ENM531: Data-driven modeling and probabilistic scientific computing

Lecture #17: Sampling Methods

Paris Perdikaris
April 7, 2020

UNIVERSITY *of* PENNSYLVANIA

The Metropolis algorithm

- Choose a symmetric proposal matrix Q. So, $Q_{ab} = Q_{ba}$.
- Initialize $x_o \in X$.
- for $i \in 0, 1, 2, \ldots, n-1$:
  - Sample proposal x from $Q(x_i, x)$ if x is discrete, otherwise, $p(x \mid x_i)$.
  - Sample r from Uniform$(0, 1)$.
  - If

    $$r < \frac{\tilde{\pi}(x)}{\tilde{\pi}(x_i)},$$

    accept and $x_{i+1} = x$.
  - Otherwise, reject and $x_{i+1} = x_i$.

Output: $x_o, x_1, \ldots, x_n$

- Symmetric proposals include:

$$J(\theta^* \mid \theta^{(s)}) = \text{Uniform}(\theta^{(s)} - \delta, \theta^{(s)} + \delta)$$

and

$$J(\theta^* \mid \theta^{(s)}) = \text{Normal}(\theta^{(s)}, \delta^2).$$

The Metropolis algorithm for Bayesian inference

Goal: We want to sample from

$$p(\theta \mid y) = \frac{f(y \mid \theta)\pi(\theta)}{m(y)}.$$

Typically, we don't know $m(y)$.

The notation is a bit more complicated, but the set up is the same.

We'll approach it a bit differently, but the idea is exactly the same.

We know $\pi(\theta)$ and $f(y \mid \theta)$, so we can can draw samples from these.

Our notation here will be that we assume parameter values $\theta_1, \theta_2, \ldots, \theta_s$ which are drawn from $\pi(\theta)$.

We assume a new parameter value comes in that is θ^*.

The Metropolis algorithm for Bayesian inference

The Metropolis algorithm proceeds as follows:

1. Sample $\theta^* \sim J(\theta \mid \theta^{(s)})$.
2. Compute the acceptance ratio (r):

$$r = \frac{p(\theta^*|y)}{p(\theta^{(s)}|y)} = \frac{p(y \mid \theta^*)p(\theta^*)}{p(y \mid \theta^{(s)})p(\theta^{(s)})}.$$

3. Let

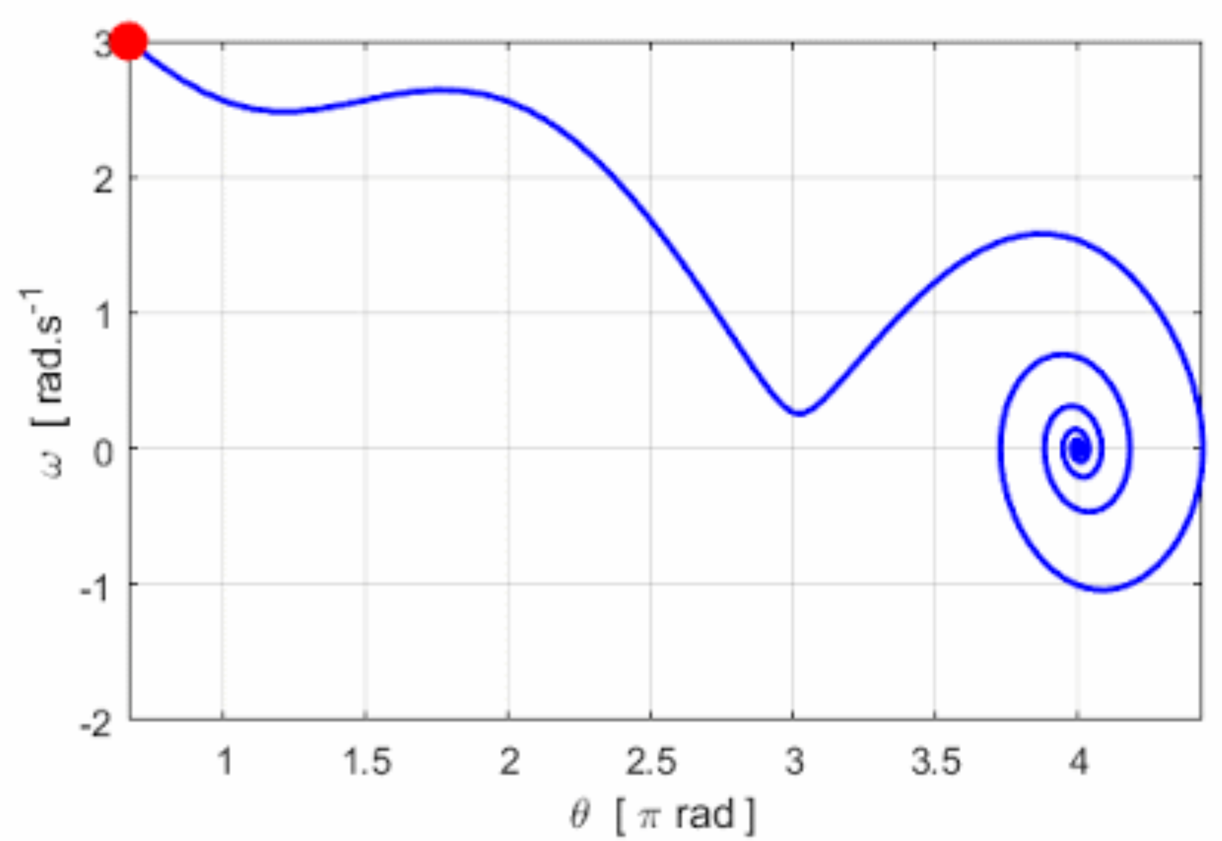
$$\theta^{(s+1)} = \begin{cases} \theta^* & \text{with prob min(r,1)} \\ \theta^{(s)} & \text{otherwise.} \end{cases}$$

