Markov Chain Monte Carlo (MCMC)

Charles J. Geyer
School of Statistics
University of Minnesota

Stat 8054 Lecture Notes

Markov Chains

X_1, X_2, ... is *Markov chain* if conditional distribution of X_{n+1} given X_1, ..., X_n depends on X_n only.

Chain has *stationary transition probabilities* if conditional distribution of X_{n+1} given X_n does not depend on n. Only kind of interest in MCMC. Always tacitly assumed.

Joint distribution of chain determined by

- marginal distribution of X_1, the *initial distribution* and
- conditional distribution of X_{n+1} given X_n, the *transition probability*.

Why MCMC?

Suppose you have a computer program

```
Initialize state $x$
repeat {
    Generate pseudorandom change to state $x$
    Output state $x$
}
```

If the state x is the entire state of the computer program exclusive of random number generator seeds (which we ignore, pretending pseudorandom is random), this is MCMC.

x must be entire state. Otherwise need not be Markov.

Stationarity

An initial distribution is *stationary* or *invariant* for a transition probability if the Markov chain they specify has the same marginal distribution at all times.

We also indicate this by saying the transition probability *preserves* the initial distribution.

Note: different from stationary transition probabilities. Every chain we consider has that, but not all are stationary.

Consider initial distributions concentrated at one point.

Reversibility

A transition probability is *reversible* with respect to an initial distribution if the Markov chain X_1, X_2, ..., they specify has the distribution of pairs (X_n, X_{n+1}) exchangeable.

Reversibility implies stationarity.

Theory of GOFMC

Recall ”swindles” slides. You want to calculate

$$\mu = E\{g(X)\}$$

but you can’t exactly.

Use

$$\widehat{\mu}_n = \frac{1}{n} \sum_{i=1}^{n} g(X_i)$$

as your *Monte Carlo approximation*, where X_1, X_2, ... are IID simulations having the same distribution as X.

Theory of GOFMC (cont.)

The central limit theorem says

$$\widehat{\mu}_n \approx \text{Normal}(\mu, \sigma^2/n)$$

where

$$\sigma^2 = \operatorname{var} g(X)$$

can be estimated by

$$\widehat{\sigma}_n^2 = \frac{1}{n} \sum_{i=1}^{n} \bigl(g(X_i) - \mu \bigr)^2$$

Theory of MCMC

You want to calculate

$$\mu = E\{g(X)\}$$

but you can't exactly.

Use

$$\widehat{\mu}_n = \frac{1}{n} \sum_{i=1}^{n} g(X_i)$$

as your *Monte Carlo approximation*, where X_1, X_2, ... is a Markov chain whose stationary distribution is the same as the distribution of X.

Theory of MCMC (cont.)

The Markov chain central limit theorem says

$$\hat{\mu}_n \approx \text{Normal}(\mu, \sigma^2/n)$$

where

$$\sigma^2 = \text{var}\, g(X_i) + 2 \sum_{k=1}^{\infty} \text{cov}\Big(g(X_i), g(X_{i+k})\Big), \qquad (1)$$

X_1, X_2, ... in (1) being stationary, can be estimated by various methods.

AR(1) Example

$$X_{n+1} = \rho X_n + Y_n$$

where Y_n are IID Normal$(0, \tau^2)$ (Rweb example).

$$\text{cov}(X_{n+k}, X_n) = \rho \, \text{cov}(X_{n+k-1}, X_n) = \rho^k \gamma_0$$

For stationary chain

$$\text{var}(X_{n+1)} = \rho^2 \, \text{var}(X_n) + \text{var}(Y_n)$$

or

$$\gamma_0 = \rho^2 \gamma_0 + \tau^2$$

so

$$\gamma_0 = \frac{\tau^2}{1 - \rho^2}$$

AR(1) Example (cont.)

$$
\begin{aligned}
\gamma_0 &= \tau^2/(1-\rho^2) \\
\gamma_k &= \rho^k \gamma_0
\end{aligned}
$$

so

$$
\begin{aligned}
\sigma^2 &= \gamma_0 + 2 \sum_{k=1}^{\infty} \rho^k \gamma_0 \\
&= \gamma_0 \left(1 + 2 \frac{\rho}{1-\rho} \right) \\
&= \gamma_0 \frac{1+\rho}{1-\rho}
\end{aligned}
$$

Method of Batch Means

Divide chain into m batches of length b. Average the batches

$$\widehat{\mu}_{b,k} = \frac{1}{b} \sum_{i=bk+1}^{bk+b} g(X_i)$$

Then

$$\frac{1}{m} \sum_{k=0}^{m-1} (\widehat{\mu}_{b,k} - \widehat{\mu}_n)^2$$

estimates σ^2 / b (Rweb example).

