out-of-equilibrium behavior, it gets it by construction. This is also seen in Fig. 4c: if SGD is drastically out-of-equilibrium, it converges around the "wide" saddle point region at the origin which has a small local entropy.

Actively constructing out-of-equilibrium behavior leads to good generalization in practice. Our evidence that SGD on deep networks itself possesses out-of-equilibrium behavior then suggests that SGD for deep networks generalizes well because of such behavior.

6 Related work

SGD, variational inference and implicit regularization: The idea that SGD is related to variational inference has been seen in machine learning before (Duvenaud et al., 2016; Mandt et al., 2016)

under assumptions such as quadratic steady-states; for instance, see Mandt et al. (2017) for methods to approximate steady-states using SGD. Our results here are very different, we would instead like to understand properties of SGD itself. Indeed, in full generality, SGD performs variational inference using a new potential Φ that it implicitly constructs given an architecture and a dataset.

It is widely believed that SGD is an implicit regularizer, see Zhang et al. (2016); Neyshabur et al. (2017); Shwartz-Ziv and Tishby (2017) among others. This belief stems from its remarkable empirical performance. Our results show that such intuition is very well-placed. Thanks to the special architecture of deep networks where gradient noise is highly non-isotropic, SGD helps itself to a potential Φ with properties that lead to both generalization and acceleration.

SGD and noise: Noise is often added in SGD to improve its behavior around saddle points for non-convex losses, see Lee et al. (2016); Anandkumar and Ge (2016); Ge et al. (2015). It is also quite indispensable for training deep networks (Hinton and Van Camp, 1993; Srivastava et al., 2014; Kingma et al., 2015; Gulcehre et al., 2016; Achille and Soatto, 2016). There is however a disconnect between these two directions due to the fact that while adding external gradient noise helps in theory, it works poorly in practice (Neelakantan et al., 2015; Chaudhari and Soatto, 2015). Instead, "noise tied to the architecture" works better in practice, e.g., Dropout, noisy neurons or small mini-batches. Our results close this gap and show that SGD crucially leverages the highly degenerate noise induced by the architecture.

Yin et al. (2017) construct a scalar measure of the gradient diversity given by $\sum_k \|\nabla f_k(x)\| / \|\nabla f(x)\|$, and analyze its effect on the maximum allowed batch-size in the context of distributed optimization.

Markov Chain Monte Carlo: MCMC methods that sample from a negative log-likelihood $\Phi(x)$ have employed the idea of designing a force $j = \nabla\Phi - \nabla f$ to accelerate convergence, see Ma et al. (2015) for a thorough survey. We instead compute the potential Φ given ∇f and D, which necessitates the use of techniques from physics. In fact, our results show that very simple algorithms such as SGLD by Welling and Teh (2011) also benefit from the acceleration that their sophisticated counterparts aim for (Ding et al., 2014; Chen et al., 2016), if they use deep networks.

7 Discussion

The continuous-time point-of-view used in this paper gives access to general principles that govern SGD, such analyses are increasingly becoming popular (Wibisono et al., 2016; Chaudhari et al., 2017b). However, in practice, deep networks are trained for only a few epochs with discrete-time updates. Closing this gap is an important future direction. A promising avenue towards this is that for typical conditions in practice such as small mini-batches or large learning rates, SGD converges to the steady-state distribution quickly (Raginsky et al., 2017).

References

Achille, A. and Soatto, S. (2016). Information dropout: learning optimal representations through noise. *arXiv:1611.01353*.

Achille, A. and Soatto, S. (2017). On the emergence of invariance and disentangling in deep representations. *arXiv:1706.01350*.

Anandkumar, A. and Ge, R. (2016). Efficient approaches for escaping higher order saddle points in non-convex optimization. In *COLT*, pages 81–102.

Ao, P., Kwon, C., and Qian, H. (2007). On the existence of potential landscape in the evolution of complex systems. *Complexity*, 12(4):19–27.

Baldassi, C., Borgs, C., Chayes, J., Ingrosso, A., Lucibello, C., Saglietti, L., and Zecchina, R. (2016). Unreasonable effectiveness of learning neural networks: From accessible states and robust ensembles to basic algorithmic schemes. *PNAS*, 113(48):E7655–E7662.

Baldassi, C., Ingrosso, A., Lucibello, C., Saglietti, L., and Zecchina, R. (2015). Subdominant dense clusters allow for simple learning and high computational performance in neural networks with discrete synapses. *Physical review letters*, 115(12):128101.

