# Lecture 6: Hamiltonian Monte Carlo

## STATS305C: Applied Statistics III

mondays: models
wednesdays: algorithms

Scott Linderman

April 13, 2022

# Last Time...

- Metropolis-Hastings, Gibbs Sampling
- Probabilistic PCA, Factor Analysis, and Friends

→ read Murphy on ICA

# Today...

**Outline:**

- MCMC Diagnostics
- Hamiltonian Monte Carlo

  or Hybrid

**Reading:**

read this!

- MCMC using Hamiltonian dynamics [Neal, 2012]
- Optional: A Conceptual Introduction to Hamiltonian Monte Carlo [Betancourt, 2017]

# Trace of the Log Joint Probability

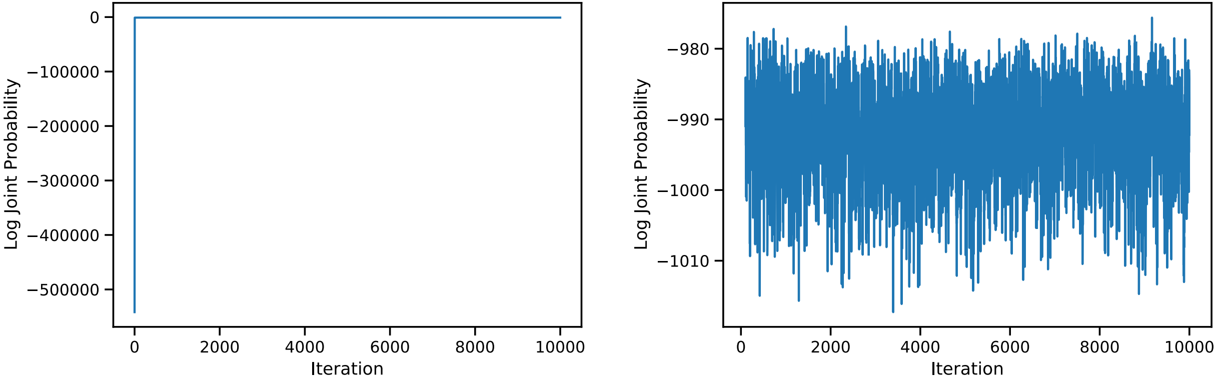


*Figure:* Log probability of all samples (left) and samples 100+ (right)

# Monte Carlo approximations

**Recall:** The idea behind ordinary Monte Carlo is to approximate expectations via sampling,

$$\mathbb{E}_{p(\boldsymbol{\theta}|\boldsymbol{X})}[f(\boldsymbol{\theta})] \approx \frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{\theta}_m) \quad \text{where} \quad \boldsymbol{\theta}_m \sim p(\boldsymbol{\theta} \mid \boldsymbol{X}). \tag{1}$$

Let $\hat{f} = \frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{\theta}_m)$ denote the Monte Carlo estimate. It is a random variable, since it's a function of random samples $\boldsymbol{\theta}_m$.

As such we can reason about its mean and variance. Clearly,

$$\mathbb{E}[\hat{f}] = \frac{1}{M}\sum_{m=1}^{M} \mathbb{E}_{p(\boldsymbol{\theta}|\boldsymbol{X})}[f(\boldsymbol{\theta})] = \mathbb{E}_{p(\boldsymbol{\theta}|\boldsymbol{X})}[f(\boldsymbol{\theta})]. \tag{2}$$

Thus, $\hat{f}$ is an *unbiased* estimate of the desired expectation.

# Variance of Monte Carlo Estimators

What about its variance?

$$\mathrm{Var}[\hat{f}] = \mathrm{Var}\left(\frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{\theta}_m)\right) \tag{3}$$

$$= \frac{1}{M^2}\left(\sum_{m=1}^{M} \mathrm{Var}[f(\boldsymbol{\theta})] + 2\sum_{1\leq m<m'\leq M} \mathrm{Cov}[f(\boldsymbol{\theta}_m), f(\boldsymbol{\theta}_{m'})]\right) \tag{4}$$

If the samples are not only identically distributed but also *uncorrelated*, then $\mathrm{Var}[\hat{f}] = \frac{1}{M}\mathrm{Var}[f(\boldsymbol{\theta})]$.

In this case, the *root mean squared error* (RMSE) of the estimate is $\sqrt{\mathrm{Var}[\hat{f}]} = O(M^{-\frac{1}{2}})$.

However, in MCMC, the samples will be correlated!

transition probability depends on previous state of markov chain!

# Autocovariance and Autocorrelation

For MCMC,

$$\operatorname{Var}[\hat{f}] = \frac{1}{M^2}\left(\sum_{m=1}^{M} \operatorname{Var}[f(\boldsymbol{\theta})] + 2 \sum_{1\leq m<m'\leq M} \operatorname{Cov}[f(\boldsymbol{\theta}_m), f(\boldsymbol{\theta}_{m'})]\right) \tag{5}$$

$$\approx \frac{1}{M}\left(\operatorname{Var}[f(\boldsymbol{\theta})] + 2\sum_{\ell=1}^{M} \operatorname{Cov}[f(\boldsymbol{\theta}_m), f(\boldsymbol{\theta}_{m+\ell})]\right) \tag{6}$$

since the covariance is only a function of the lag $\ell$ once the chain has reached stationarity.

