On Transformations in Stochastic Gradient MCMC

Soma Yokoi\textsuperscript{1,2}, Takuma Otsuka\textsuperscript{3}, Issei Sato\textsuperscript{1,2}
1 The University of Tokyo
2 RIKEN
3 NTT Communication Science Laboratories, NTT Corporation

Abstract

Stochastic gradient Langevin dynamics (SGLD) is a widely used sampler for the posterior inference with a large scale dataset. Although SGLD is designed for unbounded random variables, many practical models incorporate variables with boundaries such as non-negative ones or those in a finite interval. Existing modifications of SGLD for handling bounded random variables resort to heuristics without a formal guarantee of sampling from the true stationary distribution. In this paper, we reformulate the SGLD algorithm incorporating a deterministic transformation with rigorous theories. Our method transforms unbounded samples obtained by SGLD into the domain of interest. We demonstrate transformed SGLD in both artificial problem settings and real-world applications of Bayesian non-negative matrix factorization and binary neural networks.

1 Introduction

Sampling a random variable from a given target distribution is a key problem in Bayesian inference. In this study, we discuss the problem of drawing samples from a target distribution on a bounded domain using the Langevin Monte Carlo (LMC) algorithm. More precisely, let $\theta \sim \pi_{\theta}(\theta)$ be the target random variable in constrained state space $\mathbb{R}_c$ and $\varphi \sim \pi(\varphi)$ be a proxy random variable in $\mathbb{R}$, we discuss the following two-step LMC algorithm:

$$\varphi_{t+1} = \varphi_t + \epsilon \hat{\nabla} \log \pi(\varphi_t) + \sqrt{2\epsilon} \eta,$$

$$\theta_{t+1} = f(\varphi_{t+1}),$$

where $f$ is a transform function that maps the proxy to the target domain and $\hat{\nabla}$ denotes an unbiased stochastic gradient operator. This kind of algorithm is often employed when the target $\theta$ is difficult to directly sample, while its unconstrained counterpart $\varphi$ can be easily obtained. For example, when $\theta$ is non-negative, the exponential function is adopted as mapping $f$. We can see this is a common procedure in recent sampling software and probabilistic programming languages, e.g. Stan (Lee et al., 2017), Edward (Tran et al., 2016), and PyMC (Salvatier et al., 2016). Our goal is to provide a practical algorithm and theoretical justification for this framework.

1.1 Related Work

Efficient sampling via SGLD. Langevin Monte Carlo (LMC) (Roberts and Tweedie, 1996) is a well-known scheme to draw samples from a target distribution $\pi_{\theta}(\theta)$ on the basis of a discretization of the stochastic differential equation (SDE) [Welling and Teh, 2011] applied LMC to the sampling from the posterior distribution given a large dataset. Their method, stochastic gradient Langevin dynamics (SGLD), replaces accurate but expensive evaluation of log-posterior gradient with its stochastic approximation. The similarities in the formulation of SGLD and recent optimization methods (e.g. SGD and RMSprop) have widely attracted attention, and various extensions and theoretical investigations of SGLD have been conducted. Sato and Nakagawa (2014) showed weak
convergence of SGLD in terms of asymptotic invariant measure. For finite-time convergence property, Chen et al. (2015), Teh et al. (2016), Vollmer et al. (2016), and Li et al. (2016) discussed bias-variance and derived mean square error (MSE) of SGLD and its extensions.

Since the error caused by stochastic gradients often makes theoretical analysis challenging, more advanced distances than MSE have been given only for LMC assuming accurate gradients. Durmus and Moulines (2015), Dalalyan (2017), and Dalalyan and Karagulyan (2017) have given finite-sample convergence rates of an LMC algorithm in total variation distance. Dalalyan and Karagulyan (2017) also discussed bounds on the Wasserstein distance for noisy gradients. They stated that their analysis does not provide any interesting results when applied to SGLD, because their upper bound depends on dataset size. They found a valid bound can be obtained if the log-likelihood is modified not to depend on dataset size. However, such modification changes the equilibrium, and therefore fails to sample from the original target distribution.

**SGLD in constrained state space.** The above theoretical analyses have been carried out for LMC and SGLD defined on the whole real numbers. Some modification is necessary when the target variable $\theta$ is defined in a constrained space. This can be seen in many applications such as latent Dirichlet allocation (Blei et al., 2003) where $\theta$ lies in a probability simplex, non-negative matrix factorization (Cemgil, 2009) with all elements of $\theta$ being non-negative, and binary neural networks (Courbariaux et al., 2015) (Hubara et al., 2016) with $\theta \in (-1, 1)$. The difficulty comes from the interpretation that the LMC algorithm is an Euler-Maruyama discretization of the SDE whose equilibrium is a posterior distribution defined on the whole real numbers.

The mirroring trick is a straightforward heuristic to sample bounded random variables. This trick sends back outgoing samples at the domain boundaries so as not to overstep the constraint. Patterson and Teh (2013) employed it to sample from a Gamma distribution defined on $\mathbb{R}_+$, simply taking the absolute value of the generated sample. The mirroring trick is partially justified by Bubeck et al. (2015) and Bubeck et al. (2018). They extended the LMC algorithm to an SDE with a reflecting boundary condition. Their stochastic process defined on a convex body, reflected Brownian motion, is designed to be discretized into an LMC algorithm accompanied by the mirroring trick. This interpretation enables the trick to be theoretically investigated. However, Bubeck et al. (2018) also stated that the extension of their result to SGLD is an open problem for future work.

Brosse et al. (2017) developed another line of research for an LMC algorithm on a convex body. They employed the proximal MCMC (Pereyra, 2016) (Durmus et al., 2018) originally developed for log-concave non-smooth density by taking advantages of the proximal gradient method. Brosse et al. (2017) employed the Moreau-Yosida envelope to find a well-behaved regularization of the target density defined on a convex body so that it preserves convexity and Lipschitzness. This is appropriate for their theoretical investigation, finite-sample bounds in total variation and Wasserstein distance. The sampling distribution is, nevertheless, an unbounded approximation of the target distribution. That is, it still draws samples from outside the domain. Also, the limitation of log-concavity and the computing cost of proximal operator at each sample prevent its application to large datasets as well as complex models such as neural networks.

The above two methods, mirroring trick and proximal MCMC, suffer from crucial drawbacks for practical use. One is the computational efficiency for large datasets: the LMC-based methods require an expensive evaluation of log-posterior gradients over the full dataset for each iteration of MCMC. Another drawback is their inaccurate sampling near the boundaries. Indeed, our preliminary experiments in Figure 1 (see mirror) indicate that the mirroring trick fails to capture the distribution especially when the density is sparse, or concentrated at boundaries. This implies that the sampling may be inaccurate when the model uses a sparse prior that is often employed to avoid overfitting.