Chaudhari, P., Baldassi, C., Zecchina, R., Soatto, S., Talwalkar, A., and Oberman, A. (2017a). Parle: parallelizing stochastic gradient descent. *arXiv:1707.00424*.

Chaudhari, P., Choromanska, A., Soatto, S., LeCun, Y., Baldassi, C., Borgs, C., Chayes, J., Sagun, L., and Zecchina, R. (2016). Entropy-SGD: biasing gradient descent into wide valleys. *arXiv:1611.01838*.

Chaudhari, P., Oberman, A., Osher, S., Soatto, S., and Guillame, C. (2017b). Deep Relaxation: partial differential equations for optimizing deep neural networks. *arXiv:1704.04932*.

Chaudhari, P. and Soatto, S. (2015). On the energy landscape of deep networks. *arXiv:1511.06485*.

Chen, C., Carlson, D., Gan, Z., Li, C., and Carin, L. (2016). Bridging the gap between stochastic gradient MCMC and stochastic optimization. In *AISTATS*, pages 1051–1060.

Ding, N., Fang, Y., Babbush, R., Chen, C., Skeel, R., and Neven, H. (2014). Bayesian sampling using stochastic gradient thermostats. In *NIPS*, pages 3203–3211.

Duvenaud, D., Maclaurin, D., and Adams, R. (2016). Early stopping as non-parametric variational inference. In *AISTATS*, pages 1070–1077.

Frank, T. D. (2005). *Nonlinear Fokker-Planck equations: fundamentals and applications*. Springer Science & Business Media.

Ge, R., Huang, F., Jin, C., and Yuan, Y. (2015). Escaping from saddle points online stochastic gradient for tensor decomposition. In *COLT*, pages 797–842.

Goyal, P., Dollr, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., and He, K. (2017). Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. *arXiv:1706.02677*.

Gulcehre, C., Moczulski, M., Denil, M., and Bengio, Y. (2016). Noisy activation functions. In *ICML*, pages 3059–3068.

Hinton, G. E. and Van Camp, D. (1993). Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the sixth annual conference on Computational learning theory*, pages 5–13. ACM.

Kaiser, M., Jack, R. L., and Zimmer, J. (2017). Acceleration of convergence to equilibrium in Markov chains by breaking detailed balance. *Journal of Statistical Physics*, 168(2):259–287.

Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., and Tang, P. T. P. (2016). On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv:1609.04836*.

Kingma, D. P., Salimans, T., and Welling, M. (2015). Variational dropout and the local reparameterization trick. In *NIPS*, pages 2575–2583.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv:1312.6114*.

Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. Master's thesis, Computer Science, University of Toronto.

Kwon, C. and Ao, P. (2011). Nonequilibrium steady state of a stochastic system driven by a nonlinear drift force. *Physical Review E*, 84(6):061106.

Kwon, C., Ao, P., and Thouless, D. J. (2005). Structure of stochastic dynamics near fixed points. *Proceedings of the National Academy of Sciences of the United States of America*, 102(37):13029–13033.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lee, J. D., Simchowitz, M., Jordan, M. I., and Recht, B. (2016). Gradient descent only converges to minimizers. In *COLT*, pages 1246–1257.

Li, C. J., Li, L., Qian, J., and Liu, J.-G. (2017a). Batch size matters: A diffusion approximation framework on nonconvex stochastic gradient descent. *arXiv:1705.07562*.

Li, Q., Tai, C., and Weinan, E. (2017b). Stochastic modified equations and adaptive stochastic gradient algorithms. In *ICML*, pages 2101–2110.

Ma, Y.-A., Chen, T., and Fox, E. (2015). A complete recipe for stochastic gradient MCMC. In *NIPS*, pages 2917–2925.

Mandt, S., Hoffman, M., and Blei, D. (2016). A variational analysis of stochastic gradient algorithms. In *ICML*, pages 354–363.

Mandt, S., Hoffman, M. D., and Blei, D. M. (2017). Stochastic gradient descent as approximate bayesian inference. *arXiv:1704.04289*.

Neelakantan, A., Vilnis, L., Le, Q. V., Sutskever, I., Kaiser, L., Kurach, K., and Martens, J. (2015). Adding gradient noise improves learning for very deep networks. *arXiv:1511.06807*.

