Introduction to sampling algorithms

Samuel Livingstone

Department of Statistical Science, University College London

samuel.livingstone@ucl.ac.uk

CECAM summer school, July 2025

Outline

- Motivation
- Langevin dynamics
- Kinetic Langevin dynamics
- Metropolis—Hastings
- Example models in Statistics & machine learning
  - Basic probabilistic classifier (logistic regression)
  - Latent variable model (Neal's funnel)

Motivation (high level)

The sampling problem

Draw $X \sim \pi$, for some (possibly unnormalised) probability measure π.

The integration problem

Given π and $f \in L^2(\pi)$, compute $\pi(f) := \int f(x)\pi(dx)$.

This talk: Euclidean state spaces, π has a density

Motivation (more precise)

1. **The ($\epsilon-$)sampling problem.** *Given a finite measure π_u on E, generate a sample*

$$X \sim \rho$$

such that $\mathscr{D}(\rho, \pi) < \epsilon$, where $\pi := \pi_u / \pi_u(E)$ (normalised measure).

2. **The ($\epsilon-$)integration problem.** *Given a finite measure π_u and $f \in L^2(\pi_u)$, and denoting*

$$\pi(f) := \int f(x)\pi(dx),$$

compute $\hat{\pi}(f)$ such that such that $\mathbb{E}\,|\,\hat{\pi}(f) - \pi(f)\,|^2 < \epsilon$

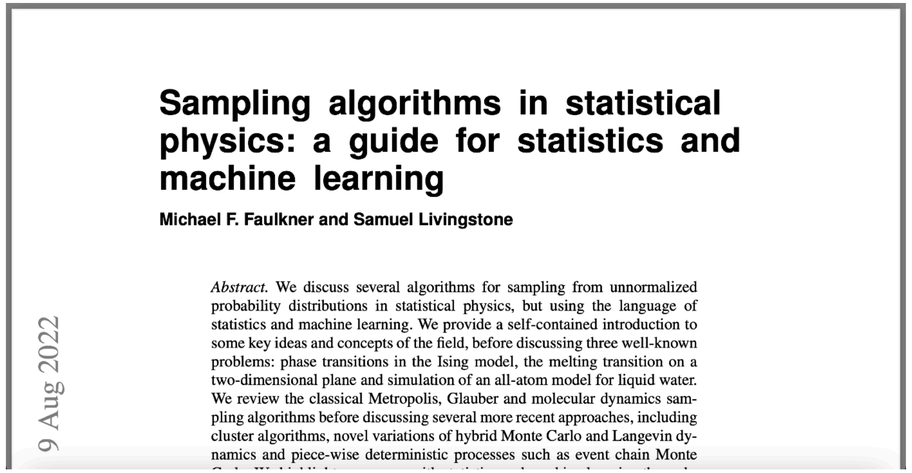
Why unnormalised?

Motivation from statistical physics

- In statistical physics interest often lies in understanding macroscopic consequences of microscopic behaviour in a system
- Particle dynamics —> interaction potential $U(x_i, x_j)$
- Boltzmann—Gibbs distribution at temperature $1/\beta$

$$\pi_\beta(x)dx \propto e^{-\beta \sum_{i \neq j} U(x_i, x_j)} dx$$

- Macroscopic *observables*: $\int f(x)\pi_\beta(x)dx$



Normalisation is a challenge!

Why unnormalised?

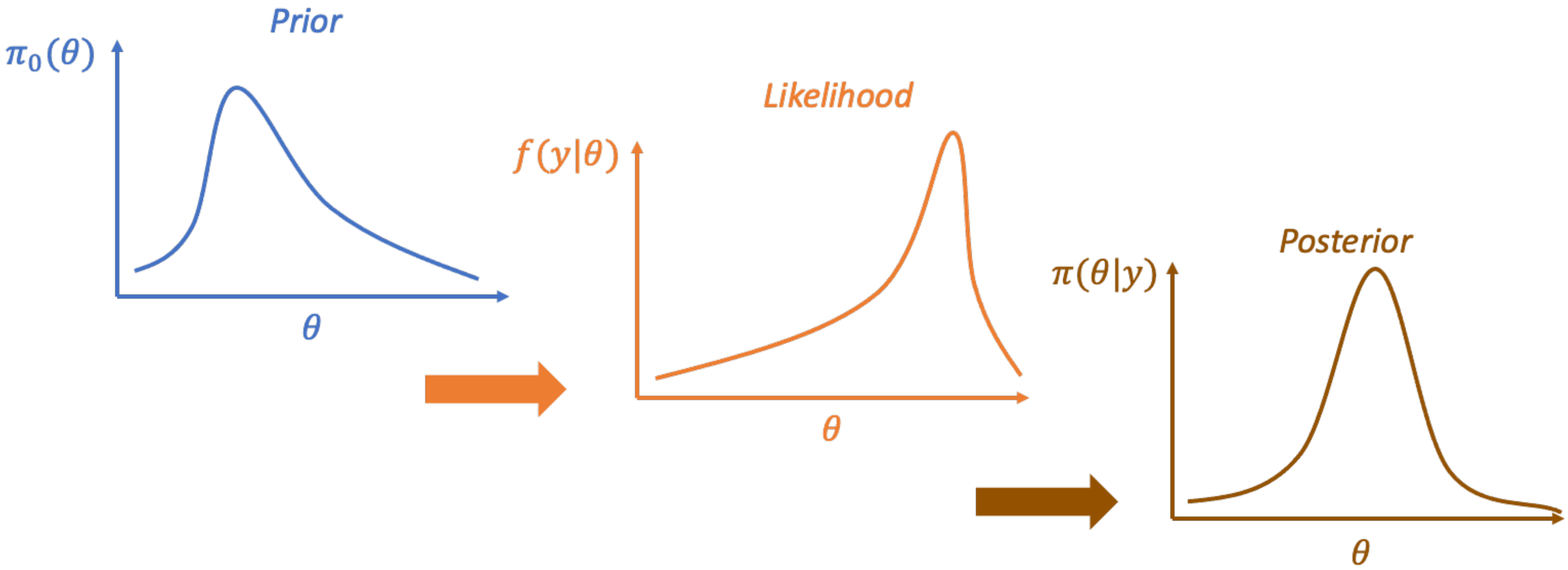
Motivation from Bayesian statistical inference