Remark: Step 3 can be accomplished by sampling $u \sim \text{Uniform}(0, 1)$ and setting $\theta^{(s+1)} = \theta^*$ if $u < r$ and setting $\theta^{(s+1)} = \theta^{(s)}$ otherwise.

Bayesian calibration of dynamical systems

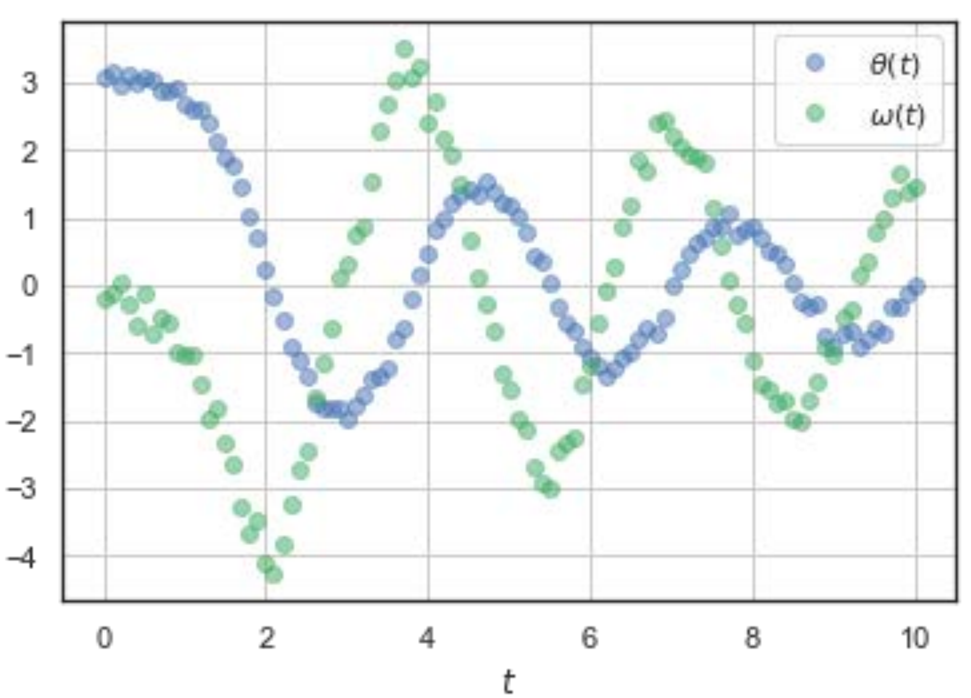
Example: Damped pendulum

$$\frac{d\theta}{dt} = \omega,$$
$$\frac{d\omega}{dt} = -b\omega - c\sin(\theta).$$