Neyshabur, B., Tomioka, R., Salakhutdinov, R., and Srebro, N. (2017). Geometry of optimization and implicit regularization in deep learning. *arXiv:1705.03071*.

Noh, J. D. and Lee, J. (2015). On the steady-state probability distribution of nonequilibrium stochastic systems. *Journal of the Korean Physical Society*, 66(4):544–552.

Ottinger, H. (2005). *Beyond equilibrium thermodynamics*. John Wiley & Sons.

Prigogine, I. (1955). *Thermodynamics of irreversible processes*, volume 404. Thomas.

Qian, H. (2014). The zeroth law of thermodynamics and volume-preserving conservative system in equilibrium with stochastic damping. *Physics Letters A*, 378(7):609–616.

Qian, H. (2015). Thermodynamics of the general diffusion process: Equilibrium supercurrent and nonequilibrium driven circulation with dissipation. *The European Physical Journal Special Topics*, 224(5):781–799.

Raginsky, M., Rakhlin, A., and Telgarsky, M. (2017). Non-convex learning via Stochastic Gradient Langevin Dynamics: a nonasymptotic analysis. *arXiv:1702.03849*.

Risken, H. (1996). The Fokker-Planck Equation. Springer.

Santambrogio, F. (2015). Optimal transport for applied mathematicians. *Birkuser, NY*.

Shwartz-Ziv, R. and Tishby, N. (2017). Opening the black box of deep neural networks via information. *arXiv:1703.00810*.

Springenberg, J., Dosovitskiy, A., Brox, T., and Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. *arXiv:1412.6806*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958.

Tel, T., Graham, R., and Hu, G. (1989). Nonequilibrium potentials and their power-series expansions. *Physical Review A*, 40(7):4065.

Tishby, N., Pereira, F. C., and Bialek, W. (1999). The information bottleneck method. In *Proc. of the 37-th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377.

Wang, J., Xu, L., and Wang, E. (2008). Potential landscape and flux framework of nonequilibrium networks: robustness, dissipation, and coherence of biochemical oscillations. *Proceedings of the National Academy of Sciences*, 105(34):12271–12276.

Welling, M. and Teh, Y. W. (2011). Bayesian learning via stochastic gradient Langevin dynamics. In *ICML*, pages 681–688.

Wibisono, A., Wilson, A. C., and Jordan, M. I. (2016). A variational perspective on accelerated methods in optimization. *PNAS*, page 201614734.

Yin, D., Pananjady, A., Lam, M., Papailiopoulos, D., Ramchandran, K., and Bartlett, P. (2017). Gradient diversity empowers distributed learning. *arXiv:1706.05699*.

Yin, L. and Ao, P. (2006). Existence and construction of dynamical potential in nonequilibrium processes without detailed balance. *Journal of Physics A: Mathematical and General*, 39(27):8593.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016). Understanding deep learning requires rethinking generalization. *arXiv:1611.03530*.

A DIFFUSION MATRIX $D(x)$

In this section we denote $g_k := \nabla f_k(x)$ and $g := \nabla f(x) = \frac{1}{N} \sum_{k=1}^{N} g_k$. Although we drop the dependence of g_k on x to keep the notation clear, we emphasize that the diffusion matrix D depends on the weights x.

A.1 WITH REPLACEMENT

Let $i_1, \ldots, i_b$ be b iid random variables in $\{1, 2, \ldots, N\}$. We would like to compute

$$\mathrm{var}\left(\frac{1}{b}\sum_{j=1}^{b} g_{i_j}\right) = \mathbb{E}_{i_1,\ldots,i_b}\left\{\left(\frac{1}{b}\sum_{j=1}^{b} g_{i_j} - g\right)\left(\frac{1}{b}\sum_{j=1}^{b} g_{i_j} - g\right)^\top\right\}.$$

Note that we have that for any $j \neq k$, the random vectors g_{i_j} and g_{i_k} are independent. We therefore have

$$\mathrm{covar}(g_{i_j}, g_{i_k}) = 0 = \mathbb{E}_{i_j,\, i_k}\left\{(g_{i_j} - g)(g_{i_k} - g)^\top\right\}$$

We use this to obtain

$$\begin{aligned}\mathrm{var}\left(\frac{1}{b}\sum_{j=1}^{b} g_{i_j}\right) = \frac{1}{b^2}\sum_{j=1}^{b}\mathrm{var}(g_{i_j}) &= \frac{1}{Nb}\sum_{k=1}^{N}\left((g_k - g)\,(g_k - g)^\top\right) \\ &= \frac{1}{b}\left(\frac{\sum_{k=1}^{N} g_k\, g_k^\top}{N} - g\, g^\top\right).\end{aligned}$$

We will set

$$D(x) = \frac{1}{N}\left(\sum_{k=1}^{N} g_k\, g_k^\top\right) - g\, g^\top. \tag{A1}$$

and assimilate the factor of b^{-1} in the inverse temperature β.

