Bayesian Computation: Posterior Sampling & MCMC

Tom Loredo
Dept. of Astronomy, Cornell University
http://www.astro.cornell.edu/staff/loredo/bayes/

Cosmic Populations @ CASt — 9–10 June 2014

Posterior Sampling & MCMC

❶ Posterior sampling

❷ Markov chain Monte Carlo

Markov chain properties

Metropolis-Hastings algorithm

Classes of proposals

❸ MCMC diagnostics

Posterior sample diagnostics

Joint distribution diagnostics

Cautionary advice

❹ Beyond the basics

Posterior Sampling & MCMC

1. **Posterior sampling**

2. **Markov chain Monte Carlo**
   - Markov chain properties
   - Metropolis-Hastings algorithm
   - Classes of proposals

3. **MCMC diagnostics**
   - Posterior sample diagnostics
   - Joint distribution diagnostics
   - Cautionary advice

4. **Beyond the basics**

Posterior sampling

$$\int d\theta\ g(\theta)p(\theta|D) \approx \frac{1}{n} \sum_{\theta_i \sim p(\theta|D)} g(\theta_i) + O(n^{-1/2})$$

When $p(\theta)$ is a posterior distribution, drawing samples from it is called *posterior sampling* (or *simulation from the posterior*):

- *One set of samples* can be used for many different calculations (so long as they don't depend on low-probability events)
- This is the most promising and general approach for Bayesian computation in *high dimensions*—though with a twist (MCMC!)

Challenge: How to build a RNG that samples from a posterior?

Accept-Reject Algorithm

1. Choose a tractable density $h(\theta)$ and a constant C so Ch bounds q
2. Draw a candidate parameter value $\theta' \sim h$
3. Draw a uniform random number, u
4. If $q(\theta') < Ch(\theta')$, record θ' as a sample
5. Goto 2, repeating as necessary to get the desired number of samples.

Efficiency = ratio of volumes, Z/C.

In problems of realistic complexity, the efficiency is intolerably low for parameter spaces of more than several dimensions.

Take-away idea: *Propose candidates that may be accepted or rejected*

Posterior Sampling & MCMC

❶ Posterior sampling

❷ **Markov chain Monte Carlo**

- Markov chain properties
- Metropolis-Hastings algorithm
- Classes of proposals

❸ MCMC diagnostics

- Posterior sample diagnostics
- Joint distribution diagnostics
- Cautionary advice

❹ Beyond the basics

Markov Chain Monte Carlo*

Accept/Reject aims to produce *independent* samples—each new θ is chosen irrespective of previous draws.

To enable exploration of complex pdfs, let's introduce *dependence*: Choose new θ points in a way that

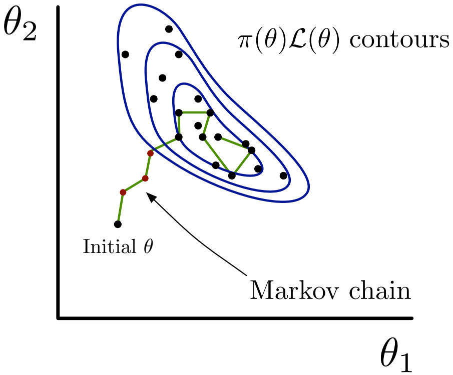
- Tends to *move toward* regions with higher probability than current
- Tends to *avoid* lower probability regions

The simplest possibility is a *Markov chain*:

$$\begin{aligned} p(\text{next location} | \textit{current } \textbf{and } \textit{previous locations}) \\ = p(\text{next location} | \textit{current location}) \end{aligned}$$

A Markov chain "has no memory."

*Chib & Greenberg (1995): "Understanding the Metropolis-Hastings Algorithm"


θ_2
$\pi(\theta)\mathcal{L}(\theta)$ contours
Initial θ
Markov chain
θ_1

Equilibrium Distributions

Start with some (possibly random) point θ_0; produce a sequence of points labeled in order by a "time" index, θ_t.

Ideally we'd like to have $p(\theta_t) = q(\theta_t)/Z$ for each t. Can we do this with a Markov chain?