The failure of mirroring trick motivates us to transform an unbounded variable into the bounded target domain. While an appropriate choice of transform function satisfies the constraints of target domain, we argue that a naive transformation results in an unstable behavior, as shown in Figure 1 (see naive) and detailed in Section 3.1. To mitigate this pitfall, we carefully design a stochastic gradient-based sampler using a transform function in the form of Eqs. (1) and (2) such that our algorithm accomplishes both the computational efficiency and accuracy in the sampling.
1.2 Contributions

Our contributions are summarized as follows:

- We propose a computationally efficient algorithm that uses stochastic gradients of the target distribution.
- Our formulation is general in that it can be applied to many common transformations, including the exponential, sigmoid, and arctangent functions.
- Our result provides a simple condition of transform functions (Assumption 2 in Section 4.1) to guarantee the stationary distribution and weak convergence even when the algorithm uses stochastic gradients.

We show the following main result for asymptotic invariant measure. Its proof is given in Section 4.

\[ \lim_{t \to \infty} p(\theta, t) = \pi_\theta(\theta), \]  

(3)

with discretization error

\[ \left| \mathbb{E}[h(\tilde{\theta}(T))] - \mathbb{E}[h(\theta(T))] \right| = O(\epsilon_0). \]  

(4)

We show the following main result for asymptotic invariant measure. Its proof is given in Section 4.

\[ \textbf{Theorem 1.} \quad \text{Let transform function } f \text{ satisfy Assumption 2. Then samples } \{\theta_t\}_{t=1}^T \text{ obtained by Algorithm 1 weakly converge to the posterior distribution:} \]

\[ \lim_{t \to \infty} p(\theta, t) = \pi_\theta(\theta), \]  

(3)

with discretization error

\[ \left| \mathbb{E}[h(\tilde{\theta}(T))] - \mathbb{E}[h(\theta(T))] \right| = O(\epsilon_0). \]  

(4)

The rest of this paper is organized as follows. Section 2 reviews SGLD and its weak convergence result for unconstrained random variables. In Section 3, we present two possible modifications of SGLD using a transformation for handling bounded random variables: one based on the direct application of Itô formula (Theorem 4 in Appendix A) and the other based on the change of random variable. We show that the former suffers from instability near the boundary whereas the latter is stable. Section 4 derives Theorem 1 for the latter transformed approach, called change-of-random-variable (CoRV) SGLD. Experimental results are shown in Section 5 using Bayesian non-negative matrix factorization and binary neural networks. Section 6 concludes the paper.

2 Stochastic Gradient Langevin Dynamics

This section reviews the SGLD algorithm in unconstrained state space. We start with defining the stochastic gradient used for computational efficiency in Section 2.1 and describe SGLD and its weak convergence result in Section 2.2.

Note that we use one-dimensional parameters for simplicity, but there is no theoretical reason to prevent us from extending the algorithms to multi-dimensional cases as necessary.

2.1 Stochastic Gradient

Given dataset \( X = \{x_i\}_{i=1}^N \), we consider parameter \( \theta \) with its prior \( p(\theta) \) and a conditionally independent likelihood \( p(X|\theta) = \prod_{i=1}^N p(x_i|\theta) \). We draw samples from an unnormalized form of the posterior \( \pi_\theta(\theta) \propto \exp(-U_\theta(\theta)) \) using the potential

\[ U_\theta(\theta) = -\log p(\theta) - \sum_{i=1}^N \log p(x_i|\theta). \]  

(5)

Assuming the differentiability of potential, the following gradient has been used in classical LMC algorithms:

\[ U'_\theta(\theta) = \frac{d}{d\theta} U_\theta(\theta). \]  

(6)

Since accurate evaluation of the gradient requires \( O(N) \) time complexity, it can be a computational bottleneck for large datasets. Stochastic approximation can be carried out using a small mini-batch \( S \subset X, |S| \ll N \) instead of the full dataset. The approximated potential of \( \theta \) is

\[ \hat{U}_\theta(\theta) = -\log p(\theta) - \frac{N}{|S|} \sum_{x_i \in S} \log p(x_i|\theta). \]  

(7)
The stochastic gradient of potential $U_\theta(\theta)$ is given by
\begin{equation}
\hat{U}_\theta(\theta) = \frac{d}{d \theta} \hat{U}_\theta(\theta).
\end{equation}
(8)

Such replacement of the gradient achieves computational benefit of $O(|S|)$ time at the cost of gradient accuracy. As with existing literature, this paper assumes that the error satisfies the following property.

**Assumption 1** (gradient error). The stochastic gradient $\hat{U}_\theta(\theta)$ is written by using the accurate gradient $U_\theta(\theta)$ and the error $\delta$ as
\begin{equation}
\hat{U}_\theta(\theta) = U_\theta(\theta) + \delta,
\end{equation}
(9)
where $\delta$ is white noise or the Wiener process of zero mean and finite variance satisfying
\begin{equation}
E_S[\delta] = 0, \quad E_S[|\delta|^l] < \infty,
\end{equation}
(10)
for some integer $l \geq 2$. $E_S$ denotes the expectation over sampling set $S$.

2.2 Stochastic Gradient Langevin Dynamics

Let us consider an Itô process described by the SDE,
\begin{equation}
d\theta(t) = a(t, \theta) dt + b(t, \theta) dW(t),
\end{equation}
(11)
where $W(t)$ denotes the Wiener process. When the drift coefficient involves the stochastic gradient $a(t, \theta) = -\hat{U}_\theta(\theta)$ and $b(t, \theta) = \sqrt{2}$, its time-discretization corresponds to the SGLD algorithm. It generates a sample sequence $\{\theta_t\}_{t=1}^T$ with an initial point $\theta_0 \in \mathbb{R}$ by
\begin{equation}
\theta_{t+1} = \theta_t - \epsilon_t \hat{U}_\theta(\theta_t) + \sqrt{2\epsilon_t} \eta_t, \quad \eta_t \sim \mathcal{N}(0, 1),
\end{equation}
(12)
where $\mathcal{N}(0, 1)$ is the standard Gaussian distribution and $\{\epsilon_t\}_{t=1}^T$ is a non-negative stepsize sequence.

Sato and Nakagawa (2014) showed that the sample sequence $\{\theta_t\}_{t=1}^T$ satisfies the following weak convergence.