A.2 WITHOUT REPLACEMENT

Let us define an indicator random variable $\mathbf{1}_{i\in b}$ that denotes if an example i was sampled in batch b. We can show that

$$\mathrm{var}(\mathbf{1}_{i\in b}) = \frac{b}{N} - \frac{b^2}{N^2},$$

and for $i \neq j$,

$$\mathrm{covar}(\mathbf{1}_{i\in b}, \mathbf{1}_{j\in b}) = -\frac{b(N-b)}{N^2(N-1)}.$$

Similar to Li et al. (2017a), we can now compute

$$\begin{aligned}\mathrm{var}\left(\frac{1}{b}\sum_{k=1}^{N} g_k\, \mathbf{1}_{k\in b}\right) &= \frac{1}{b^2}\,\mathrm{var}\left(\sum_{k=1}^{N} g_k\, \mathbf{1}_{k\in b}\right) \\ &= \frac{1}{b^2}\sum_{k=1}^{N} g_k\, g_k^\top\,\mathrm{var}(\mathbf{1}_{k\in b}) + \frac{1}{b^2}\sum_{i,j=1,\, i\neq j}^{N} g_i\, g_j^\top\,\mathrm{covar}(\mathbf{1}_{i\in b}, \mathbf{1}_{j\in b}) \\ &= \frac{1}{b}\left(1 - \frac{b}{N}\right)\left[\frac{\sum_{k=1}^{N} g_k\, g_k^\top}{N-1} - \left(1 - \frac{1}{N-1}\right) g\, g^\top\right].\end{aligned}$$

We will again set

$$D(x) = \frac{1}{N-1}\left(\sum_{k=1}^{N} g_k\, g_k^\top\right) - \left(1 - \frac{1}{N-1}\right) g\, g^\top \tag{A2}$$

and assimilate the factor of $b^{-1}\left(1 - \frac{b}{N}\right)$ that depends on the batch-size in the inverse temperature β.

B DISCUSSION ON ASSUMPTION 4

Let $F(\rho)$ be as defined in (12). In non-equilibrium thermodynamics, it is assumed that if the system is not far from equilibrium, i.e., if $\|j(x)\|$ is small, the local entropy production is a product of the force $-\nabla\left(\frac{\delta F}{\delta \rho}\right)$ from (A7) and the probability current $-J(x,t)$ from (FP). This assumption in this form was first introduced by Prigogine (1955), see Frank (2005, Sec. 4.5) for a mathematical treatment. The total entropy increase is given by

$$\beta^{-1}\,\frac{dS_i}{dt} = \int_{x\in\Omega} \nabla\left(\frac{\delta F}{\delta \rho}\right)\,J(x,t)\,dx.$$

This can now be written using (A7) again as

$$\beta^{-1}\,\frac{dS_i}{dt} = \int \rho\,D : \left(\nabla\frac{\delta F}{\delta \rho}\right)\,\left(\nabla\frac{\delta F}{\delta \rho}\right)^{\top} + \int j\rho\left(\nabla\frac{\delta F}{\delta \rho}\right)\,dx.$$

The first term in the above expression is non-negative, in order to ensure that $\frac{dS_i}{dt} \geq 0$, we require

$$\begin{aligned} 0 &= \int j\rho\left(\nabla\frac{\delta F}{\delta \rho}\right)\,dx \\ &= \int \nabla\cdot(j\rho)\,\left(\frac{\delta F}{\delta \rho}\right)\,dx; \end{aligned}$$

where the second equality again follows by integration by parts. It can be shown (Frank, 2005, Sec. 4.5.5) that the condition in Assumption 4, viz., $\nabla\cdot j(x) = 0$, is sufficient to make the above integral vanish and therefore for the entropy generation to be non-negative.

C EXPERIMENTAL SETP

The three networks used in the experiments in Section 4 are as follows.

