COMPSCI 688: Probabilistic Graphical Models

Lecture 16: Metropolis-Hastings and Practical Aspects

Dan Sheldon

Manning College of Information and Computer Sciences
University of Massachusetts Amherst

Partially based on materials by Benjamin M. Marlin (marlin@cs.umass.edu) and Justin Domke (domke@cs.umass.edu)

Metropolis-Hastings

The Metropolis-Hastings Sampler

- The Metropolis Hastings sampler is an extremely general sampler based on the idea of "proposing" a new state with a *proposal distribution* $q(\mathbf{x}'|\mathbf{x})$, and then "accepting" or "rejecting"
- Like the Gibbs sampler, it can be used with continuous or discrete distributions and avoids computation of the partition function.
- Unlike the Gibbs sampler, it doesn't require the ability to sample from the conditional distributions.

Proposal and Acceptance MCMC Illustration

Proposal and Acceptance MCMC

Initialize x

for $t = 1, 2, 3, \ldots, S$:

  Sample $x' \sim q(x'|x)$

  Look at x and x', and calculate a probability $\alpha(x, x')$ of keeping x'.

  Choose $r \in [0, 1]$ uniformly

  If $r < \alpha(x, x')$ **then**

    $x \leftarrow x'$

  $x^{(t)} \leftarrow x$

return $x^{(1)}, x^{(2)}, \ldots, x^{(S)}$

How to Choose Acceptance Probability?

The key missing step is how to set the acceptance probability $\alpha(x, x')$. It can depend on p and q. The transition probability density is

$$T(x'|x) = \begin{cases} q(x'|x)\alpha(x, x') & \text{if } x \neq x' \\ ? & \text{if } x = x' \end{cases}$$

Our goal is to satisfy detailed balance, i.e., for all x, x':

$$p(x)T(x'|x) = p(x')T(x|x')$$

$$\iff p(x)q(x'|x)\alpha(x, x') = p(x')q(x|x')\alpha(x', x)$$

We don't care about $T(x'|x)$ when $x = x'$, because the detailed balance condition is always satsified for $x = x'$.

There are different acceptance rules $\alpha(x, x')$ that ensure detailed balance. Metropolis-Hastings is based on the adjusting the larger "flow" to be equal to the smaller one.

$$\underbrace{p(x)q(x'|x)}_{x \to x' \text{ flow}} \underbrace{\alpha(x, x')}_{\text{adjustment}} = \underbrace{p(x')q(x|x')}_{x' \to x \text{ flow}} \underbrace{\alpha(x', x)}_{\text{adjustment}}$$

The rule is

- If $x \to x'$ flow $>$ $x' \to x$ flow, set $\alpha(x, x')$ equal to their ratio, and set $\alpha(x', x) = 1$
- If $p(x)q(x'|x) > p(x')q(x|x')$, set $\alpha(x, x') = \dfrac{p(x')q(x|x')}{p(x)q(x'|x)}$ and set $\alpha(x', x) = 1$.

By symmetry, the general Metropolis-Hastings acceptance rule is:

$$\boxed{\alpha(x, x') = \min\left\{1, \frac{p(x')q(x|x')}{p(x)q(x'|x)}\right\}}$$

Proof of Detailed Balance

Claim: detailed balance holds with $\alpha(x, x') = \min\left\{1, \frac{p(x')q(x|x')}{p(x)q(x'|x)}\right\}$

Proof: First, consider when $p(x)q(x'|x) > p(x')q(x|x')$. Then $\alpha(x, x') = \frac{p(x')q(x|x')}{p(x)q(x'|x)}$ and $\alpha(x', x) = 1$, and we have

$$\begin{aligned}
p(x)T(x'|x) &= p(x)q(x'|x)\alpha(x, x') \\
&= p(x)q(x'|x)\frac{p(x')q(x|x')}{p(x)q(x'|x)} \\
&= p(x')q(x|x') \\
&= p(x')q(x|x')\alpha(x', x) \\
&= p(x')T(x|x').
\end{aligned}$$

For the second case, we have $p(x')q(x|x') > p(x)q(x'|x)$. The proof is the same as the first case, with x and x' swapped.