**Definition 1** (weak convergence (Iacus, 2008)). Let $Y_\epsilon$ be a time-discretized approximation of a continuous-time process $Y$ and $\epsilon_0$ be the maximum time increment of the discretization. $Y_\epsilon$ is said to converge weakly to $Y$ if for any fixed time $T$ and any continuous differentiable and polynomial growth function $h$, it holds true that
\begin{equation}
\lim_{\epsilon \to 0} |E[h(Y_\epsilon(T))] - E[h(Y(T))]| = 0, \forall \epsilon < \epsilon_0
\end{equation}
(13)
with $\epsilon_0 > 0$.

This definition means that the discretization error of SGLD becomes zero in expectation for any fixed time where the time increment approaches zero.

3 Transformed SGLD

In this section, we describe two modifications of LMC for constrained state space based on accurate potential gradients. One in Section 3.1 is a naive application of the Itô formula, or the chain rule in SDEs. We will also see that this can result in inaccurate sampling. The other in Section 3.2 overcomes this problem based on the change-of-random-variable approach. Then we extend the latter to stochastic gradients for data-size scalability.

3.1 Naive Transformation

Here we consider the following two-step modification: first, we use the Itô formula to construct the SDE in the unconstrained domain with the corresponding transform function. Then, the SDE is discretized to obtain the desired algorithm. This derivation is straightforward and theoretically appreciated, as Bubeck et al. (2015), Brosse et al. (2017), and Bubeck et al. (2018) constructed their method.
We transform the following Itô process of $\theta(t)$,
\begin{equation}
    d\theta(t) = a(t, \theta)dt + b(t, \theta)dW(t). \tag{14}
\end{equation}

Let $g : \mathbb{R}_c \to \mathbb{R}$ be a smooth invertible function from a bounded target variable $\theta \in \mathbb{R}_c$ to an unbounded proxy variable $\varphi \in \mathbb{R}$. $\mathbb{R}_c$ is constrained state space, e.g. finite or semi-infinite interval for $\mathbb{R}$, as described in Section 1. We consider a new stochastic process $\varphi(t)$ defined by
\begin{equation}
    \varphi(t) = g(\theta(t)). \tag{15}
\end{equation}

From the Itô formula (Theorem 4 in Appendix A), $\varphi(t)$ is also an Itô process of
\begin{equation}
    d\varphi(t) = \left\{a(t, \varphi)g'(\varphi(t)) + \frac{b^2}{2} g''(\varphi(t))\right\} dt + b(t, \varphi)g'(\varphi(t))dW(t). \tag{16}
\end{equation}

Letting $a(\theta) = -U'_{\theta}(\theta)$ and $b = \sqrt{2}$ and discretizing the process, we obtain the following LMC algorithm
\begin{equation}
    \varphi_{t+1} = \varphi_t + \epsilon \left(-g'(\theta_t)U'_{\theta}(\theta_t) + g''(\theta_t)\right) + \sqrt{2}\epsilon g'(\theta_t)\eta. \tag{17}
\end{equation}

While a general connection between SDE and LMC is discussed by [Ma et al., 2015], this algorithm is distinct in that the transform step $\theta = g^{-1}(\varphi)$ is employed to keep samples in the target domain.

Unfortunately, the method of Eq. (17) is likely to draw inaccurate samples. Figure 1 demonstrates that this method (labeled as naive) fails to track the target density. We attribute this phenomenon to the intrinsic instability around the boundary regardless of the target potential and choice of the

Figure 1: Sampling results from the beta, gamma, and truncated standard normal distributions with $n = 100,000$ samples for each method. The mirroring trick (mirror) often fails at the distribution near boundaries as well as slow mixing due to a small stepsize. The change-of-random-variable formulation (CoRV) works appropriately for these distributions. The stochastic gradients were emulated by adding Gaussian noise to the exact gradients. The stepsize was chosen by the tree-structured Parzen estimator (TPE) [Bergstra et al., 2011] to maximize similarities between the true density functions and the histograms.
Algorithm 1 CoRV SGLD

Require: \( \theta_0 \) and \( \phi_0 = f^{-1}(\theta_0) \)
1: for \( t = 0 \) to \( T - 1 \) do
2: draw a mini-batch and compute \( \hat{U}'_{\theta}(\theta_t) \)
3: \( \phi_{t+1} = \phi_t - \epsilon_t \left( f'(\phi_t) \hat{U}'_{\theta}(\theta_t) - \frac{f''(\phi_t)}{f'(\phi_t)} \right) + \sqrt{2} \epsilon \eta_t \)
4: \( \theta_{t+1} = f(\phi_{t+1}) \)
5: end for
6: return \( \{\theta_t\}_{t=1}^T \)

transform function, which will be explained by Theorem\[ in Section 4.4. This theorem suggests that the stepsize must be small enough to circumvent this divergence, but it would make the sampling substantially slow to mix.

3.2 Change-of-Random-Variable Formulation

We thus introduce another formulation to employ a transformation step in LMC. The derivation methodology is the opposite of the naive transformation; we begin with a discretized algorithm and then construct the corresponding continuous-time SDE. This SDE representation is used for deriving Theorem\[ as detailed in Section 4. In addition, this algorithm overcomes the instability issue unlike the former naive method by Theorem\[ presented in Section 4.4.

Let function \( f : \mathbb{R} \rightarrow \mathbb{R}_c \) be a twice differentiable monotonic function from an unbounded proxy variable \( \phi \in \mathbb{R} \) to a bounded target variable \( \theta \in \mathbb{R}_c \):

\[
\theta = f(\phi). \tag{18}
\]

The target posterior \( \pi_{\theta}(\theta) \) and the proxy density \( \pi(\phi) \) are known to have the following relation,

\[
\pi(\phi) = \pi_{\theta}(\theta) | f'(\phi) |. \tag{19}
\]

For the proxy potential \( U(\phi) = -\log \pi(\phi) \), the gradient \( U'(\phi) \) is represented by the gradient \( U'_{\phi}(\theta) \) of the target potential:

\[
U'(\phi) = f'(\phi) U'_{\phi}(\theta) - \frac{f''(\phi)}{f'(\phi)}. \tag{20}
\]

Following Section 2.1, we can enjoy the computational gain using the stochastic gradient \( \hat{U}'_{\phi} \), and construct the SGLD algorithm for the proxy variable:

\[
\phi_{t+1} = \phi_t - \epsilon_t \left( f'(\phi_t) \hat{U}'_{\phi}(\theta_t) - \frac{f''(\phi_t)}{f'(\phi_t)} \right) + \sqrt{2} \epsilon \eta_t. \tag{21}
\]

We call this algorithm the change-of-random-variable (CoRV) SGLD. The procedures are summarized as Algorithm 1. CoRV SGLD forms a generalized class of samplers that contains the ordinary SGLD. Indeed, we recover SGLD by using the identity function as the transform \( f(\phi) = \phi \). CoRV SGLD satisfies the following advantages.

