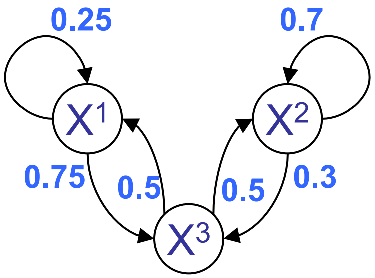
School of Computer Science
Carnegie Mellon

Probabilistic Graphical Models

Optimization in Markov Chain Monte Carlo

Avinava Dubey

Lecture 17, March 22, 2017



Recap of Monte Carlo

- Monte Carlo methods are algorithms that:
  - Generate samples from a given probability distribution $p(x)$
  - Estimate expectations of functions $E[f(x)]$ under a distribution $p(x)$

- Why is this useful?
  - Can use samples of $p(x)$ to approximate $p(x)$ itself
    - Allows us to do graphical model inference when we can't compute $p(x)$
  - Expectations $E[f(x)]$ reveal interesting properties about $p(x)$
    - e.g. means and variances of $p(x)$

Limitations of Monte Carlo

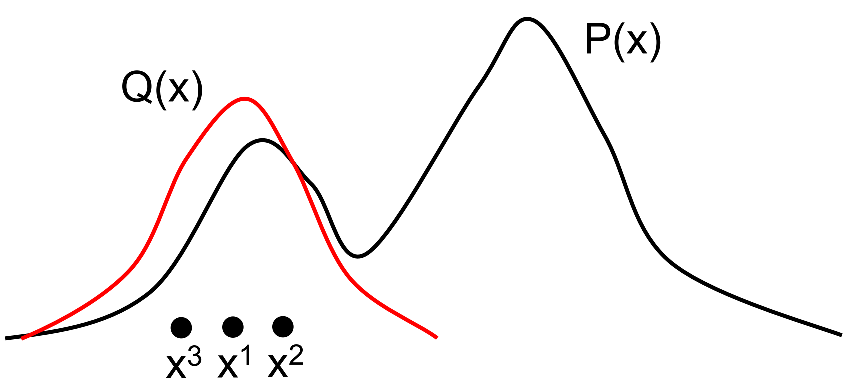
- Direct sampling
  - Hard to get rare events in high-dimensional spaces
  - Infeasible for MRFs, unless we know the normalizer Z
- Rejection sampling, Importance sampling
  - Do not work well if the proposal Q(x) is very different from P(x)
  - Yet constructing a Q(x) similar to P(x) can be difficult
    - Making a good proposal usually requires knowledge of the analytic form of P(x) – but if we had that, we wouldn't even need to sample!

- Intuition: instead of a fixed proposal Q(x), what if we could use an adaptive proposal?

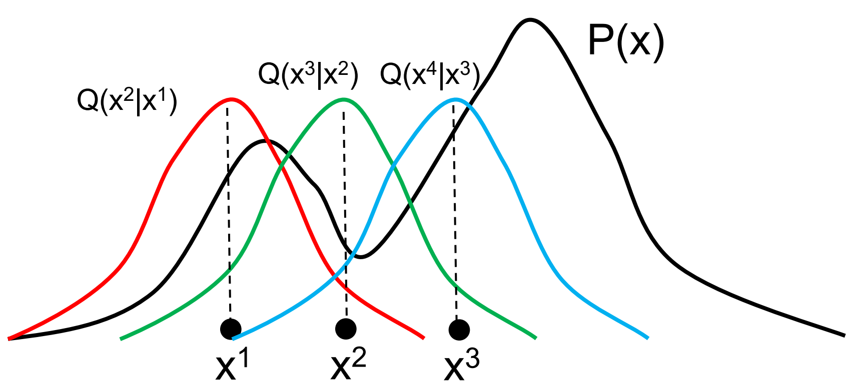
Markov Chain Monte Carlo: Recap

- MCMC algorithms feature adaptive proposals
  - Instead of Q(x'), they use Q(x'|x) where x' is the new state being sampled, and x is the previous sample
  - As x changes, Q(x'|x) can also change (as a function of x')

Importance sampling with a (bad) proposal Q(x)



MCMC with adaptive proposal Q(x'|x)



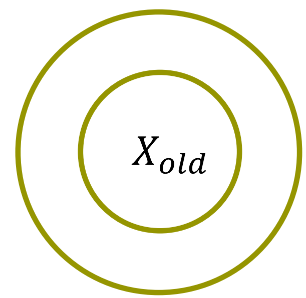
MCMC: Recap

- Distribution to sample from $P(X)$
- Proposal distribution $Q(X_{new}|X_{old})$
- Accept X_{new} with probability

$$\min\{\ 1,\quad \frac{P(X_{new})Q(X_{old}|X_{new})}{P(X_{old})Q(X_{new}|X_{old})}\}$$

MCMC: Recap

- Simple Example



$P(X)$

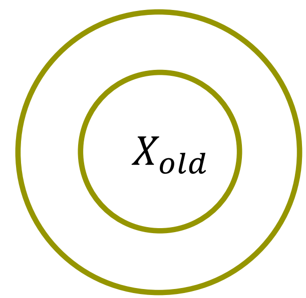
$Q(X_{new} | X_{old})$

$$\min\{\ 1, \quad \frac{P(X_{new})Q(X_{old}|X_{new})}{P(X_{old})Q(X_{new}|X_{old})}\}$$

