Improving Langevin Monte Carlo with Irreversible Diffusion

Sida Li LISTAR2000@UCHICAGO.EDU
Department of Statistics
The University of Chicago

Advisor: Daniel Sanz-Alonso

Abstract

This is a very early draft version of Sida Li's master thesis for the M.S. Statistics program at UChicago. The format, content, and organization of the final version might vary hugely. This version contains necessary context and proofs for the Langevin Monte Carlo (LMC) problem, as well as experiment results up to January 2024.

Keywords: Langevin Monte Carlo, Irreversible Diffusion, Ensemble Methods

1. Monte Carlo Using Langevin Dynamics

1.1 The Langevin Equation

In the field of physics, the Langevin equation, attributed to Paul Langevin, represents a type of stochastic differential equation that models the evolution of a system under the influence of both predictable (deterministic) and unpredictable (random) forces. Consider the one-dimensional case first, one of the commonly used expression is given by

$$\lambda \frac{dX_t}{dt} = -\nabla_x V(X_t) + \eta_t \tag{1}$$

where $X_t \in \mathbb{R}$ denotes the particle position at time t, λ is a fixed damping term that decides the "friction" for the particle movement, $V(x)$ is a potential energy function, and η_t is a noise term, often representing thermal fluctuations.

For practical sampling purpose, we can simplify the damping effect (set $\lambda = 1$) and model η_t as white noise, i.e. the derivative of a 1-d Brownian motion B_t, by

$$\eta_t = \sqrt{2\beta^{-1}} \frac{dB_t}{dt} \tag{2}$$

where $\beta = 1/(k_B T)$ in physics denotes an inverse "temperature" determined by temperature T at thermal equilibrum and the Boltzmann constant. We will show that it is also trivial in our application of interest. Rearranging terms will give us the more familiar (overdamped) Langevin stochastic differential equation (SDE):

$$dX_t = -\nabla_x V(X_t)dt + \sqrt{2\beta^{-1}} dB_t \tag{3}$$

where the negative gradient of the potential serves as the drift term and β controls the magnitude of the diffusion effect.

1.2 The Steady State Distribution

To demonstrate how the SDE in (3) is related to a Monte Carlo sampling method, we need to show that there exists a steady-state distribution p_∞ such that if $X_t \sim p_\infty$ at some time t, then following the above dynamics yields stationary $X_s \sim p_\infty$ for all $s \geq t$.

Recall the 1-d Fokker-Planck equation: for a SDE of the form $dX_t = \mu(X_t, t)dt + \sigma(X_t, t)dB_t$, the probability density of X_t, denoted as $p(x, t)$, satisfies

$$\frac{\partial}{\partial t}p(x,t) = -\frac{\partial}{\partial x}\left[\mu(x,t)p(x,t)\right] + \frac{\partial^2}{\partial^2 x}\left[D(x,t)p(x,t)\right] \tag{4}$$

where $D(x, t) = \sigma^2(x, t)/2$. In our case, we simply have $\mu(x, t) = -\nabla_x V(x)$ and $D(x, t) = \beta^{-1}$. For a steady state distribution p_∞, it needs to satisfy $\frac{\partial}{\partial t}p_\infty(x, t) = 0$. We can plug in a candidate (unnormalized) solution $\hat{p}_\infty(x, t) = \exp\left[-\beta V(x)\right]$ and check

$$\begin{aligned}
\frac{\partial}{\partial t}\hat{p}_\infty(x,t) &= \frac{\partial}{\partial x}\left[\nabla_x V(x)\exp\left[-\beta V(x)\right]\right] + \frac{\partial^2}{\partial^2 x}\left[\beta^{-1}\exp\left[-\beta V(x)\right]\right] && (5)\\
&= \frac{\partial}{\partial x}\left[\nabla_x V(x)\exp\left[-\beta V(x)\right]\right] + \frac{\partial}{\partial x}\left[\frac{\partial}{\partial x}\left[\beta^{-1}\exp\left[-\beta V(x)\right]\right]\right] && (6)\\
&= \frac{\partial}{\partial x}\left[\nabla_x V(x)\exp\left[-\beta V(x)\right]\right] + \frac{\partial}{\partial x}\left[\beta^{-1}(-\beta)\exp\left[-\beta V(x)\right]\nabla_x V(x)\right] && (7)\\
&= \frac{\partial}{\partial x}\left[\nabla_x V(x)\exp\left[-\beta V(x)\right]\right] - \frac{\partial}{\partial x}\left[\nabla_x V(x)\exp\left[-\beta V(x)\right]\right] && (8)\\
&= 0 && (9)
\end{aligned}$$

which verifies that $\hat{p}_\infty(x, t)$ serves as a stationary-state solution and we can drop its dependence on t. Even though a valid probability density also need to integrate to 1, we already know the steady state distribution up to proportionality, i.e.