- The algorithm is computationally efficient. Equation (21) requires only \( \mathcal{O}(|S|) \) time complexity.
- The samples are always in the target constrained space \( \mathbb{R}_c \). Equation (21) generates a proxy sample \( \phi_t \in \mathbb{R} \) and then Eq. (18) transforms it into a target sample \( \theta_t \in \mathbb{R}_c \).
- Any transform functions \( f \) can be employed in Eq. (21) if it is twice differentiable monotonic and \( f'(\phi) \) exists. Many common functions satisfy this condition, such as exponential, sigmoid, and softmax functions.

4 Theoretical Analysis

In this section, we analyze asymptotic properties and discretizing error of CoRV SGLD by deriving Theorem\[ through Sections 4.1 to 4.5.
### Table 1: Transform functions

| Name   | Definition                               | Range     | Asm |
|--------|------------------------------------------|-----------|-----|
| sigmoid | \( \frac{1}{1 + \exp(-\varphi)} \)     | \( \mathbb{R} \rightarrow (0, 1) \) | Yes |
| arctan  | \( \tan^{-1}(\varphi)/\pi + 1/2 \)     | \( \mathbb{R} \rightarrow (0, 1) \) | Yes |
| softsign| \( \varphi/2(1 + |\varphi|) + 1/2 \)   | \( \mathbb{R} \rightarrow (0, 1) \) | Yes |
| exp     | \( \exp(\varphi) \)                    | \( \mathbb{R} \rightarrow \mathbb{R}^+ \) | No  |
| softplus| \( \log(1 + \exp(\varphi)) \)         | \( \mathbb{R} \rightarrow \mathbb{R}^+ \) | Yes |
| ICLL    | \( \varphi - \text{Ei}(-\exp(\varphi)) + \gamma \) | \( \mathbb{R} \rightarrow \mathbb{R}^+ \) | Yes |

The SDE corresponding to our target variable is given as

\[
d\theta(t) = -\hat{U}'(\theta(t))dt + \sqrt{2}dW(t),
\]

but directly drawing samples using this SDE is difficult because \( \theta \) lies in a bounded domain. We consider the following SDE of proxy variable \( \varphi \) as the continuous counterpart of our algorithm in Eq. (21)

\[
d\varphi(t) = -\hat{U}'(\varphi(t))dt + \sqrt{2}dW(t),
\]

so as to apply the tools of stochastic analysis. We will establish the existence and uniqueness of the weak solution and obtain its equilibrium.

The gap between the SDE (23) and the algorithm given in Eq. (21) is that we can directly obtain stochastic gradients of the target potential but not of the proxy potential. We derive Theorem 1 with the following steps: Section 4.1 promises that certain conditions are fulfilled for the existence and uniqueness of the weak solution of the SDE (23). Section 4.2 confirms Algorithm 1 does not break the unique weak solution. Section 4.3 shows the stationary distribution.

### 4.1 Existence and Uniqueness

We assume the following class of transform functions.

**Assumption 2 (transform function).** Let \( f \) be a Lipschitz and monotonically increasing function. Namely, for any \( \varphi \in \mathbb{R} \), there exists constant \( L > 0 \) such that

\[
0 \leq f'(\varphi) \leq L.
\]

The boundary value of target domain denoted by \( \partial S \) corresponds to the infinity in the proxy space: \( \lim_{\varphi \to \infty} f(\varphi) = \partial S \), and \( \lim_{\varphi \to \infty} f'(\varphi) \) exists.

All functions in Table 1 satisfy this assumption except the exponential. Depending on the constraints in the target domain, \( f \) may be a decreasing or upper- and lower-bounded function. Though our discussion also applies to these cases in the same way, we continue with Assumption 2 for simplicity.

The following lemma is essential for showing the proxy potential and the solution of proxy SDE (23).

**Lemma 1 (limit of transform derivative).** Under Assumption 2 we have

\[
\lim_{\varphi \to \infty} f'(\varphi) = 0.
\]

**Proof.** See Appendix B.

Unlike the unconstrained case, a constrained target distribution \( \pi_\theta(\theta) \) often has nonzero density at a domain boundary. The following lemma is required so that the unnormalized proxy distribution \( \int f \exp(-U(\varphi))d\varphi \) does not diverge.

**Lemma 2 (proxy potential).** Let \( f \) satisfy Assumption 2 and a target pdf \( \pi_\theta(\theta) \) have a finite limit as \( \theta \) goes to the boundary. Then for any \( U(\varphi) \), we have

\[
\lim_{\varphi \to \infty} U(\varphi) = \infty.
\]

**Proof.** See Appendix C.
Lemma 2 is enough for some cases (e.g. truncated normal). However, in order to show it for distributions that has infinite density at a boundary (e.g. beta and gamma), we need the following additional assumption.

**Assumption 3.** For $\pi_\theta(\theta)$ of interest, $f$ satisfies
\[
\lim_{\varphi \to \infty} \pi_\theta(f(\varphi)) |f'(\varphi)| = 0.
\] (27)

We check the existence of the solution of the SDE (23). The following result is well-known.

**Lemma 3** (solution existence). Let $\hat{U}'(\varphi)$ be a continuous function of $\varphi$. Then the solution of the SDE (23) exists.

Next, we confirm the uniqueness of the solution. We employ the weak uniqueness (Stroock and Varadhan, 1979) (Theorem 5 in Appendix D) for the uniqueness in the sense of a distribution law. The solution is unique under the following condition of $\hat{U}'(\varphi)$.

**Lemma 4** (solution uniqueness). Let the proxy potential gradient $\hat{U}'(\varphi)$ be a bounded function. Then the solution of the SDE (23) is unique in the sense of a distribution law.

**Proof.** See Appendix E.

### 4.2 Weak Convergence

We also check Algorithm 1 does not break the unique weak solution by confirming that the error of stochastic gradient is bounded.

**Lemma 5** (proxy gradient error). Let $\delta$ be a noise of the stochastic gradient of the target potential that satisfies Assumption 2, and let $f$ satisfy Assumption 2. Then for any noise $\delta_\varphi$ of the stochastic gradient of the proxy potential
\[
\hat{U}'(\varphi) = U'(\varphi) + \delta_\varphi,
\] (28)
we have:
\[
\mathbb{E}[\delta_\varphi] = 0, \quad \mathbb{E}[|\delta_\varphi|^l] < \infty,
\] (29)
for some integer $l \geq 2$.

**Proof.** See Appendix F.

From Lemmas 2 and 5, the weak convergence is derived.

