Lecture 10: Hamiltonian Monte-Carlo, slice sampling

$\tilde{p}(\vec{q})$ - unnormalized density

Our goal: sample from it

Metropolis-Hastings MCMC suffers from random walk behaviour and requires careful tweaking of an MCMC algorithm

---

Hamiltonian dynamics

A state of a system at time t is described by a pair $(\vec{q}, \vec{p}) = (\vec{q}(t), \vec{p}(t))$

$\vec{q} \in \mathbb{R}^d$ - the position (our parameters)

$\vec{p} \in \mathbb{R}^d$ - the momentum (e.g. $\vec{p} = m\vec{v}$)

Hamiltonian of the system $H(\vec{q}, \vec{p})$ - the total energy of the system

The Hamiltonian has the form

$$H(\vec{q}, \vec{p}) = \underbrace{U(\vec{q})}_{\text{potential energy}} + \underbrace{K(\vec{p})}_{\text{kinetic energy}}$$

The system evolves in time according to Hamiltonian equations:

$$\begin{cases} \dfrac{d\vec{p}}{dt} = -\dfrac{\partial H}{\partial \vec{q}} \end{cases}$$

$$\frac{d\vec{q}}{dt} = \frac{\partial H}{\partial \vec{p}}$$

determines the path of a particle in the space $\mathbb{R}^d \times \mathbb{R}^d$

<u>Note</u>: $\frac{d}{dt} H(\vec{q}, \vec{p}) = 0$ - the total energy doesn't change, the system is closed

<u>Numerical simulation of Hamilt. dynamics</u>

<u>Leapfrog integrator</u>: $(\vec{q}(t), \vec{p}(t)) \to (\vec{q}(t+\varepsilon), \vec{p}(t+\varepsilon))$

$i = 1, \dots, d$

$$p_i(t + \varepsilon/2) = p_i(t) - \frac{\varepsilon}{2} \nabla U(\vec{q}(t))$$

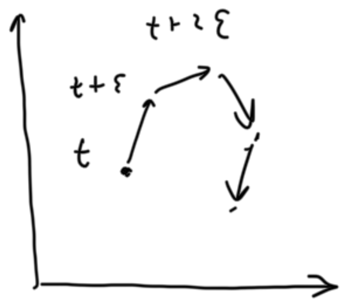
$$q_i(t + \varepsilon) = q_i(t) + \varepsilon\, p_i(t + \varepsilon/2)$$

$$p_i(t + \varepsilon) = p_i(t + \varepsilon/2) - \frac{\varepsilon}{2} \nabla U(\vec{q}(t+\varepsilon))$$

Parameters of leapfrog:

the step size ε

the number of steps L



ε - small enough, L - in between, good

<u>Canonical distribution</u>

$$C(\vec{x}) \propto \exp(-E(\vec{x})/T),$$

$E(\vec{x})$ - the energy function, $\vec{x}$ - a state,

T - temperature ($T = 1$ for us)

$$H(\vec{q}, \vec{p}) = U(\vec{q}) + K(\vec{p})$$

$$C(\vec{q}, \vec{p}) \propto \exp(-H(\vec{q}, \vec{p})) =$$

$$= \exp(-U(\vec{q})) \exp(-K(\vec{p}))$$

$U(\vec{q}) = -\log P(\vec{q}))$, P is prob. density (unnormalized)

$K(\vec{p}) = \frac{\langle \vec{p}, \vec{p} \rangle}{2}$

$C(\vec{q}, \vec{p}) = P(\vec{q}) \exp(-\langle \vec{p}, \vec{p} \rangle / 2)$

1. $\vec{q}, \vec{p}$ are independent
2. Marginal for $\vec{q}$ is our $P(\vec{q})$

Hamiltonian MC

$\vec{q} \in \mathbb{R}^d$ - our parameters, e.g. $\vec{\Theta}$

$\vec{p} \in \mathbb{R}^d$ - auxiliary variables

We sample from $C(\vec{p}, \vec{q})$.