MCMC: Recap

- Simple Example

Might reject a lot of samples



$P(X)$

$Q(X_{new}|X_{old})$

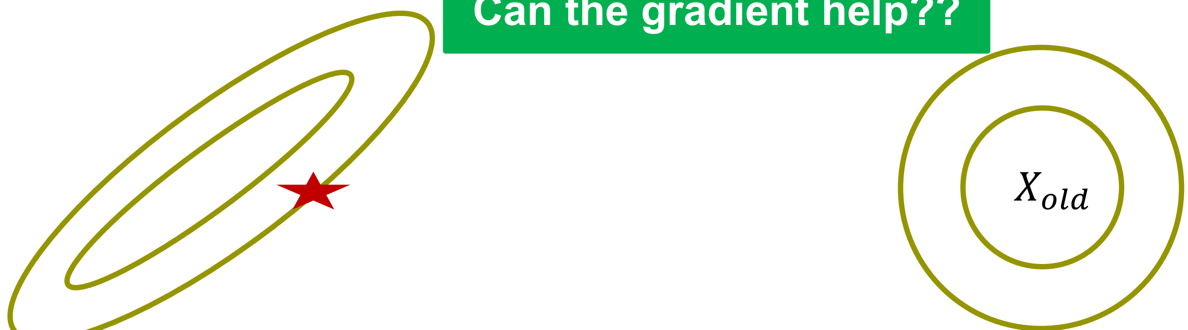
$$\min\{\ 1,\ \frac{P(X_{new})Q(X_{old}|X_{new})}{P(X_{old})Q(X_{new}|X_{old})}\}$$

MCMC: Recap

- Simple Example

Might reject a lot of samples

Can the gradient help??



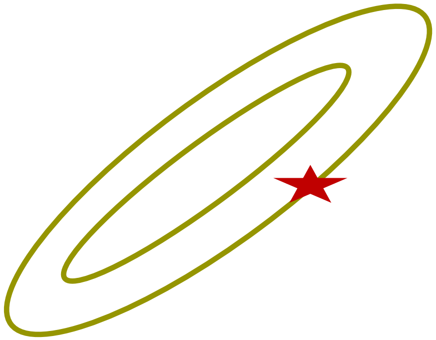
$P(X)$ $\qquad\qquad$ $Q(X_{new}|X_{old})$

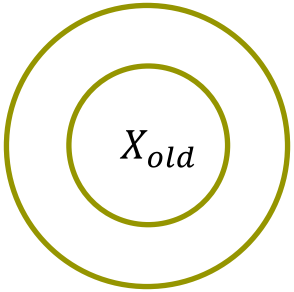
$$\min\{\ 1,\ \frac{P(X_{new})Q(X_{old}|X_{new})}{P(X_{old})Q(X_{new}|X_{old})}\}$$

MCMC: Recap

- Simple Example

If variance of Q is small then next sample might be very correlated to the previous one





$P(X)$

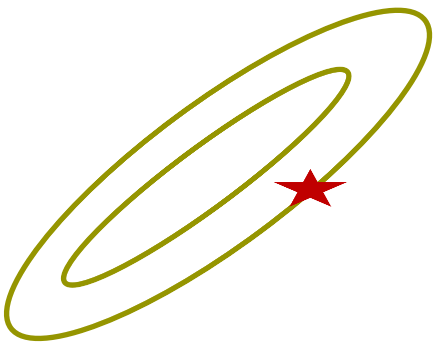
$Q(X_{new}|X_{old})$

$$\min\{\ 1, \quad \frac{P(X_{new})Q(X_{old}|X_{new})}{P(X_{old})Q(X_{new}|X_{old})}\}$$

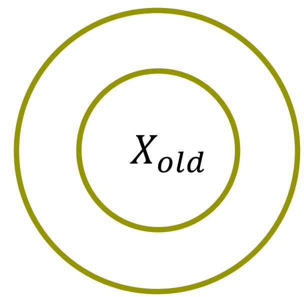
MCMC: Recap

- Simple Example

If variance of Q is large then next sample might be rejected



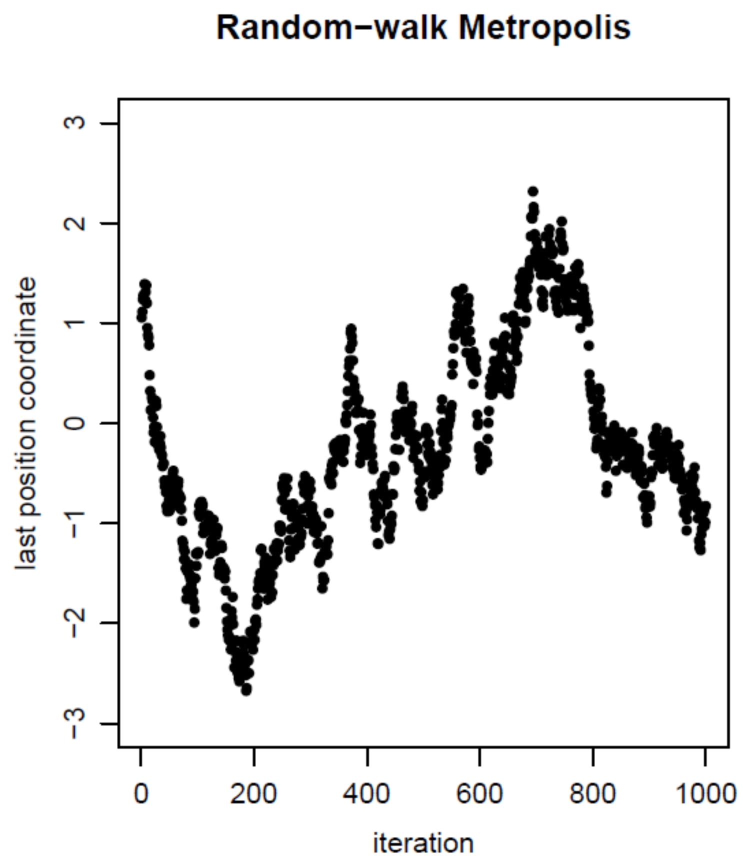
$P(X)$



$Q(X_{new}|X_{old})$

$$\min\{\ 1,\quad \frac{P(X_{new})Q(X_{old}|X_{new})}{P(X_{old})Q(X_{new}|X_{old})}\}$$