To simplify discussion, discretize parameter space into a countable number of *states*, which we'll label by x or y (i.e., cell numbers). If θ_t is in cell x, we say state $S_t = x$.

Focus on *homogeneous Markov chains*:

$$p(S_t = y|S_{t-1} = x) = T(y|x), \quad \text{transition probability (matrix)}$$

Note that $T(y|x)$ is a probability distribution over y, and does not depend on t.

Aside: There is no standard notation for any of this—including the order of arguments in T!

What is the probability for being in state y at time t? How does it evolve?

$$\begin{aligned}
p(S_t = y) &= \sum_x p(S_t = y, S_{t-1} = x) \\
&= \sum_x p(S_{t-1} = x)\, p(S_t = y | S_{t-1} = x) \\
&= \sum_x p(S_{t-1} = x) T(y|x) \\
&= p(S_{t-1} = y) T(y|y) + \sum_{x \neq y} p(S_{t-1} = x) T(y|x) \\
&= p(S_{t-1} = y) \left[1 - \sum_{x \neq y} T(x|y) \right] + \sum_{x \neq y} p(S_{t-1} = x) T(y|x) \\
&= p(S_{t-1} = y) \\
&\qquad + \sum_{x \neq y} \left[p(S_{t-1} = x) T(y|x) - p(S_{t-1} = y) T(x|y) \right] \\
&= p(\text{was at } y) + p(\text{move to } y) - p(\text{move from } y)
\end{aligned}$$

What is the probability for being in state y at time t?

$$\begin{aligned} p(S_t = y) &= p(\text{was at } y) + p(\text{move to } y) - p(\text{move from } y) \\ &= p(S_{t-1} = y) \\ &\quad + \sum_{x \neq y} \left[p(S_{t-1} = x)\, T(y|x) - p(S_{t-1} = y)\, T(x|y) \right] \end{aligned}$$

If the sum vanishes, then there is an *equilibrium distribution*:

$$p(S_t = y) = p(S_{t-1} = y) \equiv p_{\mathrm{eq}}(y)$$

If we *start* in a state drawn from p_{eq}, every subsequent sample will be a (dependent) draw from p_{eq}.

Reversibility/Detailed Balance

A sufficient (but not necessary!) condition for there to be an equilibrium distribution is for *each* term of the sum to vanish:

$$\begin{aligned} p_{\mathrm{eq}}(x)T(y|x) &= p_{\mathrm{eq}}(y)T(x|y) \qquad \textit{or} \\ \frac{T(y|x)}{T(x|y)} &= \frac{p_{\mathrm{eq}}(y)}{p_{\mathrm{eq}}(x)} \end{aligned}$$

the *detailed balance* or *reversibility* condition

If we set $p_{\mathrm{eq}} = q/Z$, and we build a reversible transition distribution for this choice, then *the equilibrim distribution will be the posterior distribution*

Convergence

Problem: What about $p(S_0 = x)$?

If we start the chain with a draw from the posterior, every subsequent draw will be from the posterior. But we can't do this!

Convergence

If the chain produced by $T(y|x)$ satisifies two conditions:

- It is *irreducible*: From any x, we can reach any y with finite probability in a finite $\#$ of steps
- It is *aperiodic*: The transitions never get trapped in cycles

then $p(S_t = s) \rightarrow p_{\mathrm{eq}}(x)$

Early samples will show evidence of whatever procedure was used to generate the starting point → discard samples in an initial "burn-in" period

Designing Reversible Transitions

Set $p_{\rm eq}(x) = q(x)/Z$; how can we build a $T(y|x)$ with this as its EQ dist'n?

Steal an idea from accept/reject: Start with a *proposal* or candidate distribution, $k(y|x)$. Devise an accept/reject criterion that leads to a reversible $T(y|x)$ for q/Z.

Using any $k(y|x)$ as T will not guarantee reversibility. E.g., from a particular x, the transition rate to a particular y may be too large:

$$q(x)k(y|x) > q(y)k(x|y) \qquad \textit{Note: } Z \textit{ dropped out!}$$

When this is true, we should use rejections to reduce the rate to y.

