# Advanced MCMC: Hamiltonian Monte Carlo

Ed Herbst

November 19, 2020

# Hamiltonian Monte Carlo

We previously spoke about how much the efficacy of MCMC algorithms depended on the shape of the posterior.

We're going to talk about posterior simulators that make that idea explicit.

In particular, the Hamiltonian Monte Carlo sampler described in Neal (2011).

# Details

- Hamiltonian Monte Carlo adapts methods from the study of molecular dynamics: *simulate the motion of molecules based on Newton's laws*.

- The systems which describe the evolution of molecules over time exhibit so-called *Hamiltonian dynamics*

- The state of the system at any point in time is summarized by a pair $(\theta, p)$. $\theta$ is the location of the molecule, while $p$ gives its momentum (mass times velocity).

# Some Math

The evolution $\theta$ and $p$ is governed by set of differential equations. These differential equations are determined by the *Hamiltonian*, $H(\theta, p)$:

$$H(\theta, p) = \text{Kinetic Energy } + \text{Potential Energy.} \tag{1}$$

In physical systems, potential and kinetic energy are obviously instrinsic features of the environment.

Our world:

- the potential energy will be pinned down by the density $\pi(\theta)$
- The kinetic energy $p$ is unrestricted, look at different kinds

# Explicit Formulation of Hamiltonian

$$\begin{aligned} H(\theta, p) &= -\log \pi(p|\theta) - \log \pi(\theta). \\ &= K(p, \theta) + V(\theta) \end{aligned}$$

Then the joint density of $(\theta, p)$ is given by $e^{-H(\theta,p)}$

As mentioned above, the $\theta, p$ evolve over time according to the Hamiltonian equations:

$$\frac{d\theta}{dt} = \frac{\partial H}{\partial p} = \frac{\partial K}{\partial p} \tag{2}$$

$$\frac{dp}{dt} = -\frac{\partial H}{\partial \theta} = -\frac{\partial K}{\partial \theta} - \frac{\partial V}{\partial \theta} \tag{3}$$

Note that $-\partial V/\partial\theta = \nabla \log \pi(\theta)$, the gradient of the log of the target density.

# A proposal distribution

Consider now the mapping $Q_{\bar{t}}(p, \theta) = (p(\bar{t}), \theta(\bar{t}))$, the mapping of the Hamiltonian for initial state $(p, \theta)$ at time $t = \bar{t}$.

The basic idea is to use this as a proposal distribution in an Metropolis-Hastings chain which targets $e^{-H(p,\theta)}$.

# Some Facts About Q

- $Q_{\bar{t}}(p, \theta)$ is deterministic and one-to-one. So $Q$ has an inverse. Neal makes the point that when $K(p) = K(-p)$, the inverse can be obtained via $-Q_{\bar{t}}(-p, \theta)$.

- Prior to approximation $H(p, \theta) = H(Q_{\bar{t}}(p, \theta))$, so the proposal is always accepted, for any $\bar{t}$. This is because of the principal of the conservation of energy (again, high school physics.) In implementation, the discretization necessary to solve the ODEs in a general setting means $H(p, \theta)$ will not necessarily be constant.

- The jacobian of transformation associated with $Q_{\bar{t}}(p, \theta) = 1$.

# Some Intuition

The Hamiltonian Monte Carlo generates a Markov chain for the pair $(\theta^i, p^i)$, $i = 1, \ldots, N$.

Start with the case in which $\pi(p|\theta)$ does not depend on $\theta$. Thus,

$$H(\theta, p) = -\log \pi(p) - \log \pi(\theta).$$

The Hamiltonian equations describe the evolution of $\theta$ and $p$ subject to the constraint that $H(\theta, p)$ (the sum of kinetic and potential energy) stays constant.