(i) **small-lenet:** a smaller version of LeNet (LeCun et al., 1998) on MNIST with batch-normalization and dropout (0.1) after both convolutional layers of 8 and 16 output channels, respectively. The fully-connected layer has 128 hidden units. This network has 13338 weights and reaches about 0.75% training and validation error.

(ii) **small-fc:** a fully-connected network with two-layers, batch-normalization and rectified linear units that takes 7×7 down-sampled images of MNIST as input and has 64 hidden units. Experiments in Section 4.2 use a smaller version of this network with 16 hidden units and 5 output classes ($30,000$ input images); this is called **tiny-fc**.

(iii) **small-allcnn:** this a smaller version of the fully-convolutional network for CIFAR-10/100 introduced by Springenberg et al. (2014) with batch-normalization and $12, 24$ output channels in the first and second block respectively. It has $26,982$ weights and reaches about 11% and 17% training and validation errors, respectively.

We train the above networks with SGD with appropriate learning rate annealing and Nesterov's momentum set to 0.9. We do not use any data-augmentation and pre-process data using global contrast normalization with ZCA for CIFAR-10 and CIFAR-100.

D SOME PROPERTIES OF THE FORCE $j(x)$

The Fokker-Planck equation (FP) can be written in terms of the probability current as

$$\begin{aligned} \rho_t^{\text{ss}} &= \nabla\cdot\left(-j\,\rho^{\text{ss}} + D\,\nabla\Phi\,\rho^{\text{ss}} + \beta^{-1}D\,\nabla\rho^{\text{ss}}\right) \\ &= \nabla\cdot\,J^{\text{ss}}. \end{aligned}$$

Since we have $\rho^{\text{ss}} \propto e^{-\beta\Phi(x)}$, we also have that

$$0 = \rho_t^{\text{ss}} = \nabla\cdot\left(D\,\nabla\Phi\,\rho^{\text{ss}} + \beta^{-1}D\,\nabla\rho^{\text{ss}}\right),$$

which gives

$$0 = \nabla \cdot (j\, \rho^{\text{ss}})$$
$$\Rightarrow \quad j(x) = \frac{J^{\text{ss}}(x)}{\rho^{\text{ss}}(x)}. \tag{A3}$$

In other words, the conservative force is non-zero only if detailed balance is broken, i.e., $J^{\text{ss}} \neq 0$. We also have

$$\begin{aligned} 0 &= \nabla \cdot (j\, \rho^{\text{ss}}) \\ &= \rho^{\text{ss}} \left(\nabla \cdot j \; - \; j \cdot \nabla \Phi \right), \end{aligned}$$

which shows using Assumption 4 and $\rho^{\text{ss}}(x) > 0$ for all $x \in \Omega$ that $j(x)$ is always orthognal to the gradient of the potential

$$\begin{aligned} 0 &= j(x) \cdot \nabla \Phi(x) \\ &= j(x) \cdot \nabla \rho^{\text{ss}}. \end{aligned} \tag{A4}$$

Using the definition of $j(x)$ in (9), we have detailed balance when

$$\nabla f(x) = D(x)\, \nabla \Phi(x). \tag{A5}$$

E PROOFS

E.1 THEOREM 5

The KL-divergence is non-negative $F(\rho) \geq 0$ with equality if and only if $\rho = \rho^{\text{ss}}$. The expression in (12) follows after writing

$$\log \rho^{\text{ss}} = -\beta \Phi - \log Z(\beta).$$

We now show that $\frac{dF(\rho)}{dt} \leq 0$ with equality only at $\rho = \rho^{\text{ss}}$ when $F(\rho)$ reaches its minimum and the Fokker-Planck equation achieves its steady-state. The first variation (Santambrogio, 2015) of $F(\rho)$ computed from (12) is

$$\frac{\delta F}{\delta \rho}(\rho) = \Phi(x) + \beta^{-1} \big(\log \rho + 1 \big), \tag{A6}$$

which helps us write the Fokker-Planck equation (FP) as

$$\rho_t = \nabla \cdot \left(-\, j\, \rho + \rho\, D\, \nabla \left(\frac{\delta F}{\delta \rho} \right) \right). \tag{A7}$$

