MCMC Methods: Gibbs and Metropolis

Patrick Breheny

February 28

Introduction

- As we have seen, the ability to sample from the posterior distribution is essential to the practice of Bayesian statistics, as it allows Monte Carlo estimation of all posterior quantities of interest
- Typically however, direct sampling from the posterior is not possible either
- Today, we discuss two mechanisms that allow us to carry out this sampling when a direct approach is not possible (Gibbs sampling and the Metropolis-Hastings algorithm), as well as discuss *why* these approaches work

Markov chains

- A sequence of random variables $X^{(0)}, X^{(1)}, X^{(2)}, \ldots$ is said to form a *Markov chain* if, for all t,

$$p(X^{(t+1)} = x) = t(x|X^{(t)});$$

  in other words, the distribution of $X^{(t+1)}$ depends only on the previous draw, and is independent of $X^{(0)}, X^{(1)}, \ldots, X^{(t-1)}$
- The function $t(x|X^{(t)})$ defines the *transition probabilities* or *transition distribution* of the chain

Stationary distributions

- A distribution $\pi(x)$ is *stationary* with respect to a Markov chain if, given that $X^{(t)} \sim \pi$, $X^{(t+1)} \sim \pi$
- Provided that a Markov chain is positive recurrent, aperiodic, and irreducible (next slide), it will converge to a unique stationary distribution, also known as an *equilibrium distribution*, as $t \to \infty$
- This stationary distribution is determined entirely by the transition probabilities of the chain; the initial value of the chain is irrelevant in the long run
- In Bayesian statistics, we will be interested in constructing Markov chains whose equilibrium is the posterior distribution

Conditions

The following conditions are required for a Markov chain to converge to a unique stationary distribution (below, I use "set" to refer to a set with nonzero probability $\pi(A)$):

- *Irreducibile*: Any set A can be reached from any other set B with nonzero probability
- *Positive recurrent*: For any set A, the expected number of steps required for the chain to return to A is finite
- *Aperiodic*: For any set A, the number of steps required to return to A must not always be a multiple of some value k
- Thankfully, these conditions are typically met in Bayesian statistics

Metropolis-Hastings

Suppose that a Markov chain is in position x; the *Metropolis-Hastings algorithm* is as follows:

(1) Propose a move to y with probability $q(y|x)$

(2) Calculate the ratio

$$r = \frac{p(y)q(x|y)}{p(x)q(y|x)}$$

(3) Accept the proposed move with probability

$$\alpha = \min\{1, r\};$$

otherwise, remain at x (*i.e.*, $X^{(t+1)} = X^{(t)}$)

Stationary distribution

- The description on the previous slide allows asymmetric proposals; if the proposal is symmetric, *i.e.*, $q(y|x) = q(x|y)$, the ratio is simply $r = p(y)/p(x)$
- **Theorem (detailed balance):** For any sets A and B, $P(A)T(B|A) = P(B)T(A|B)$, where $T(A|B)$ is the transition probability from $B \to A$ imposed by the Metropolis-Hastings algorithm
- **Theorem:** The Markov chain with transition probabilities arising from the Metropolis-Hastings algorithm has the posterior distribution p as a stationary distribution

Example

- As an example of how the Metropolis-Hastings algorithm works, let's sample from the following posterior:

$$Y \sim t_5(\mu, 1)$$
$$\mu \sim t_5(0, 1)$$

- The following code can be used to calculate the posterior density:

```
p <- function(mu) {
  dt(mu, 5) * prod(dt(y, 5, mu))
}
```

- In practice, it is better to work with probabilities on the log scale to avoid numerical overflow/underflow, but the above will be sufficient for our purposes today

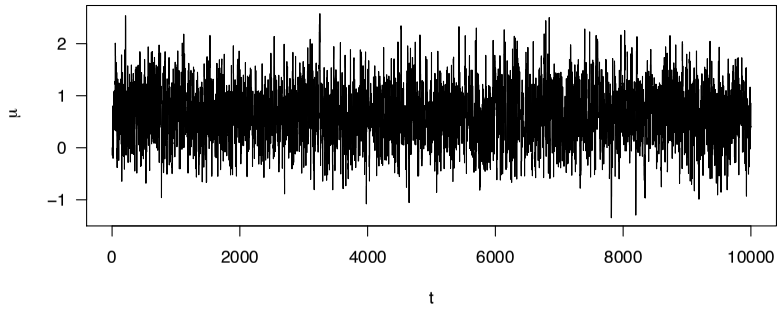
Example

The Metropolis algorithm can be coded as follows:

```
N <- 10000
mu <- numeric(N)
for (i in 1:(N-1)) {
  proposal <- mu[i] + rnorm(1)
  r <- p(proposal)/p(mu[i])
  accept <- rbinom(1, 1, min(1,r))
  mu[i+1] <- if (accept) proposal else mu[i]
}
```