- if $\pi(p^i) > \pi(p^{i-1})$, it has to be the case that $\pi(\theta^i) < \pi(\theta^{i-1})$.
- One way of thinking about this relationship is that $p^i$ determines the level of the posterior contour from which we are sampling and $\theta^i$ is the location on the contour.
- As the algorithm iterates over $i$ we are first changing the level of the contour and then we adjust $\theta^i$.

# An Example

Suppose that $\pi(\theta)$ corresponds to a $N(\mu, \sigma^2)$. Thus,

$$\log \pi(\theta) = -\frac{1}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}(\theta - \mu)^2.$$

Suppose we would generate the sequence $\theta^i$ by direct sampling. Then the level of the log posterior density would vary according to

$$\log \pi(\theta^i) = -\frac{1}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}(\theta^i - \mu)^2$$

with

$$\begin{aligned} \mathbb{E}[\log \pi(\theta^i)] &= -\frac{1}{2} \ln(2\pi\sigma^2) - \frac{1}{2} \\ \mathbb{V}[\log \pi(\theta^i)] &= \frac{1}{2} \end{aligned}$$

# Example, Continued

Thus, if $\theta^i$ is generated by direct sampling, we should observe that the draws are *iid* and that the variance of the log density is $1/2$.

In the algorithm below, we will generate $p^i$ by direct sampling from $\pi(p)$, which we set to a $N(0, M)$, where $M = \mathbb{V}_\pi[\theta]^{-1} = \sigma^{-2}$.

Note that in this simplified setting, the choice of $M$ only affects the level

$$\mathbb{E}[\log \pi(p^i)] = \frac{1}{2}\ln(2\pi\sigma^2) - \frac{1}{2},$$

but not the variance in the fluctuations around this level.

# Example, Continued

We are now in a position to solve the Hamiltonian equations, which take the form:

$$\begin{aligned}\dot{\theta} &= \frac{1}{M} p \\ \dot{p} &= -\frac{1}{\sigma^2}(\theta - \mu)\end{aligned}$$

It can be verified that the solution to this system of differential equations takes the form

$$\begin{aligned}\theta(t) &= \mu + \rho \cos\left(\alpha + t/\sqrt{\sigma^2 M}\right) \\ p(t) &= -\rho \frac{\sqrt{M}}{\sigma} \sin\left(\alpha + t/\sqrt{\sigma^2 M}\right).\end{aligned}$$

# Example, Continued

Here the parameters $(\rho, \alpha)$ are determined by the initial conditions:

$$\theta(0) = \mu + \rho \cos(\alpha), \quad p(0) = -\rho \frac{\sqrt{M}}{\sigma} \sin(\alpha),$$

which leads to

$$\rho = -p(0) \frac{\sigma}{\sqrt{M}} \frac{1}{\sin \alpha}, \quad \alpha = \arctan \left[ -\frac{p(0)/\sqrt{M}}{(\theta(0) - \mu)/\sigma} \right].$$

# Example, Continued

We can now specify the function $Q_{\bar{t}}(p, \theta)$. Using the fact that

$$\begin{aligned} \cos(x+y) &= \cos x \cos y - \sin x \sin y \\ \sin(x+y) &= \sin x \cos y + \cos x \sin y, \end{aligned}$$

we obtain

$$\begin{aligned} \theta(t) - \mu &= \rho \cos(\alpha) \cos(t/\sqrt{\sigma^2 M}) - \rho \sin(\alpha) \sin(t/\sqrt{\sigma^2 M}) \\ &= (\theta(0) - \mu) \cos(t/\sqrt{\sigma^2 M}) + p(0) \frac{\sigma}{\sqrt{M}} \sin(t/\sqrt{\sigma^2 M}) \\ p(t) &= -\rho \frac{\sqrt{M}}{\sigma} \sin(\alpha) \cos(t/\sqrt{\sigma^2 M}) - \rho \frac{\sqrt{M}}{\sigma} \cos(\alpha) \sin(t/\sqrt{\sigma^2 M \\ &= p(0) \cos(t/\sqrt{\sigma^2 M}) - (\theta(0) - \mu) \frac{\sqrt{M}}{\sigma} \sin(t/\sqrt{\sigma^2 M}). \end{aligned}$$