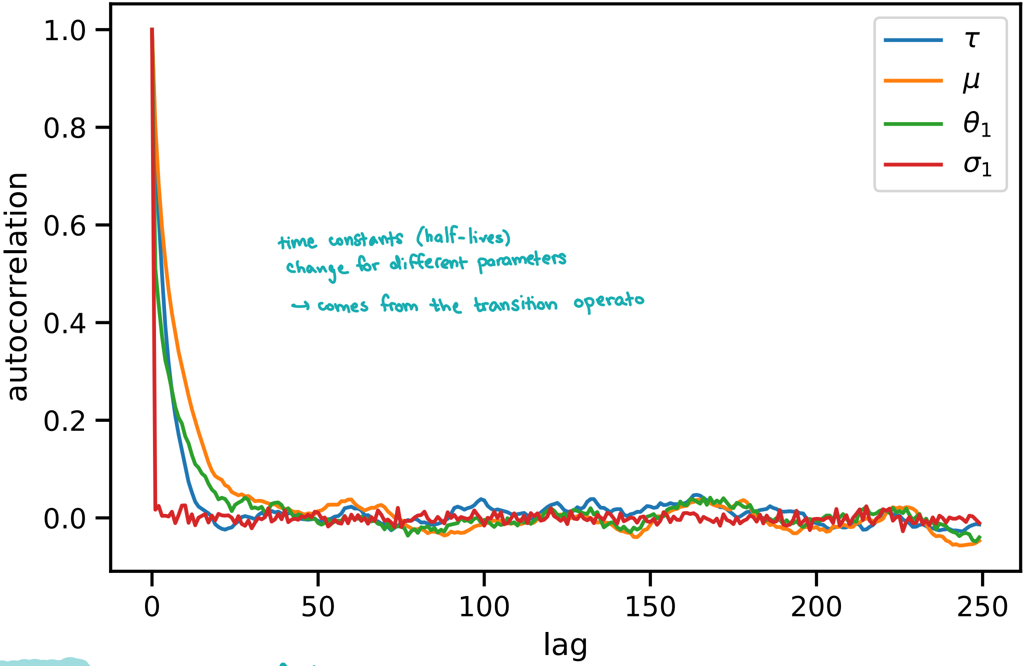
**Note:** At stationarity, the samples are identically distributed but still correlated!

$\operatorname{Cov}[f(\boldsymbol{\theta}_m), f(\boldsymbol{\theta}_{m+\ell})]$ is called the *autocovariance*. It's a function of the lag $\ell$ (and the function $f$).

The *autocorrelation function* (ACF) is defined as, $\operatorname{acf}_f(\ell) = \operatorname{Cov}[f(\boldsymbol{\theta}_m), f(\boldsymbol{\theta}_{m+\ell})]/\operatorname{Var}[f(\theta)]$ so

$$\operatorname{Var}[\hat{f}] \approx M^{-1} \operatorname{Var}[f(\boldsymbol{\theta})]\left(1 + 2\sum_{\ell=1}^{M} \operatorname{acf}_f[\ell]\right). \tag{7}$$

# Autocorrelation Plots for Gibbs Sampling in the Hierarchical Gaussian Model



time constants (half-lives) change for different parameters

→ comes from the transition operato

available in notebook for lecture 4.

$\underbrace{\text{→ throwing away samples}}_{\text{binning}}$ increases the variance of the Monte-Callo estimator

*Figure:* Autocorrelation function for $\tau$, $\mu$, $\theta_1$, and $\sigma_1^2$

# Effective sample size

The *effective sample size* (ESS) approximates the effective number of independent samples you get from an autocorrelated chain, in terms of the variance of the Monte Carlo estimate,

$$M_{\text{eff},f} = M \frac{\text{Var}[f(\theta)]}{\text{Var}[f(\theta)](1 + 2\sum_{\ell=1}^{\infty} \text{acf}_f[\ell])} = \frac{M}{1 + 2\sum_{\ell=1}^{\infty} \text{acf}_f[\ell]} \tag{8}$$

and we let $M_{\text{eff}}$ denote the ESS of the identity functional $f(\theta) = \theta$.

You have to be a bit careful when estimating the ESS—for large values of $\ell$ the sample correlation is too noisy. Typically, we stop when the sample acf is negative. See Section 11.5 of BDA3 for more details.

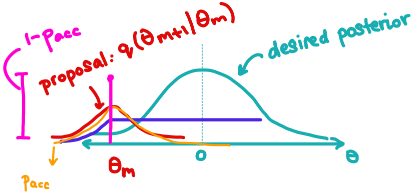
In practice, there are already good implementations in Python (c.f. `pyro.ops.stats.effective_sample_size`) and R (c.f. the `coda` package).

# Effective Sample Size of the Gibbs Sampler for the Hierarchical Gaussian Model

```
Effective sample sizes (in 10000 Gibbs steps):
tausq:     tensor(1563.3734)
mu:        tensor(625.2454)
theta1:    tensor(1027.9817)
sigamsq1:  tensor(8996.0518)
```

higher the half-life, the smaller the ESS becomes

# Metropolis-Hastings Illustrated



$$p(\theta) = N(\theta|0,1) \propto \exp\left(-\tfrac{1}{2}\theta^2\right)$$

$$q(\theta_{m+1}|\theta_m) = N(\theta_{m+1}|\theta_m, \sigma^2)$$

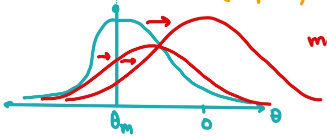
$$a(\theta_m \to \theta_{m+1}) = \min\left\{1, \frac{p(\theta_{m+1})}{p(\theta_m)} \cdot \cancel{\frac{q(\theta_m)}{q(\theta_{m+1})}}\right\}$$

q is symmetric

$$= \min\left\{1, \exp\left[-\tfrac{1}{2}(\theta_{m+1}^2 - \theta_m^2)\right]\right\}$$

$$P_{accept} = \int q(\theta_{m+1}|\theta_m) \cdot a(\theta_m \to \theta_{m+1}) \, d\theta_{m+1}$$

$$\pi(\theta_{m+1}|\theta_m) = \begin{cases} q(\theta_{m+1}|\theta_m) \cdot a(\theta_m \to \theta_{m+1}) \\ (1 - p_{acc}) \cdot \delta_{\theta_m} \end{cases}$$



marginal distribution converges to the stationary distribution and the $(1-p_{acc}) \cdot \delta_{\theta_m}$ vanishes

the Metropolis-Hasting is a Gaussian random walk.