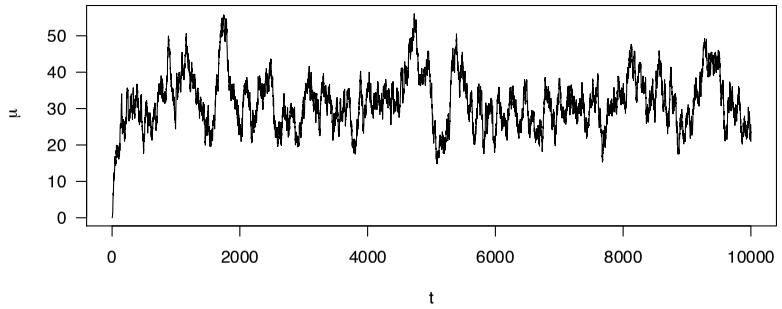
Results

Suppose we observe $y = c(-1, 1, 5)$:



Results

Now suppose we observe $y = c(39, 41, 45)$:



Remarks

- The preceding plots are known as *trace plots*; we will discuss trace plots further next week when we discuss convergence diagnostics
- The preceding examples illustrate the notion of Markov chains converging to the posterior distribution regardless of where they start
- Note, however, that this may take a while (≈ 100 draws in the second example; this can be much larger in multidimensional problems)
- Thus, it might be desirable to discard the beginning values of the Markov chain – for the second example, only considering draws from 101 onward to be draws from the posterior

Burn-in

- This idea is known as *burn-in*
- Both BUGS and JAGS (and `R2OpenBUGS` and `R2jags`) allow you to set the number of burn-in iterations – to run the Markov chain for a while before you begin to record draws
- The advantage of this is to eliminate dependency on the initial values (which are arbitrary) from the results
- Strictly speaking, burn-in is not necessary, since if you simply run the chain long enough, the impact of the initial values will gradually diminish and achieve the same result
- Discarding the initial values of the chain, however, is often faster, especially if you are interested in estimating quantities pertaining to the tails of the distribution

Autocorrelation

- Note that although the marginal distribution of $X^{(t)}$ converges to the posterior, that doesn't mean that the chain converges to a chain producing IID draws from the posterior
- Indeed, in the second example, consecutive draws were quite highly correlated (this is known as *autocorrelation*, which we will discuss in greater depth next week)

Proposal distribution: Tradeoffs

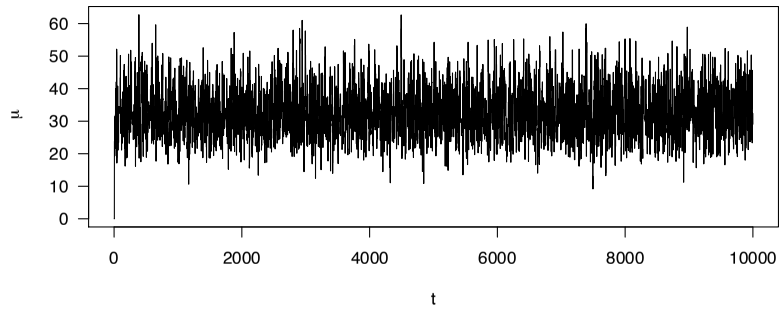
- The high degree of autocorrelation is a consequence of the proposal distribution
- Newcomers to the Metropolis-Hastings algorithm often feel that rejecting a proposal is a bad outcome and that we should minimize the probability that it occurs
- However, while an excessive amount of rejection is indeed bad, too little rejection is also bad, as it indicates that the proposals are too cautious and represent only very small movements around the posterior distribution (giving rise to high autocorrelation)

Proposal distribution: Tradeoffs

- Our original proposal had $\sigma = 1$; for the first example, this led to an acceptance rate of 53.5%, for the second example it led to an acceptance rate of 95.5%
- Informally, it certainly appeared that the Markov chain worked better in the first example than in the second
- Formally, there are theoretical arguments indicating that the optimal acceptance rate is 44% for one dimension, and has a limit of 23.4% as the dimension goes to infinity
- We can achieve these targets by modify the proposal; specifically, by increasing or decreasing σ^2, the variance of the normal proposal

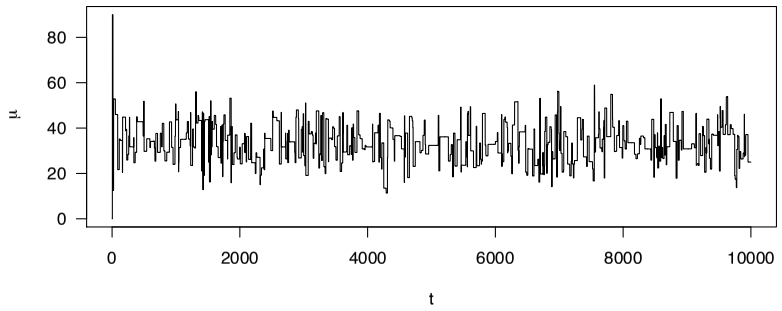
Results: $y = c(39, 41, 45)$, $\sigma = 15$