Acceptance probability: Accept y with probability $\alpha(y|x)$; reject it with probability $1 - \alpha(y|x)$ and stay at x:

$$T(y|x) = k(y|x)\alpha(y|x) + [1 - \alpha(y|x)]\delta_{y,x}$$

The detailed balance condition is a requirement for $y \neq x$ transitions, for which $\delta_{y,x} = 0$; it gives a condition for α:

$$q(x)k(y|x)\alpha(y|x) = q(y)k(x|y)\alpha(x|y)$$

Suppose $q(x)k(y|x) > q(y)k(x|y)$; then we want to suppress $x \to y$ transitions, but we want to maximize $y \to x$ transitions. So we should set $\alpha(x|y) = 1$, and the condition becomes:

$$\alpha(y|x) = \frac{q(y)k(x|y)}{q(x)k(y|x)}$$

If instead $q(x)k(y|x) < q(y)k(x|y)$, the situation is reversed: we want $\alpha(y|x) = 1$, and $\alpha(x|y)$ should suppress $y \to x$ transitions

We can summarize the two cases as:

$$\alpha(y|x) = \begin{cases} \frac{q(y)k(x|y)}{q(x)k(y|x)} & \text{if } q(y)k(x|y) < q(x)k(y|x) \\ 1 & \text{otherwise} \end{cases}$$

or equivalently:

$$\alpha(y|x) = \min\left[\frac{q(y)k(x|y)}{q(x)k(y|x)}, 1\right]$$

Metropolis-Hastings algorithm

Given a target quasi-distribution $q(x)$ (it need not be normalized):

1. Specify a proposal distribution $k(y|x)$ (make sure it is irreducible and aperiodic).
2. Choose a starting point x; set $t = 0$ and $S_t = x$
3. Increment t
4. Propose a new state $y \sim k(y|x)$
5. If $q(x)k(y|x) < q(y)k(x|y)$, set $S_t = y$; goto (3)
6. Draw a uniform random number u
7. If $u < \frac{q(y)k(x|y)}{q(x)k(y|x)}$, set $S_t = y$; else set $S_t = x$; goto (3)

The art of MCMC is in *specifying the proposal distribution $k(y|x)$*

We want:

- New proposals to be accepted, so there is movement
- Movement to be significant, so we explore efficiently

These desiderata compete!

Random walk Metropolis (RWM)

Propose an *increment*, z, from the current location, not dependent on the current location, so $y = x + z$ with a specified PDF $K(z)$, corresponding to

$$k(y|x) = K(y - x)$$

The proposals would give rise to a *random walk* if they were all accepted; the M-H rule modifies them to be a kind of directed random walk

Most commonly, a symmetric proposal is adopted:

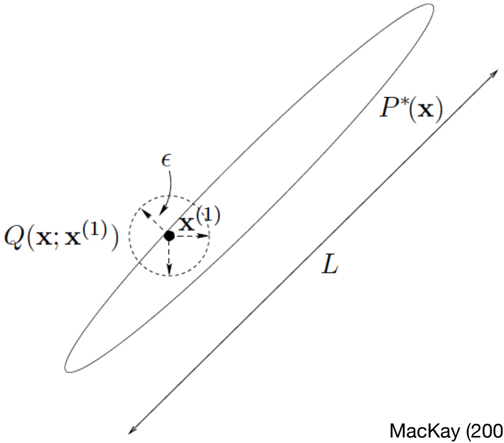
$$k(y|x) = K(|y - x|)$$

The acceptance probability simplifies:

$$\alpha(y|x) = \min\left[\frac{q(y)}{q(x)}, 1\right]$$

Key issues: shape and scale (in all directions) of $K(z)$

RWM in 2-D



MacKay (2003)

Small step size → good acceptance rate, but slow exploration

Independent Metropolis (IM)

Propose a new point independently of the current location:

$$k(y|x) = K(y)$$

The acceptance probability is now

$$\alpha(y|x) = \min\left[\frac{q(y)K(x)}{q(x)K(y)}, 1\right]$$

Note if $K(\cdot) \propto q(\cdot)$, proposals are from the target and are always accepted