# Autocorrelation of a Random Walk

[From Geyer [2011]] To get some intuition, consider the following Markov chain,

$$\theta_m = \rho\theta_{m-1} + \epsilon_m \tag{9}$$

where $\epsilon_m \sim \mathcal{N}(0, \tau^2)$.

MH isn't quite a mean-reverting random walk, but that's not a terrible model.

In this toy example, we can calculate the autocovariance of the identity functional $f(\theta) = \theta$,

$$\mathrm{Cov}[\theta_m, \theta_{m+\ell}] = \rho\mathrm{Cov}[\theta_m, \theta_{m+\ell-1}] = \rho^{\ell-1}\mathrm{Cov}[\theta_m, \theta_{m+1}] = \rho^{\ell}\mathrm{Var}[\theta_m] \tag{10}$$

so the autocorrelation function decays geometrically as $\mathrm{acf}(\ell) = \rho^{\ell}$.

At stationarity,

$$\mathrm{Var}[\theta_m] = \mathrm{Var}[\theta_{m+1}] = \rho^2\mathrm{Var}[\theta_m] + \tau^2 \Rightarrow \mathrm{Var}[\theta_m] = \frac{\tau^2}{1-\rho^2} \tag{11}$$

as $\rho^2 \to 0$: $\mathcal{N}(0, \tau^2)$

For $\rho^2 < 1$, the stationary distribution exists and is $\mathcal{N}(0, \frac{\tau^2}{1-\rho^2})$.

as $\rho^2 \to 1$: stationary distribution does not exist as samples are highly correlated.

# Autocorrelation of a Random Walk II

Letting $f(\theta) = \theta$, we have,

$$\begin{aligned}
\mathrm{Var}[\hat{f}] &= \frac{1}{M}\left(\mathrm{Var}[\theta] + 2\sum_{\ell=1}^{M}\mathrm{Cov}[\theta_m, \theta_{m+\ell}]\right) && (12)\\
&= \frac{1}{M}\cdot\mathrm{Var}[\theta]\left(1 + 2\sum_{\ell=1}^{M}\rho^{\ell}\right) && (13)\\
&\approx \frac{1}{M}\cdot\mathrm{Var}[\theta]\left(1 + 2\frac{\rho}{1-\rho}\right) \qquad \text{(for large } M\text{)} && (14)\\
&= \frac{1}{M}\cdot\mathrm{Var}[\theta]\cdot\frac{1+\rho}{1-\rho} && (15)
\end{aligned}$$

MH can be thought of as a random walk.

and

$$M_{\text{eff}} = M\cdot\frac{1-\rho}{1+\rho}. \qquad (16)$$

As $\rho \rightarrow 0$, we recover ordinary Monte Carlo. As $\rho \rightarrow 1$, the autocorrelation causes the variance of the estimator to blow up and the effective sample size to go to zero.

# What About Bias?

The mean squared error (MSE) of the estimator is determined by both the bias and the variance.

Ordinary Monte Carlo estimates are unbiased by construction, but MCMC estimates are only *asymptotically unbiased*.

Bias is introduced whenever the initial distribution $\pi_1(\boldsymbol{\theta})$ differs from the stationary distribution; i.e. in all practical cases!

Fortunately, the bias decays as $O(M^{-1})$ whereas the variance decays as $O(M^{-1/2})$, so asymptotically the MSE is dominated by the variance.

If you want to learn more, see Levin and Peres [2017], Meyn and Tweedie [2012], and STATS 318.

# How Can We Make Smarter Proposals?

- Metropolis-Hastings with a symmetric Gaussian proposal behaves (kind of) like a random walk.
- Neal [2012] argues that in $D$ dimensions, random walk MH needs $\underbrace{O(D^2)}_{\text{bad!}}$ iterations to get an independent sample.
- Can we develop more efficient transition distributions?
  - **Yes**! If we have more information about the log probability.
- For example, suppose that the log probability $\log p(\theta)$ is differentiable. We can use the gradient to make proposals that move farther and are more likely to be accepted.

→ can do this with continuous RVs.

# Metropolis Adjusted Langevin Algorithm (MALA)

The *Metropolis-Adjusted Langevin Algorithm* uses the gradient of the log probability to make asymmetric proposals,

$$q(\boldsymbol{\theta}' \mid \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta} + \tau \nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}, \boldsymbol{X}), 2\tau^2 \boldsymbol{I}) \tag{17}$$

**Note:** $q(\boldsymbol{\theta}' \mid \boldsymbol{\theta}) \neq q(\boldsymbol{\theta} \mid \boldsymbol{\theta}')$! To calculate the acceptance probability, you need the gradient at both points.

MALA can be motivated as a discrete-time approximation to the *Langevin* diffusion, a continuous-time stochastic differential equation for modeling molecular dynamics.