Highly correlated samples



MCMC: Recap

- Random walk can have poor acceptance rate
- The samples can have high correlation between themselves reducing the effective sample size

MCMC: Recap

- Random walk can have poor acceptance rate
- The samples can have high correlation between themselves reducing the effective sample size
- Can we have a better proposal
  - Using gradient information
  - Using approximation of the given probability distribution

Hamiltonian Monte Carlo

- Hamiltonian Dynamics (1959)
  - Deterministic System
- Hybrid Monte Carlo (1987)
  - United MCMC and molecular Dynamics
- Statistical Application (1993)
  - Inference in Neural Networks
  - Improves acceptance rate
  - Uncorrelated Samples

Hamiltonian Dynamics

- Position vector q, Momentum vector p
- Kinetic Energy $K(p)$
- Potential Energy $U(q)$
- Define $H(p,q) = K(p) + U(q)$

Hamiltonian Dynamics

- Position vector q, Momentum vector p
- Kinetic Energy $K(p)$
- Potential Energy $U(q)$
- Define $H(p,q) = K(p) + U(q)$
- Hamiltonian Dynamic

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}$$

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}$$

Hamiltonian Dynamics: Frictionless puck

$$K(p) = \frac{|p|^2}{2m}$$

$$U(q)$$

$$H(p,q) = K(p) + U(q)$$

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}$$

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}$$

Hamiltonian Dynamics: Example

- Kinetic Energy $K(p) = \frac{|p|^2}{2m}$
- Potential Energy $U(q) = \frac{q^2}{2}$
- Define $H(p,q) = K(p) + U(q)$
- Hamiltonian Dynamic

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}$$

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}$$

Hamiltonian Dynamics: Example

- Kinetic Energy $K(p) = \frac{|p|^2}{2}$
- Potential Energy $U(q) = \frac{q^2}{2}$
- So $\frac{dq}{dt} = p, \quad \frac{dp}{dt} = -q$
- And

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}$$

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}$$

$$q(t) = r\cos(a+t), \quad p(t) = -r\sin(a+t)$$

Properties of Hamiltonian

- Reversibility
- Conservation of Hamiltonian
- Mapping preserves volume

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}$$

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}$$

How to get solution

- Discretization
  - Euler's Method
  - Leapfrog Method
  - etc

Euler's Method

$$p_i(t+\varepsilon) = p_i(t) + \varepsilon \frac{dp_i}{dt}(t) = p_i(t) - \varepsilon \frac{\partial U}{\partial q_i}(q(t))$$

$$q_i(t+\varepsilon) = q_i(t) + \varepsilon \frac{dq_i}{dt}(t) = q_i(t) + \varepsilon \frac{p_i(t)}{m_i}$$

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}$$

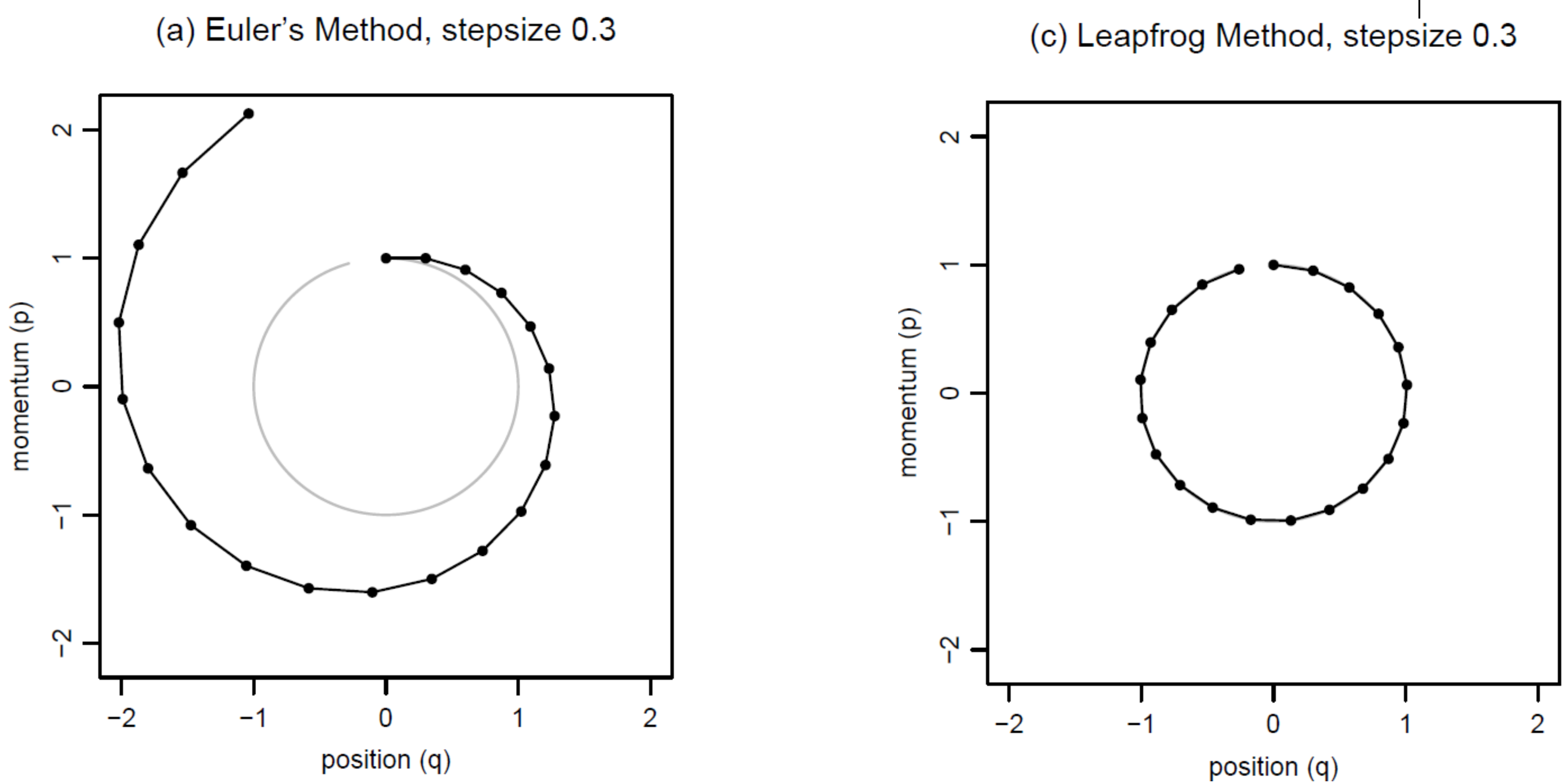
$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}$$

