Monte Carlo Simulation
5-Page Essay:
Implementation Issues and Convergence Diagnostics for MCMC

Jake Singleton

Department of Statistics, The University of Chicago

April 28, 2022

1 Opening Remarks

“It is not possible to say with certainty that a finite sample from an MCMC algorithm is representative of an underlying stationary distribution.” - Cowles and Carlin 1996

Although impossible to declare convergence with certainty, statisticians have worked extensively to create tools to assess convergence effectively. In this essay I will discuss two approaches that apply to any MCMC sampler, the first being a recipe for creating and diagnosing a multi-sequence sampler, and the second being more of a quick graphical heuristic for individual chains. We begin with the former.

2 Approach 1: Recipe and Diagnosis

2.1 MCMC Recipe

The two overarching ideas of this approach are 1) create an overdispersed starting distribution covering the target from which we sample starting values for multiple chains, and 2) use the sequences to perform inference on the target distribution. Note the notion of finding a distribution "covering the target" is in the style of rejection sampling. The recipe is as follows.

1. Locate the mode(s) of the target distribution P using a numerical optimizer (the EM algorithm is a common choice). The optimizer should be run multiple times from different starting points to find all modes of the target if the target is multi-modal. Suppose the number of modes is K. Estimate the Hessian at each of the K modes. We approximate the regions of high target density with a mixture of K multivariate normals, where each normal is $N_n(\mu_k, \Sigma_k)$ fit to its own Hessian. Then the associated mixture model approximating P is

$$\widehat{P}(x) = \sum_{k=1}^{K} \pi_k (2\pi)^{-d/2} |\Sigma_k|^{-1/2} \text{exp}\Big(-\frac{1}{2}(x-\mu_k)^t \Sigma_k^{-1}(x-\mu_k)\Big)$$

with $\sum_k \pi_k = 1$. The π_k are called the mixture components and can be calculated by setting $\widehat{P}(\mu_k) = P(\mu_k)$, and d is the dimension of x.

2. Make the approximation overdispersed. First, we draw from the normal mixture and divide each sample by a random χ^2_α variate normalized by α. The resulting distribution

$$\tilde{P}(x) \propto \sum_{k=1}^{K} \pi_k (2\pi)^{-d/2} |\Sigma_k|^{-1/2} (\alpha + (x-\mu_k)^t \Sigma_k^{-1}(x-\mu_k))^{-(d+\alpha)/2}$$

is a mixture of multivariate t-distributions. A common choice in practice is $\alpha = 4$. Transforming our samples from a multivariate normal to a multivariate t in this way has the effect of yielding overdispersed samples.

3. Downweight regions with low target density using *importance resampling* (SIR). This algorithm generates starting points for the multiple sequences.

Algorithm 2.1 Importance Resampling (SIR)

```
**Input**: Sample size $N$, target P, t mixture $\tilde{P}$, sample size $M < N$
For $n = 1, \dots, N$ do:
Draw $X^{(n)} \sim \tilde{P}$
Set $\mathrm{IR}^{(n)} = P(X^{(n)})/\tilde{P}(X^{(n)})$, called the *importance ratio*
Stop after iteration $N$ and set $S = \{X^{(1)}, \dots, X^{(N)}\}$.
For $m = 1, \dots, M$ do:
Pick $Y^{(m)}$ from $S$, each $X^{(n)}$ has probability of being picked proportional to $\mathrm{IR}^{(n)}$
Discard the selected $X^{(n)}$ from $S$
**Output**: Starting points $Y^{(1)}, \dots, Y^{(M)}$
```

As $N \to \infty$, $Y^{(m)} \sim P$ under mild regularity conditions. In practice $N = 1,000$ and $M = 10$ are reasonable choices. The M SIR starting points are closer to P than any draws from $\tilde{P}$ would otherwise be, thereby speeding convergence of the MCMC sampler.