Method of Overlapping Batch Means

Divide chain into $n - b + 1$ overlapping batches of length b. Average the batches

$$\hat{\mu}_{b,k} = \frac{1}{b} \sum_{i=k+1}^{k+b} g(X_i)$$

Then

$$\frac{1}{n - b + 1} \sum_{k=0}^{n-b} (\hat{\mu}_{b,k} - \hat{\mu}_n)^2$$

estimates σ^2/b (Rweb example).

Time Series Methods

Define

$$\gamma_k = \text{cov}\Big(g(X_i), g(X_{i+k})\Big)$$

where X_1, X_2, ... are stationary, so

$$\sigma^2 = \gamma_0 + 2 \sum_{k=1}^{\infty} \gamma_k$$

Estimate γ_k

$$\widehat{\gamma}_k = \frac{1}{n} \sum_{i=1}^{n-k} \Big(g(X_i) - \widehat{\mu}_n\Big)\Big(g(X_{i+k}) - \widehat{\mu}_n\Big)$$

Plug in ????

Naive Plug-In Doesn't Work

For large k

$$\operatorname{var} \widehat{\gamma}_k \approx \frac{1}{n} \left(\gamma_0^2 + 2 \sum_{j=1}^{\infty} \gamma_j^2 \right)$$

does not go to zero as $k \to \infty$ (Rweb example).

Infinite sum of random noise is a bad idea.

Time Series Methods (cont.)

Theorem: For reversible chain

$$\Gamma_k = \gamma_{2k} + \gamma_{2k+1}$$

is strictly positive, strictly decreasing, strictly convex function of k.

Can use to estimate

$$\sigma^2 = -\gamma_0 + 2 \sum_{k=0}^{\infty} \Gamma_k$$

(Rweb examples).

Which Variance Estimate?

Any of them!

Most users of MCMC cannot be bothered to figure out the accuracy of their MCMC estimates.

If they don't care about their numbers, why should you?

Don't follow their example.

Creating Markov Chains

You have a distribution that you want to study by MCMC.

How do you set up a Markov chain having that as its stationary distribution?

Basically, only one idea, Metropolis-Hastings-Green algorithm.

Metropolis Update

Desired stationary distribution has unnormalized density h.

At x propose move to y with density $q(x, \cdot\,)$ which is symmetric $q(x,y) = q(y,x)$.

Accept proposed move with probability

$$a(x,y) = \text{min}\Big(1, r(x,y)\Big)$$

where

$$r(x,y) = \frac{h(y)}{h(x)} \tag{2}$$

Otherwise reject proposed move, and chain *stays at the same position* $(X_{n+1} = X_n)$. Not like rejection sampling !!!!!

Metropolis-Hastings Update

Same as Metropolis except proposal density does not need to be symmetric and

$$r(x, y) = \frac{h(y)q(y, x)}{h(x)q(x, y)} \tag{3}$$

(clearly Metropolis is special case).

Metropolis-Hastings Theorem

Want to prove reversibility with respect to h.

If X_n is current state and Y_n is proposal, have $X_n = X_{n+1}$ whenever proposal is rejected. Clearly, the distribution of (X_n, X_{n+1}) given rejection is exchangeable.

So only need to work on part given acceptance. Need to show

$$E\{f(X_n, Y_n)a(X_n, Y_n)\} = E\{f(Y_n, X_n)a(X_n, Y_n)\}$$

for any function f that has expectation (assuming X_n has desired stationary distribution).

Metropolis-Hastings Theorem (cont.)

That is, must show can interchange arguments of f in

$$\iint f(x, y) h(x) a(x, y) q(x, y) \, dx \, dy$$

and that follows if can interchange x and y in

$$h(x) a(x, y) q(x, y)$$

Say $r(x, y) \leq 1$, hence $r(x, y) = a(x, y)$ and $a(y, x) = 1$. Then

$$\begin{aligned} h(x) a(x, y) q(x, y) &= h(x) r(x, y) q(x, y) \\ &= h(y) q(y, x) \\ &= h(y) q(y, x) a(y, x) \end{aligned}$$

QED (quite easily derived).