Given some noisy time-series data

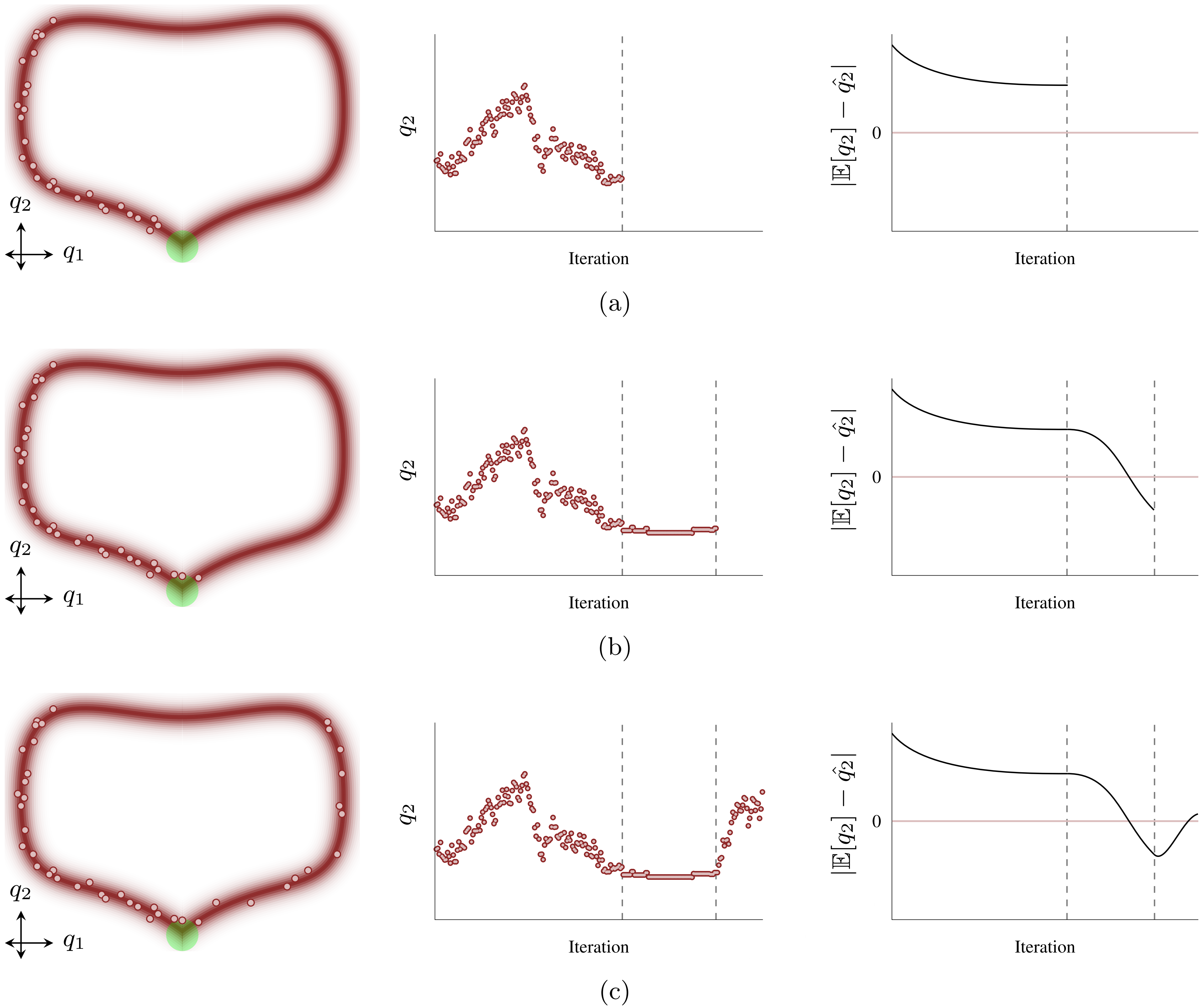
$$\mathcal{D} := \{\theta(t_i), \omega_i(t_i)\}, \quad i = 1, \ldots, n$$



Infer a posterior distribution over the unknown parameters

$$p(b, c|\mathcal{D})$$

Hamiltonian Monte Carlo



In practice, pathological regions of the typical set usually cause Markov chains to get "stuck".