Good acceptance requires $K(\cdot)$ to resemble the posterior $\rightarrow$ IM is typically only useful in low-D problems where we can construct a good $K(\cdot)$ (e.g., MVN at the mode)

Blocking and Gibbs sampling

Basic idea: Propose moves of only subsets of the parameters at a time in an effort to improve rate of exploration

Suppose $x = (x_1, x_2)$ is 2-D. If we alternate *two* 1-D M-H samplers,

- Targeting $p_{\mathrm{eq}}(x_1) \propto p(x_1|x_2)$
- Targeting $p_{\mathrm{eq}}(x_2) \propto p(x_2|x_1)$

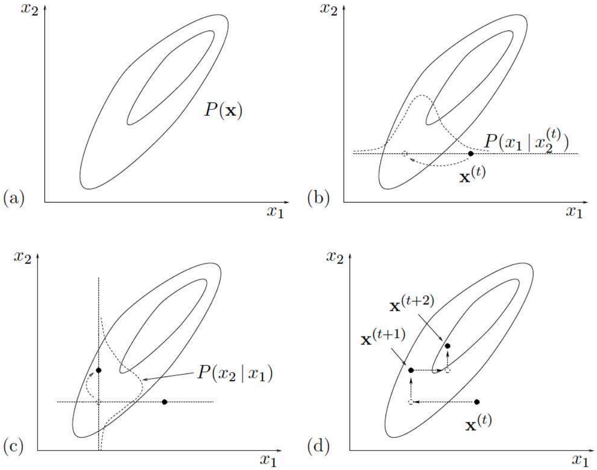
then the resulting chain produces samples from $p(x)$

The simplest case is *Gibbs sampling*, which alternates proposals drawn directly from *full conditionals*:

$$p(x_1|x_2) = \frac{p(x_1, x_2)}{p(x_2)} \propto p(x_1, x_2) \text{ with } x_2 \text{ fixed}$$
$$p(x_2|x_1) = \frac{p(x_1, x_2)}{p(x_1)} \propto p(x_1, x_2) \text{ with } x_1 \text{ fixed}$$

M-H with these proposals *always accepts*

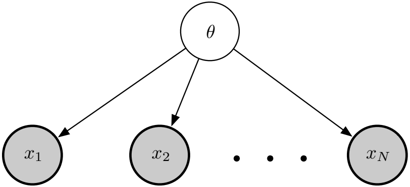
Gibbs sampling in 2-D



(a) The joint density $P(\mathbf{x})$ from which samples are required. (b) Starting from a state $\mathbf{x}^{(t)}$, x_1 is sampled from the conditional density $P(x_1 \,|\, x_2^{(t)})$. (c) A sample is then made from the conditional density $P(x_2 \,|\, x_1)$. (d) A couple of iterations of Gibbs sampling.

MacKay (2003)

Metropolis-within-Gibbs



For MLMs with *conditional independence* structure:

- Full conditionals for latent parameters are often low-D and straightforward to sample from
- Full conditionals for upper-level parameters (hyperparameters) are *not* easy to sample from

$\Rightarrow$ block-update upper- and lower-level parameters:

- Use Gibbs sampling for latent parameters (always accepts)
- Use another M-H algorithm for upper-level parameters

Posterior Sampling & MCMC

1. Posterior sampling

2. Markov chain Monte Carlo
   - Markov chain properties
   - Metropolis-Hastings algorithm
   - Classes of proposals

3. **MCMC diagnostics**
   - Posterior sample diagnostics
   - Joint distribution diagnostics
   - Cautionary advice

4. Beyond the basics

The Good News

The Metropolis-Hastings algorithm enables us to draw a few time series realizations $\{\theta_t\}$, $t = 0$ to N, from a Markov chain with a specified stationary distribution $p(\theta)$

The algorithm works for any $f(\theta) \propto p(\theta)$, i.e., Z *needn't be known*