$$p_\infty(x) \propto \hat{p}_\infty(x) = \exp\left[-\beta V(x)\right] \tag{10}$$

1.3 Sampling from Langevin Dynamics

Consider the sampling problem for a target distribution which is known only up to some unnormalized density $\pi(x) = \exp[-U(x)]/Z$ (Z might be intractable; this is often denoted as an energy-based model, or EBM). Equation (10) shows that if we set $V(x) = U(x)$ and $\beta = 1$, then the steady-state distribution in the Langevin SDE satisfies $p_\infty(x) \propto \exp[-U(x)]$ and thus equals to $\pi(x)$. In general, we can rewrite $\pi(x) = \exp[\log \pi(x)]$ and

$$dX_t = \nabla_x \log \pi(X_t)\, dt + \sqrt{2}\, dB_t \tag{11}$$

yields an SDE with $\pi(x)$ being its steady state solution. An important takeaway is that: combined with the discretization approaches mentioned below, we are able to sample from any EBM while only knowing its score function.

To sample from Eq (11), the SDE must be discretized using some numerical method. A common choice is the **Euler-Maruyama method**, which gives the approximation

$$X_{t+\epsilon} = X_t + \epsilon \nabla_x \log \pi(X_t) + \sqrt{2\epsilon}\xi \tag{12}$$

with ϵ being the step size and $\xi \sim \mathcal{N}(0,1)$ a standard normal noise. When $\epsilon \to 0$, the E-M discretization can approximate the SDE arbitrarily well at the expense of increasing the number of iterations to reach some time t. In the below discussions, we treat ϵ as a fixed hyper-parameter that does not vary over time.

Starting from some initialization X_t (usually $t = 0$), if we directly take $X_{(i)} = X_{t+i\epsilon}$ and collect $\{X_{(1)}, X_{(2)}, ...\}$ as the Monte Carlo samples, we have the **Unadjusted Langevin Algorithm (ULA)**. Theoretically, under certain regularity conditions (e.g. smoothness of $\nabla_x \log \pi(x)$ and bound of ϵ), ULA approximates the target distribution exponentially fast [1]. In practice, however, choosing ϵ can be hard and we might employ the **Metropolis-Hastings algorithm** to accept or reject a proposed sample.

Algorithm 1 Metropolis-adjusted Langevin Algorithm (MALA)

Input: Initial point x_0, step size ϵ, target distribution $\pi(x)$, number of sample N.
Output: A sample $\hat{x}$ from the distribution $\pi(x)$.
procedure MALA(x_0, ϵ, $\pi(x)$)
    $k \leftarrow 0$
    $x_k \leftarrow x_0$
    while $k < N$ **do**
        Sample $\xi \sim \mathcal{N}(0,1)$
        Propose $\hat{x}_{k+1} \leftarrow x_k + \epsilon \nabla \log \pi(x_k) + \sqrt{2\epsilon}\xi$
        Compute acceptance probability α as follows:
        $\alpha \leftarrow \min\left(1, \frac{\pi(\hat{x}_{k+1}) q(x_k|\hat{x}_{k+1})}{\pi(x_k) q(\hat{x}_{k+1}|x_k)}\right)$
        Accept or reject the new sample based on α:
        With probability α, accept $\hat{x}_{k+1}$
        if accept **then**
            $x_{k+1} \leftarrow \hat{x}_{k+1}$
        else
            $x_{k+1} \leftarrow x_k$
        end if
        $k \leftarrow k + 1$
    end while
    return $x_0, x_1, ..., x_{N-1}$
end procedure

where $q(\cdot|\cdot)$ is the proposal density given by

$$q(\hat{x}_{k+1}|x_k) \propto \exp\left(-\frac{\|\hat{x}_{k+1} - x_k - \epsilon \nabla_x \log \pi(x)\|^2}{4\epsilon}\right) \tag{13}$$

which is a natural consequence of the E-M transition dynamics.