Gibbs Update

Proposal is from a conditional distribution of the desired stationary distribution.

Now proof is trivial: marginal times conditional equals joint.

Suppose X_n has the desired stationary distribution.

Suppose conditional distribution X_{n+1} given $f(X_n)$ is same as the conditional distribution of X_n given $f(X_n)$.

Then the pair (X_n, X_{n+1}) is conditionally exchangeable given $f(X_n)$. Hence unconditionally exchangeable.

Mixing and Matching

Composition

Let P_1, ..., P_k be update mechanisms (computer code) and let $P_1 P_2 \cdots P_k$ denote the composite update that consists of these updates done in order.

If each P_i preserves a distribution, then so does $P_1 P_2 \cdots P_k$.

Palindromic Composition

Note $P_1 P_2 \cdots P_k$ not reversible unless equal in distribution to $P_k P_{k-1} \cdots P_1$.

Then we call it *palindromic*.

Mixing and Matching (cont.)

State-Independent Mixing

Let P_y be update mechanisms (computer code) and let $E(P_Y)$ denote the update that consists of doing a random one of these updates: generate Y from some distribution and do P_Y.

Clearly, if Y is independent of the current state and each P_y preserves the same distribution, then so does $E(P_Y)$. (If X_n and X_{n+1} both have the distribution π conditional on Y_n, then both have the distribution π unconditionally.)

“Mixture” is used here in the sense of mixture models.

Mixing and Matching (cont.)

Subsampling

P^k is the update that consists of the update P repeated k times.

If

$$X_1, X_2, X_3, \ldots$$

is a Markov chain with update P, then

$$X_k, X_{2k}, X_{3k}, \ldots$$

is a Markov chain with update P^k.

Mixing and Matching (cont.)

Random Subsampling

Define P^0 is to be the identity update that does nothing.

Let K be a nonnegative integer random variable and consider $E(P^K)$.

If K_1, K_2, ... are IID random variables with the same distribution as K and

$$X_0, X_1, X_2, X_3, \ldots$$

is a Markov chain with update P, then

$$X_{K_1}, X_{K_1+K_2}, X_{K_1+K_2+K_3}, \ldots$$

is a Markov chain with update $E(P^K)$.

Mixing and Matching (cont.)

One Component at a Time

The traditional way to do Gibbs updates is to sample from the conditional distribution of one component of the state given the rest. This gives k distinct updates if there are k components.

Combine by composition, mixing, or both.

Despite popularity, one-component-at-a-time has no computational virtues.

Gibbs Sampler

An MCMC scheme consisting entirely of Gibbs updates combined by composition, mixing, or both is called a "Gibbs sampler."

Despite popularity, Gibbs sampler has no computational virtues.

Peter Clifford discussing the afternoon of the 11 Bayesians said

> Currently [1993], there are many statisticians trying to reverse out of this historical *cul-de-sac*.
>
> To use the Gibbs sampler, we have to be good at manipulating conditional distributions . . . this rather brings back the mystique of the statisticians.

Metropolis-Hastings Algorithm

An MCMC scheme consisting entirely of Metropolis-Hastings updates combined by composition, mixing, or both is called an instance of the "Metropolis-Hastings algorithm."

Acceptance Rate

Generally, one can make the acceptance rate as high as one pleases (propose little baby steps) or as low (propose big giant steps). Neither is a good idea. It's a Goldilocks problem.

Two different groups studying two different toy problems concluded that 20% acceptance is about right. In non-toy problems your mileage may vary.

One Long Run

If one long run of the Markov chain doesn't "work" — adequately represent the stationary distribution — then many short runs certainly won't work; it's merely IID sampling from the initial distribution slightly fuzzed.

The subject of several rants.

Not to say you can't make as many runs as you please. But all actual inference should be from one long run.

Burn In

Idea of throwing away an initial segment of the Markov chain — part before it “reaches equilibrium.”

Just a different kind of initial distribution. No magic.

Neither necessary nor sufficient for good MCMC.

Also subject of a rant.

MCMC package for R

The R contributed package `mcmc` (on-line help) has just two functions

- `metrop` (on-line help) and

- `olbm` (on-line help).

It also has a (package vignette) that gives a complete discussion of one problem, which was on a qualifying exam.

Design of MCMC Package

The `mcmc` package also has a (design document) that gives the rationale for the `metrop` being the way it is.