- Given some data $y := \{y_i\}_{i=1}^n$ assume a probabilistic model $f(y \mid \theta)$
- Inference: given y learn θ. Prediction: given y predict y^* (via θ)
- Bayesian approach
  - Construct prior $\pi_0(\theta)$
  - Bayes' rule: model + prior —> posterior

$$\pi(\theta \mid y) = \frac{f(y \mid \theta)\pi_0(\theta)}{\int f(y \mid \theta)\pi_0(\theta)d\theta} \propto f(y \mid \theta)\pi_0(\theta)$$

Interesting quantities are usually integrals wrt posterior

Bayes in pictures



Integration via sampling + Monte Carlo

- We can solve Problem 2 by solving Problem 1 to generate $X_1, \ldots, X_N$ and computing

$$\hat{\pi}(f) := \frac{1}{N} \sum_{i=1}^{N} f(X_N).$$

- Most popular solution to Problem 1 is to generate $X_1, \ldots, X_N$ by simulating a Markov chain -> MCMC

The Markov chain Monte Carlo solution

Construct a Markov chain with equilibrium distribution π, then simulate from it.

Simulate for a long time to generate samples

Ergodic averages to approximate integrals

THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

I. INTRODUCTION

THE purpose of this paper is to describe a general method, suitable for fast electronic computing machines, of calculating the properties of any substance which may be considered as composed of interacting individual molecules. Classical statistics is assumed, only two-body forces are considered, and the potential field of a molecule is assumed spherically symmetric.

II. THE GENERAL METHOD FOR AN ARBITRARY POTENTIAL BETWEEN THE PARTICLES

In order to reduce the problem to a feasible size for numerical work, we can, of course, consider only a finite number of particles. This number N may be as high as several hundred. Our system consists of a square† containing N particles. In order to minimize the surface effects we suppose the complete substance to be periodic, consisting of many such squares, each square contain-

Biometrika (1970), **57**, 1, *p.* 97
Printed in Great Britain

Monte Carlo sampling methods using Markov chains and their applications

BY W. K. HASTINGS
University of Toronto

SUMMARY

A generalization of the sampling method introduced by Metropolis *et al.* (1953) is presented along with an exposition of the relevant theory, techniques of application and methods and difficulties of assessing the error in Monte Carlo estimates. Examples of the methods, including the generation of random orthogonal matrices and potential applications of the methods to numerical problems arising in statistics, are discussed.

1. INTRODUCTION

For numerical problems in a large number of dimensions, Monte Carlo methods are often more efficient than conventional numerical methods. However, implementation of the

Why Markov processes?

- There is no inherent reason why a sampling algorithm should use a Markov process
- BUT
  - Independent sampling seems very hard in high-dimensions!
  - We have many tools to assess the equilibrium distribution & mixing times
- It's also surprising how much flexibility Markovianity gives us
  - (e.g. if state space is augmented)

We will discuss both continuous and discrete time processes

Setup

- Let $X_0 \sim \mu$, and define the transition function/kernel (continuous-time process $t \in [0,\infty)$, discrete-time $t \in \{0,1,2,\dots\}$)

$$P^t(x, A) := \mathbb{P}(X_{s+t} \in A \,|\, X_s = x).$$

- We write $X_t \sim \mu P^t$, where

$$\mu P^t(A) := \int P^t(x, A)\mu(dx)$$

- We can also define the process through the semi-group $(P_t)_{t\geq 0}$, where for $f \in L^2(\pi)$

$$P_t f(x) := \int f(y)P^t(x, dy) = \mathbb{E}[f(X_{s+t}) \,|\, X_s = x]$$

- The generator of the semi—group is $Lf := \lim_{\delta \to 0} \delta^{-1}(P_\delta f - f)$ for appropriate f

Equilibrium & Invariant distributions

- We say $(X_t)_{t\geq 0}$ has equilibrium distribution π if

$$\lim_{t\to\infty} \mathscr{D}(\mu P^t, \pi) = 0$$

  where $\mathscr{D}$ is some metric between distributions (e.g. Total Variation distance).