4. Run the MCMC sampler from each of the M starting points.

2.2 MCMC Diagnosis

In this section we will discuss how to assess whether our sampler converges, bearing in mind it is impossible to say so with complete certainty. We denote x a scalar estimand of interest as opposed to the (potentially) multi-dimensional random variable being simulated. With our M independent sequences in hand, we discard up to half of the iterations of each sequence as burn-in, leaving n iterations for each sequence. Next, we calculate two quantities: $B/n = \sum_{i=1}^{M}(\bar{x}_{i\cdot} - \bar{x}_{\cdot\cdot})^2/(M-1)$, where $\bar{x}_{\cdot\cdot}$ = the average of the $M *$ n iterates, $\bar{x}_{i\cdot}$ = the average of the n iterates for each of the M sequences, and $W = \sum_{i=1}^{M} s_i^2/M$, where $s_i^2 = \sum_{j=1}^{n}(x_{ij} - \bar{x}_{i\cdot})^2/n - 1, i = 1, \dots, M$. We can interpret these metrics in an ANOVA sense: B measures the variability between the M sequences, and W measures the average within-sequence variability. The key difference is that, when we see a treatment effect in ANOVA, B is significantly larger than W. However, in MCMC, it is actually W that we would like to be much larger than B, as if the within-sequence variance is high, then the sequence has "forgotten" its starting point and mixed well (i.e. explored sufficiently) with the target. Next, we quite naturally use $\bar{x}_{\cdot\cdot}$ as our estimator of the target mean, and also use $\widehat{\sigma}^2 = \frac{n-1}{n}W + \frac{B}{n}$, a weighted average of W and B, as our estimate of the target variance. Our approximation of the target using our M sequences is $x \sim$ a t-distribution centered at $\widehat{\mu} = \bar{x}_{\cdot\cdot}$., scale $\sqrt{\widehat{V}} = \sqrt{\widehat{\sigma}^2 + B/mn}$, and $\mathrm{df} = 2\widehat{V}^2/\widehat{\mathrm{var}}(\widehat{V})$, where $\widehat{\mathrm{var}}(\widehat{V}) = \left(\frac{n-1}{n}\right)^2 \frac{1}{M}\widehat{\mathrm{var}}(s_i^2) + \left(\frac{M+1}{Mn}\right)^2 \frac{2}{M-1}B^2 + 2\frac{(M+1)(n-1)}{Mn^2} \cdot \frac{n}{M}(\widehat{\mathrm{cov}}(s_i^2, \bar{x}_{i\cdot}^2) - 2\bar{x}_{\cdot\cdot}\widehat{\mathrm{cov}}(s_i^2, \bar{x}_{i\cdot}))$. Note the estimated variances and covariances are merely sample variances and covariances obtained from the M sample values of $\bar{x}_{i\cdot}$ and s_i^2.

Now, the **key convergence diagnostic** of interest to us is $\sqrt{\widehat{R}} = \sqrt{(\widehat{V}/W)\mathrm{df}/(\mathrm{df}-2)}$ (observe that large W shrinks $\widehat{R}$). This is the factor by which the scale of our current distribution on x (the t-distribution $\tilde{P}$) would decrease if $n \to \infty$. Intuitively, if n were infinite, then $\widehat{R}$ would be 1, as we would have certain convergence to the target, and indeed one can show $\widehat{R} \to 1$ as $n \to \infty$. If $\sqrt{\widehat{R}}$ is close to 1, we can be fairly sure our iterative sampler has converged. If not, we have evidence suggesting we would benefit from increasing n. Ultimately, once $\widehat{R}$ is close to 1 for all scalar estimands of interest, we can feel confident in convergence and may use the remaining $M * n$ iterates to summarize the target however we wish.

Advantages

- Using multiple chains at overdispersed starting points allows us to find all regions of high target density.
- Using a t-distribution approximation to the target is conservative and accounts for sampling variability.
- $\widehat{R}$ will not be small unless the sequences have moved away from their starting neighborhoods, leading to large W.
- Works for any MCMC sampler.

Disadvantages

- Using multiple chains means we cannot run a single, very long chain. If a single chain is run for 10,000 iterations while 10 independent chains are each run for 1,000 iterations, then the last 9,000 iterates of the single, long chain are more likely to be closer to the target than those reached by any of the shorter chains.
- This method requires us to find a suitable $\tilde{P}$ that indeed covers the target, which may be difficult.
- Univariate.