Acceptance rate: 49.6%



Results: $y = c(39, 41, 45)$, $\sigma = 200$

Acceptance rate: 4.9%



Remarks

- Traceplots should look like "fat, hairy caterpillars", as they do in slides 10 and 15; not like they do on slides 11 or 16
- Both BUGS and JAGS allow for "adapting phases" in which they try out different values of σ (or other such tuning parameters) to see which ones work the best before they actually start the "official" Markov chain; we will discuss these as they come up

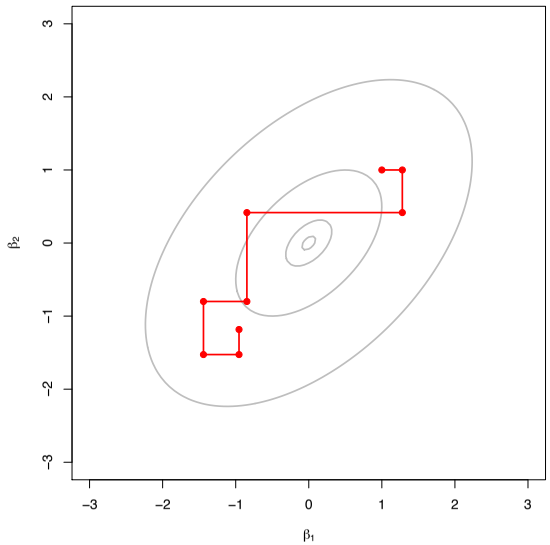
Idea

- Another extremely useful technique for sampling multidimensional distributions is *Gibbs sampling*, which we have already encountered
- The basic idea is to split the multidimensional $\boldsymbol{\theta}$ into blocks (often scalars) and sample each block separately, conditional on the most recent values of the other blocks
- The beauty of Gibbs sampling is that it simplifies a complex high-dimensional problem by breaking it down into simple, low-dimensional problems

Formal description

- Formally, the algorithm proceeds as follows, where θ consists of k blocks $\theta_1, \theta_2, \ldots, \theta_k$: at iteration (t),
  - Draw $\theta_1^{(t+1)}$ from
  $$p(\theta_1 | \theta_2^{(t)}, \theta_3^{(t)}, \ldots, \theta_k^{(t)})$$
  - Draw $\theta_2^{(t+1)}$ from
  $$p(\theta_2 | \theta_1^{(t+1)}, \theta_3^{(t)}, \ldots, \theta_k^{(t)})$$
  - $\ldots$
- This completes one iteration of the Gibbs sampler, thereby producing one draw $\theta^{(t+1)}$; the above process is then repeated many times
- The distribution $p(\theta_1 | \theta_2^{(t)}, \theta_3^{(t)}, \ldots, \theta_k^{(t)})$ is known as the *full conditional* distribution of θ_1

Gibbs: Illustration



Justification for Gibbs sampling

- Although they appear quite different, Gibbs sampling is a special case of the Metropolis-Hasting algorithm
- Specifically, Gibbs sampling involves a proposal from the full conditional distribution, which always has a Metropolis-Hastings ratio of 1 – *i.e.*, the proposal is always accepted
- Thus, Gibbs sampling produces a Markov chain whose stationary distribution is the posterior distribution, for all the same reasons that the Metropolis-Hastings algorithm works

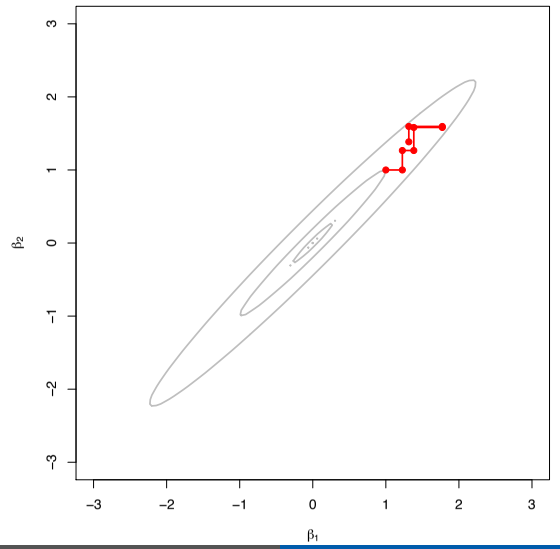
Gibbs vs. Metropolis

- Thus, there is no real conflict as far as using Gibbs sampling or the Metropolis-Hastings algorithm to draw from the posterior
- In fact, they are frequently used in combination with each other
- As we have seen, semi-conjugacy leads to Gibbs updates with simple, established methods for drawing from the full conditional distribution
- In other cases, we may have to resort to Metropolis-Hastings to draw from some of the full conditionals