- We say $(X_t)_{t\geq 0}$ is $\pi-$invariant if

$$X_s \sim \pi \implies X_{s+t} \sim \pi \quad \forall t \geq 0$$

- NOTE: Unless stated otherwise assume π has a density with potential $U(x)$, i.e.

$$\pi(x) \propto e^{-U(x)}$$

A recipe for sampling based on Markov processes

- Find a Markov process $(X_t)_{t \geq 0}$ that has equilibrium distribution π
  - *Usually continuous time*
  - *Usually not possible to simulate exactly*
- Approximate the dynamics to simulate a (discrete-time) Markov chain
  - *Typically the equilibrium is no longer π, but hopefully not far off*
  - *Optionally, we can use a Metropolis—Hastings filter to enforce π-equilibrium (more later)*

Langevin sampling

Diffusion processes

A first candidate for stochastic process with user-specified equilibrium

$$dX_t = b(X_t)dt + \sigma(X_t)dW_t$$

- $b : \mathbb{R}^d \to \mathbb{R}^d$ drift
- $\sigma : \mathbb{R}^d \to \mathbb{R}^{d\times m}$ volatility
- $(W_t)_{t\geq 0}$ Brownian motion on $\mathbb{R}^m$
  - $W_0 = 0$, $W_t - W_s \sim N_m(0,(t-s)I)$ for $t > s$
- Intuition:

$$\mathbb{E}[X_{t+\delta t} - X_t \,|\, X_t = x] = b(x)\delta t + o(\delta t)$$

$$\mathrm{Var}[X_{t+\delta t} - X_t \,|\, X_t = x] = \sigma(x)\sigma(x)^T\delta t + o(\delta t)$$

Example diffusion sample paths

- https://samuel-livingstone.shinyapps.io/app_bm/

The density of X_t

Fokker—Planck equation (FPK)

- Recall $X_t \sim \mu P^t$, let μP^t have density $u(x,t)$
- Then FPK states

$$\partial_t u = L^\dagger u$$

with $L^\dagger$ the 'flat' $L^2(\mathbb{R}^d)$ adjoint of L

- For diffusions $L^\dagger$ expressed in terms of b and σ, giving for $A(x) := \sigma(x)\sigma(x)^T$

$$\partial_t u(x,t) = -\sum_i \partial_i [b_i(x) u(x,t)] + \frac{1}{2} \sum_{i,j} \partial_i \partial_j [A_{ij}(x) u(x,t)]$$

Invariant distributions for diffusion processes

Fokker—Planck equation

- Setting $u(x,t) := \pi(x)$ implies $\partial_t \pi(x) = 0$, so to be π-invariant we need

$$2\sum_i \partial_i[b_i(x)\pi(x)] = \sum_{i,j} \partial_i \partial_j [A_{ij}(x)\pi(x)]$$

- A sufficient condition is that $\forall i$

$$2b_i(x)\pi(x) = \sum_j \partial_j [A_{ij}(x)\pi(x)]$$

Any b and σ for which this is satisfied will produce a $\pi-$invariant diffusion process!

The (overdamped) Langevin diffusion

- Setting $A_{ij}(x) = 2\mathbb{I}(i = j)$ for simplicity

$$b_i(x)\pi(x) = \partial_i \pi(x)$$

- Rearranging gives

$$b_i(x) = \frac{1}{\pi(x)} \partial_i \pi(x) = \partial_i \log \pi(x)$$

- Writing $\pi(x) \propto e^{-U(x)}$, meaning $\nabla \log \pi(x) = -\nabla U(x)$, this leads to

$$dX_t = -\nabla U(X_t)dt + \sqrt{2}dW_t$$

Properties of the Langevin diffusion

$$dX_t = -\nabla U(X_t)dt + \sqrt{2}dW_t$$

- Only unnormalised π need be known!
- Drift $-\nabla U(x)$ moves towards local mode, dW_t injects noise
- The $\sqrt{2}$ factor is important! *Tells us the right balance between drift and noise*
- Under mild conditions $\mathscr{D}_{TV}(\mu P^t, \pi) \to 0$ as $t \to \infty$
  - If U exhibits e.g. tail convexity convergence happens at an exponential rate
  - If e.g. $\nabla^2 U \succeq mI$ for $m > 0$ then rate is quantitative $e^{-m/2}$

Numerical simulation

Langevin Monte Carlo/Unadjusted Langevin algorithm

- Euler—Maruyama scheme for a general diffusion process, for $\xi_n \sim N(0, I_{d\times d})$, $h > 0$ step-size

$$X_{n+1} = X_n + hb(X_n) + \sqrt{h}\sigma(X_n)\xi_n$$

- Applied to Langevin gives

$$X_{n+1} = X_n - h\nabla U(X_n) + \sqrt{2h}\xi_n$$

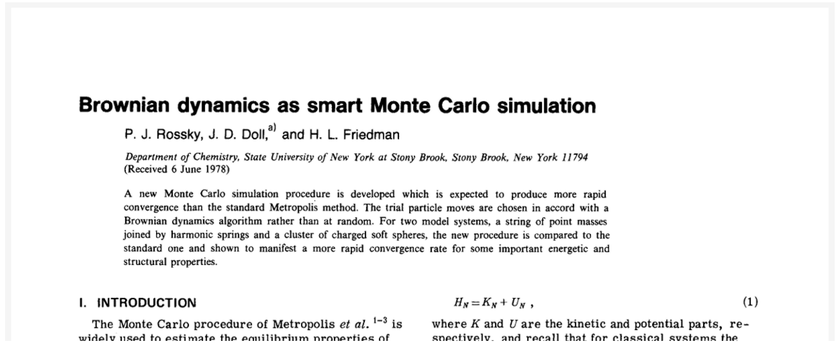
The **Unadjusted Langevin algorithm** (ULA) or **Langevin Monte Carlo**

Example numerical simulation

- https://samuel-livingstone.shinyapps.io/app_ou/

Some key references

Rossky, Doll & Friedman (1978).
An early use in the Physics literature



Brownian dynamics as smart Monte Carlo simulation

P. J. Rossky, J. D. Doll,[a] and H. L. Friedman

Department of Chemistry, State University of New York at Stony Brook, Stony Brook, New York 11794
(Received 6 June 1978)

A new Monte Carlo simulation procedure is developed which is expected to produce more rapid convergence than the standard Metropolis method. The trial particle moves are chosen in accord with a Brownian dynamics algorithm rather than at random. For two model systems, a string of point masses joined by harmonic springs and a cluster of charged soft spheres, the new procedure is compared to the standard one and shown to manifest a more rapid convergence rate for some important energetic and structural properties.