**Theorem 1-a** (weak convergence). Let transform function $f$ satisfy Assumption 2. For any test functions $h$ and $h_\theta$ those are continuous differentiable and polynomial growth, we have:
\[
|\mathbb{E}[h(\hat{\varphi}(T))] - \mathbb{E}[h(\varphi(T))]| = O(\epsilon_0),
\] (30)
and
\[
|\mathbb{E}[h_\theta(\hat{\theta}(T))] - \mathbb{E}[h_\theta(\theta(T))]| = O(\epsilon_0),
\] (31)
where $\varphi(T)$ and $\theta(T)$ denote the random variables at fixed time $T$, $\hat{\varphi}(T)$ and $\hat{\theta}(T)$ denote discretized samples at fixed time $T$ by CoRV SGLD, and $\epsilon_0 > 0$ is the initial stepsize.

**Proof.** See Appendix G.

### 4.3 Stationary Distribution

The distribution of proxy variable is specified as the unique solution of SDE (23) by Lemmas 3 and 4. Finally, we derive its stationary distribution as follows.

**Theorem 1-b** (stationary distribution). Let transform function $f$ satisfy Assumption 2. For transition probability density functions $p(\varphi, t)$ and $p(\theta, t)$ of the variables at time $t$, we have:
\[
\lim_{t \to \infty} p(\varphi, t) = \pi(f(\varphi)) |f'(\varphi)|,
\] (32)
and
\[
\lim_{t \to \infty} p(\theta, t) = \pi_\theta(\theta).
\] (33)

**Proof.** See Appendix H.
4.4 Numerical Stability

This section describes the numerical stability of two transformation approaches in Sections 3.1 and 3.2. First, the following theorem suggests the inaccurate sampling of the naive transformation by showing a divergent behavior near the boundary.

**Theorem 2** (instability of the naive transformation). Let \( f = g^{-1} : \mathbb{R} \to \mathbb{R} \), satisfy Assumption 2. Then for any \( \epsilon > 0 \), and \( U'_\theta(\theta) \), and any \( \theta \in S \) approaching \( \partial S \) from the inside, the single-step difference of the naive transformation method diverges almost surely:

\[
\lim_{\theta \to \partial S} |\varphi_{t+1} - \varphi_t| = \infty
\]

**Proof.** See Appendix I.

By contrast, the following theorem explains the stability of CoRV SGLD by showing that the transformation does not cause an abrupt movement in the dynamics.

**Theorem 3** (stability of CoRV). Let transform function \( f \) satisfy Assumption 2. Then for a gradient error \( \delta \varphi \) and for any \( \theta \in S \) approaching \( \partial S \) from the inside, we have:

\[
\lim_{\theta \to \partial S} \delta \varphi = 0.
\]

**Proof.** See Appendix J.

5 Experiments

In this section, we show the usefulness of our method using a range of models for many application scenarios. Results demonstrate a practical efficacy of our method on top of the theoretical justifications that have been discussed.

5.1 Bayesian NMF

For a typical application that uses a probability distribution supported on a finite or semi-infinite interval, we considered Bayesian non-negative matrix factorization (Cemgil, 2009). In the experiments, we compared three methods: (1) our CoRV, (2) the state-of-the-art SGLD-based method (Ahn et al., 2015) modified for non-negative values, and (3) SGRDL (Patterson and Teh, 2013) using natural gradient with diagonal preconditioning. Methods (2) and (3) used the mirroring trick. We also compared three transform functions for constraining to non-negative variables: exp, softplus, and ICLL in Table 1.

Given the observed \( I \times J \) matrix \( X \), whose components take non-negative discrete values, we approximated it with a low-rank matrix product \( WH \), where \( W \) is \( I \times R \) and \( H \) is \( R \times J \) non-negative matrix. The prior distribution and likelihood are

\[
W_{ir} \sim \text{Exponential}(\lambda_W), \quad H_{rj} \sim \text{Exponential}(\lambda_H),
\]

\[
X_{ij}| W_i, H_j \sim \text{Poisson} \left( \sum_{r=1}^{R} W_{ir}H_{rj} \right),
\]

where \( \lambda_W \) and \( \lambda_H \) are hyper-parameters. See Appendix K for the derivation and algorithms.

We used the MovieLens dataset, a commonly used benchmark for matrix factorization tasks (Ahn et al., 2015). The data matrix consists of \( I = 71,567 \) users and \( J = 10,681 \) items with in total \( 10,000,054 \) non-zero entries. 25% of data points were held out for the test data. We set hyper-parameter \( \lambda_W, \lambda_H \) to 1.0, the number of dimensions of latent variables \( R \) to 20 and 50, and the size of the mini-batch \( |S| \) to 10,000. The root mean square error (RMSE) of the prediction on the test data was used as the performance metric.
Figure 2: Test RMSE of Bayesian non-negative matrix factorization on the MovieLens dataset. Vertical and horizontal axes indicate RMSE and iteration number, respectively. Broken lines (SGLD and SGRLD) used the mirroring trick. Solid lines indicate CoRV method with respective transform functions. CoRV methods consistently outperformed existing methods with quick decrease in the error, while the choice of transform function slightly influenced the performance.

Figure 3: Test loss and accuracy of Bayesian binary neural networks on the MNIST and Fashion-MNIST dataset. The mirroring trick (mirror) showed a slower learning curve for both datasets. The CoRV formulation worked appropriately for the neural networks with the sigmoid, arctan, and softsign transforms. The naive formulation was omitted from the figures due to a significant numerical instability. The cross-entropy loss was evaluated with binarized weights. The accuracy was evaluated with the Bayesian predictive mean computed by a moving average of binarized weights at each epoch. The stepsize was chosen by TPE to minimize the validation loss.

Result. Figure 2 shows the curves of RMSE values as a function of iterations. SGLD and SGRLD are existing methods with the mirroring trick whereas exp, softplus, and ICLL indicates our method with the specified transform function. We observed that CoRV SGLD made better predictions with smaller iterations than the other two algorithms. When $R = 20$ (Figure 2 left), SGLD took 10,000 iterations to reach an RMSE of 0.90, whereas CoRV SGLD (softplus and ICLL) achieves it with only 3,000 iterations. While the choice of transform functions may influence the performance, CoRV outperformed the best performing baseline SGRLD. Our method has a computational overhead regarding the transform function, as discussed in Appendix L. In this experiment, we found that at most 10% computation time was necessary to run our method.