# Example, Continued

In matrix form, the equations can be written as

$$\left[\begin{array}{c}\theta(t)-\mu \\ p(t)\end{array}\right]=\left[\begin{array}{cc}\cos(t/\sqrt{\sigma^2 M}) & \frac{\sigma}{\sqrt{M}}\sin(t/\sqrt{\sigma^2 M}) \\ -\frac{\sqrt{M}}{\sigma}\sin(t/\sqrt{\sigma^2 M}) & \cos(t/\sqrt{\sigma^2 M})\end{array}\right]\left[\begin{array}{c}\theta(0)-\mu \\ p(0)\end{array}\right]$$

# Example, Continued

It is now easy to see that the jacobian of transformation associated with $Q_{\bar{t}}(p,\theta) = 1$. Moreover,

$$
\begin{aligned}
&\left[\begin{array}{cc} \cos(t/\sqrt{\sigma^2 M}) & \frac{\sigma}{\sqrt{M}}\sin(t/\sqrt{\sigma^2 M}) \\ -\frac{\sqrt{M}}{\sigma}\sin(t/\sqrt{\sigma^2 M}) & \cos(t/\sqrt{\sigma^2 M}) \end{array}\right]\left[\begin{array}{c} \theta(t)-\mu \\ -p(t) \end{array}\right] \\
&= \left[\begin{array}{cc} \cos(t/\sqrt{\sigma^2 M}) & \frac{\sigma}{\sqrt{M}}\sin(t/\sqrt{\sigma^2 M}) \\ -\frac{\sqrt{M}}{\sigma}\sin(t/\sqrt{\sigma^2 M}) & \cos(t/\sqrt{\sigma^2 M}) \end{array}\right] \\
&\quad\times\left[\begin{array}{cc} \cos(t/\sqrt{\sigma^2 M}) & \frac{\sigma}{\sqrt{M}}\sin(t/\sqrt{\sigma^2 M}) \\ \frac{\sqrt{M}}{\sigma}\sin(t/\sqrt{\sigma^2 M}) & -\cos(t/\sqrt{\sigma^2 M}) \end{array}\right]\left[\begin{array}{c} \theta(0)-\mu \\ p(0) \end{array}\right] \\
&= \left[\begin{array}{cc} 1 & 0 \\ 0 & -1 \end{array}\right]\left[\begin{array}{c} \theta(0)-\mu \\ p(0) \end{array}\right].
\end{aligned}
$$

# Example, Continued

Given the solution to the Hamiltonian equation, we can verify that $H(\theta, p)$ stays indeed constant over time:

$$\begin{aligned} H\big(\theta(t), p(t)\big) &= \frac{1}{2}\ln(2\pi M) + \frac{1}{2M}\rho^2\frac{M}{\sigma^2}\sin^2(\cdot) + \frac{1}{2}\ln(2\pi\sigma^2) + \frac{1}{2\sigma^2}\rho^2 \\ &= \frac{1}{2}\left(\ln(2\pi M) + \frac{1}{2}\ln(2\pi\sigma^2) + \frac{\rho^2}{\sigma^2}\right). \end{aligned}$$

Note that the initial values enter the level of “energy” through the parameter $\rho$.