Leapfrog Method

- The updates looks like

$$
\begin{aligned}
p_i(t+\varepsilon/2) &= p_i(t) - (\varepsilon/2)\frac{\partial U}{\partial q_i}(q(t)) \\
q_i(t+\varepsilon) &= q_i(t) + \varepsilon\frac{p_i(t+\varepsilon/2)}{m_i} \\
p_i(t+\varepsilon) &= p_i(t+\varepsilon/2) - (\varepsilon/2)\frac{\partial U}{\partial q_i}(q(t+\varepsilon))
\end{aligned}
$$

Leapfrog Vs Euler



$$q(t) = r\cos(a+t), \quad p(t) = -r\sin(a+t)$$

MCMC from Hamiltonian Dynamics

- Let q be variable of interest
- Define:

$$P(q,p) = \frac{1}{Z}\exp(-U(q)/T)\exp(-K(p)/T)$$

- And

$$U(q) = -\log\left[\pi(q)L(q|D)\right] \qquad K(p) = \sum_{i=1}^{d}\frac{p_i^2}{2m_i}$$

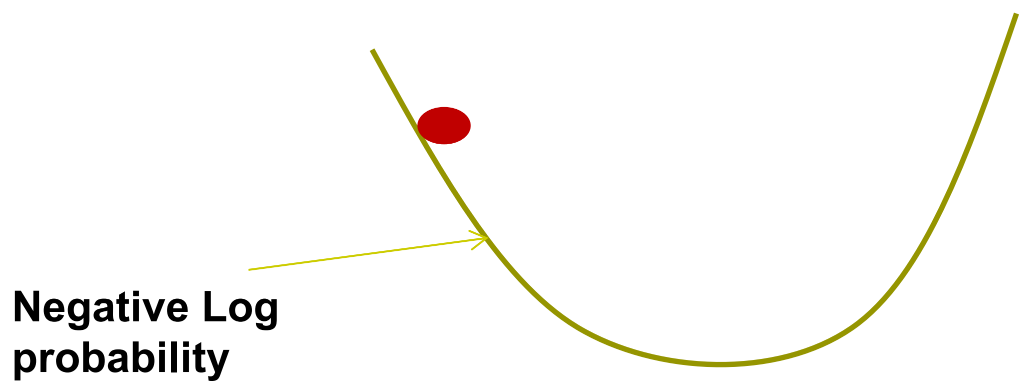
- Key Idea: Use Hamiltonian dynamics to propose next step.

MCMC from Hamiltonian Dynamics

- Let q be variable of interest

$$P(q,p) = \frac{1}{Z}\exp(-U(q)/T)\exp(-K(p)/T)$$

$$U(q) = -\log\left[\pi(q)L(q|D)\right] \quad K(p) = \sum_{i=1}^{d}\frac{p_i^2}{2m_i}$$



MCMC from Hamiltonian Dynamics

- Given q_0 (starting state)
- Draw $p \sim N(0,1)$
- Use L steps of leapfrog to propose next state
- Accept / reject based on change in Hamiltonian

MCMC from Hamiltonian Dynamics

```
p = rnorm(length(q),0,1)
```

MCMC from Hamiltonian Dynamics

```
p = rnorm(length(q),0,1)
p = p - epsilon * grad_U(q) / 2
```

MCMC from Hamiltonian Dynamics

```
p = rnorm(length(q),0,1)
p = p - epsilon * grad_U(q) / 2
# Alternate full steps for position and momentum
for (i in 1:L)
{
      q = q + epsilon * p
      if (i!=L) p = p - epsilon * grad_U(q)
 }
```