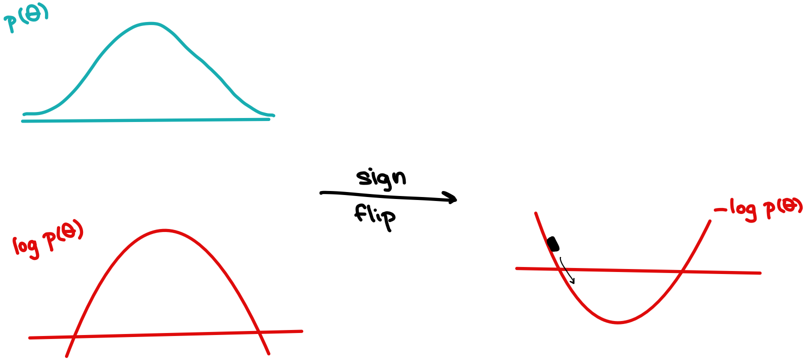
In high dimensions, the extra information provided by the gradient can lead to much more efficient chains. Neal argues that MALA needs $O(D^{4/3})$ computation to produce an independent sample.

But why stop at one gradient step?

# Hamiltonian Monte Carlo

**Reference:** Neal [2012] *MCMC using Hamiltonian dynamics.*

**Idea:** *Think of negative log probability as an energy landscape. Now imagine a puck sliding around on this bumpy surface. Give it random kicks; it will tend to slide downhill toward points of low potential energy (high probability). Each kick can displace the puck by a large amount. Done properly, the puck will visit points with probability proportional to the posterior probability.*



# Notation

Following Neal [2012], let

- $\boldsymbol{q} \in \mathbb{R}^D$ denote the *position*; i.e. the current parameters (previously $\theta$)
- $\boldsymbol{p} \in \mathbb{R}^D$ denote the *momentum*; auxiliary variables that we don't care about, but which are necessary for HMC.
- $\boldsymbol{z} = [\boldsymbol{q}, \boldsymbol{p}]^\top \in \mathbb{R}^{2D}$ denote the combined *state of the system*.
- $\boldsymbol{M}$ denote the *mass matrix*, another artificial construct. Typically, this will be $m\boldsymbol{I}$
- $U(\boldsymbol{q})$ denote the *potential energy*
- $K(\boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}^\top \boldsymbol{M}^{-1}\boldsymbol{p}$ denote the *kinetic energy*

# Hamiltonian Dynamics

The *Hamiltonian* is the sum of the potential $H(\boldsymbol{q}, \boldsymbol{p}) = U(\boldsymbol{q}) + K(\boldsymbol{p}) = U(\boldsymbol{q}) + \frac{1}{2}\boldsymbol{p}^\top \boldsymbol{M}^{-1}\boldsymbol{p}$.

The partial derivatives determine how the state evolves over time,

$$\frac{\mathrm{d}q_d}{\mathrm{d}t} = \frac{\partial H}{\partial p_d} = [\boldsymbol{M}^{-1}\boldsymbol{p}]_d \tag{18}$$

$$\frac{\mathrm{d}p_d}{\mathrm{d}t} = -\frac{\partial H}{\partial q_d} = -\frac{\partial U}{\partial q_d} \tag{19}$$

for $d = 1, \dots, D$.

Compactly,

$$\frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t} = \boldsymbol{J}\nabla H(z) \tag{20}$$

where

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{0}, \boldsymbol{I} \\ -\boldsymbol{I}, \boldsymbol{0} \end{bmatrix} \tag{21}$$

# One Dimensional Example

Consider the case where $D = 1$ and $U(q) = \frac{1}{2}q^2$ and $K(p) = \frac{1}{2}p^2$.

The partial derivatives are

$$\frac{\partial H}{\partial p} = p \tag{22}$$

$$-\frac{\partial H}{\partial q} = -q \tag{23}$$

so

$$\frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t} = \boldsymbol{Jz}. \tag{24}$$

This is a linear dynamical system, and the state at time $t + \Delta t$ is $\boldsymbol{z}(t + \Delta t) = \overbrace{e^{J\Delta t}}^{\text{matrix exponential}}\boldsymbol{z}(t)$.

Since $\boldsymbol{J}\Delta t$ is skew-symmetric, the matrix exponential $e^{J\Delta t}$ is orthogonal. More precisely, $\boldsymbol{z}(t + \Delta t)$ is a rotation about the origin of $\boldsymbol{z}(t)$.

# Properties of Hamiltonian Dynamics

1. **Reversibility:** The mapping from $z(t) \to z(t + \Delta t)$ is one-to-one and invertible. To go from $z(t + \Delta t)$ to $z(t)$, negate $p(t + \Delta t)$, apply the the Hamiltonian dynamics for $\Delta t$ time, and negate the momentum again.

2. **Conservation of energy:** The Hamiltonian (which is the total energy in a closed system) is conserved,

$$\begin{aligned} \frac{\mathrm{d}H}{\mathrm{d}t} &= \sum_{d=1}^{D} \frac{\mathrm{d}q_d}{\mathrm{d}t}\frac{\partial H}{\partial q_d} + \frac{\mathrm{d}p_d}{\mathrm{d}t}\frac{\partial H}{\partial p_d} \quad \Bigg] \text{ Chain rule in action} \qquad (25) \\ &= \sum_{d=1}^{D} \frac{\partial H}{\partial p_d}\frac{\partial H}{\partial q_d} - \frac{\partial H}{\mathrm{d}q_d}\frac{\partial H}{\partial p_d} = 0. \qquad (26) \end{aligned}$$

# Properties of Hamiltonian Dynamics II

3. **Volume preserving:** A set in $(q, p)$ space will have the same volume after being mapped through Hamiltonian dynamics. This follows from the fact that the divergence of the vector field is zero everywhere:

$$\operatorname{div}\frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t} = \sum_{d=1}^{D}\frac{\partial}{\partial q_d}\frac{\mathrm{d}q_d}{\mathrm{d}t} + \frac{\partial}{\partial p_d}\frac{\mathrm{d}p_d}{\mathrm{d}t} = \sum_{d=1}^{D}\frac{\partial}{\partial q_d}\frac{\partial H}{\partial p_d} - \frac{\partial}{\partial p_d}\frac{\partial H}{\partial q_d} = \sum_{d=1}^{D}\frac{\partial^2 H}{\partial q_d \partial p_d} - \frac{\partial^2 H}{\partial q_d \partial p_d} = 0. \tag{27}$$