# The Algorithm

The subsequent algorithm generates the proposal distribution using the Hamiltonian equations. In particular, it will start from

$$\theta(0) = \theta^{i-1}, \quad p(0) \sim N(0, M).$$

Using the Hamiltonian equations over a time interval $t$, we obtain (in slight abuse of notation)

$$\begin{aligned}
\theta^* - \mu &= \cos(t/\sqrt{\sigma^2 M})(\theta^{i-1} - \mu) + \sigma \sin(t/\sqrt{\sigma^2 M}) N(0,1) \\
p^* &= -\frac{\sqrt{M}}{\sigma} \sin(t/\sqrt{\sigma^2 M})(\theta^{i-1} - \mu) + \sqrt{M} \cos(t/\sqrt{\sigma^2 M}) \cdot N(0,1)
\end{aligned}$$

# The Chain

The acceptance probability for the proposed draw depends on the Hamiltonian. But by construction, the dynamics of $\theta$ and $p$ are such that the Hamiltonian is constant, meaning the draw is always accepted. So, we obtain the following autoregressive law of motion:

$$\theta^i - \mu = \cos(t/\sqrt{\sigma^2 M})(\theta^{i-1} - \mu) + \sigma \sin(t/\sqrt{\sigma^2 M})\epsilon^i, \quad \epsilon^i \sim N(0,1)$$

Because $\sin^2 + \cos^2 = 1$, it is easy to see that the stationary distribution of this AR(1) process is the target posterior $N(\mu, \sigma^2)$. The persistence of the chain depends on the time period $t$. When $t$ is set to

$$t^{iid} = \frac{\pi}{2}\sqrt{\sigma^2 M},$$

the chain produces uncorrelated draws of $\theta$.

# Discretization

To actual obtain (an approximate) $\hat{Q}_{\bar{t}}(p, \theta)$, Hamilton's equations are discretized in time.

To discretize, pick a step size $\epsilon = \Delta t$. Then applying $\bar{t}/\epsilon_t = L$ steps (assumed to be an integer) of the discretized Hamiltonian obtains a draw from $\hat{Q}_{\bar{t}}(p, \theta)$.

There are many ways approaches to discretize the ODEs!

We're going to use the leapfrog method because it has nice simulation properties and the resulting density maintains the important statistical properties

# The Leapfrog Simulator

Under the assumption that $p|\theta$ is normally distributed with mean 0 and variance $M^{-1}$:

$$\log \pi(p|\theta) \propto -\frac{p' M^{-1} p}{2}, \tag{4}$$

$\partial K/\partial\theta = 0$ and the calculations are considerably simplified. *(But it's worth reiterating that $\pi(p|\theta)$ is choice.)* Given $\theta(t)$ and $p(t)$, a step of size $\epsilon$ is taken using the following three steps.

$$\begin{aligned}
p(t+\epsilon/2) &= p(t) + \frac{\epsilon}{2}\nabla \log \pi(\theta(t)) \\
\theta(t+\epsilon) &= \theta(t) + \epsilon M^{-1} p(t+\epsilon/2) \\
p(t+\epsilon) &= p(t+\epsilon/2) + \frac{\epsilon}{2}\nabla \log \pi(\theta(t+\epsilon)).
\end{aligned}$$

# The Details

This leads to the following algorithm (the user needs to specify $\epsilon$ and $L$):

## Algorithm

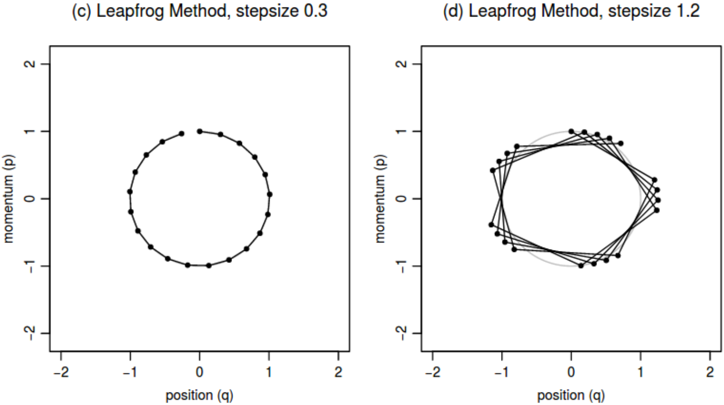
*Modified Leapfrog Simulator, $\hat{Q}_{L\epsilon}(\cdot,\cdot)$. This algorithm takes as an inputs $(\theta, p)$ and returns $(\theta^*, p^*)$, after approximating Hamiltonion dynamics using step size $\epsilon$ for L steps.*