Metropolis-Hastings Algorithm

Initialize x
for $t = 1, 2, 3, \ldots, S$:
    Sample $x' \sim q(x'|x)$
    Choose $r \in [0, 1]$ uniformly
    if $r < \frac{p(x')Q(x|x')}{p(x)Q(x'|x)}$ **then**
        $x \leftarrow x'$
    $x^{(t)} \leftarrow x$
return $x^{(1)}, x^{(2)}, \ldots, x^{(S)}$

Gaussian Random Walk Sampler

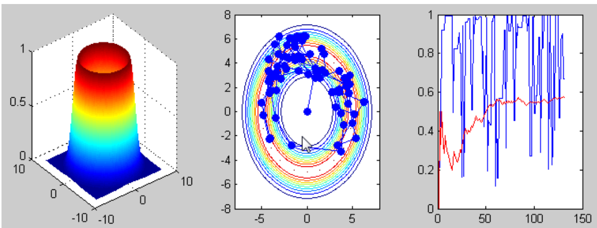
A simple proposal uses a Gaussian random walk as the proposal distribution:

$$\mathbf{x}' \sim \mathcal{N}(\mathbf{x}'|\mathbf{x}, \sigma^2 I)$$

By symmetry, the acceptance probability simplifies

$$\alpha(\mathbf{x}', \mathbf{x}) = \frac{p(\mathbf{x}')\mathcal{N}(\mathbf{x}|\mathbf{x}', \sigma^2 I)}{p(\mathbf{x})\mathcal{N}(\mathbf{x}'|\mathbf{x}, \sigma^2 I)} = \frac{p(\mathbf{x}')}{p(\mathbf{x})}$$

Demo: Gaussian Random Walk Sampler



MCMC Practical Aspects

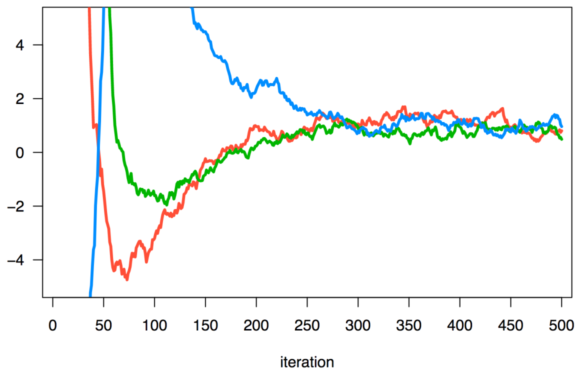
Issues with MCMC

- **Burn-in:** The underlying Markov chains take time to converge to the distribution of interest. The time needed to reach the stationary distribution of the chain is called the *burn-in time*.
- **Autocorrelation:** Consecutive samples drawn from the chain at equilibrium may be highly correlated with each other. The time lag between samples that are approximately independent of each other is called the *autocorrelation time* of the chain.

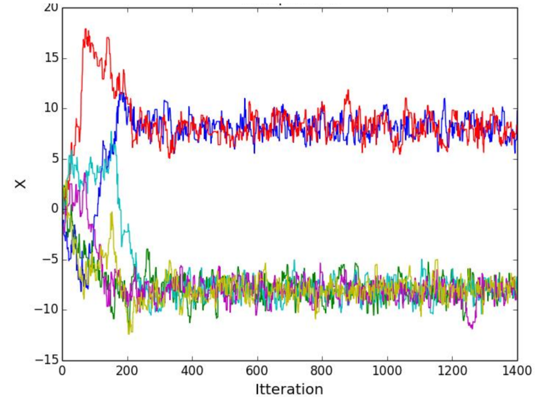
Burn-in Time

- The most fundamental issue with burn-in is that, in the absence of a theoretical lower bound, you can never be exactly sure that the chain has converged to the equilibrium distribution.
- MCMC practitioners usually rely on heuristic convergence diagnostics to assess burn-in time.
- One of the most useful heuristics is to run multiple chains from different starting points and track one or more scalar functions of the state of the chain (the log probability of the data is often a good choice).
- The distribution of values of these functions will all converge to the same mean and variance at equilibrium.

Example: Burn-in Time



Example: Burn-in Time



Autocorrelation Time

- At or near equilibrium, different samplers can traverse the state space at different rates.
- The autocorrelation time of a sampler is the number of sampling iterations we must apply at equilibrium to obtain two samples that are approximately independent.
- Practitioners sometimes use estimates of autocorrelation at different lags to estimate an "effective sample size" of S MCMC samples

Practical Aspects Summary

- There are many different diagnostics
- It's often easy to diagnose clear failures
- It's basically impossible to diagnose success
- Some practitioners advocate just running one chain for a very long time