Together, we can now write

$$\begin{aligned} \frac{dF(\rho)}{dt} &= \int_{x \in \Omega} \rho_t\, \frac{\delta F}{\delta \rho}\, dx \\ &= \int_{x \in \Omega} \frac{\delta F}{\delta \rho}\, \nabla \cdot (-j\, \rho)\; dx + \int_{x \in \Omega} \frac{\delta F}{\delta \rho}\, \nabla \cdot \left(\rho\, D\, \nabla \left(\frac{\delta F}{\delta \rho} \right) \right) dx. \end{aligned}$$

As we show in Appendix B, the first term above is zero due to Assumption 4. Under suitable boundary condition on the Fokker-Planck equation which ensures that no probability mass flows across the boundary of the domain $\partial \Omega$, after an integration by parts, the second term can be written as

$$\begin{aligned} \frac{dF(\rho)}{dt} &= -\int_{x \in \Omega} \rho\, D : \left(\nabla\, \frac{\delta F}{\delta \rho}(\rho) \right) \left(\nabla\, \frac{\delta F}{\delta \rho}(\rho) \right)^{\top} dx \\ &\leq 0. \end{aligned}$$

In the above expression, $A : B$ denotes the matrix dot product $A : B = \sum_{ij} A_{ij} B_{ij}$. The final inequality with the quadratic form holds because $D(x) \succeq 0$ is a covariance matrix. Moreover, we have from (A6) that

$$\frac{dF(\rho^{\text{ss}})}{dt} = 0.$$

E.2 LEMMA 6

The forward implication can be checked by substituting $\rho^{\text{ss}}(x) \propto e^{-c\,\beta f(x)}$ in the Fokker-Planck equation (FP) while the reverse implication is true since otherwise (A4) would not hold.

E.3 LEMMA 7

The Fokker-Planck operator written as

$$L\,\rho = \nabla \cdot \left(-j\,\rho + D\,\nabla\Phi\,\rho + \beta^{-1} D\,\nabla\rho\right)$$

from (9) and (FP) can be split into two operators

$$L = L_S + L_A,$$

where the symmetric part is

$$L_S\,\rho = \nabla \cdot \left(D\,\nabla\Phi\,\rho + \beta^{-1} D\,\nabla\rho\right) \tag{A8}$$

and the anti-symmetric part is

$$\begin{aligned} L_A\,\rho &= \nabla \cdot (-j\rho) \\ &= \nabla \cdot (-D\,\nabla\Phi\,\rho + \nabla f \rho) \\ &= \nabla \cdot \left(\beta^{-1}\,D\,\nabla\rho + \nabla f \rho\right). \end{aligned} \tag{A9}$$

We first note that L_A does not affect $F(\rho)$ in Theorem 5. For solutions of $\rho_t = L_A\,\rho$, we have

$$\begin{aligned} \frac{d}{dt} F(\rho) &= \int_\Omega \frac{\delta F}{\delta \rho}\,\rho_t\,dx \\ &= \int_\Omega \frac{\delta F}{\delta \rho}\,\nabla \cdot (-j\,\rho)\,dx \\ &= 0, \end{aligned}$$

by Assumption 4. The dynamics of the anti-symmetric operator is thus completely deterministic and conserves $F(\rho)$. In fact, the equation (A9) is known as the Liouville equation (Frank, 2005) and describes the density of a completely deterministic dynamics given by

$$\dot{x} = j(x) \tag{A10}$$

where $j(x) = J^{\text{ss}}/\rho^{\text{ss}} = -\nabla f + D\,\Phi(x)$ from Appendix D.

E.4 THEOREM 17

All the matrices below depend on the weights x; we suppress this to keep the notation clear. Our original SDE is given by

$$dx = -\nabla f\,dt + \sqrt{2\beta^{-1}\,D}\,dW(t).$$

The matrix G again satisfies the the same conditions as (16) that it is the symmetric part of D and $\nabla\Phi(x) = G^{-1}\,\nabla f(x)$ is the gradient of a scalar function, i.e., its curl is zero:

$$\begin{aligned} D &= \frac{G + G^\top}{2} \\ 0 &= \nabla \times \left(G^{-1}\,\nabla f(x)\right). \end{aligned} \tag{A11}$$

The curl is an anti-symmetric matrix $(\nabla \times v)_{ij} = \partial_i v_j - \partial_j v_i$ for any vector field v. Note that G is defined uniquely using the two conditions in (A11). The first condition that D is the symmetric part of G adds $d(d+1)/2$ constraints while the second condition on the curl adds another $d(d-1)/2$ constraints.