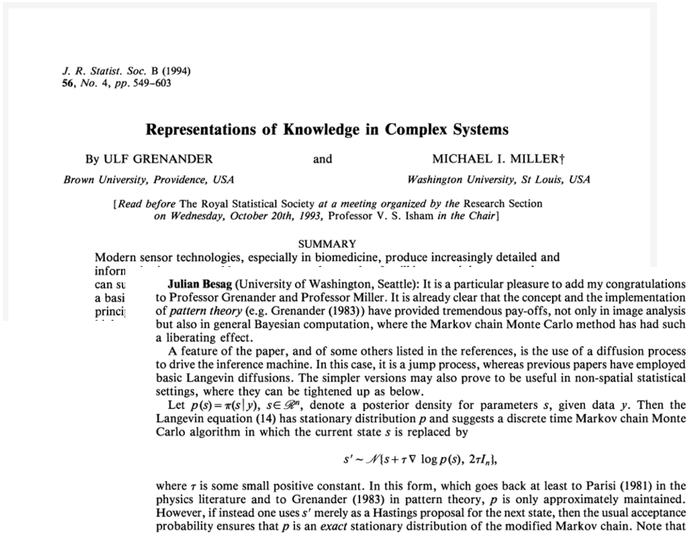
I. INTRODUCTION

The Monte Carlo procedure of Metropolis *et al.*[1–3] is

$$H_N = K_N + U_N \,, \qquad (1)$$

where K and U are the kinetic and potential parts, respectively, and recall that for classical systems the

Besag (1994). First mention of Metropolised form in Statistics literature



J. R. Statist. Soc. B (1994)
56, *No.* 4, *pp.* 549–603

Representations of Knowledge in Complex Systems

By ULF GRENANDER and MICHAEL I. MILLER†

Brown University, Providence, USA *Washington University, St Louis, USA*

[*Read before* The Royal Statistical Society *at a meeting organized by the* Research Section *on Wednesday, October 20th, 1993,* Professor V. S. Isham *in the Chair*]

SUMMARY

Modern sensor technologies, especially in biomedicine, produce increasingly detailed and inform
can su
a basi
princi

Julian Besag (University of Washington, Seattle): It is a particular pleasure to add my congratulations to Professor Grenander and Professor Miller. It is already clear that the concept and the implementation of *pattern theory* (e.g. Grenander (1983)) have provided tremendous pay-offs, not only in image analysis but also in general Bayesian computation, where the Markov chain Monte Carlo method has had such a liberating effect.

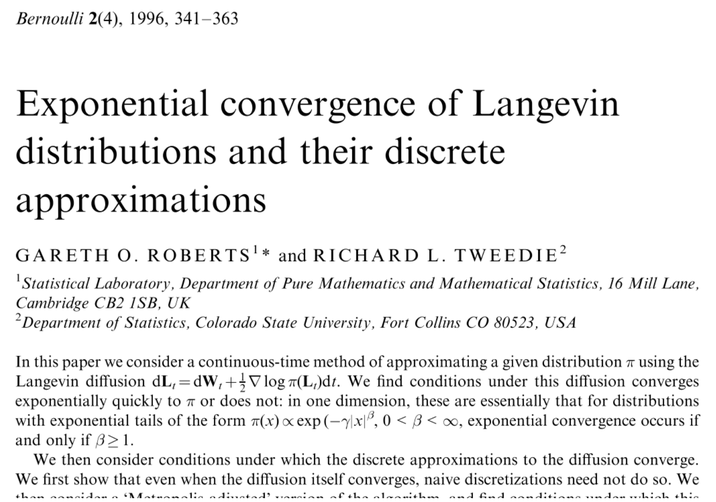
A feature of the paper, and of some others listed in the references, is the use of a diffusion process to drive the inference machine. In this case, it is a jump process, whereas previous papers have employed basic Langevin diffusions. The simpler versions may also prove to be useful in non-spatial statistical settings, where they can be tightened up as below.

Let $p(s) = \pi(s|y)$, $s \in \mathscr{R}^n$, denote a posterior density for parameters s, given data y. Then the Langevin equation (14) has stationary distribution p and suggests a discrete time Markov chain Monte Carlo algorithm in which the current state s is replaced by

$$s' \sim \mathscr{N}\{s + \tau \nabla \log p(s),\ 2\tau I_n\},$$

where τ is some small positive constant. In this form, which goes back at least to Parisi (1981) in the physics literature and to Grenander (1983) in pattern theory, p is only approximately maintained. However, if instead one uses s' merely as a Hastings proposal for the next state, then the usual acceptance probability ensures that p is an *exact* stationary distribution of the modified Markov chain. Note that

Roberts & Tweedie (1996). Influential analysis of Diffusion, ULA and Metropolised version



Bernoulli **2**(4), 1996, 341–363

Exponential convergence of Langevin distributions and their discrete approximations

GARETH O. ROBERTS[1]* and RICHARD L. TWEEDIE[2]

[1]*Statistical Laboratory, Department of Pure Mathematics and Mathematical Statistics, 16 Mill Lane, Cambridge CB2 1SB, UK*
[2]*Department of Statistics, Colorado State University, Fort Collins CO 80523, USA*

In this paper we consider a continuous-time method of approximating a given distribution π using the Langevin diffusion $d\mathbf{L}_t = d\mathbf{W}_t + \frac{1}{2}\nabla \log \pi(\mathbf{L}_t)dt$. We find conditions under this diffusion converges exponentially quickly to π or does not: in one dimension, these are essentially that for distributions with exponential tails of the form $\pi(x) \propto \exp(-\gamma|x|^\beta$, $0 < \beta < \infty$, exponential convergence occurs if and only if $\beta \geq 1$.