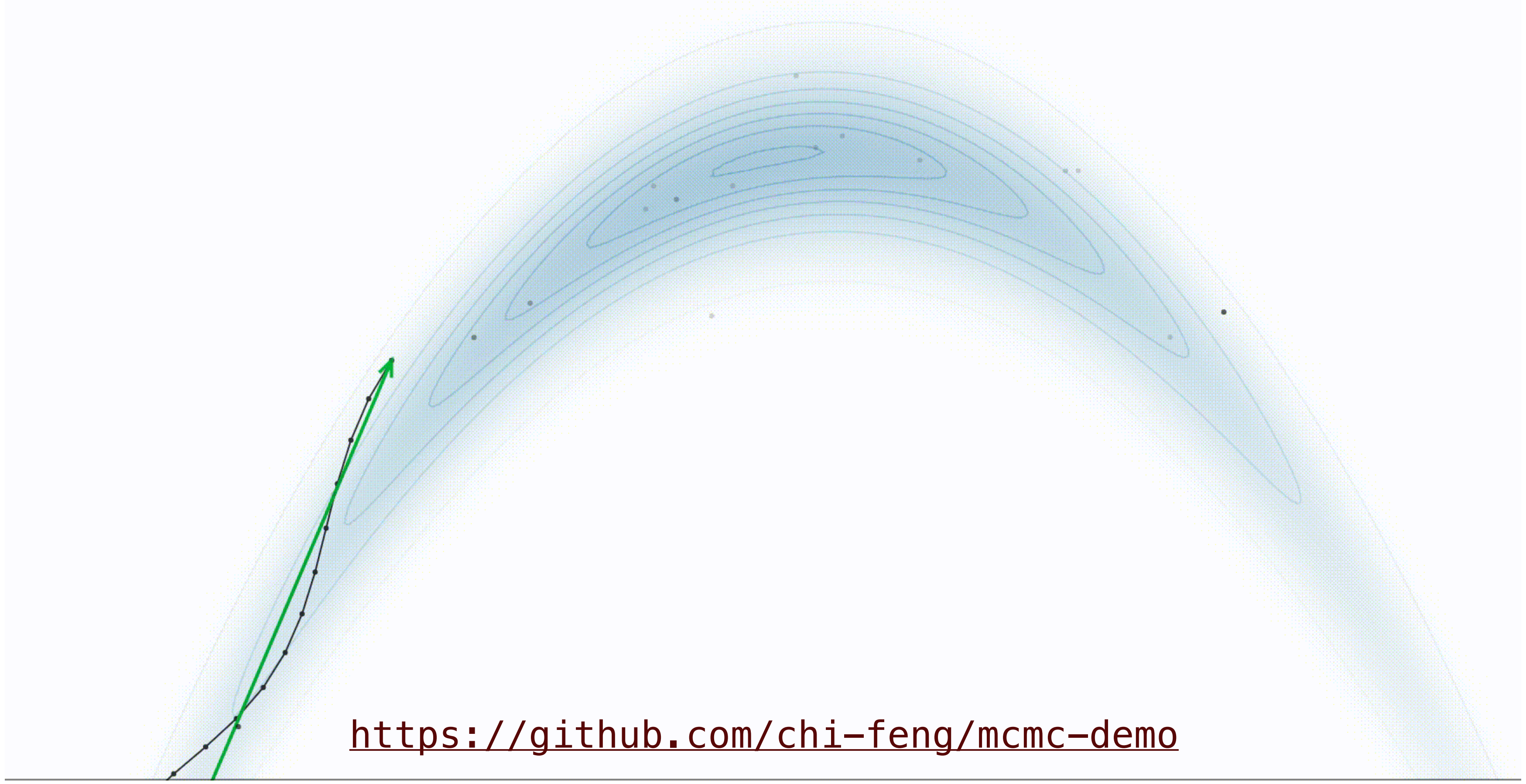
(a) Initially the Markov chain explores well-behaved regions of the typical set, avoiding the pathological neighbor- hood entirely and biasing Markov chain Monte Carlo estimators.

(b) If the Markov chain is run long enough then it will get stuck near the boundary of the pathological region, slowly correcting the Markov chain Monte Carlo estimators.