5.2 Bayesian Binary Neural Network

Neural networks have been one of the most active research fields in recent years. A binary neural network, whose parameters are restricted to binary, is expected to achieve high performance on small devices in terms of memory efficiency (Courbariaux et al., 2015) (Hubara et al., 2016). We evaluated each sampling methods through the Bayesian prediction accuracy of binary neural network model. The parameters were trained as continuous variables and binarized at prediction time to construct a Bayesian predictive distribution. The weight parameter $w \in (-1, +1)$ had a prior of translated beta distribution, with hyper-parameter $\alpha, \beta$ and beta function $B(\alpha, \beta)$,

$$p(w) = \frac{1}{2B(\alpha, \beta)} \left( \frac{1}{2} \left( w + \frac{1}{2} \right) \right)^{\alpha-1} \left( -\frac{1}{2} w + \frac{1}{2} \right)^{\beta-1}.$$  (38)
We used the cross-entropy loss of the softmax classifier since we deal with classification problems: MNIST (Lecun et al., 1998) and Fashion-MNIST (Xiao et al., 2017). We used a standard three-layer feed forward network containing 50 hidden units with the ReLU activation. We trained the network for 100 epochs with MNIST and for 300 epochs with Fashion-MNIST dataset. CoRV SGLD using the sigmoid, arctangent, and softsign functions was compared with a standard SGLD with the mirroring trick. The stepsize was chosen by the tree-structured Parzen estimator of 100 trials to minimize the validation loss.

**Results.** Figure 3 presents the loss values and accuracy for each dataset MNIST (left two) and Fashion-MNIST (right two). Note that the purpose of this experiment is to compare sampling methods on the same model rather than to propose a state-of-the-art network. The learning curves show that CoRV achieves better prediction than mirroring heuristics. It is effective in practice that transformation enables stable computation with a large stepsize.

### 6 Conclusion

SGLD has resorted to some heuristics for sampling bounded random variables since SGLD is designed for unbounded ones. We demonstrated such heuristics may sacrifice the sampling accuracy. To deal with such random variables, we generalized SGLD using the change-of-random-variable (CoRV) formulation and analyzed its weak error convergence. Empirical evaluations showed that our CoRV SGLD outperformed existing heuristic alternatives on Bayesian non-negative matrix factorization and neural networks.

### Acknowledgement

Iseei Sato was supported by KAKENHI 17H04693.

### References

S. Ahn, A. Korattikara, N. Liu, S. Rajan, and M. Welling. Large-scale distributed Bayesian matrix factorization using stochastic gradient MCMC. In *KDD*, pages 9–18. ACM, 2015.

J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for hyper-parameter optimization. In *NIPS*, pages 2546–2554. 2011.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *JMLR*, 3:993–1022, 2003.

N. Brosse, A. Durmus, É. Moulines, and M. Pereyra. Sampling from a log-concave distribution with compact support with proximal Langevin Monte Carlo. In *COLT*, volume 65, pages 319–342, 2017.

S. Bubeck, R. Eldan, and J. Lehec. Finite-time analysis of projected Langevin Monte Carlo. In *NIPS*, pages 1243–1251. 2015.

S. Bubeck, R. Eldan, and J. Lehec. Sampling from a log-concave distribution with projected Langevin Monte Carlo. *Discrete & Computational Geometry*, 59(4):757–783, 2018.

A. T. Cemgil. Bayesian inference for nonnegative matrix factorization models. *Intell. Neuroscience*, 2009:4:1–4:17, 2009.

C. Chen, N. Ding, and L. Carin. On the convergence of stochastic gradient MCMC algorithms with high-order integrators. In *NIPS*, pages 2278–2286, 2015.

M. Courbariaux, Y. Bengio, and J. David. Binaryconnect: Training deep neural networks with binary weights during propagations. 2015. arXiv:1511.00363.

A. S. Dalalyan. Theoretical guarantees for approximate sampling from a smooth and log-concave density. *Journal of the Royal Statistical Society, Series B*, 79:651–676, 2017.

A. S. Dalalyan and A. G. Karagulyan. User-friendly guarantees for the Langevin Monte Carlo with inaccurate gradient. 2017. arXiv:1710.00095.

A. Durmus and E. Moulines. Non-asymptotic convergence analysis for the unadjusted Langevin algorithm. 2015. arXiv:1507.05021.
A. Durmus, É. Moulines, and M. Pereyra. Efficient Bayesian computation by proximal Markov chain Monte Carlo: When Langevin meets Moreau. *SIAM Journal on Imaging Sciences*, 11(1):473–506, 2018.

I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio. Binarized neural networks. In *NIPS*, pages 4107–4115, 2016.

S. M. Iacus. *Simulation and inference for stochastic differential equations*. Springer Series in Statistics. Springer-Verlag New York, 2008.

Kiyoshi Itô. Stochastic integral. In *Proc. Imperial Acad. Tokyo*, pages 519–524, 1944.

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

D. Lee, B. Carpenter, P. Li, M. Morris, M. Betancourt, maverickg, M. Brubaker, R. Trangucci, M. Inacio, A. Kucukelbir, Stan buildbot, bgoodri, seantalts, J. Arnold, D. Tran, M. Hoffman, C. Margossian, M. Modrák, A. Adler, K. Sakrejda, A. Stukalov, M. Lawrence, R. J Goedman, K. S. Van Horn, A. Vehtari, J. Gabry, J. S. Casallas, and B. Bales. Stan, 2017. URL https://doi.org/10.5281/zenodo.1101116.

C. Li, C. Chen, D. Carlson, and L. Carin. Preconditioned stochastic gradient Langevin dynamics for deep neural networks. In *AAAI*, pages 1788–1794, 2016.

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

S. Patterson and Y. W. Teh. Stochastic gradient Riemannian Langevin dynamics on the probability simplex. In *NIPS*, pages 3102–3110, 2013.

M. Pereyra. Proximal Markov chain Monte Carlo algorithms. *Statistics and Computing*, 26(4):745–760, 2016.

G. O. Roberts and R. L. Tweedie. Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 12 1996.

J. Salvatier, T. V. Wiecki, and C. Fonnesbeck. Probabilistic programming in python using PyMC3. *PeerJ Computer Science*, 2:e55, 2016.

I. Sato and H. Nakagawa. Approximation analysis of stochastic gradient Langevin dynamics by using Fokker-Planck equation and Ito process. In *ICML*, pages 982–990, 2014.

D.W. Stroock and S.R.S. Varadhan. *Multidimensional Diffusion Processes*. Springer, 1979.

Y. W. Teh, A. H. Thiery, and S. J. Vollmer. Consistency and fluctuations for stochastic gradient Langevin dynamics. *JMLR*, 17(7):1–33, 2016.

D. Tran, A. Kucukelbir, A. B. Dieng, M. Rudolph, D. Liang, and D. M. Blei. Edward: A library for probabilistic modeling, inference, and criticism. *arXiv preprint arXiv:1610.09787*, 2016.