We then consider conditions under which the discrete approximations to the diffusion converge. We first show that even when the diffusion itself converges, naive discretizations need not do so. We

Some known results about ULA

- If ∇U is Lipschitz then under mild regularity conditions $\{X_0, X_1, \dots\}$ has an **equilibrium distribution** π_h
- If U is tail-convex and ∇U is Lipschitz (+details), then ULA is **geometrically ergodic**, i.e.
  $\exists \lambda > 0, M : \mathscr{P}(\mathbb{R}^d) \to [0,\infty)$ s.t. if $X_n \sim \nu_n$

$$\mathscr{D}_{TV}(\mu P^n, \pi_h) \leq M(\mu) e^{-\lambda n}$$

  where $\mathscr{D}_{TV}$ is total variation distance

- If U is m-strongly convex and L-smooth, setting $\kappa := L/m$ then various bounds of form (exc. polylog. terms)

$$\inf\{n > 0 : \mathscr{D}(\mu P^n, \pi) \leq \epsilon\} \leq C_\epsilon \kappa^\alpha d^{1/2}$$

  where d is dimension, $\alpha \geq 1$ (h chosen to control $\mathscr{D}(\pi_h, \pi)$)

But when ∇U is not Lipschitz or U not tail-convex, **it can perform poorly!**

The case of non-Lipschitz ∇U (light tails)

Toy example

- Consider $\pi(x) \propto e^{-x^4/4}$, $\nabla U(x) = x^3$
- ULA scheme is

$$X_{n+1} = X_n - hX_n^3 + \sqrt{2h}\xi_n$$

- If e.g. $X_0 = 10, h = 0.1$, then

$$\mathbb{E}[X_1] \approx -90, \quad \mathbb{E}[X_2] \approx 72900, \quad \mathbb{E}[X_3] \approx \dots$$

- In this instance the ULA scheme is **transient** (although non-asymptotic results for small h exist)

NOTE: The true diffusion is actually very fast for this example!

The case of heavy tails (non-convex $U(x)$)

- Consider a simple Cauchy target distribution (heavy-tailed model in Statistics)

$$\pi(x) \propto \frac{1}{1+x^2} \implies \nabla U(x) = \frac{2x}{1+x^2}$$

- Here as $|x| \to \infty$ then $|\nabla U(x)| \to 0$, meaning

$$X_{n+1} \approx X_n + \sqrt{h}\xi_n$$

"Random walk behaviour" (in the tails the scheme will be very slow)

More Langevin diffusions

- We set $A_{ij}(x) = 2\mathbb{I}(i = j)$, but other choices are possible, and sometimes preferred
- **Tempered Langevin:** $A_{ij}(x) \propto \pi_u^{-1}(x)\mathbb{I}(i = j)$

$$dX_t = \pi_u^{-1/2}(X_t)dW_t$$

- For generic $A_{ij}(x)$ we have

$$dX_t = -A(x)\nabla U(x)dt + \Lambda(x)dt + \sqrt{2A(x)}dW_t, \quad \Lambda_i(x) := \sum_j \partial_j A_{ij}(x)$$

(e.g. **manifold MALA**, $A_{ij}(x) \approx \partial_i\partial_j U(x)$)

- Additional drift terms can also be added *(non-reversibility)*

Kinetic Langevin sampling

Augmenting the state space

- A different class of diffusion processes can be arrived at by considering a larger state space

$$(X, V) \in \mathbb{R}^{d \times d}$$

- Auxiliary V often called **velocity** or **momentum**
- The resulting processes are non-reversible *(often means faster but harder to analyse)*

Kinetic Langevin diffusion

- For $\alpha > 0$

$$dX_t = V_t dt$$

$$dV_t = -\nabla U(X_t)dt - \alpha V_t dt + \sqrt{2\alpha} dW_t$$

- This solves Fokker—Planck for density

$$\omega(x, v) \propto \pi(x) \cdot \exp\left(-\frac{1}{2} v^T v\right)$$

- NOTE: noise is degenerate, meaning diffusion is **hypoelliptic** (Langevin is uniformly elliptic)

Hamiltonian flow (conservative) + mean reverting velocity dynamics (dissipative)

Integration via Splitting schemes

System H
(Hamiltonian dynamics)

$$dX_t = V_t dt,$$
$$dV_t = -\nabla U(X_t) dt$$

Conservative

System O
(Ornstein—Uhlenbeck process)

$$dX_t = 0,$$
$$dV_t = -\alpha V_t dt + \sqrt{2\alpha} dW_t$$

Dissipative

The Hamiltonian part

System H
(Hamiltonian dynamics)

$$dX_t = V_t dt,$$
$$dV_t = -\nabla U(X_t)dt$$

- We can employ a further splitting into

$$(B)\ dX_t = 0, \quad dV_t = -\nabla U(X_t)dt$$
$$(A)\ dX_t = V_t dt, \quad dV_t = 0$$