Variable-at-a-time Metropolis

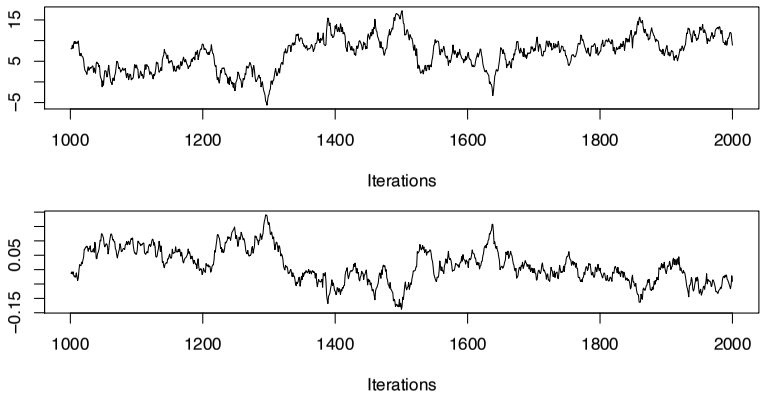
- This sort of approach is known as *variable-at-a-time Metropolis-Hastings* or *Metropolis-within-Gibbs*
- The advantage of the variable-at-a-time approach, as mentioned earlier, is that it is much easier to propose updates for a single variable than for many variables
- The disadvantage, however, is that when variables are highly correlated, it may be very difficult to change one without simultaneously changing the other

Gibbs with highly correlated parameters



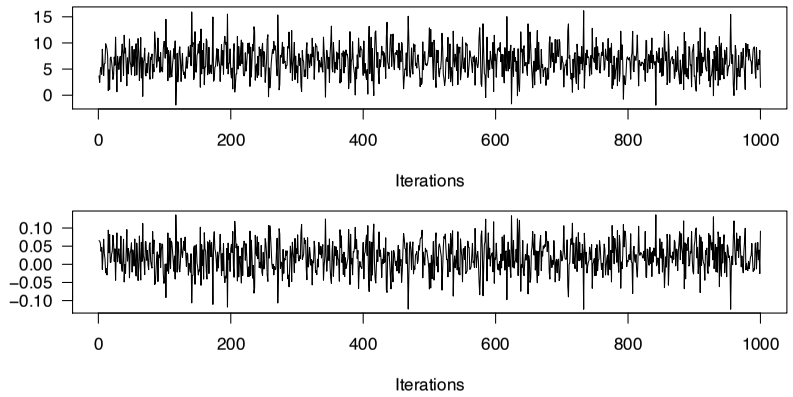
Regression: BUGS

Correlated posteriors can often arise in regression (see code for details)



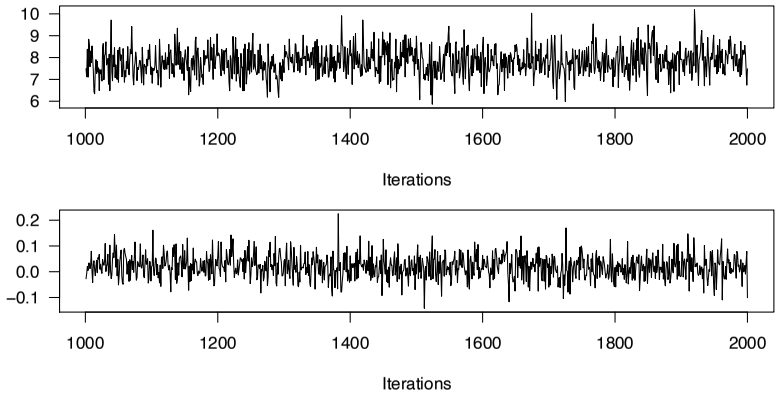
Regression: JAGS

JAGS avoids this problem (for regression) by updating the block (β_1, β_2) at once:



Regression: BUGS

We can also avoid this problem in BUGS by centering the variables or by specifying `beta` as a multivariate parameter



Conclusion

- Note that JAGS is not, in general, impervious to correlated posteriors: it merely recognizes this situation for linear models and GLMs
- Certainly, there is a large body of work on other computational approaches to sampling (slice sampling, adaptive rejection sampling, Hamiltonian Monte Carlo, etc.); covering such methods is beyond the scope of this course
- Nevertheless, I hope that by exploring the two most widely known methods (Metropolis and Gibbs), I have conveyed a sense of how such MCMC sampling methods work, why they work, and in what situations they may fail to work