If we drop $\vec{p}$, we obtain marginal $P(\vec{q})$

Naive algorithm HMC

$(\vec{q}_n, \vec{p}_n)$ - the current step

ε - step size, L - step number

1. $i = 1:L$ do: $(\vec{q}^1, \vec{p}^1) = (\vec{q}_n, \vec{p}_n)$

Leap frog
$$\begin{cases} \vec{p}^{i+1/2} = \vec{p}^i - \frac{\varepsilon}{2} \nabla U(\vec{q}^i) \\ \vec{q}^{i+1} = \vec{q}^i + \varepsilon \vec{p}^{i+1/2} \\ \vec{p}^{i+1} = \vec{p}^{i+1/2} - \frac{\varepsilon}{2} \nabla U(\vec{q}^{i+1}) \end{cases}$$

2. Calculate

$r_n^L = r(\vec{q}_n, \vec{p}_n, \vec{q}^L, \vec{p}^L) = \min(1, \exp(-H(\vec{q}^L, \vec{p}^L) + H(\vec{q}_n, \vec{p}_n)))$ - acceptance prob.

Note: $r_n^L = \min\left(1, \frac{\exp(-H(\vec{q}^L, \vec{p}^L))}{\exp(-H(\vec{q}_n, \vec{p}_n))}\right) =$

$= \min\left(1, \frac{P(\vec{q}^L)\exp(-\langle \vec{p}^L, \vec{p}^L\rangle/2)}{P(\vec{q}_n)\exp(-\langle \vec{p}_n, \vec{p}_n\rangle/2)}\right)$

3. Draw $u \sim \mathcal{U}([0,1])$

$u < r_n^L : (\vec{q}_{n+1}, \vec{p}_{n+1}) = (\vec{q}^L, \vec{p}^L),$

$u \geq r_n^L : (\vec{q}_{n+1}, \vec{p}_{n+1}) = (\vec{q}_n, \vec{p}_n)$

Note: If leapfrog is correct, we'll always have an accept.

Actual HMC algorithm

Input: the current step $\vec{q}_n$, ε, L

1. $\vec{p}_n \sim \mathcal{N}(\vec{p} \mid \vec{0}, I_d)$ - draw

2. $(\vec{q}^1, \vec{p}^1) = (\vec{q}_n, \vec{p}_n)$

for $i = 1:L$ do:

Leap-frog
$$\begin{cases} \vec{p}^{i+1/2} = \vec{p}^i - \frac{\varepsilon}{2} \nabla \mathcal{U}(\vec{q}^i) \\ \vec{q}^{i+1} = \vec{q}^i + \varepsilon \vec{p}^{i+1/2} \\ \vec{p}^{i+1} = \vec{p}^{i+1} - \frac{\varepsilon}{2} \nabla \mathcal{U}(\vec{q}^{i+1}) \end{cases}$$

3. $r = r(\vec{q}_n, \vec{p}_n, \vec{q}^L, \vec{p}^L) =$

$= \min\left(1, \exp\left(-H(\vec{q}^L, \vec{p}^L) + H(\vec{q}_n, \vec{p}_n)\right)\right),$

4. $u \sim \mathcal{U}([0,1])$ - accept step

$u < r : \vec{q}_{n+1} = \vec{q}^L$

$u \geq r : \vec{q}_{n+1} = \vec{q}_n$

We again need only unnormalized density to sample from

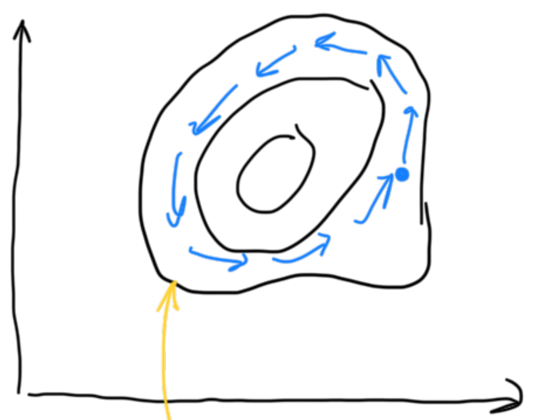
$O(d^{5/4})$ steps is better than $O(d^2)$ "r.w."