MCMC from Hamiltonian Dynamics

```
p = rnorm(length(q),0,1)
p = p - epsilon * grad_U(q) / 2

# Alternate full steps for position and momentum
for (i in 1:L)
{
     q = q + epsilon * p

     if (i!=L) p = p - epsilon * grad_U(q)
}

p = p - epsilon * grad_U(q) / 2        p = -p
```

MCMC from Hamiltonian Dynamics

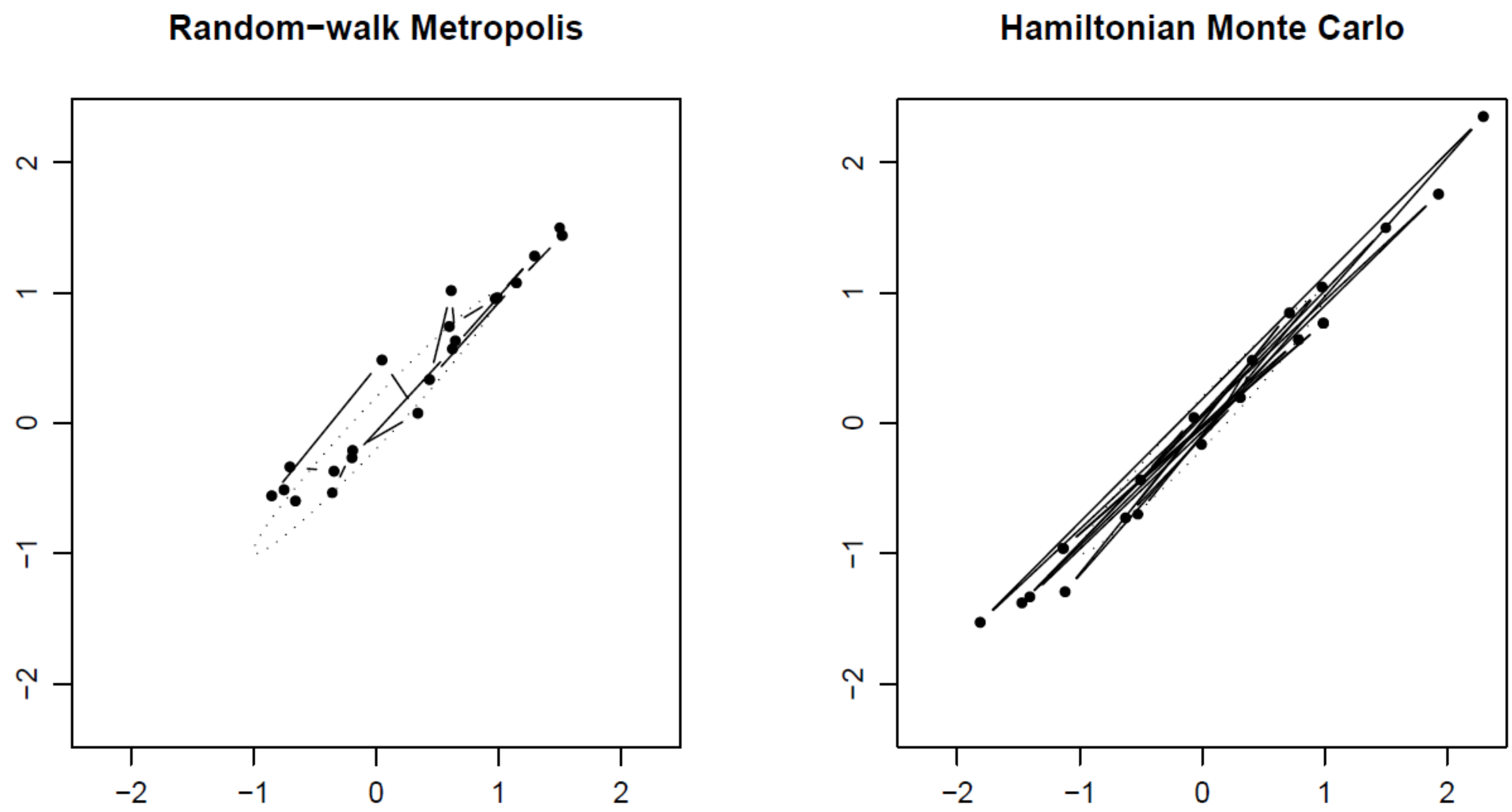
```
p = rnorm(length(q),0,1)
p = p - epsilon * grad_U(q) / 2
# Alternate full steps for position and momentum
for (i in 1:L)
{
     q = q + epsilon * p
     if (i!=L) p = p - epsilon * grad_U(q)
}
p = p - epsilon * grad_U(q) / 2      p = -p
Accept or reject the state at end of trajectory
```