Design Criteria

User supplies R function that evaluates log unnormalized density (LUD), simulate Markov chain having that LUD as equilibrium distribution.

Nothing user can screw up except that function — wrong LUD function, wrong equilibrium distribution.

Output averages arbitrary function of state $f(X_n)$ where f is R function supplied by user.

Testing MCMC

Monte Carlo algorithms are almost impossible to test. MCMC even worse.

Output is random. Often nothing is known about equilibrium distribution except what is learned from MCMC sampler. If sampler is buggy, know nothing!

Hats

You've got three hats: *statistics*, *MCMC*, *computer*.

When you are wearing your *statistics* hat, you think statistical issues, you think about the statistical meaning of the state variables.

When you are wearing your *MCMC* hat, you ignore statistical meaning. Given a meaningless problem, think how to construct an effective MCMC sampler for it and about MCMC error.

When you are wearing your *computer* hat, you ignore the "random" in "pseudorandom". Does this code correctly implement an instance of the Metropolis-Hastings-Green (MHG) algorithm?

Extended State

Let X_n denote state at time n, Y_n proposal at time n, and U_n a Uniform$(0, 1)$ random variable independent of X_n and Y_n.

MHG algorithm sets

$$X_{n+1} = \begin{cases} Y_n & U_n < r(X_n, Y_n) \\ X_n & \text{otherwise} \end{cases}$$

where $r(x, y)$ is the MHG ratio (2) or (3).

Let A_n denote the indicator (zero-or-one) of event $U_n < r(X_n, Y_n)$.

Extended state is (X_n, Y_n, U_n, A_n). Extended chain still Markov. Y_{n+1}, U_{n+1}, and A_{n+1} are conditionally independent of past history conditional on X_{n+1}, which is function of (X_n, Y_n, U_n).

Using Extended State

Acceptance rate is

$$\widehat{\alpha}_n = \frac{1}{n} \sum_{i=1}^{n} A_i$$

is just another MCMC estimate (of the true unknown acceptance rate). Use batch means or other MCMC variance estimator to get accuracy.

Can use extended state to investigate many properties of sampler that cannot be studied from ordinary state.

Testing MCMC (cont.)

Given particular pseudorandom number generators that sample the U_n and the conditional distribution of Y_n given X_n, the algorithm is entirely deterministic.

Testing need not deal with randomness.

Testing MCMC (cont.)

Suppose have (hopefully fast) MCMC scheme written in C that uses the R random number generators (either C dynamically loaded into R or uses R standalone library).

Also write (slow) implementation in R, ideally in very simple transparently correct R that does bit-for-bit identical computations to the C version.

Then can believe that C is correct (to the extent that the R is *transparently* correct)!

Testing MCMC (cont.)

Alternative testing methodology. Two implementers, without consultation (!) implement two implementations.

If answers agree to within Monte Carlo standard errors (MCSE) when sample sizes are so large that MCSE are negligible, then perhaps both are correct.

Doesn't work for two implementations by same person (possible common failure mode: wrong ideas).

Debugging MCMC

Without extended state, debugging MCMC is hopeless.

With extended state, all details of the algorithm can be checked.

- Does proposal have correct conditional distribution?
- Is LUD calculated correctly?
- Is MHG ratio calculated correctly?
- Are decisions correctly made in Metropolis rejection?

Tempering: Parallel, Serial, and Umbrella

MCMC problem is “hard” when obvious samplers don’t work.

Need better sampler.

One idea: embed problem in sequence of problems. Solve all simultaneously, using easier problems to help with harder ones.

Write h_1, . . . , h_m for the unnormalized densities of the sequence.

May have $h_i(x) = h(x)^{\beta_i}$ like simulated annealing. Not necessary.

Parallel Tempering

If S is state space of problem of interest, domain of each h_i, then S^m is state space of parallel tempering (PT) chain.

Unnormalized density of stationary distribution of PT chain is product of h_i, so components x_1, ..., x_m of PT state are asymptotically independent and and x_i has stationary distribution h_i.

Note: subscripts here are for sequence of problems *not time*. Holds until further notice.

Parallel Tempering Updates

Single Component

Update x_i preserving h_i.

Swap

Choose a random pair (i, j) of indices from some symmetric mechanism — equally likely to choose (i, j) and (j, i).

Propose swap $x_i \longleftrightarrow x_j$.