1. *For $l = 1, \ldots, L$. Set*

   1. $p \leftarrow p + \epsilon \nabla \log \pi(\theta)/2$
   2. $\theta \leftarrow \theta + \epsilon M^{-1} p$
   3. $p \leftarrow p + \epsilon \nabla \log \pi(\theta)/2$

2. *Return $\theta^* = \theta$ and $p^* = -p$. (The negation is not a typo!)*

# Leapfrog Simulator



# Back to the Example

For the example given above, we have

$$\begin{bmatrix} \theta^* - \mu \\ p^* \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 - \frac{\epsilon^2}{2M\sigma^2} & \frac{\epsilon}{M} \\ -\frac{\epsilon}{\sigma^2} + \frac{\epsilon^3}{4M\sigma^4} & 1 - \frac{\epsilon^2}{2M\sigma^2} \end{bmatrix}^L \begin{bmatrix} \theta^{i-1} - \mu \\ \hat{p} \end{bmatrix}$$

The stability of this system is determined by the eigenvalues $\lambda_1$ and $\lambda_2$, where

$$\begin{aligned} \lambda_1 &= \tfrac{2M\sigma^2 - \epsilon^2}{2M\sigma^2} + \tfrac{\epsilon}{2M\sigma^2}\sqrt{-4M\sigma^2 + \epsilon^2}, \\ \lambda_2 &= \tfrac{2M\sigma^2 - \epsilon^2}{2M\sigma^2} - \tfrac{\epsilon}{2M\sigma^2}\sqrt{-4M\sigma^2 + \epsilon^2} \end{aligned}$$

# Example, Continued

Case 1: $\epsilon > \sqrt{4M\sigma^2}$. Then

$$\begin{aligned}
\lambda_2 &= 1 - \frac{\epsilon^2}{2M\sigma^2} - \frac{\epsilon}{4M\sigma^2}\sqrt{-4M\sigma^2 + \epsilon^2} \\
&< 1 - \frac{\epsilon^2}{2M\sigma^2} \\
&< 1 - \frac{4M\sigma^2}{2M\sigma^2} \\
&< -1
\end{aligned}$$

Case 2: $\epsilon \leq \sqrt{4M\sigma^2}$. Then the eigenvalues are complex conjugates. A quick calculation reveals that $|\lambda_i| = 1$ for $i = 1, 2$. So the system is stable.

# Remarks:

- It is clear from the above calculations that if the step size is too big ($> 2\sqrt{M\sigma^2}$), $|\lambda_2| > 1$, so the leapfrog simulator will explode.
- On the other hand, step size can't really be "too small" in the sense that the system will be stable for any $\epsilon < 4\sqrt{M\sigma^2}$.
- Setting $M = \sigma^{-2}$, is optimal in the sense that the stability of the system is so no longer dependent on any features of the posterior.
- Finally, note that the number of steps, $L$, is irrelevant with respect to the stability properties of the simulator.

# What is the Right Acceptance Rate?

For any Metropolis-style algorithm for $x \sim \pi$ with proposal density $q(x^*|x)$, note the expected (unbounded) acceptance probability is given by:

$$
\begin{aligned}
E\left[\frac{\pi(x^*)q(x|x^*)}{\pi(x)q(x^*|x)}\right] &= \int\int \frac{\pi(x^*)q(x|x^*)}{\pi(x)q(x^*|x)} q(x^*|x)\pi(x)dxdx^* \\
&= \int \pi(x^*)\left[\int q(x|x^*)dx\right]dx^* \\
&= \int \pi(x^*)dx^* \\
&= 1. \qquad (5)
\end{aligned}
$$