3 Approach 2: Graphical Diagnostic

3.1 The Cusum Path Plot

The diagnostic tool we turn to now is a graphical one called the cusum path plot, much more heuristic than the aforementioned strategy. Before we define the plot, know that this tool works for individual chains from any MCMC sampler and is applied to, again, a *univariate* summary statistic. Given such a chain and statistic of interest, the first order of business is to sequentially plot values of the summary statistic over the entire duration of the chain and discard some initial observations as burn-in. For example, if $T(X)$ is our univariate summary statistic, then we plot $T(X^{(i)})$ vs. i, $i = 1, \dots, N$ where N is the number of iterations in our chain. Suppose we discard the first N_0 iterations, leaving us with iterations $N_0 + 1$ to N.

Having discarded burn-in, now we can turn to the cusum plot itself. The estimate of the mean of the summary statistic of interest is naturally $\widehat{\mu} = \frac{1}{N-N_0} \sum_{i=N_0+1}^{N} T(X^{(i)})$. We define the observed *cusum* or *partial sum* as $\widehat{S}_t = \sum_{i=N_0+1}^{t} \Big(T(X^{(i)}) - \widehat{\mu} \Big), t = N_0 + 1, \dots, N$. That is, each $\widehat{S}_t$ is the cumulative sum of the differences between our summary statistic of interest and its overall average after t iterations minus burn-in. Then we create the cusum path plot itself by simply plotting $\widehat{S}_t$ vs. t, $t = N_0 + 1, \ldots, N$. Note the plot ends at 0 by definition of $\widehat{S}_t$.

Let's see what one might look like. The target density in this example is a mixture of two univariate normals: $P(x) = \pi_1 N(x; 0, 1) + \pi_2 N(x; 3, 1)$ where $N(x; \mu, \sigma^2)$ denotes the normal density with mean μ and variance σ^2. An independence sampler (unknown proposal) is run alongside two Metropolis-Hastings samplers, one with proposal kernel $N(x; x, 4.146)$ and the other $N(x; x, 3)$. $N = 2000$, and the starting distribution is $N(0, 1)$. The first 1000 iterates were discarded as burn-in. The target is bimodal, and the sample statistic of interest in this case then is just the scalar iterate $X^{(i)}$. Their cusums are:

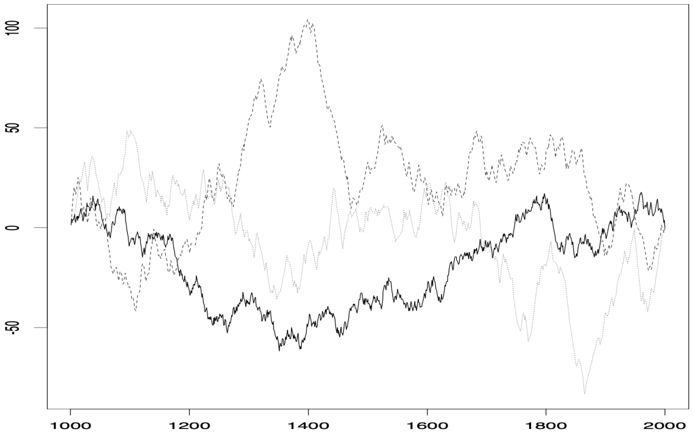

Figure 1 Solid line: Indep. Sampler, Dotted: $N(x; x, 4.146)$, Dashed: $N(x; x, 3)$

The interpretation of the plot is that, if the $\widehat{S}_t$ continue to either increase or decrease far from 0, then our sampler is slow-mixing. So any type of smooth behavior and/or systematic wandering from 0 is bad. If the plot appears "hairy" or zig-zags up and down then the sampler is mixing well. A slightly more mathematical explanation is that, the more "zig-zaggy" the plot is, the more uncorrelated the iterates of the chain are, indicating good mixing. The independence sampler looks best to my eye, followed by the dotted $N(x; x, 4.146)$ kernel and last the dashed $N(x; x, 3)$ kernel.