Limitations:

- gradient of the log-posterior
- sampling from multimodal distributions

Gradient-free MCMC [NeurIPS, 2015]

How to select L and ε?



U-turn, it is better to stop here

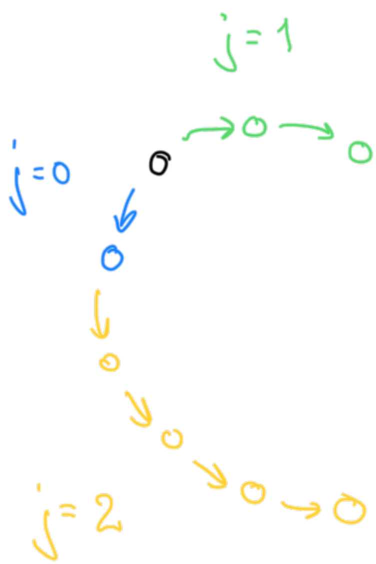
Naive approach doesn't work:

$$< \vec{q}_n - \vec{q}^{\,k}, \vec{p}^{\,k} > < 0$$

doesn't preserve the distribution

$\Rightarrow$ <u>No-U-turns sampler</u> (NUTs)

doesn't break detailed balance and stops



- We do leap frog steps forward and backward 2^j, $j = 0, 1, \dots, k$ steps until we meet a stopping criterion

$\rightarrow$ L selection

Slice sampling [Neal, 2003]

Adaptive step size selection

$\vec{q}$ - our parameters

u - additional variable, a new thing

Let us define joint $\hat{p}(\vec{q}, u) = \begin{cases} 0, \text{ if } u > \tilde{p}(\vec{q}) \\ 1/Z_p, \text{ if } 0 \leq u \leq \tilde{p}(\vec{q}), \end{cases}$

$$Z_p = \int \tilde{p}(\vec{q})\, d\vec{q}$$

Stat. Marginal $\hat{p}(\vec{q}) = \int \hat{p}(\vec{q}, u)\, du = p(\vec{q})$

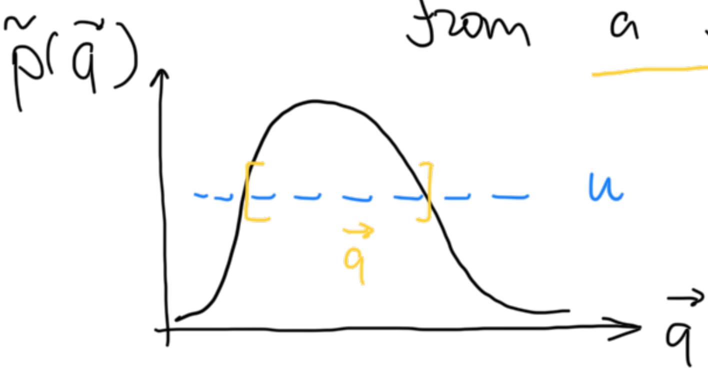
$$\int \hat{p}(\vec{q}, u)\, du = \int_0^{\tilde{p}(\vec{q})} \frac{1}{Z_p}\, du = \frac{1}{Z_p} \tilde{p}(\vec{q}) =$$
$$= p(\vec{q})$$

Algorithm:

1. Sample from $\hat{p}(\vec{q}, u)$
2. Ignore u

Given $\vec{q} \to$ sampe $u \sim \mathcal{U}([0, \tilde{p}(\vec{q})])$

Given $u \to$ sample $\vec{q}$ s.t. $\tilde{p}(\vec{q}) > u$ from a slice



We can find slice iteratively