The marginal distribution at each time is $p_t(\theta)$

- *Stationarity*: If $p_0(\theta) = p(\theta)$, then $p_t(\theta) = p(\theta)$
- *Convergence*: If $p_0(\theta) \neq p(\theta)$, eventually

  $$||p_t(\theta), p(\theta)|| < \epsilon$$

  for an appropriate norm between distributions
- *Ergodicity*:

  $$\bar{g} \equiv \frac{1}{N}\sum_i g(\theta_i) \to \langle g \rangle \equiv \int d\theta\; g(\theta)p(\theta)$$

The Bad News

- We never have $p_0(\theta) = p(\theta)$: we have to figure out how to initialize a realization, and we are always in the situation where $p_t(\theta) \neq p(\theta)$

- "Eventually" means $t < \infty$; that's not very comforting!

- After convergence at time $t = c$, $p_t(\theta) \approx p(\theta)$, but θ values at different times are dependent; the Markov chain CLT says

$$\bar{g} \sim N(\langle g \rangle, \sigma^2/N)$$

$$\sigma^2 = \mathrm{var}[g(\theta_c)] + 2\sum_{k=1}^{\infty} \mathrm{cov}[g(\theta_c), g(\theta_{c+k})]$$

- We have to learn about $p_t(\theta)$ from just a few time series realizations (maybe just one)

Posterior sample diagnostics

Posterior sample diagnostics use single or multiple chains, $\{\theta_t\}$, to diagnose:

- **Convergence:** How long until starting values are forgotten? (Discard as "burn-in," or run long enough so averages "forget" initialization bias.)
- **Mixing:** How long until we have fairly sampled the full posterior? (Make finite-sample Monte Carlo uncertainties small.)

Two excellent R packages with routines, descriptions, references:

- **boa**
  `http://cran.r-project.org/web/packages/boa/index.html`
- **coda**
  `http://cran.r-project.org/web/packages/coda/index.html`

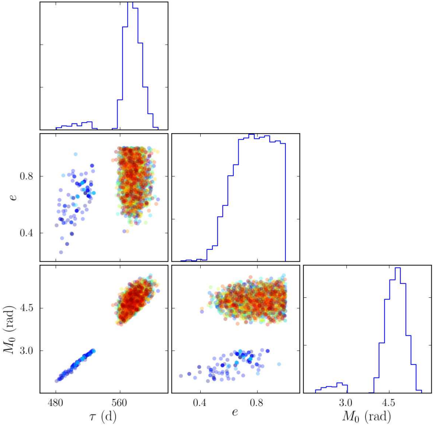
They also supply other *output analyses*: estimating means, variances, marginals, HPD regions. . .

Diagnosing convergence

Qualitative

- Trace plots—trends?
- Diagnostic plots; e.g., running mean
- Color-coded pair plots

Exoplanet parameter estimation using RV data from HD 222582 and Ter Braak's differential evolution MCMC



Quantitative

- Gelman-Rubin-Brooks potential scale-reduction statistic $\sqrt{R}$: multiple chains, compare within- and between-chain variance
- Geweke: single chain, consistency of early/late means
- Heidelberger & Welch: single chain, checks for Brownian motion signature of stationarity, estimates burn-in
- Fan-Brooks-Gelman score statistic:

$$U_k(\theta) = \frac{\partial \log p(\theta)}{\partial \theta_k}$$

Uses $\langle U_k \rangle_p = 0$ (but requires derivatives)

Use diagnostics for all quantities of interest!
Check all parameters, and functions of them

Diagnosing mixing

Qualitative

- Trace plots—does chain get stuck, have slow trends?
- Diagnostic plots; e.g., running mean, autocorrelation function

Quantitative

- Batch means (Murali's CASt summer school lab)
- AR and spectral analysis estimators (SCMA 2011 tutorials)

Bayesian Inference and the Joint Distribution

Recall that Bayes's theorem comes from the *joint distribution for data and hypotheses* (parameters/models):

$$\begin{aligned} p(\theta, D|M) &= p(\theta|M)\, p(D|\theta, M) \\ &= p(D|M)\, p(\theta|D, M) \end{aligned}$$

Bayesian inference takes $D = D_{\text{obs}}$ and solves RHS for the posterior:

$$\rightarrow p(\theta|D_{\text{obs}}, M) = \frac{p(\theta|M) p(D_{\text{obs}}|\theta, M)}{p(D_{\text{obs}}|M)}$$

MCMC is nontrivial technology for building RNGs to sample θ values from the *intractable posterior*, $p(\theta|D_{\text{obs}}, M)$.