If one has one chain and one chain only, it is best practice to include a "benchmark" plot as a reference point. The benchmark plot is created by drawing $Y_{N_0+1}, \ldots, Y_N \sim$ i.i.d. $N(\widehat{\mu}, \widehat{\sigma}^2)$ where $\widehat{\mu}$ is the sample mean of the $T(X^{(i)})$ as defined earlier and $\widehat{\sigma}^2 = \frac{1}{N-(N_0+1)-1} \sum_{i=N_0+1}^{N} (T(X^{(i)}) - \widehat{\mu})^2$ is the sample variance of the $T(X^{(i)})$. Next, we form quite naturally $\widehat{S}_t^b = \sum_{i=N_0+1}^{N} (Y_j - \widehat{\mu}_Y)$, where $\widehat{\mu}_Y$ is the sample mean of the Y_i. Then, in the same plot with the $\widehat{S}_t$, we plot $\widehat{S}_t^b$ vs. t, $t = N_0 + 1, \ldots, N$. We view this benchmark plot as what an idealized cusum plot would look like, as it approximates the cumulative sums of an i.i.d. sequence from the target distribution. We have an example from the same setting as earlier:

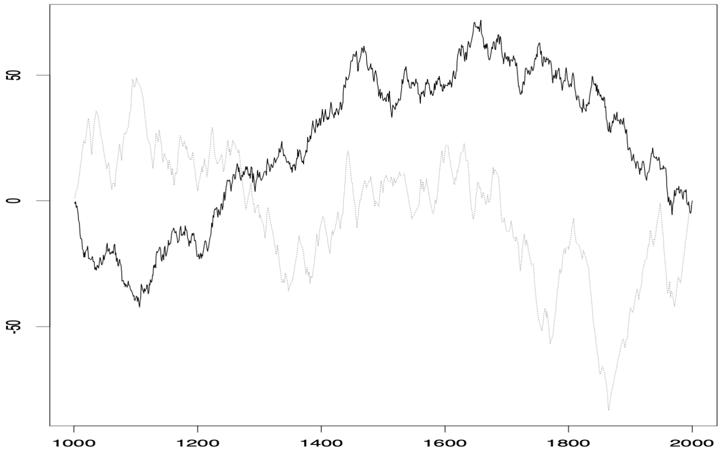

Figure 2 Solid line: Benchmark, Dotted: $N(x; x, 4.146)$ kernel

In this case the cusum plot for the $N(x; x, 4.146)$ kernel looks good. We should emphasize that the cusum path need not follow the same trajectory as the benchmark plot; what we should look out for is the smoothness/zig-zag nature of the paths and how far they wander from 0.

Advantages

- Easy to make.
- Works for any MCMC sampler.

Disadvantages

- When multiple modes exist, the plot may falsely indicate good mixing when in fact the chain is stuck in a local mode.
- Univariate and subjective.

4 Closing Remarks

In this essay we have seen one complete walkthrough of implementing an effective MCMC sampler and two ways of diagnosing MCMC results. The first diagnostic, $\widehat{R}$ in particular, is quantitative and pairs naturally with the walkthrough. Meanwhile, the cusum plot is entirely graphical. One key advantage of these approaches is they work for any MCMC sampler, and one key disadvantage is they're both univariate (plenty of multivariate diagnostics exist). In general, we should always use multiple tools to assess convergence, and these approaches provide excellent starting points. Last, know that diagnostics sometimes give contradictory answers. As statisticians, it is our job to proceed as best we can.

5 Bibliography

"Approach 1: Recipe and Diagnosis" is primarily attributed to:
Gelman, Andrew & Rubin, Donald. (1992). Inference From Iterative Simulation Using Multiple Sequences. Statistical Science. 7. 10.1214/ss/1177011136.

"Approach 2: Graphical Diagnostic" is primarily attributed to:
Yu, Bin & Mykland, Per. (1998). Looking at Markov samplers through cusum path plots: A simple diagnostic idea. Statistics and Computing. 8. 275-286. 10.1023/A:1008917713940.

Additional support on both approaches:
Cowles, M. K., & Carlin, B. P. (1996). Markov chain Monte Carlo convergence diagnostics: a comparative review. Journal of the American Statistical Association, 91(434), 883+.

With a high-level overview from:
Brooks, Stephen & Gelman, Andrew. (1998). Some Issues in Monitoring Convergence of Iterative Simulations. J. Comput. Graph. Stat. 7.