# Some Analytics

Note that for both the Random Walk Metropolis-Hastings (RWMH) algorithm ($x = \theta$) and the Hamiltonian Monte Carlo algorithm ($x = (\theta, p)$), we have $q(x|x^*) = q(x^*|x)$. This means we can deduce that

$$E\left[\frac{\pi(x^*)}{\pi(x)}\right] = 1. \tag{6}$$

If we write the posterior in the form, $\pi(x) = \exp(-f(x))/Z$, then using (7), we deduce that

$$E\left[\exp\left\{-(f(x^*) - f(x))\right\}\right] = 1. \tag{7}$$

# Analytics, Continued

Defining $\Delta = f(x^*) - f(x)$ and using Jensen's inequality, we have:

$$E[\Delta] \geq 0. \tag{8}$$

In most interesting cases, the inequality is strict. Next, note also that $\alpha(x^*|x)$, the Metropolis-Hastings acceptance probability is given by:

$$\alpha(x^*|x) = \min\{1, \exp(-\Delta)\}. \tag{9}$$

# Analytics, Continued

The expected acceptance rate for an MH algorithm can be written as

$$
\begin{aligned}
\bar{\alpha} &= \int \alpha(x^*|x)q(x^*|x)\pi(x)dx^*dx \\
&= \int_{\Delta(x^*,x)<0} \alpha(x^*|x)q(x^*|x)\pi(x)dx^*dx + \int_{\Delta(x^*,x)>0} \alpha(x^*|x)q(x^*|x)\pi(x)dx^*dx \\
&= \int_{\Delta(x^*,x)<0} \alpha(x^*|x)q(x^*|x)\pi(x)dx^*dx + \int_{\Delta(x^*,x)>0} \alpha(x|x^*)q(x|x^*)\pi(x^*)dx^*dx \\
&= \int_{\Delta(x^*,x)<0} \alpha(x^*|x)q(x^*|x)\pi(x)dx^*dx + \int_{\Delta(x,x^*)<0} \alpha(x|x^*)q(x|x^*)\pi(x^*)dx^*dx \\
&= \int_{\Delta(x^*,x)<0} q(x^*|x)\pi(x)dx^*dx + \int_{\Delta(x,x^*)<0} q(x|x^*)\pi(x^*)dx^*dx \\
&= 2 \times \mathbb{P}(\Delta < 0). \qquad (10)
\end{aligned}
$$

# Properties of Delta

In what follows, we first derive a restriction on the relationship between the mean and variance of $\Delta$. If we assume the elements of $x$, $x_i$, are independent we can write,

$$f(x) = \sum_{d=1}^{D} f_d(x_d) \text{ and } \Delta = \sum_{d=1}^{D} \Delta_d, \tag{11}$$

where $D$ is the size of x. Taking a second-order expansion around $\exp(-\Delta_d)$, we have:

$$\exp(-\Delta_d) \approx 1 - \Delta_d + \Delta_d^2/2. \tag{12}$$

Using (11), $E[\Delta_d] \approx E[\Delta^{2}_{d}]/2.$ This means the mean of $\Delta_d$ is about half of the variance of $\Delta_d^2$. If we further assume that the proposals are independent, this means that $E[\Delta] \approx E[\Delta^2]/2$.