4. **Sympleticness** Let $B$ be the Jacobian of the transformation from $\boldsymbol{z}(t) \to \boldsymbol{z}(t + \Delta t)$. It turns out that,

$$\boldsymbol{B}^\top \boldsymbol{J}^{-1}\boldsymbol{B} = \boldsymbol{J}^{-1} \tag{28}$$

which implies that $|\boldsymbol{B}^\top||\boldsymbol{J}^{-1}||\boldsymbol{B}| = |\boldsymbol{J}^{-1}|$ and thus $|\boldsymbol{B}| = 1$. I.e. the dynamics preserve volume.

# Discretizing Hamilton's Equations

The properties above apply to the *continuous time* Hamiltonian dynamics. Can we maintain them in practice?

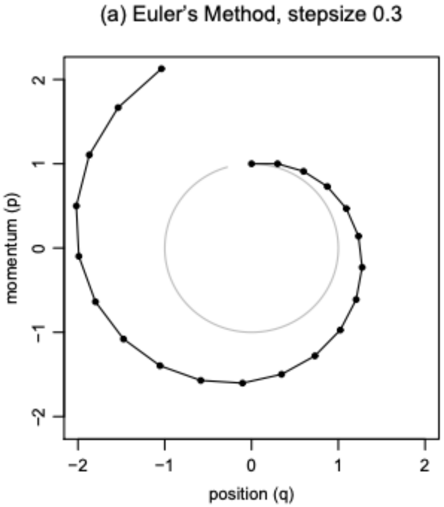
**Idea:** In practice, to simulate $\Delta t$ elapsed time, we break it down into steps of size $\Delta t/\epsilon$.

**Euler's method:** Update the state as,

$$\boldsymbol{z}(t+\epsilon) = \boldsymbol{z}(t) + \epsilon \frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t}\Big|_{\boldsymbol{z}(t)} \tag{29}$$

$$\Rightarrow p_d(t+\epsilon) = p_d(t) - \epsilon \frac{\partial U}{\partial q_d}\Big|_{\boldsymbol{q}(t)} \tag{30}$$

$$q_d(t+\epsilon) = q_d(t) + \epsilon \frac{p_d(t)}{m_d} \tag{31}$$



Simple Euler integration does not preserve volume: trajectories eventually diverge, even with small $\epsilon$.

# The Leapfrog Integrator

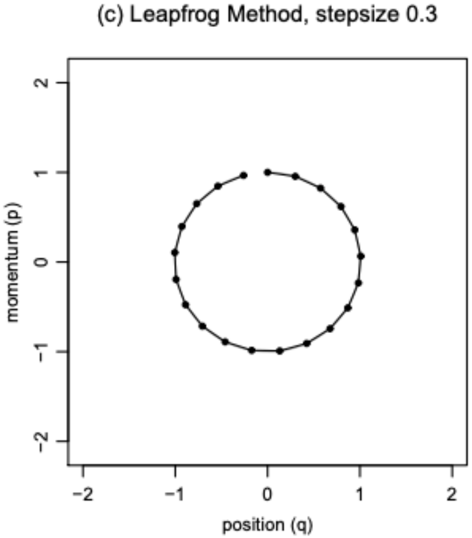
Instead, alternate updates of $p$ and $q$

$$p_d(t+\tfrac{\epsilon}{2}) = p_d(t) - \frac{\epsilon}{2}\frac{\partial U}{\partial q_d}\bigg|_{\boldsymbol{q}(t)} \tag{32}$$

$$q_d(t+\epsilon) = q_d(t) + \epsilon \frac{p_d(t+\tfrac{\epsilon}{2})}{m_d} \tag{33}$$

$$p_d(t+\epsilon) = p_d(t+\tfrac{\epsilon}{2}) - \frac{\epsilon}{2}\frac{\partial U}{\partial q_d}\bigg|_{\boldsymbol{q}(t+\epsilon)} \tag{34}$$

$$\tag{35}$$



Each update is a *shear transformation* in which only some variables change, by amounts that depend on the other, fixed variables. The determinant of such a transformation is one, so it preserves volume.

# Using Hamiltonian Dynamics for Posterior Inference

Define a joint distribution on positions and momenta as,

$$p(\boldsymbol{q}, \boldsymbol{p}) \propto \exp\left\{-H(\boldsymbol{q}, \boldsymbol{p})\right\} \propto \exp\left\{-U(\boldsymbol{q}) - K(\boldsymbol{p})\right\}. \tag{36}$$

Now let $U(\boldsymbol{q}) = -\log p(\boldsymbol{\theta} = \boldsymbol{q}, \boldsymbol{X})$ be the *negative* log joint probability. Then,

$$p(\boldsymbol{q}, \boldsymbol{p}) = p(\boldsymbol{\theta} = \boldsymbol{q} \mid \boldsymbol{X}) \times p(\boldsymbol{p}) \tag{37}$$

Samples of $\boldsymbol{q}$ will be marginally distributed according to the posterior $p(\boldsymbol{\theta} = \boldsymbol{q} \mid \boldsymbol{X})$.

Samples of $\boldsymbol{p}$ will be marginally distributed $p(\boldsymbol{p}) = \frac{\exp\{-K(\boldsymbol{p})\}}{\int_{\mathbb{R}^D} \exp\{-K(\boldsymbol{p})\}\,\mathrm{d}\boldsymbol{p}}$. These are *auxiliary variables* that we don't really care about—they're just there to help us construct MH proposals.