$$\min\left[1, \exp(-U(q^*) + U(q) - K(p^*) + K(p))\right]$$

MCMC from Hamiltonian Dynamics

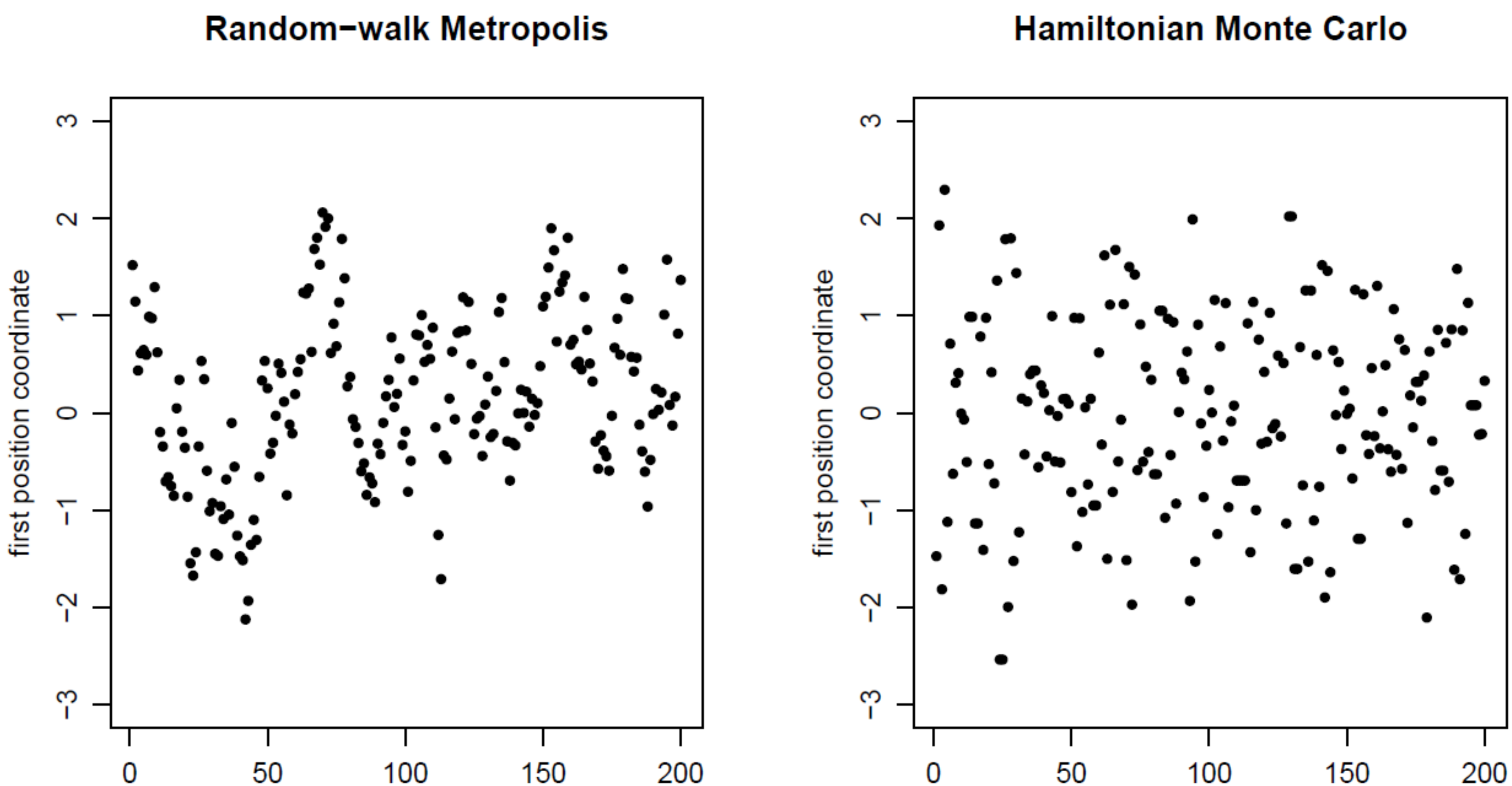
- Detailed balance satisfied
- Ergodic
- canonical distribution invariant