# Asymptotics

write $\mu = E[\Delta]$,

$$\Delta \sim N(\mu, 2\mu).$$

Using this, we can write:

$$\begin{aligned} \bar{\alpha} &= 2 \times \mathbb{P}(\Delta < 0) \\ &= 2\Phi\left(\frac{0-\mu}{\sqrt{2\mu}}\right) \\ &= 2\Phi(-\sqrt{\mu/2}). \end{aligned} \tag{13}$$

Finally, we use this to construct a (heuristic) cost of an algorithm:

$$\begin{aligned} \text{Cost of Alg.} &\propto \text{Avg. Number of Proposals Before Acceptance} \\ &\quad \times \text{Proposal Steps to "Independent" Point} \\ &= \frac{1}{\bar{\alpha}} \times \text{ Proposal Steps to "Independent" Point} \end{aligned} \tag{14}$$

# $\mu$ for RWMH.

Consider the RWMH applied to a independent multivariate normal distribution of size $D$. So $x \sim N(0, I_D)$ and $q(x^*|x) \sim N(x, c^2 I_D)$. Then:

$$E[\Delta] = E\left[\sum_{d=1}^{D} \Delta_d\right] = \sum_{d=1}^{D} E[\Delta_d] = \frac{c^2}{2} D.$$

Informally, this probability is decreasing as $D$ increases, since the expected difference $E[\Delta] \propto D$ with fixed $c$.

This means to maintain a given acceptance rate as $D$ increases, $c$ must shrink at rate $D^{-1/2}$.

# RWMH Continued

Finally for random walk processes, it is well known that the number of steps needed to reach a “nearly independent” point will be proportional to $c^{-2}$. We can then write the cost, $C_{RWMH}$, as:

$$C_{RWMH} = \frac{1}{2\Phi(-\sqrt{\mu/2})} \times \frac{1}{\mu}. \tag{15}$$

This cost is minimized when $\mu = 2.83$, which implies as $\bar{\alpha} = 0.234$, in accordance with well known results.

# $\mu$ for HMC.

Now $f(\theta, p)$ can be written as:

$$f(\theta, p) = \frac{\theta^2}{2} + \frac{p^2}{2}$$

Applying *many* leapfrog steps $L$ leads to:

$$E[\Delta] \approx D\epsilon^4.$$

This means that as $D$ increases, $\epsilon$ must shrink at rate $D^{-1/4}$ in order to maintain a reasonable acceptance rate.

The number of leapfrog updates to reach a nearly independent point will grow at rate $\epsilon$.

# HMC Acceptance Rate

Using the above relationship, $\mu \propto \epsilon^{1/4}$. Thus:

$$C_{HMC} = \frac{1}{2\Phi(-\sqrt{\mu/2})} \times \frac{1}{\mu^{1/4}}. \tag{16}$$

This cost is minimized when $\mu = 0.41$, which implies as $\bar{\alpha} = 0.65$. Hence, one should target an acceptance rate of about 65%.

# MCMC with (Euclidean) Hamiltonian Proposal.

We now combine the components to obtain a Hamiltonian Monte Carlo algorithm. Our goal is to obtain draws from $\pi(\theta)$. To do so, we target the augmented distribution:

$$\pi(\theta, p) = \pi(\theta) \underbrace{\pi(p|\theta)}_{\mathcal{N}(0,M).} = \exp\{-H(\theta, p)\}.$$

Given $(\theta^{i-1}, p^{i-1})$, we apply two Markov transition kernels in succession to get $(\theta^i, p^i)$. The two-step process first operates on $p$ and then $(\theta, p)$ jointly.

# A Two Step Process

1. First, draw $\hat{p} \sim \mathcal{N}(0, M)$. This draw is always “accepted” because this in fact coincides with true marginal distribution of $\pi(p|\theta)$.
2. Second, given $(\theta^{i-1}, \hat{p})$. Draw $(\theta^{*}, p^{*})$ by applying the (approximate) Hamiltonian mapping using the $\hat{Q}_{\bar{t}=L\epsilon}(\theta, p)$. Note that this is deterministic. We use the leap frog method with a twist.

Of course, in practice this is implemented in a one-step procedure. In fact, there is no need to store draws of $p$. In our applications, we should try to have $M$ approximate $\mathbb{V}_{\pi}[\theta]^{-1}$. And, of course, we also need an initial $\theta$.