We will transform the original SDE into a new SDE

$$G\,dx = -\nabla\Phi\,dt + \sqrt{2\beta^{-1}\,S}\,dW(t) \tag{A12}$$

where S and A are the symmetric and anti-symmetric parts of G^{-1},

$$S = \frac{G^{-1} + G^{-T}}{2},$$
$$A = G^{-1} - S.$$

Since the two SDEs above are equal to each other, both the deterministic and the stochastic terms have to match. This gives

$$\nabla f(x) = G\, \nabla \Phi(x).$$

Using the above expression, we can now give an explicit, although formal expression for the potential as

$$\Phi(x) = \int_0^1 \left(G^{-1}(\Gamma(s))\, \nabla f(\Gamma(s)) \right) \cdot d\Gamma(s) \tag{A13}$$

where $\Gamma : [0,1] \to \Omega$ is any curve such that $\Gamma(1) = x$ and $\Gamma(0) = x(0)$ which is the initial condition of the dynamics in (3). Note that $\Phi(x)$ does not depend on β because $G(x)$ does not depend on β.

Now *define* the difference

$$Q = G - D.$$

This is really Assumption 16 in action, such a Q may not be anti-symmetric. This is a technical trick, not uncommon in physics, and its validation comes from experimental evidence of its implications.

We now write the modified SDE (A12) as a second-order Langevin system after introducing a velocity variable p with $q \triangleq x$ and mass m:

$$\begin{aligned} dq &= \frac{p}{m}\, dt \\ dp &= -\,(S+A)\, \frac{p}{m}\, dt - \nabla_q \Phi(q)\, dt + \sqrt{2\beta^{-1}\, S}\, dW(t). \end{aligned} \tag{A14}$$

The key idea in Yin and Ao (2006) is to compute the Fokker-Planck equation of the system above and take its zero-mass limit. The steady-state distribution of this equation, which is also known as the Klein-Kramer's equation, is

$$\rho^{\text{ss}}(q,p) = \frac{1}{Z'(\beta)}\, \exp\left(-\beta\, \Phi(q) - \frac{\beta p^2}{2m} \right); \tag{A15}$$

where the position and momentum variables are decoupled. The zero-mass limit is given by

$$\begin{aligned} \rho_t &= \nabla \cdot G \left(\nabla \Phi\, \rho + \beta^{-1}\, \nabla \rho \right), \\ &= \nabla \cdot \left(D\, \nabla \Phi\, \rho + Q\, \nabla \Phi\, \rho + (D+Q)\, \beta^{-1}\, \nabla \rho \right) \\ &= \nabla \cdot \left(D\, \nabla \Phi\, \rho + Q\, \nabla \Phi\, \rho \right) + \nabla \cdot \left(D\, \beta^{-1}\, \nabla \rho \right) + \beta^{-1} \underbrace{\nabla \cdot (Q\ \nabla)\ \rho}_{*} \end{aligned} \tag{A16}$$

We now exploit the fact that Q is defined to be an anti-symmetric matrix. Rewrite the third term on the last step $(*)$ as

$$\begin{aligned} \nabla \cdot (Q\, \nabla \rho) &= (\nabla \cdot Q)\ \cdot (\nabla \rho) \\ &= \sum_{ij} \partial_j \left(\frac{\partial Q_{ij}}{\partial x_i}\, \rho \right) \\ &= -\sum_{ij} \partial_j \left(\frac{\partial Q_{ji}}{\partial x_i}\, \rho \right) \\ &= -\nabla \cdot \left(\nabla \cdot Q^{\top} \right) \rho. \end{aligned} \tag{A17}$$

We have assumed that the steady-state distribution of (3) exists and this also means that the potential $\Phi = -\beta^{-1} \log \rho^{\text{ss}}$ exists. Now consider the Fokker-Planck equation for the force decomposed like (19),

$$\begin{aligned} \rho_t &= \nabla \cdot \left(\nabla f\, \rho + \beta^{-1}\, D\, \nabla \rho \right) \\ &= \nabla \cdot \left(D\, \nabla \Phi\, \rho + Q\, \nabla \Phi\, \rho - j_{\text{off}}\, \rho + \beta^{-1}\, D\, \nabla \rho \right). \end{aligned} \tag{A18}$$

Since both (A15) and (A18) have the same steady-state, we can compare the two equations to get

$$j_{\text{off}} = \beta^{-1}\, \nabla \cdot Q^{\top}. \tag{A19}$$