1.4 Langevin MC in Higher Dimensions

Previous discussion and equations involve the simple case when the partcile X_t is a 1-d scalar. In general, when we are interested in the sampling from target density $\pi(x) : \mathbb{R}^d \to \mathbb{R}$,

1. I should probably dedicate a section to discuss discretization choices and proofs?

the "vanilla" Langevin SDE extends to

$$dX_t = \nabla_x \log \pi(X_t)\, dt + \sqrt{2}\mathbf{I}_d\, dB_t \tag{14}$$

where B_t is now a d-dimensional Brownian motion. The E-M discretization, ULA and MALA algorithms follow suits and are all similar to the scalar version.

2. A General Recipe for Non-reversible Langevin Diffusion

The vanilla Langevin SDE in (14) can be further recognized as a special case (i.e. the **reversible diffusion**) of a larger framework that incorporates most continuous Markov processes for sampling purpose [2].

$$dX_t = [D + Q]\nabla_x \log \pi(X_t)\, dt + \sqrt{2D}\, dB_t \tag{15}$$

with Q being a skew-symmetric curl matrix and D being a positive semi-definte diffusion matrix. We also assume that both D and Q are fixed across time [3]. It has been shown in **?** that this generalized SDE still preserves $p_\infty(x) \propto \pi(x)$ as a steady-state solution (and the unique solution when the dynamics is ergodic). Therefore, the "vanilla" Langevin dynamics can be viewed as the special case when setting $D = \mathbf{I}_d$ and $Q = \mathbf{0}_d$. The following experiments will demonstrate how different choices of D and Q would affect the convergence speed of the LMC algorithm.

3. Various Preliminary Experiment Results

3.1 Gaussian Mixture Models

Also known as mixture of Gaussians, the **Gaussian Mixture Models (GMMs)** are a form of mixture model wherein the components are assumed to follow a Gaussian distribution, characterized by their means and variances. This assumption allows for a natural way to model datasets with multiple modes, capturing the essence of complex, overlapped data structures through a combination of multiple Gaussian densities.

Consider a GMM with n Gaussian components (assumed uni-variate now), each with mean μ_i and variances σ_i^2, as well as the weights $w_1, ..., w_n$ ($\sum_i w_i = 1$). The log density of this model is:

$$\log \pi(x) = \log \left(\sum_{i=1}^{n} w_i \cdot \phi(x; \mu_i, \sigma_i^2) \right) \tag{16}$$

where ϕ represents the Gaussian pdf. The gradient of log density (i.e. score) is slightly more complicated:

$$\nabla_x \log \pi(x) = \frac{\sum_{i=1}^{n} w_i \cdot \phi(x; \mu_i, \sigma_i^2) \cdot (x - \mu_i)/\sigma_i^2}{\sum_{i=1}^{n} w_i \cdot \phi(x; \mu_i, \sigma_i^2)} \tag{17}$$

this allows us to use LMC to generate samples from GMMs.

2. I might include an additional section talking in details about "reversibility"
3. I don't know whether this assumption should be dropped since, e.g. in the automatic-differentiated optimization case, these matrices are treated as varying parameters

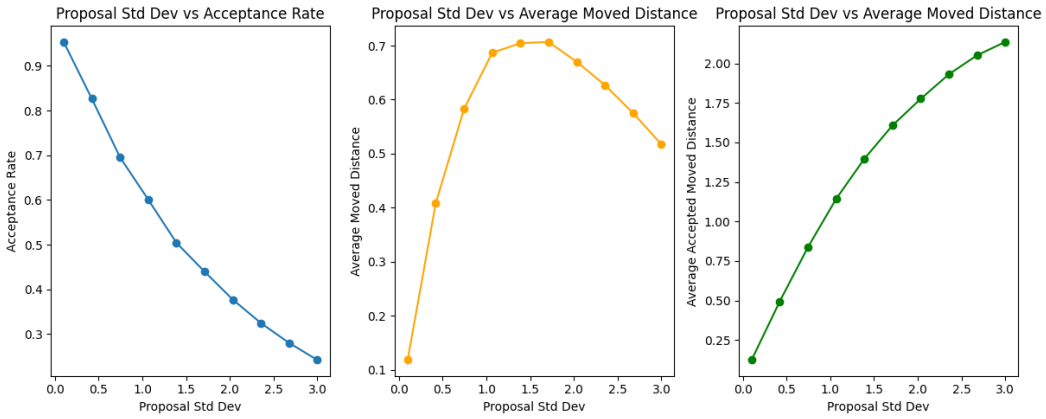


Figure 1: the different metrics for a vanilla MH-MCMC algorithm

Metrics: to measure the "speed" of a Monte Carlo sampling algorithm, we introduce the following metrics that we can use for quantitative comparisons.