We choose $K(\boldsymbol{p})$ so $p(\boldsymbol{p})$ is convenient; e.g. if $K(\boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}^\top \boldsymbol{M}^{-1}\boldsymbol{p}$ then

$$p(\boldsymbol{p}) = \mathcal{N}(\boldsymbol{p} \mid \mathbf{0}, \boldsymbol{M}). \tag{38}$$

# Hamiltonian Monte Carlo (HMC)

**Hamiltonian Monte Carlo (HMC)** is Metropolis-Hastings on the joint distribution of $(\boldsymbol{q}, \boldsymbol{p})$ with proposals based on Hamiltonian dynamics.

Starting at point $(\boldsymbol{q}', \boldsymbol{p}')$, sample the proposal distribution:

1. Throw away $\boldsymbol{p}'$ and sample new momenta from their marginal distribution $\boldsymbol{p} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{M})$.
2. Approximate Hamiltonian dynamics on $(\boldsymbol{q}, \boldsymbol{p})$ for $\Delta t$ time using $L = \Delta t / \epsilon$ Leapfrog steps each of size $\epsilon$. Call the resulting point $(\boldsymbol{q}, \boldsymbol{p})$.
3. Flip the momentum $\boldsymbol{p} \leftarrow -\boldsymbol{p}$ to make the proposal symmetric.

Then accept the proposed point $(\boldsymbol{q}, \boldsymbol{p})$ with probability,

$$a((\boldsymbol{q}', \boldsymbol{p}') \to (\boldsymbol{q}, \boldsymbol{p})) = \min\left\{1, \frac{\exp\{-H(\boldsymbol{q}, \boldsymbol{p})\}\, q(\boldsymbol{q}', \boldsymbol{p}' \mid \boldsymbol{q}, \boldsymbol{p})}{\exp\{-H(\boldsymbol{q}', \boldsymbol{p}')\} q(\boldsymbol{q}, \boldsymbol{p} \mid \boldsymbol{q}', \boldsymbol{p}')}\right\} = \min\left\{1, \frac{\exp\{-H(\boldsymbol{q}, \boldsymbol{p})\}}{\exp\{-H(\boldsymbol{q}', \boldsymbol{p}')\}}\right\}. \quad (39)$$

If the Hamiltonian dynamics were simulated exactly, HMC would always accept. In practice, differences arise from numerical integration errors.

# HMC Dynamics on a Correlated 2D Gaussian

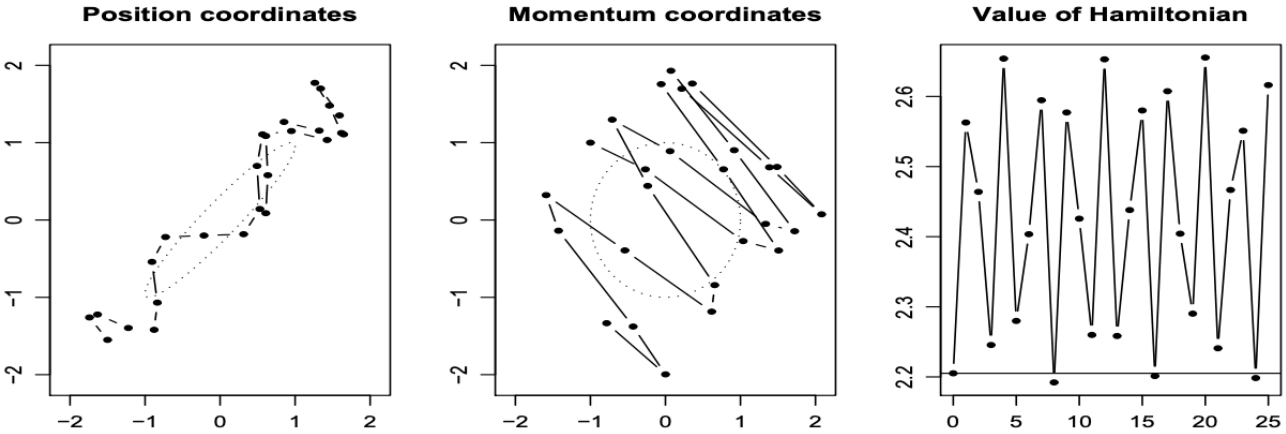


Figure 3: A trajectory for a 2D Gaussian distribution, simulated using 25 leapfrog steps with a stepsize of 0.25. The ellipses plotted are one standard deviation from the means. The initial state had $q = [-1.50, -1.55]^T$ and $p = [-1, 1]^T$.