S. J. Vollmer, K. C. Zygalakis, and Y. W. Teh. Exploration of the (non-)asymptotic bias and variance of stochastic gradient Langevin dynamics. *JMLR*, 17(159):1–48, 2016.

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

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. 2017. arXiv:1708.07747.
A Itô Formula

In the stochastic differential equation, we have the following formula.

**Theorem 4 (Itô Formula [Itô, 1944])** \(X(t)\) satisfies the stochastic differential equation

\[
dX(t) = a(t, X(t))dt + b(t, X(t))dW(t). \tag{39}
\]

Let \(h(t, X(t))\) be a given bounded function in \(C^2((0, \infty) \times \mathbb{R})\). Then, \(h(t, X(t))\) satisfies the stochastic differential equation

\[
dh(t, X(t)) = L_1 h(t, X(t))dt + L_2 h(t, X(t))dW(t), \tag{40}
\]

where \(L_1\) and \(L_2\) are linear operators defined by

\[
L_1 = \frac{\partial}{\partial t} + a \frac{\partial}{\partial X} + \frac{1}{2} b^2 \frac{\partial^2}{\partial X^2}, \quad L_2 = b \frac{\partial}{\partial X}. \tag{41}
\]

B Proof of Lemma 1

**Proof.** Using the L'Hôpital's rule,

\[
\lim_{\varphi \to \infty} f' (\varphi) = 0. \tag{43}
\]

C Proof of Lemma 2

**Proof.** From the assumption of the target pdf, and \(\varphi \to \infty\) as \(\theta \to \partial S\),

\[
\lim_{\varphi \to \infty} \pi (f(\varphi)) < C, \tag{44}
\]

for constant \(C > 0\). From Lemma 1,

\[
\lim_{\varphi \to \infty} \pi (f(\varphi)) f'(\varphi) = 0. \tag{45}
\]

Using

\[
U(\varphi) = - \log \left(\pi(\theta) |f'(\varphi)|\right), \tag{46}
\]

we have

\[
\lim_{\varphi \to \infty} U(\varphi) = \infty. \tag{47}
\]

D Weak uniqueness

We have the following condition for the uniqueness of the solution in the sense of a distribution law.

**Theorem 5 (weak uniqueness [Stroock and Varadhan, 1979])** Consider a \(d\)-dimensional SDE of \(X \in \mathbb{R}^d\),

\[
dX(t) = a(X(t))dt + b(X(t))dW(t). \tag{48}
\]

Let \(a(x)\) be a bounded measurable function for \(x \in \mathbb{R}^d\). Let \(B(x) = b(x)^\top b(x)\) be a bounded, continuous function where constant \(K > 0\) exists such that

\[
\sum_{i,j=1}^d B_{ij}(x) \zeta_i \zeta_j \geq K |\zeta|^2, \tag{49}
\]

for \(\zeta = (\zeta_1, \ldots, \zeta_d) \in \mathbb{R}^d\). Then the uniqueness in the sense of a distribution law holds for the solution of the SDE (48).
E Proof of Lemma 4

Proof. From Theorem 5, the condition of the diffusion coefficient is straightforwardly confirmed by letting $b = \sqrt{2}$ and $\zeta \in \mathbb{R}$, there exists constant $K > 0$ such that
\begin{equation}
 b^2\zeta^2 \geq K\zeta^2.
\end{equation}

F Proof of Lemma 5

Proof. Since $\tilde{U}'_\theta(\theta)$ satisfies Assumption 1,
\begin{equation}
 \tilde{U}'_\theta(\theta) = U'_\theta(\theta) + \delta,
\end{equation}
as in Eq. (20), the stochastic gradient of the proxy potential is
\begin{equation}
 \tilde{U}'(\varphi) = f'(\varphi)(U'_\theta(\theta) + \delta) - \frac{f''(\varphi)}{f'(\varphi)} = U'(\varphi) + \delta_\varphi,
\end{equation}
by letting $\delta_\varphi = f'(\varphi)\delta$. Since Assumption 2 suggests that the derivative of transform is always finite, $\delta_\varphi$ also satisfies zero mean and finite variance
\begin{equation}
 \mathbb{E}_S[\delta_\varphi] = 0, \quad \mathbb{E}_S[|\delta_\varphi|^2] < \infty.
\end{equation}

G Proof of Theorem 1-a

Proof. Let us consider stochastic differential equation
\begin{equation}
 d\varphi(t) = a(\varphi(t))dt + b(\varphi(t))dW(t), \quad 0 \leq t \leq T
\end{equation}
and its approximation in time $t_{k-1} \leq t \leq t_k$
\begin{equation}
 d\tilde{\varphi}(t) = \tilde{a}(\varphi(t))dt + \tilde{b}(\varphi(t))dW(t),
\end{equation}
where $\tilde{a}(\varphi(t)) = a(\varphi(t)) + \delta_{\varphi,t}$. Using Lemma 5 and Theorem 6 of Sato and Nakagawa (2014), for the test function $h$, we have
\begin{equation}
 |\mathbb{E}[h(\tilde{\varphi}(T))] - \mathbb{E}[h(\varphi(T))]| = \left| \int_0^T \mathbb{E} \left[ \tilde{a}(\varphi(t)) - a(\varphi(t)) \right] \frac{\partial}{\partial \varphi} \mathbb{E}[h(\tilde{\varphi}(t))] dt 
 + \int_0^T \frac{1}{2} \mathbb{E} \left[ \tilde{b}(\varphi(t))^2 - b(\varphi(t))^2 \right] \frac{\partial^2}{\partial \varphi^2} \mathbb{E}[h(\tilde{\varphi}(t))] dt \right|.
\end{equation}
From the Weierstrass theorem, there exists constant $C_k > 0$ such that
\begin{equation}
 \mathbb{E} \left[ \tilde{a}(\varphi(t)) - a(\varphi(t)) \right] \frac{\partial}{\partial \varphi} \mathbb{E}[h(\tilde{\varphi}(t))] \leq C_k \epsilon_{t_{k-1}}
\end{equation}
and
\begin{equation}
 \mathbb{E} \left[ \tilde{b}(\varphi(t))^2 - b(\varphi(t))^2 \right] \frac{\partial^2}{\partial \varphi^2} \mathbb{E}[h(\tilde{\varphi}(t))] \leq C_k \epsilon_{t_{k-1}}
\end{equation}
for time $t_{k-1} \leq t \leq t_k$. Letting the maximum value of $C_k$ be $C_{\max}$ and $\epsilon_{t_{k-1}}$ be $\epsilon_0$, we have
\begin{equation}
 |\mathbb{E}[h(\tilde{\varphi}(T))] - \mathbb{E}[h(\varphi(T))]| < TC_{\max}^2 \epsilon_0.
\end{equation}
That is, the sample of proxy variable $\varphi$ generated by Eq. (21) weakly converges
\begin{equation}
 |\mathbb{E}[h(\tilde{\varphi}(T))] - \mathbb{E}[h(\varphi(T))]| = O(\epsilon_0).
\end{equation}
Let test function $h$ be a composition of transform function $f$ and test function $h_\theta$ in the target domain: $h(\cdot) = h_\theta(f(\cdot))$. Thus, $h(\varphi(T)) = h_\theta(\theta(T))$ and $h(\tilde{\varphi}(T)) = h_\theta(\tilde{\theta}(T))$. The sample of target variable $\theta$ satisfies
\begin{equation}
 |\mathbb{E}[h(\tilde{\varphi}(T))] - \mathbb{E}[h(\varphi(T))]| = O(\epsilon_0).
\end{equation}
H Proof of Theorem 1-b