1. **Acceptance rate** (ρ): this is the ratio of accepted particles ("transitions") over all proposed particles in the Metropolis-Hastings procedure.

2. **Average (accepted) moved distance** (d): this is the average difference between pairs of proposed state and the previous state, i.e. $d = N^{-1}\sum_{i=1}^{N}\|\hat{X}_{(i)} - X_{(i-1)}\|^2$, that have been accpeted in M-H procedure.

3. **Average moved distance** (d^*): this is the average difference between the accepted state and the previous state, i.e. the actual distance in transitions $d^* = N^{-1}\sum_{i=1}^{N}\|X_{(i)} - X_{(i-1)}\|^2$. If a proposal is rejected, then the actual moved distance is considered 0.

Overall, the acceptance rate and the average moved distance are of the greatest importance – the former measures sampling efficiency and the latter indicates how fast the Markov Chain is transitioning when the proposals are accepted.

Consider a basic example of running the vanilla M-H MCMC algorithm (which is our baseline) for the GMM problem, the three metrics corresponding to different proposal distributions (i.e. zero-mean Gaussian proposals with different standard deviations) are summarized in Figure 1.

Experiment 1: different step sizes and curl matrices for Langevin diffusion

We consider a simple case with $n = 2$ and each Gaussian component is bivariate. The diffusion matrix D is fixed as identity and we explore different curl matrix Q in the form:

$$Q = \begin{bmatrix} \alpha & 0 \\ 0 & -\alpha \end{bmatrix} \tag{18}$$

i.e. parametrized by a scalar α. We run multiple grid-searches over both ϵs and αs and measure the metrics mentioned, each trial with $N = 10000$ samples. One of the grid-searches is displayed in Fig 2

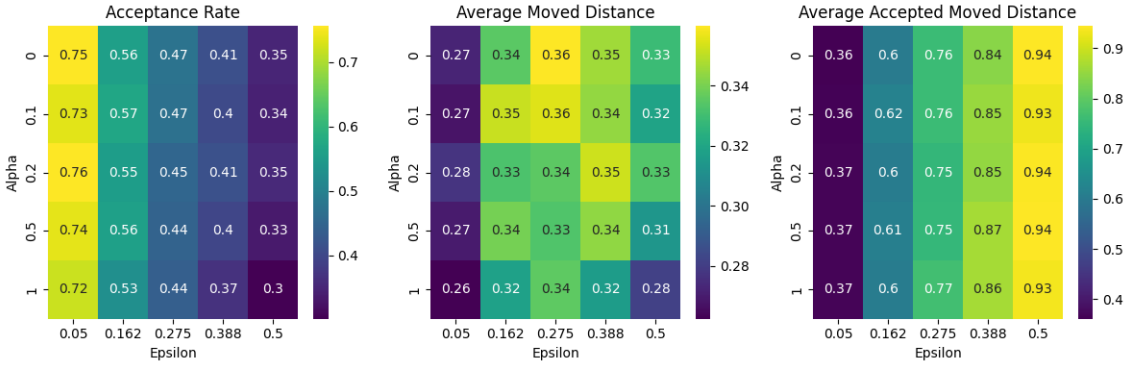


Figure 2: the grid search for finding curl matrix parameter

We observe that the M-H acceptance rate drops as either α or ϵ increases, which is reasonable since both of them are positively correlated to the magnitude of a proposed transition. The interesting part happens with the **averaged accepted moved distance**, which shows that for multiple levels of ϵ (e.g. 0.5 and 0.05 cases), **the optimial choice of α is a non-zero one**, showing settings when irreversible diffusion is non-trivial.