# HMC vs Random Walk MH

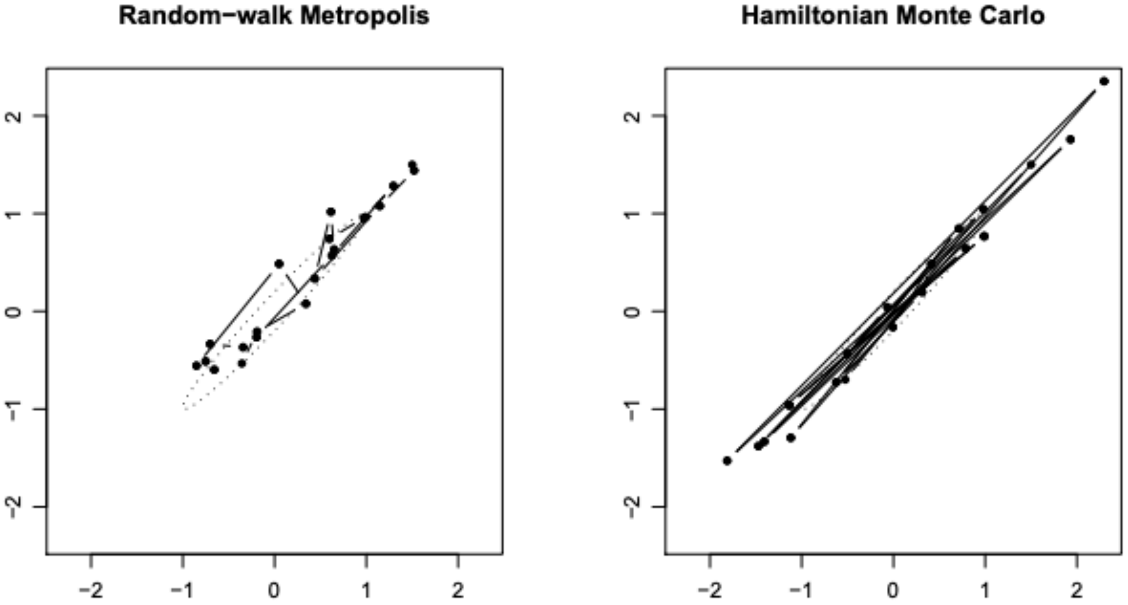


Figure 4: Twenty iterations of the random-walk Metropolis method (with 20 updates per iteration) and of the Hamiltonian Monte Carlo method (with 20 leapfrog steps per trajectory) for a 2D Gaussian distribution with marginal standard deviations of one and correlation 0.98. Only the two position coordinates are plotted, with ellipses drawn one standard deviation away from the mean.

# HMC vs Random Walk MH II

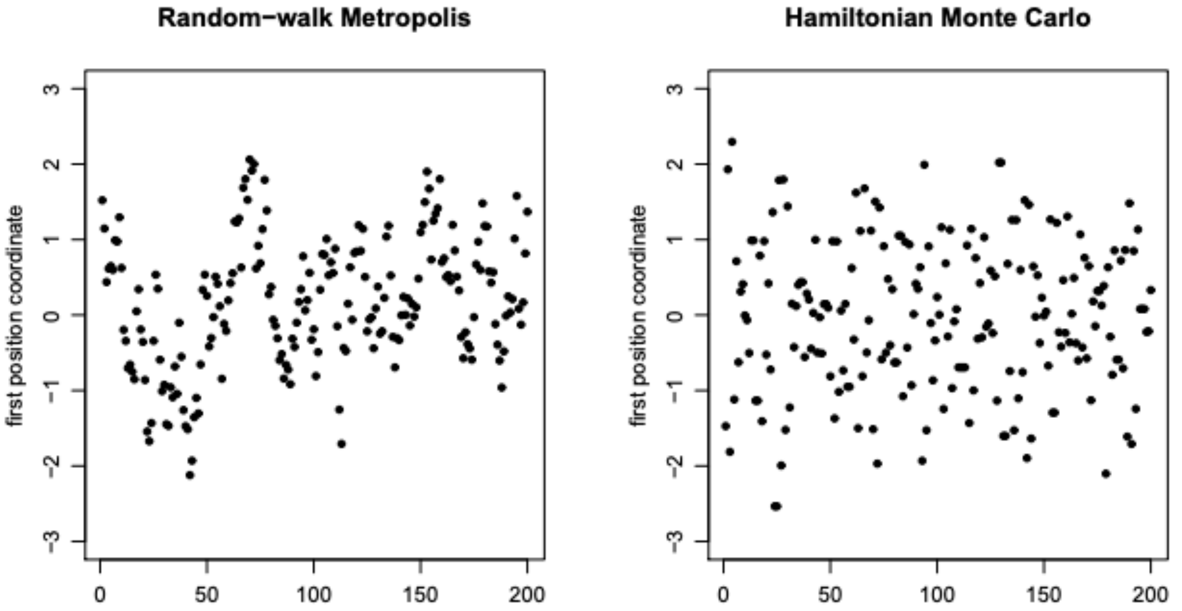


Figure 5: Two hundred iterations, starting with the twenty iterations shown above, with only the first position coordinate plotted.