Proof. From Lemma 3 and 4, there exists a unique solution in the sense of a distribution law. From Lemma 5, the SDE (23) satisfies the same assumption that Sato and Nakagawa (2014) used for SGLD in unconstrained state space. The transition probability density function \( p(\varphi, t) \) follows the Fokker-Planck equation

\[
\frac{\partial}{\partial t} p(\varphi, t) = -\frac{\partial}{\partial \varphi} \left( U'(\varphi)p(\varphi, t) \right) + \frac{\partial^2}{\partial \varphi^2} p(\varphi, t),
\]

and its stationary distribution is

\[
\lim_{t \to \infty} p(\varphi, t) = \exp(-U(\varphi)) = \pi(\varphi).
\]

Note that \( f'(\varphi(t)) \) is always finite from Assumption 2. Applying Eq. (19), we obtain the stationary distribution as

\[
\lim_{t \to \infty} p(\theta, t) = \pi(\theta) \cdot f'(\varphi).
\]

I Proof of Theorem 2

Proof. From Lemma 1,

\[
\lim_{\varphi \to \infty} \frac{d}{d\varphi} g^{-1}(\varphi) = 0.
\]

This implies that

\[
\lim_{\theta \to \partial S} |g'(\theta)| = \lim_{\varphi \to \infty} \frac{1}{d\varphi g^{-1}(\varphi)} = \infty.
\]

From Eq. (17), the single-step difference is given by

\[
|\varphi_{t+1} - \varphi_t| = |\epsilon_t (-g'(\theta_t) U_0'(\theta_t) + g''(\theta_t)) + \sqrt{2}\epsilon_t g'(\theta_t) \eta_t|.
\]

Considering \( \eta_t \sim \mathcal{N}(0, 1) \), the factor \( g'(\theta) \) almost surely dominates this quantity. Therefore,

\[
\lim_{\theta \to \partial S} |\varphi_{t+1} - \varphi_t| = \infty.
\]

J Proof of Theorem 3

Proof. From Eq. (52), we have

\[
\hat{U}'(\varphi) = U'(\varphi) + \delta_{\varphi},
\]

where \( \delta_{\varphi} = f'(\varphi)\delta \) and \( \delta \) satisfies Assumption 1. From Lemma 1,

\[
\lim_{\theta \to \partial S} \delta_{\varphi} = \lim_{\varphi \to \infty} f'(\varphi)\delta = 0.
\]

K Algorithm for Bayesian NMF

SGLD generated samples as follows using the stochastic gradient evaluated with a mini-batch:

\[
W_{i}^* = W_{i} - \epsilon_t \hat{U}'_{W_i} + \sqrt{2}\eta,
\]
where noise $\eta$ conforms to $\mathcal{N}(0, I)$ with $I$ being the $R \times R$ identity matrix. $W^*_t$ denotes the sample at time $t+1$ given $W_t$, is the sample at time $t$. The absolute value is taken in an element-wise manner, which corresponds to the mirroring trick. The stochastic gradient is

$$\hat{U}^*_W = -\frac{N}{|S|} \sum_{X_k \in S} H_{j_k} \left( \frac{X_k}{\tilde{X}_k} - 1 \right) + \lambda W,$$

where $j_k \in \{1, \cdots, J\}$ is the index of the $k$th data point in mini-batch $S$, $X_k$ is a discrete value of the $k$th data, and $\tilde{X}_k = \sum_{r=1}^R W_{kr} H_{rk}$ is its estimate.

CoRV SGLD updates proxy variables by

$$\varphi^*_W = \varphi W - \epsilon_t \left( f'(\varphi W) \hat{U}^*_W - \frac{f''(\varphi W)}{f'(\varphi W)} \right) + \sqrt{2\eta}.$$

Here $f'$ and $f''$ are applied element-by-element. The sample of $\varphi_H$ is obtained in the same manner. Note that Eq. (73) bypasses the mirroring trick because proxy variables $\varphi_W$ and $\varphi_H$ are in the entire domain $\mathbb{R}$. Matrices $W, H$ are always non-negative via transform $f : \mathbb{R} \rightarrow \mathbb{R}_+$. The algorithms are shown below.

### Algorithm 2 SGLD for Bayesian NMF

- **initialize** $W^{(0)}, H^{(0)}$
- **for** time $t \in 1, \cdots, T$ **do**
  - subsample mini-batch $S_t$ from dataset
  - obtain new sample of $W^{(t)}$ by Eq. (71)
  - obtain new sample of $H^{(t)}$
- **end for**
- output $W^{(1)}, \cdots, W^{(T)}, H^{(1)}, \cdots, H^{(T)}$

### Algorithm 3 Transformed SGLD for Bayesian NMF

- **initialize** $W^{(0)}, H^{(0)}, \varphi_W^{(0)}, \varphi_H^{(0)}$
- **for** time $t \in 1, \cdots, T$ **do**
  - subsample mini-batch $S_t$ from dataset
  - obtain new sample of $\varphi_W^{(t)}$ by Eq. (73)
  - obtain new sample of $\varphi_H^{(t)}$
  - transform $W^{(t)} = f(\varphi_W^{(t)})$
  - transform $H^{(t)} = f(\varphi_H^{(t)})$
- **end for**
- output $W^{(1)}, \cdots, W^{(T)}, H^{(1)}, \cdots, H^{(T)}$

### L Computational Complexity

CoRV SGLD requires additional computation of the transformation step compared to the vanilla SGLD (see Algorithm 3 in Appendix K). In most cases, gradient computation is dominant in the SGLD calculation, which is proportional to the number of data in each mini-batch. CoRV SGLD depends only on the number of parameters and does not change the complexity of gradient computation. The influence on the computation time is limited, as the measured execution time was up to $+10\%$ at the maximum.