- (B) has explicit solution

$$(X_t, V_t) = \Psi_t^{(B)}(X_0, V_0) = (X_0, V_0 - t\nabla U(X_0))$$

- (A) has explicit solution

$$(X_t, V_t) = \Psi_t^{(A)}(X_0, V_0) = (X_0 + tV_0, V_0)$$

- Full 'splitting' integrator is some combination, e.g.

$$\Psi_h(X_0, V_0) := \Psi_{h/2}^{(B)} \circ \Psi_h^{(A)} \circ \Psi_{h/2}^{(B)}(X_0, V_0)$$

The Ornstein—Uhlenbeck part

System O
(Ornstein—Uhlenbeck process)

$$dX_t = 0,$$

$$dV_t = -\alpha V_t dt + \sqrt{2\alpha} dW_t$$

- This can be solved exactly, giving

$$(X_t, V_t) = \Phi_{OU}(X_0, V_0)$$

$$= (X_0, e^{-\alpha t} V_0 + \sqrt{1 - e^{-2\alpha t}}\,\xi)$$

where $\xi \sim N(0, I)$.

Putting it together

- Splitting integrators are extremely popular for this type of system, e.g.

$$(X_{t+h}, V_{t+h}) = \Psi^{(A)}_{h/2} \circ \Psi^{(B)}_{h/2} \circ \Phi^{(O)}_{h} \circ \Psi^{(B)}_{h/2} \circ \Psi^{(A)}_{h/2}(X_t, V_t)$$

- This = **ABOBA**, alternatives **BAOAB, OBABO** + other ways to split
- The Hamiltonian solver is **symplectic**
  - *(good for long time preservation of* $H(x, v) = U(x) + v^T v/2$*)*
- Full system solver sometimes called **quasi-symplectic**

Kinetic Langevin vs ULA

- ULA is easier to implement (only need step-size h)
  - The friction α must also be chosen for kinetic
- Kinetic Langevin often faster in high-dimensions
  - Mixing times can be $d^{1/4}$ rather than $d^{1/2}$ for ULA
- Kinetic Langevin still struggles with:
  - Heavy tails *(process itself is slow, just like ULA)*
  - Non-Lipschitz gradients *(integrator unstable, again just like ULA)*

Metropolis—Hastings

Metropolis—Hastings

Starting point: an arbitrary Markov transition kernel $Q(x, dy) := q(x, y)dy$.

Metropolis—Hastings Algorithm

At iteration $i + 1$, with current state X_i

1. Draw $X' \sim Q(X_i, \cdot)$ *(propose a move to a new state)*
2. Set $X_{i+1} = \begin{cases} X' & \text{w.p. } \alpha(X_i, X') \\ X_i & \text{otherwise} \end{cases}$ *(accept or reject the move)*

$$\alpha(X_i, X') := \min\left(1, \frac{\pi(X')q(X', X_i)}{\pi(X_i)q(X_i, X')}\right)$$

Why does Metropolis—Hastings work?

1. Often easy to show that π is equilibrium distribution *(π-reversibility = trivial)*

$$\pi(x)q(x,y)\min\left(1,\frac{\pi(y)q(y,x)}{\pi(x)q(x,y)}\right) = \pi(y)q(y,x)\min\left(\frac{\pi(x)q(x,y)}{\pi(y)q(y,x)},1\right)$$

2. Only unnormalised π needed (only need ratio $\pi(y)/\pi(x)$)

How well does Metropolis—Hastings work?

And what is meant by this...

- *Working well* generally means
  - Markov chain converges quickly to equilibrium
  - Ergodic averages have good properties (e.g. low bias & variance)
- Essentially this depends on the choice of Q

Metropolis—Hastings is really a whole family of algorithms

Metropolising the Langevin diffusion

MALA

- One choice for Q is the ULA transition kernel
- This leads to the Metropolis-adjusted Langevin algorithm (MALA)
- PROs of MALA vs ULA:
  - π is now correct equilibrium distribution *('asymptotically exact')*
  - Step-size h often easier to choose (set $\mathbb{E}\alpha \approx 0.57$)
- CONS
  - More expensive (two gradients & π must be evaluated)
  - Metropolis filter adds complexity to transition $\Longrightarrow$ harder to analyse

For heavy tails MALA ~ ULA, for non-Lipschitz gradients ULA drifts to ∞ while MALA gets stuck

Metropolising kinetic Langevin

- Kinetic Langevin diffusion is not reversible *(unlike overdamped)*
  - This means that naive Metropolisation a bad idea
- Process does, however, satisfy a *skew-detailed balance* condition
  - For smarter Metropolis filter, upon rejection we can flip velocity

$$v \to -v$$

- I would say it's more common to Metropolis a related algorithm called **Hamiltonian Monte Carlo**