MHG ratio

$$r(i, j) = \frac{h_i(x_j)h_j(x_i)}{h_i(x_i)h_j(x_j)}$$

Parallel Tempering (cont.)

Swaps connect state $(x_1, \ldots, x_m)$ of PT chain. One Markov chain with state space S^m.

Mixing properties of PT chain intermediate between those of chains for separate problems.

If distribution of interest is slowly mixing, then PT improves it.

Easy to do. Popular. Works well.

Parallel Tempering (cont.)

Inference is easy. If we only consider the output for one distribution of the m distributions being sampled. Then we can consider it a representative sample from that distribution.

We are looking at functions $f(x_1, \ldots, x_m)$ whose expectation we want to calculate. If we are only interested in the j-th distribution then we only look at functions $g(x_j)$ of that component only.

All of our theory about variance estimation applies.

Problems with Parallel Tempering

Doesn’t estimate normalizing constants for separate distributions h_i. Doesn’t need to. Just like MHG for one distribution.

Remembers too much state, for m distributions rather than one.

Neither is fatal, but limit applicability.

Serial Tempering

Not the established name. First parallel tempering was invented but not called that. Then *simulated tempering* was invented. Then parallel tempering was named by analogy with simulated tempering.

But anybody that knows anything about electronic circuits knows that the opposite of parallel is *serial*. Hence the name used here. Nice that ST can stand for either serial or simulated.

Serial Tempering (cont.)

If S is state space of problem of interest, domain of each h_i, then $S \times \{1, \ldots, m\}$ is state space of ST chain.

Unnormalized density of stationary distribution of ST chain is

$$h(x, i) = h_i(x) c_i$$

where the c_i are arbitrary constants chosen by user (more on this later).

Asymptotic distribution of ST state (X, I) — both bits random — is such that conditional distribution of X given $I = i$ is distribution with unnormalized density h_i.

Serial Tempering Updates

Single Component (Update X)

Update x_i preserving h_i.

Swap (Update I)

Choose a random index j for new value of I from some symmetric mechanism — equally likely to choose j when $I = i$ and choose i when $I = j$.

MHG ratio

$$r(i, j) = \frac{h_j(x) c_j}{h_i(x) c_i}$$

Serial Tempering (cont.)

Inference is a bit more complicated than for PT.

The ST chain can be written (X_t, I_t), $t = 1$, 2, ...

If we are only interested in a particular distribution h_k, then we want to only look at X_t values when $I_t = k$.

Hence we look at

$$\widehat{\nu}_n = \frac{1}{n} \sum_{t=1}^{n} f(X_t) 1(I_t = k)$$

Serial Tempering (cont.)

Each h_i has its own normalizing constant

$$d_i = \int h_i(x) \, dx$$

and the unnormalized density

$$h(x, i) = h_i(x) c_i$$

has normalizing constant

$$d = \sum_i \int h_i(x) c_i \, dx = \sum_i d_i c_i$$

So h / d and h_i / d_i are normalized densities.

Serial Tempering (cont.)

By the law of large numbers, $\widehat{\nu}_n$ converges to its expectation under the stationary distribution.

$$\begin{aligned}
\nu &= \sum_i \int f(x) 1(i = k) \frac{h_i(x) c_i}{d} \, dx \\
&= \frac{c_k}{d} \int f(x) h_k(x) \, dx \\
&= \frac{c_k d_k}{d} E_k\{f(X)\}
\end{aligned}$$

Serial Tempering (cont.)

Summary:

$$\widehat{\nu}_n = \frac{1}{n} \sum_{t=1}^{n} f(X_t) 1(I_t = k) \to \frac{c_k d_k}{d} E_k\{f(X)\}$$

and similarly

$$\widehat{\delta}_n = \frac{1}{n} \sum_{t=1}^{n} 1(I_t = k) \to \frac{c_k d_k}{d}$$

and

$$\widehat{\nu}_n / \widehat{\delta}_n \to E_k\{f(X)\}$$

Serial Tempering (cont.)

End of “inference is a bit more complicated than for PT.”

To estimate expectations with respect to one of the distributions having density h_k, we use a ratio estimator $\widehat{\nu}_n / \widehat{\delta}_n$.

Hence we have to use the delta method.

Tuning Serial Tempering

From

$$\frac{1}{n} \sum_{t=1}^{n} 1(I_t = k) \to \frac{c_k d_k}{d}$$

it is clear that ST doesn't work unless $c_i d_i$ is about the same size for all i. This must be accomplished by trial and error.