# HMC vs Random Walk MH in 100D

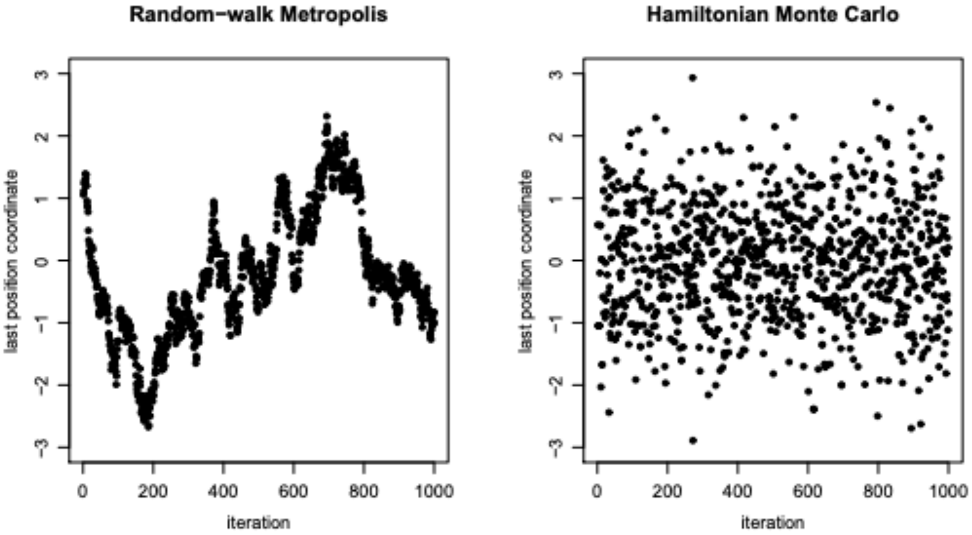
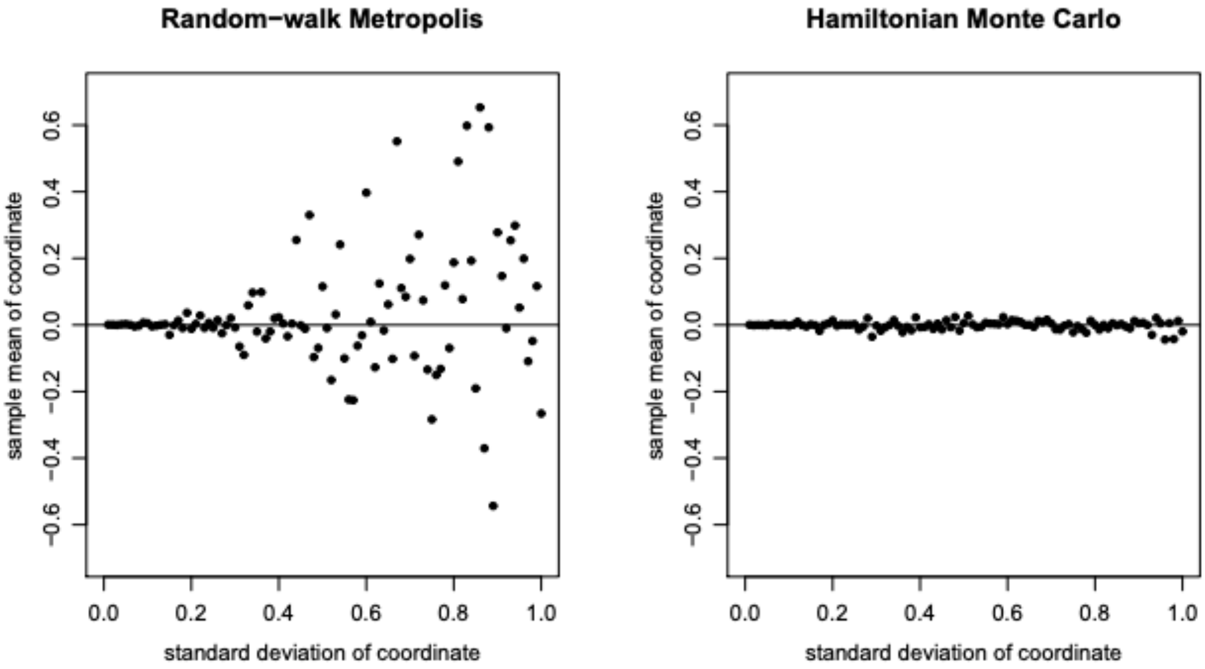


Figure 6: Values for the variable with largest standard deviation for the 100-dimensional example, from a random-walk Metropolis run and an HMC run with $L = 150$. To match computation time, 150 updates were counted as one iteration for random-walk Metropolis.

# HMC vs Random Walk MH in 100D II



# Benefits of Avoiding Random Walks

- To maintain reasonably high acceptance probability, random walk MH needs proposal standard deviation (s.d.) comparable to the s.d. in the most constrained dimension (0.14 in the 2D Gaussian example and 0.01 in the 100D example).

- Num. iterations needed for RW-MH to reach an approximately independent state is proportional to the *square* of the largest standard deviation to the smallest; i.e. to the condition number of the covariance matrix.

- In contrast, integrating the Hamiltonian makes many steps in the same direction. The number of integration steps to reach an independent state is about the ratio of the largest s.d. to the smallest; i.e. the square root of the condition number.

- Neal [2012] argues that the number of leapfrog updates to reach an independent point scales as $O(D^{5/4})$, better than the $O(D^2)$ and $O(D^{4/3})$ estimates for random walk MH and MALA, respectively.

- However, we still need to tune the step size $\epsilon$ to be comparable to the smallest s.d.

# Adapting the step size

- A simple strategy is to tune the step size adaptively during the initial run of the Markov chain.
- For example, set a target acceptance rate (Neal argues that it should be around 0.65), then increase the step size if you're accepting too often and decrease if you're rejecting too often.
- Andrieu and Thoms [2008] proposed a widely-used multiplicative update scheme; it is the default in `tfp.mcmc.SimpleStepSizeAdaptation`. Pyro defaults to a similar "dual averaging" scheme.
- The **No U-Turn Sampler (NUTS)** [Hoffman and Gelman, 2014] adapts the distance traveled in response to the curvature of the target density. Conceptually, it continues until the trajectory turns back on itself (hence the name, "No U-Turn")
- More details can be found in Betancourt [2017].

# Demos

https://chi-feng.github.io/mcmc-demo/app.html

# References I

Radford M Neal. MCMC using Hamiltonian dynamics. June 2012.

Michael Betancourt. A conceptual introduction to Hamiltonian Monte Carlo. January 2017.

Charles J Geyer. Introduction to Markov chain Monte Carlo. *Handbook of Markov chain Monte Carlo*, 20116022:45, 2011.

David A Levin and Yuval Peres. *Markov chains and mixing times*. American Mathematical Soc., 2017.

Sean P Meyn and Richard L Tweedie. *Markov chains and stochastic stability*. Springer Science & Business Media, 2012.

Christophe Andrieu and Johannes Thoms. A tutorial on adaptive MCMC. *Stat. Comput.*, 18(4):343–373, December 2008.

Matthew D Hoffman and Andrew Gelman. The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, 2014.