# Some Details

An important thing to note about this proposal is that,

$$\hat{Q}_{L\epsilon}(\hat{Q}_{L\epsilon}(\theta^{i-1}, \hat{p})) = (\theta^{i-1}, \hat{p}).$$

In this sense the mapping is symmetric.

Moreover, the Jacobian associated with $Q_{L\epsilon}$ is 1. The upshot of this is that we don't need to account for $Q_{L\epsilon}$ in the Metropolis-Hastings acceptance probability, which is given by:

$$\alpha = \max\left\{1, \exp\left(H(\theta^{i-1}, \hat{p}) - H(\theta^*, p^*)\right)\right\} = \max\left\{1, \frac{\pi(\theta^*, p^*)}{\pi(\theta^{i-1}, \hat{p})}\right\}.(1$$

Note that the Hamiltonian equations ensure that the ratio $\pi(\theta^*, p^*)/\pi(\theta^{i-1}, \hat{p})$, and therefore the acceptance ratio, is close to one. Because of the discretization, however, in the practical implementation the ratio is not equal to one.

# Example

Consider the circular posterior density

$$\pi(\theta) \propto \exp\left\{-\frac{1}{2}\psi_n\big((\theta_1-1)^2+(\theta_2-1)^2-1\big)^2\right\}\mathbf{1}\left\{\theta\in[-1,3]^2\right\}$$

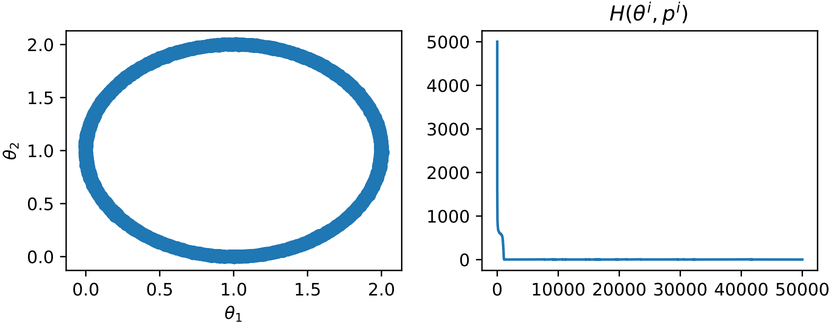
This means that:

$$\nabla\log\pi(\theta)=\Bigg[-2\psi_n\,(\theta_1-1)\left((\theta_1-1)^2+(\theta_2-1)^2-1\right),$$
$$-2\psi_n\,(\theta_2-1)\left((\theta_1-1)^2+(\theta_2-1)^2-1\right)\Bigg].$$

With $\psi_n=1000$, the “posterior” covariance of this distribution is:

$$\mathbb{V}_\pi[\theta]=\begin{bmatrix}0.5 & 0\\ 0 & 0.5\end{bmatrix}.$$

Let $N = 50000, L = 10, \epsilon = 0.01$.



# Going Beyond Hamiltonian

- Sometimes the leapfrog set is inefficient. Going around in circles!
- Modification: “No U-Turn Sampler” (NUTS) of Hoffman and Gelman (2014)
- This is the backbone of the popular “probabilistic programming language” Stan.
- Provide interface (and automatic differentiate) for many popular random variables (and hence Bayesian models)

# Can this work for DSGE models?

It seems like this should be great for DSGE models right?

Well, we need to compute $\nabla \log \pi(\theta)$. Not to easy for a DSGE!

- Numerical? Noisy, Slow
- Analytic? Slow, Explosive
- Automatic? Accurate but slow, issues with complex roots

Also, multimodality is still present! Can combine SMC with HMC!

# References

HOFFMAN, M. D. AND A. GELMAN (2014): “The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo,” *Journal of Machine Learning Research*, 15, 1351–1381.

NEAL, R. (2011): “MCMC Using Hamiltonian Dynamics,” *Handbook of Markov Chain Monte Carlo*.