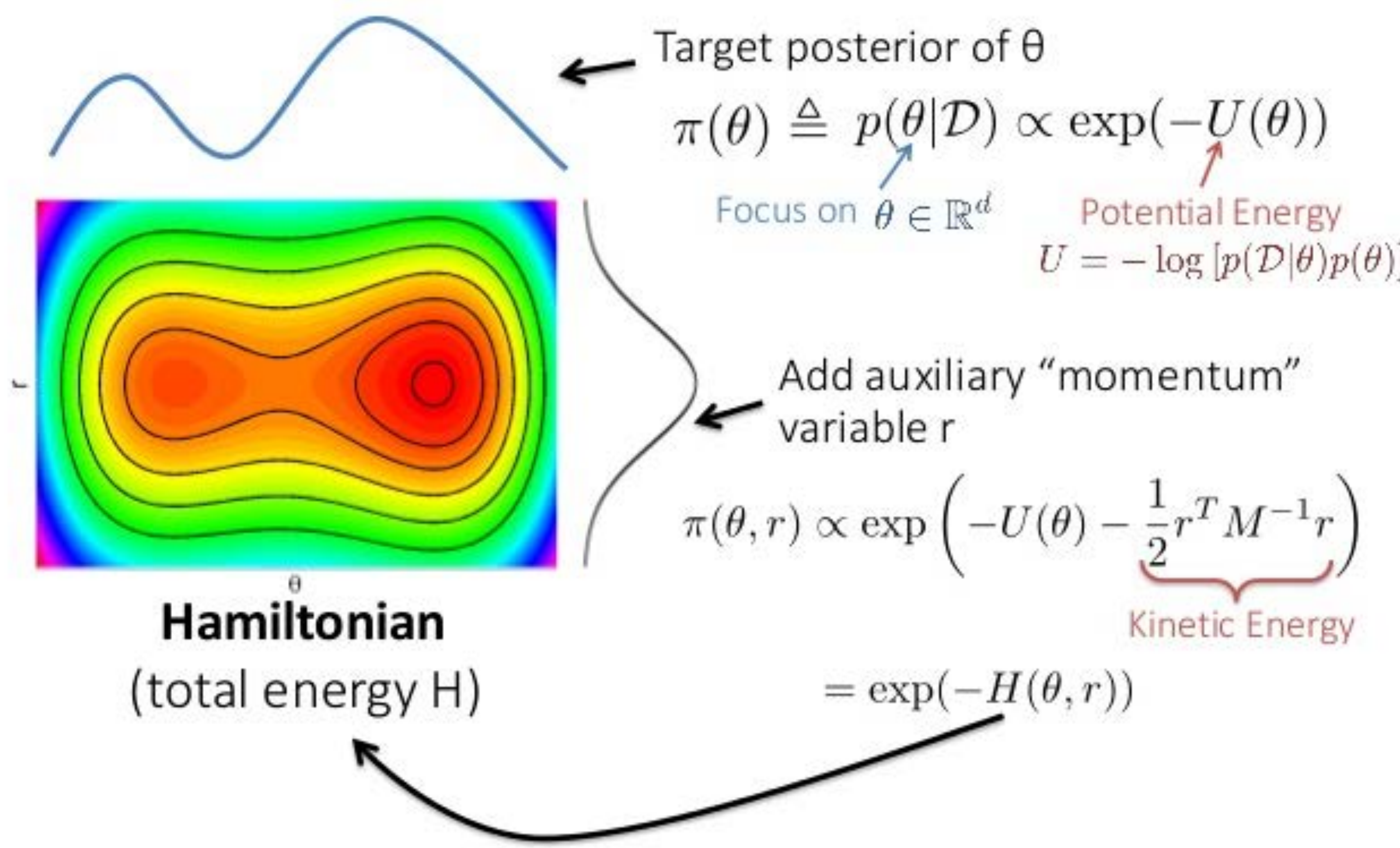
(c) Eventually the Markov chain escapes to explore the rest of the typical set. This process repeats, causing the resulting estimators to oscillate around the true expectations and inducing strong biases unless the chain is improbably stopped at exactly the right time.

Betancourt, M. (2017). A conceptual introduction to Hamiltonian Monte Carlo. arXiv preprint arXiv:1701.02434.

Hamiltonian Monte Carlo



https://github.com/chi-feng/mcmc-demo



Probabilistic programming

Stan

http://mc-stan.org/

PyMC3

https://github.com/pymc-devs/pymc3

Edward

http://edwardlib.org/

https://github.com/uber/pyro