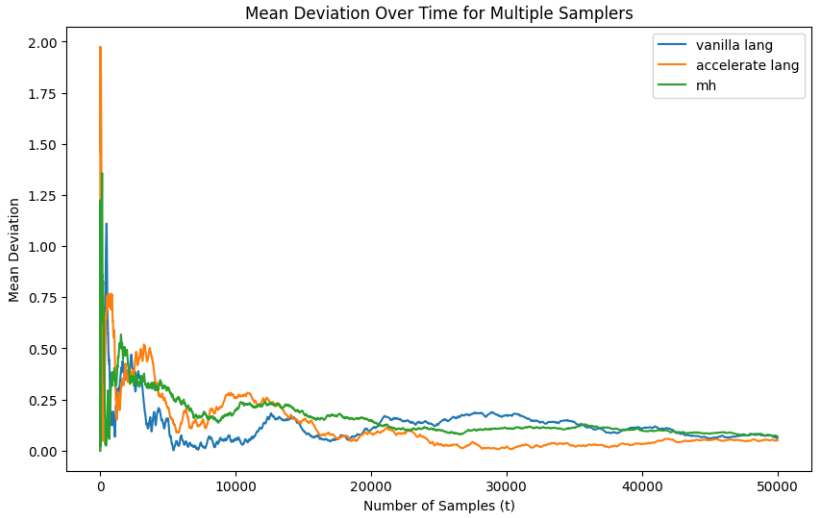


Figure 3: deviation between sample versus ground-truth statistics

Other than simple distance metrics, we can directly compare the **empirical statistics** from the samples (at different iterations) with the ground-truth to reflect information such as how quickly does the sampling process converges. In this problem, we calculate the mean deviation (l_2 norm) between the sample mean and the ground-truth mean $\sum_{i=1}^{n} w_i \cdot \mu_i$. For the vanilla Langevin method, we use the optimal ϵ in terms of averaged moved distance from the grid-search ($\alpha = 0$ cases), and the "accelerated" Langevin method uses the best combination of α and ϵ. Finally the M-H MCMC method is used as a baseline. Each method is run to generate $N = 50000$ samples with 1000 burn-ins. The experiments demonstrate minor advantage of the accelerated method.

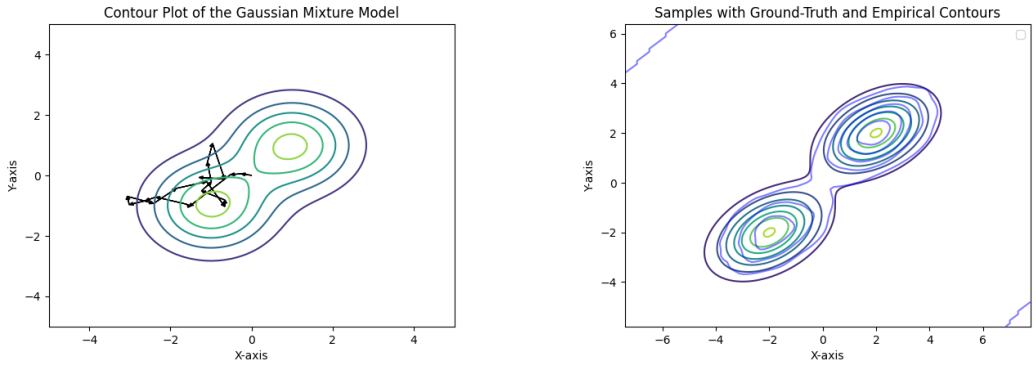


Figure 4: *Left:* a less separated target landscape; *Right:* a more separated target landscape where irreversibility becomes useful

An intruguing question that we wish to investigate is **which target landscape is irreversible diffusion good at?** Since the curl Q matrix applies a deterministic traversing effect for transiting along the contour lines of the target, one assumption we can make is that:

The Q matrix will play a significant role when the target is multi-modal and somewhat separated – with the contour lines serving as easy connectors between the modes.

To verify this claim, we rerun the above experiment but in a different GMM model – one that is again bi-modal yet the modes are more separated, as shown in Fig 4. The grid-search and deviation analysis for this new model is demonstrated below:

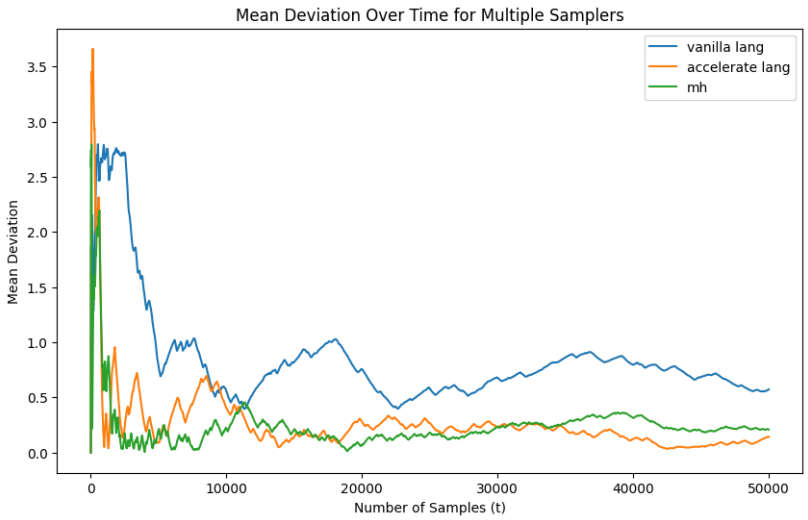


Figure 5: the grid search for finding curl matrix parameter

This time, with the same experiment setup and procedures, we see that there exists a much larger gap between the best reversible LMC and the irreversible version.

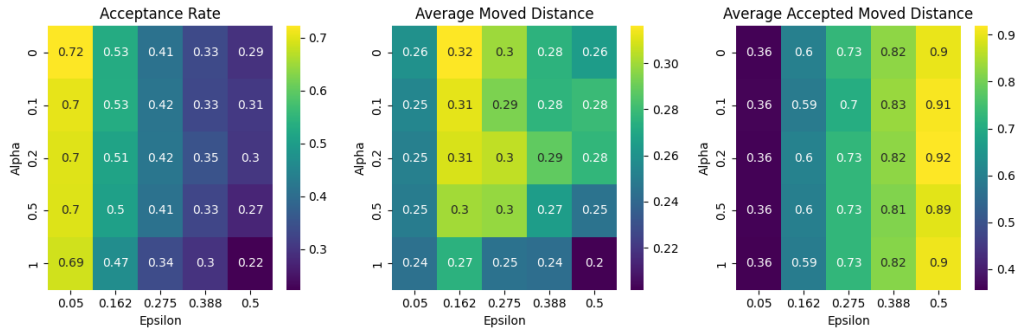


Figure 6: deviation between sample versus ground-truth statistics

3.2 Bayesian Learning

The second model for experiment is adapted from the the SGLD paper by **?**. It is a Bayesian posterior inference problem with the below setup:

$$\theta_1 \sim N(0, \sigma_1^2)\,; \quad \theta_2 \sim N(0, \sigma_2^2) \tag{19}$$

$$x_i \sim \frac{1}{2}N(\theta_1, \sigma_x^2) + \frac{1}{2}N(\theta_1 + \theta_2, \sigma_x^2) \tag{20}$$

where the hyperparameters $\sigma_1^2 = 10, \sigma_2^2 = 1, \sigma_x^2 = 2$ are known. When generating 100 data points $x_1, ..., x_{100}$ from $\theta_1 = 0, \theta_2 = 1$, the ground-truth posterior is multimodal with the other mode at $\theta_1 = 1, \theta_2 = -1$. Our goal is to run our Monte Carlo sampling algorithm to (1) recover the posterior mean $(0.5, 0)$ and (2) recover the bimodal posterior landscape.

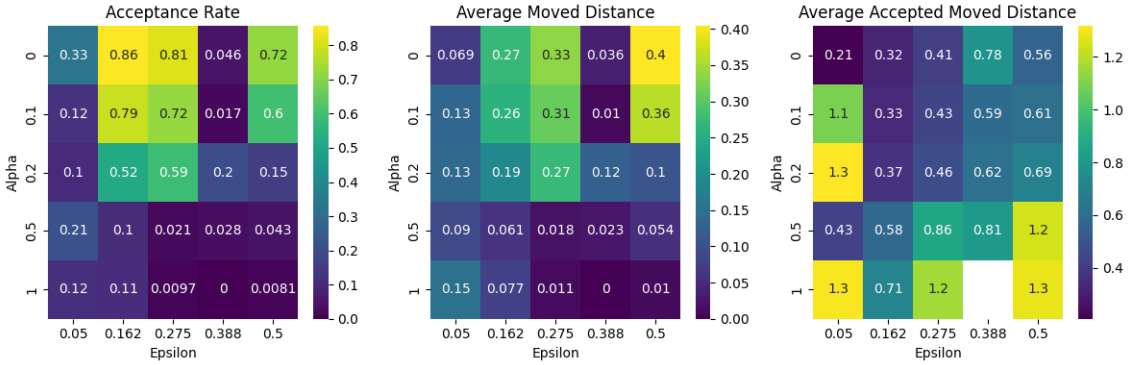
In addition to the above MALA algorithm, we additionally make use of **ensembling method** in this model. The pseudo-code for this revised algorithm is:

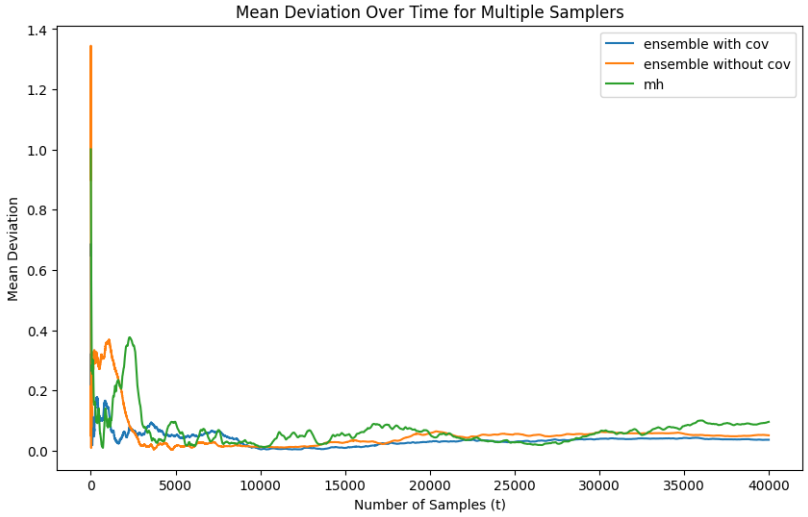
Algorithm 2 Ensemble Metropolis-adjusted Langevin Algorithm (E-MALA)

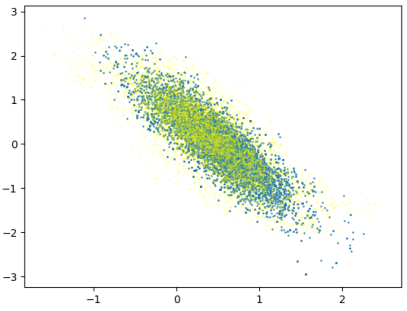
Input: Ensemble size m, initial particles $\{x_0^{(1)}, ..., x_0^{(m)}\}$, step size ϵ, target distribution $\pi(x)$, number of sample N, curl matrix parameter α.
Output: N samples from the distribution $\pi(x)$.
procedure E-MALA
    $k \leftarrow 0$
    $x_k \leftarrow x_0$
    Construct fixed curl Q matrix
    while $k < N/m$ **do**
        $j \leftarrow 1$
        Estimate ensemble covariance matrix
        $\bar{x}_k \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_k^{(i)}$
        $C_k \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_k^{(i)} - \bar{x}_k) \otimes (x_k^{(i)} - \bar{x}_k)$
        for $j \leq m$ **do**
            Sample $\xi \sim \mathcal{N}(0, 1)$
            Propose $\hat{x}_{k+1} \leftarrow x_k^{(j)} + \epsilon[C_k + Q]\nabla \log \pi(x_k^{(j)}) + \sqrt{2\epsilon C_k}\xi$
            Compute acceptance probability α as follows:
            $\alpha \leftarrow \min\left(1, \frac{\pi(\hat{x}_{k+1})q(x_k^{(j)}|\hat{x}_{k+1})}{\pi(x_k^{(j)})q(\hat{x}_{k+1}|x_k^{(j)})}\right)$
            Accept or reject the new sample based on α:
            With probability α, accept $\hat{x}_{k+1}$
            if accept **then**
                $x_{k+1}^{(j)} \leftarrow \hat{x}_{k+1}$
            else
                $x_{k+1}^{(j)} \leftarrow x_k$
            end if
        end for
        $k \leftarrow k + 1$
    end while
    return $x_0^{(1)}, ..., x_0^{(m)}, ..., x_k^{(1)}, ..., x_k^{(m)}$
end procedure

A few things to notice are:

- in this implementation, the only use of the ensembles is to calculate an **empirical covariance matrix** that serves as a diffusion matrix. The ensembles could be used as gradient-free approximators of $\nabla \log \pi(x)$ but will make the model more complicated.
- the total number of sample N comes from the ensembles at different iterations instead of the last few iterations. Therefore, reasonable amount of burn-ins might be needed to decouple different ensembles from their initial distribution.







(a) blue points: E-MALA with covariance, yellow points: MH-MCMC