Example models in Statistics & Machine Learning

Logistic regression

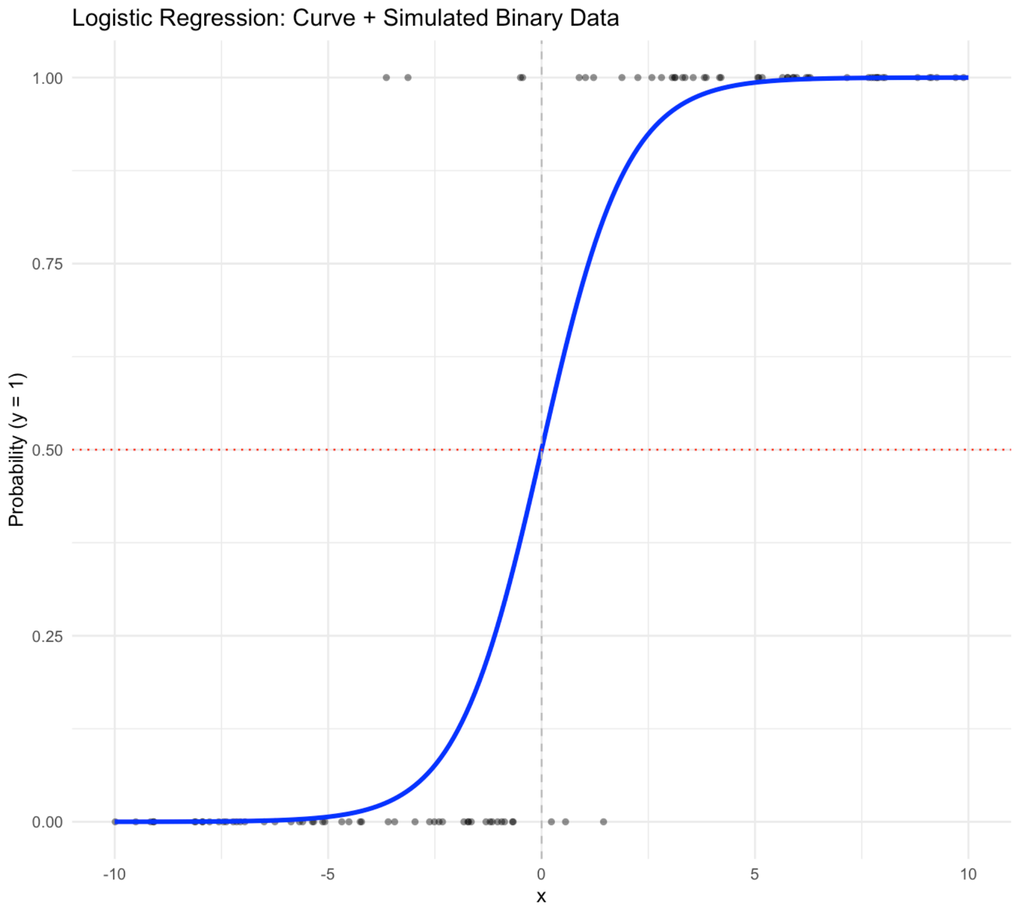
Probabilistic classification

- Data $(x_i, y_i)_{i=1}^n,\ x_i \in \mathbb{R}^d,\ y_i \in \{0,1\}$
- Model is

$$\mathbb{P}(Y_i = 1) = \frac{1}{1 + e^{-x_i^T \beta}}$$

- Setting $\pi_0(\beta)$ as Gaussian, then

$$U(\beta) = \sum_{i=1}^{n} \left[\log\left(1 + e^{x_i^\top \beta}\right) - y_i (x_i^\top \beta) \right] + \frac{1}{2\sigma^2} \beta^T \beta$$



Why a benchmark model?

- One of the simplest real Bayesian models for which sampling algorithms are required (no conjugate prior)
- Challenging benchmark datasets (e.g. many more 1 than 0, $\dim \theta \gg n$)
  - See e.g. Chopin & Ridgway, 2017

Statistical Science
2017, Vol. 32, No. 1, 64–87
DOI: 10.1214/16-STS581

Leave Pima Indians Alone: Binary Regression as a Benchmark for Bayesian Computation

Nicolas Chopin and James Ridgway

Abstract. Whenever a new approach to perform Bayesian computation is introduced, a common practice is to showcase this approach on a binary regression model and datasets of moderate size. This paper discusses to which extent this practice is sound. It also reviews the current state of the art of Bayesian computation, using binary regression as a running example. Both sampling-based algorithms (importance sampling, MCMC and SMC) and fast approximations (Laplace, VB and EP) are covered. Extensive numerical results are provided, and are used to make recommendations to both end users and Bayesian computation experts. Implications for other problems (variable selection) and other models are also discussed.