2D Gaussian Example



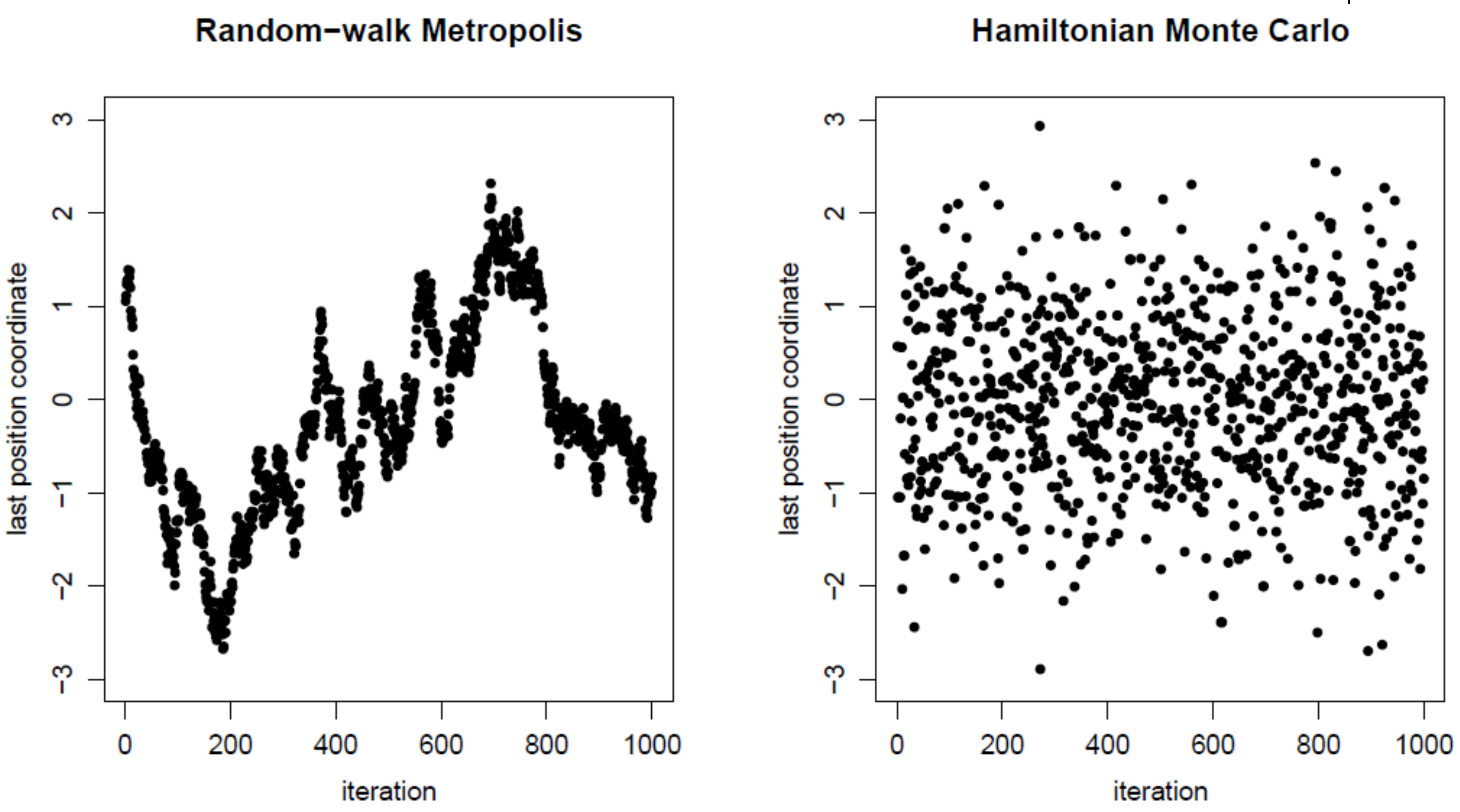
Twenty iterations of the random-walk Metropolis method (with 20 updates per iteration) and of the Hamiltonian Monte Carlo method (with 20 leapfrog steps per trajectory) for a 2D Gaussian distribution with marginal standard deviations of one and correlation 0.98. Only the two position coordinates are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

2D Gaussian Example



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

100D Gaussian Example



Acceptance Rate

- 2D example HMC : 91% Random Walk: 63%
- 100D example HMC: 87% Random Walk: 25%

MCMC from Hamiltonian Dynamics

```
p = rnorm(length(q),0,1)
p = p - epsilon * grad_U(q) / 2

# Alternate full steps for position and momentum
for (i in 1:L)
{
      q = q + epsilon * p

      if (i!=L) p = p - epsilon * grad_U(q)
}

p = p - epsilon * grad_U(q) / 2        p = -p
```

Accept or reject the state at end of trajectory

$$\min\left[1, \exp(-U(q^*) + U(q) - K(p^*) + K(p))\right]$$

Langevin Dynamics

$$q_i^* = q_i - \frac{\varepsilon^2}{2}\frac{\partial U}{\partial q_i}(q) + \varepsilon p_i$$

$$p_i^* = p_i - \frac{\varepsilon}{2}\frac{\partial U}{\partial q_i}(q) - \frac{\varepsilon}{2}\frac{\partial U}{\partial q_i}(q^*)$$

accept q^* as the new state with probability

$$\min\left[1,\ \exp\left(-\left(U(q^*) - U(q)\right) - \frac{1}{2}\sum_i \left((p_i^*)^2 - p_i^2\right)\right)\right]$$

Leapfrog

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

$$q_i(t+\varepsilon) = q_i(t) + \varepsilon\frac{p_i(t+\varepsilon/2)}{m_i}$$

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

Stochastic Langevin Dynamics

- For large datasets hard to compute the whole gradient

$$q_i^* = q_i - \frac{\varepsilon^2}{2}\frac{\partial U}{\partial q_i}(q) + \varepsilon p_i$$

$$U(q) = -\log\left[\pi(q)L(q|D)\right]$$

Stochastic Gradient Langevin Dynamics

- For large datasets hard to compute the whole gradient

$$q_i^* = q_i - \frac{\varepsilon^2}{2}\underbrace{\frac{\partial U}{\partial q_i}(q)} + \varepsilon p_i$$

Calculate using subset of data

$$U(q) = -\log\left[\pi(q)L(q|D)\right]$$

Stochastic Gradient Langevin Dynamics: Bayesian Models

- Posterior $p(\theta|\mathbf{X}) \propto p(\theta) \prod_{i=1}^{N} p(x_i|\theta)$

- SGLD update:

$$\Delta\theta_t = \frac{h_t}{2}\left(\nabla \log p(\theta_t) + \frac{N}{n}\sum_{i=1}^{n} \nabla \log p(x_{ti}|\theta_t)\right) + \eta_t$$

$$\eta_t \sim N(0, h_t)$$

$$q_i^* = q_i - \frac{\varepsilon^2}{2}\frac{\partial U}{\partial q_i}(q) + \varepsilon p_i$$

$$U(q) = -\log\left[\pi(q) L(q|D)\right]$$

Stochastic Gradient Langevin Dynamics

- High variance in stochastic gradient

- Take help from the optimization community

Conclusion

- HMC can improve acceptance rate and give better mixing
- Stochastic variants can be used to improve performance in large dataset scenarios
- HMC may not be used for discrete variable

Towards better proposal

- $Q(X_{new}|X_{old})$ determines when the chain converges
- Idea: Variational approximation of P(X) be the proposal distribution