Posterior sampling is hard, but sampling from the other distributions is often easy:

- Often easy to draw θ^* from $\pi(\theta)$
- Typically easy to draw $D_{\rm sim}$ from $p(D|\theta, M)$
- Thus we can sample the joint for (θ, D) by sequencing:

$$
\begin{aligned}
&\theta^* \sim \pi(\theta) \\
&D_{\rm sim} \sim p(D|\theta^*, M)
\end{aligned}
$$

- $\{D_{\rm sim}\}$ from above are samples from prior predictive,

$$p(D|M) = \int d\theta\, \pi(\theta) p(D|\theta, M)$$

Now note that $\{D_{\rm sim}, \theta\}$ with $\theta \sim p(\theta|D_{\rm sim}, M)$ are also samples from the joint distribution

Joint distribution methods check the consistency of these two joint samplers to validate a posterior sampler

Example: “Calibration” of credible regions

How often may we expect an HPD region with probability P to include the true value if we analyze many datasets? I.e., what’s the frequentist coverage of an interval rule $\Delta(D)$ defined by calculating the Bayesian HPD region each time?

Suppose we generate datasets by picking a parameter value from $\pi(\theta)$ and simulating data from $p(D|\theta)$.

The fraction of time θ will be in the HPD region is:

$$Q = \int d\theta \, \pi(\theta) \int dD \; p(D|\theta) \; [\![\theta \in \Delta(D)]\!]$$

Note $\pi(\theta)p(D|\theta) = p(\theta, D) = p(D)p(\theta|D)$, so

$$Q \;\; = \;\; \int dD \int d\theta \; p(\theta|D) \, p(D) \; [\![\theta \in \Delta(D)]\!]$$

$$
\begin{aligned}
Q &= \int dD \int d\theta \; p(\theta|D)\, p(D) \; [\![\theta \in \Delta(D)]\!] \\
&= \int dD \; p(D) \int d\theta \; p(\theta|D) \; [\![\theta \in \Delta(D)]\!] \\
&= \int dD \; p(D) \; \int_{\Delta(D)} d\theta \; p(\theta|D) \\
&= \int dD \; p(D) P \\
&= P
\end{aligned}
$$

The HPD region includes the true parameters $100P\%$ of the time

This is exactly true for any problem, even for small datasets

Keep in mind it involves drawing θ from the prior; credible regions are "calibrated with respect to the prior"

A Tangent: Average Coverage

Recall the original Q integral:

$$\begin{aligned} Q &= \int d\theta\, \pi(\theta) \int dD\; p(D|\theta)\; [\![\theta \in \Delta(D)]\!] \\ &= \int d\theta\, \pi(\theta) C(\theta) \end{aligned}$$

where $C(\theta)$ is the (frequentist) coverage of the HPD region when the data are generated using θ

This indicates Bayesian regions have accurate *average coverage*

The prior can be interpreted as quantifying how much we care about coverage in different parts of the parameter space

Basic Bayesian Calibration Diagnostics

Encapsulate your sampler: Create an MCMC posterior sampling algorithm for model M that takes data D as input and produces posterior samples $\{\theta_i\}$, and a 100 P% credible region $\Delta_P(D)$

Initialize counter $Q = 0$
Repeat $N \gg 1$ times:

1. Sample a "true" parameter value θ^* from $\pi(\theta)$
2. Sample a dataset $D_{\rm sim}$ from $p(D|\theta^*)$
3. Use the encapsulated posterior sampler to get $\Delta_P(D_{\rm sim})$ from $p(\theta|D_{\rm sim}, M)$
4. If $\theta^* \in \Delta_P(D)$, increment Q

Check that $Q/N \approx P$

Easily extend the idea to check *all* credible region sizes:

Initialize a list that will store N probabilities, P
Repeat $N \gg 1$ times:

1. Sample a "true" parameter value θ^* from $\pi(\theta)$
2. Sample a dataset D_{sim} from $p(D|\theta^*)$
3. Use the encapsulated posterior sampler to get $\{\theta_i\}$ from $p(\theta|D_{\text{sim}}, M)$
4. Find P so that θ^* is on the boundary of $\Delta_P(D)$; append to list $[P = \text{fraction of } \{\theta_i\} \text{ with } q(\theta_i) > q(\theta^*)]$

Check that the Ps follow a uniform distribution on $[0, 1]$

Other Joint Distribution Tests

- Geweke 2004: Calculate means of scalar functions of (θ, D) two ways; compare with z statistics
- Cook, Gelman, Rubin 2006: Posterior quantile test, expect $p[g(\theta) > g(\theta^*)] \sim$ Uniform (HPD test is special case)

What Joint Distribution Tests Accomplish

Suppose the prior and sampling distribution samplers are well-validated

- **Convergence verification:** If your sampler is bug-free but was not run long enough → unlikely that inferences will be calibrated
- **Bug detection:** An incorrect posterior sampler implementation will not converge to the correct posterior distribution → unlikely that inferences will be calibrated, even if the chain converges

Cost: Prior and data sampling is often cheap, but posterior sampling is often expensive, and joint distribution tests require you run your MCMC code *hundreds* of times

Compromise: If MCMC cost grows with dataset size, running the test with smaller datasets provides a good bug test, and *some* insight on convergence

Experts Speak

All the methods can fail to detect the sorts of convergence failure they were designed to identify. We recommend a combination of strategies. . . it is not possible to say with certainty that a finite sample from an MCMC algorithm is representative of an underlying stationary distribution.

— Cowles & Carlin review of 13 diagnostics

[A]ll methods based solely upon sampler output can be fooled. . . and multiple-chain-based diagnostics, while safer than single-chain-based diagnostics, can still be highly dependent upon the starting points of the simulations. . . . in practice, it may be useful to combine a number of the alternative approaches. . . .

— Brooks & Gelman 1998

In more than, say, a dozen dimensions, it is difficult to believe that a few, even well-chosen, scalar statistics give an adequate picture of convergence of the multivariate distribution.

— Peter Green 2002

Handbook of Markov Chain Monte Carlo (2011)

Your humble author has a dictum that *the least one can do is to make an overnight run*. What better way for your computer to spend its time? In many problems that are not too complicated, this is millions or billions of iterations. *If you do not make runs like that, you are simply not serious about MCMC.* Your humble author has another dictum (only slightly facetious) that one should start a run when the paper is submitted and keep running until the referees' reports arrive. This cannot delay the paper, and may detect pseudo-convergence.

— Charles Geyer

When all is done, compare inferences to those from simpler models or approximations. Examine discrepancies to see whether they represent programming errors, poor convergence, or actual changes in inferences as the model is expanded.

— Gelman & Shirley

Posterior Sampling & MCMC

❶ **Posterior sampling**

❷ **Markov chain Monte Carlo**

- Markov chain properties
- Metropolis-Hastings algorithm
- Classes of proposals

❸ **MCMC diagnostics**

- Posterior sample diagnostics
- Joint distribution diagnostics
- Cautionary advice

❹ **Beyond the basics**

Much More to Computational Bayes

Fancier MCMC

- Sophisticated proposals (e.g., with auxiliary variables)
- Adaptive proposals (use many past states)
- Population-based MCMC (e.g., differential evolution MCMC)

Sequential Monte Carlo

- Particle filters for dynamical models (posterior tracks a changing state)
- Adaptive importance sampling ("evolve" posterior via annealing or on-line processing)

Model uncertainty

- Marginal likelihood computation: Thermodynamic integration, bridge sampling, nested sampling
- MCMC in *model* space: Reversible jump MCMC, birth/death
- Exploration of large (discrete) model spaces (e.g., variable selection): Shotgun stochastic search, Bayesian adaptive sampling

This is just a small sampling!