Neal's funnel

A benchmark for hierarchical models

- Typical structure

$$y_i \,|\, \mu_{c(i)} \sim \mathscr{M}(\mu_{c(i)}) \quad \text{model}$$

$$\mu_{c(i)} \,|\, \nu \sim N(0, e^{\nu}) \quad \text{local parameters}$$

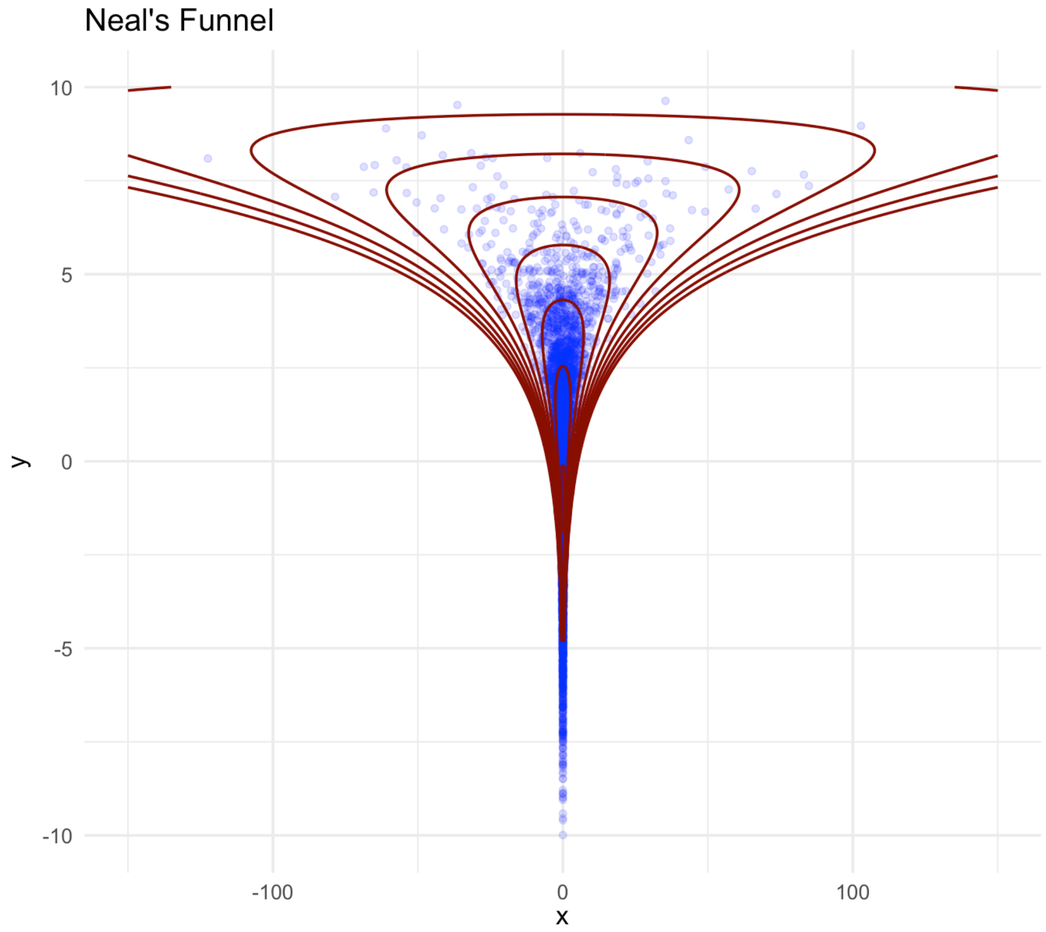
$$\nu \sim N(0, \sigma^2) \quad \text{global parameters}$$

- Neal's funnel:

$$Y_i \,|\, (X = x) \sim N(0, e^{x})$$

$$X \sim N(0, 3^2)$$

- Non-Lipschitz gradients, very hard to move between neck and mouth!
- Typically $c(n) = O(n)$



Discussion

Discussion

- I have focused on two key sampling algorithms & how to Metropolise them
- I have not covered several other things, such as:
  - Sampling on discrete spaces/distributions with atoms *(see e.g. PDMPs)*
  - Diagnosing convergence/the quality of ergodic averages
  - Improving sampling via pre-conditioning *(e.g. mass matrix)*
  - More robust integrators for e.g. non-Lipschitz gradients
  - Detailed analysis of mixing times etc.
  - Multi-model sampling *(e.g. parallel tempering)*
- It's a huge field! Nonetheless I hope this was useful

Thanks for listening!

Thank you for listening!

References (non-exhaustive)

- Bou-Rabee, N., & Hairer, M. (2013). Nonasymptotic mixing of the MALA algorithm. IMA Journal of Numerical Analysis, 33(1), 80-110.
- Chopin, N., & Ridgway, J. (2017). Leave Pima Indians alone: binary regression as a benchmark for Bayesian computation.
- Faulkner, M. F., & Livingstone, S. (2024). Sampling algorithms in statistical physics: a guide for statistics and machine learning. *Statistical Science*, *39*(1), 137-164.
- Grenander, U., & Miller, M. I. (1994). Representations of knowledge in complex systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, *56*(4), 549-581.
- Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications.
- Leimkuhler, B., & Matthews, C. (2013). Robust and efficient configurational molecular sampling via Langevin dynamics. *The Journal of chemical physics*, *138*(17).
- Leimkuhler, B., & Reich, S. (2004). *Simulating hamiltonian dynamics* (No. 14). Cambridge university press.
- Lelievre, T., Rousset, M., & Stoltz, G. (2010). *Free Energy Computations: A Mathematical Perspective*.
- Livingstone, S., Faulkner, M. F., & Roberts, G. O. (2019). Kinetic energy choice in Hamiltonian/hybrid Monte Carlo. Biometrika, 106(2), 303-319.
- Livingstone, S., Nüsken, N., Vasdekis, G., & Zhang, R. Y. (2024). Skew-symmetric schemes for stochastic differential equations with non-Lipschitz drift: an unadjusted Barker algorithm. arXiv preprint arXiv:2405.14373.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, *21*(6), 1087-1092.
- Pavliotis, G. A. (2014). Diffusion Processes. In Stochastic Processes and Applications: Diffusion Processes, the Fokker-Planck and Langevin Equations (pp. 29-54). New York, NY: Springer New York.
- Robert, C. P., Casella, G., & Casella, G. (1999). *Monte Carlo statistical methods* (Vol. 2). New York: Springer.
- Roberts, G. O., & Tweedie, R. L. (1996). Exponential convergence of Langevin distributions and their discrete approximations. Bernoulli, 2(3), 341-363.
- Rossky, P. J., Doll, J. D., & Friedman, H. L. (1978). Brownian dynamics as smart Monte Carlo simulation. *The Journal of Chemical Physics*, *69*(10), 4628-4633.