Variational Inference: Recap

- Interested in posterior of parameters $P(\theta|x)$
- Using Jensen's Inequality

$$\boldsymbol{log(p(x|\theta) \geq E_{q(z)}[log(p(x|\theta)] - E_{q(z)}[log\big(q(z)\big)]}$$

- Choose $q(z|\lambda)$ where λ is the variational parameter
- Replace $p(x|\theta)$ with $p(x|\theta,\xi)$ where ξ is another set of variational parameters
- Using this we can easily obtain un-normalized bound for posterior

$$\boldsymbol{P(\theta|x\,) \geq P^{est}(\theta|x,\lambda,\xi)}$$

Variational MCMC

- Idea: Variational approximation of P(X) be the proposal distribution

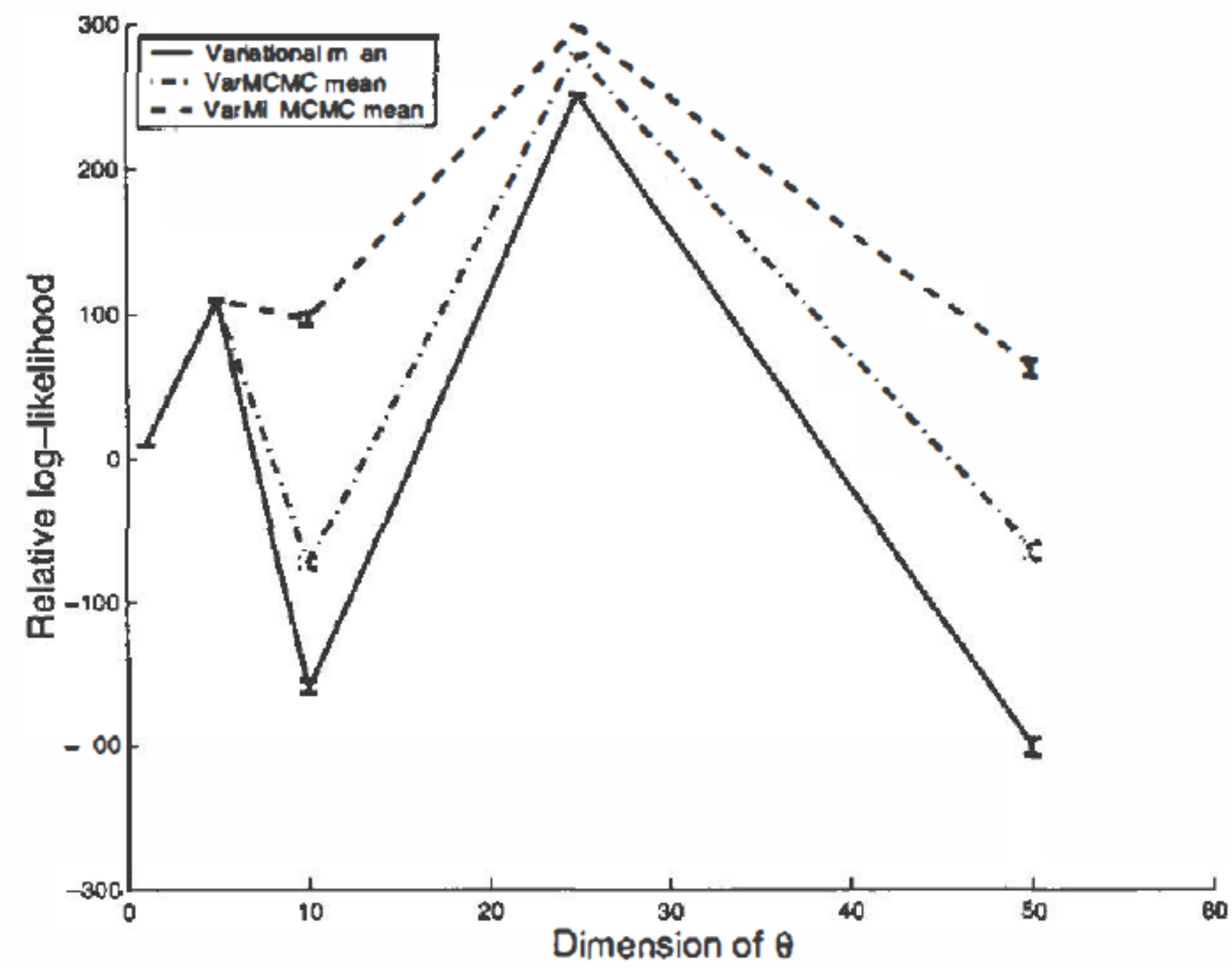
- $Q(\theta_{new}|\theta_{old}) = \boldsymbol{P}^{\boldsymbol{est}}(\boldsymbol{\theta}|\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\xi})$

- Issues:
  - Low acceptance in high dimensions
  - Works well if $\boldsymbol{P}^{\boldsymbol{est}}$ is close to P

Variational MCMC

- Design the proposal in blocks to take care of correlated variables
- Use a mixture of random walk and variational approximation as a proposal distribution
- Now can use stochastic variational methods in estimating $\boldsymbol{P}^{est}(\boldsymbol{\theta}|\boldsymbol{x},\boldsymbol{\lambda},\boldsymbol{\xi})$

Variational MCMC



Conclusion

- Adapting proposal distribution can be helpful in
  - Increasing mixing
  - Decreasing time to convergence
  - Increasing acceptance rate
  - Getting uncorrelated information