Increase c_i for i that don't appear very often.

Repeat adjustment until all i appear roughly equally often.

Tuning Serial Tempering (cont.)

c_i don't have to be perfectly adjusted.

So long as all i appear in sample I_t, $t = 1$, 2, ... with reasonable frequency, ST works.

(Works for any c_i eventually, but only in reasonable time when c_i adjusted well.)

Estimating Normalizing Constants

From

$$\frac{1}{n} \sum_{t=1}^{n} 1(I_t = k) \to \frac{c_k d_k}{d}$$

also see that ST estimates relative normalizing constants d_k / d.

This is usually all that is needed.

No need for delta method here.

Importance Sampling

Recall ”swindles” slides. You want to calculate

$$E_\theta\{f(X)\} = E_h \left\{ f(X) \frac{h_\theta(X)}{h(X)} \right\}$$

valid so long as no division by zero and h_θ and h are normalized densities.

GOFMC and MCMC are no different here

$$\frac{1}{n} \sum_{t=1}^{n} f(X_t) \frac{h_\theta(X_t)}{h(X_t)} \to E_\theta\{f(X)\}$$

simultaneously for all θ where X_1, X_2, ... is a Markov chain with stationary distribution having density h.

Fairly useless in MCMC because requires h_θ and h normalized.

Importance Sampling (cont.)

Same trick used in ST allows use of unnormalized densities. If h_θ and h are unnormalized, then

$$\frac{1}{n} \sum_{t=1}^{n} f(X_t) \frac{h_\theta(X_t)}{h(X_t)} \longrightarrow d(\theta) E_\theta\{f(X)\}$$

where $d(\theta)$ is ratio of normalizing constants of h_θ and h, and

$$\frac{\frac{1}{n} \sum_{t=1}^{n} f(X_t) \frac{h_\theta(X_t)}{h(X_t)}}{\frac{1}{n} \sum_{t=1}^{n} \frac{h_\theta(X_t)}{h(X_t)}} \longrightarrow E_\theta\{f(X)\}$$

Importance Sampling (cont.)

Unnormalized importance weights $h_\theta(x) / h(x)$.

Make sum to one

$$w_\theta(x) = \frac{h_\theta(x) / h(x)}{\sum_{t=1}^n h_\theta(X_t) / h(X_t)}$$

get *normalized importance weights*.

Then

$$\sum_{t=1}^n f(X_t) w_\theta(X_t) \to E_\theta\{f(X)\}$$

is the same formula as before in different notation.

Importance Sampling (cont.)

Method of normalized importance nice to use even in GOFMC with normalized densities, because then empirical estimates obey all laws of probability. Other estimate

$$\frac{1}{n} \sum_{t=1}^{n} f(X_t) \frac{h_\theta(X_t)}{h(X_t)}$$

does not obey the complement rule and expectation of a constant is that constant, because importance weights fail to sum to one.

Umbrella Sampling

Sometimes want to sample from mixture of distributions or to interpolate between distributions of ST or PT. Very hard to do with PT. Easy with ST. There are two methods to apply importance sampling to ST.

Umbrella Sampling (Method I)

Use

$$h(x) = \sum_i h(x, i) = \sum_i h_i(x) c_i$$

as unnormalized importance sampling density. Importance sampling formula now

$$\frac{\displaystyle \frac{1}{n} \sum_{t=1}^{n} f(X_t) \frac{h_\theta(X_t)}{\sum_i h_i(X_t) c_i}}{\displaystyle \frac{1}{n} \sum_{t=1}^{n} \frac{h_\theta(X_t)}{\sum_i h_i(X_t) c_i}} \to E_\theta\{f(X)\}$$

Umbrella Sampling (Method II)

$$\frac{\frac{1}{n}\sum_{t=1}^{n} f(X_t)\frac{h_\theta(X_t)}{h_{I_t}(X_t)c_{I_t}}}{\frac{1}{n}\sum_{t=1}^{n} \frac{h_\theta(X_t)}{h_{I_t}(X_t)c_{I_t}}} \to E_\theta\{f(X)\}$$

also works. Numerator converges to

$$\begin{aligned} E\left\{f(X)\frac{h_\theta(X)}{h(X,I)}\right\} &= \sum_i \int f(x)\frac{h_\theta(x)}{h(x,i)} \cdot \frac{h(x,i)}{d}\,dx \\ &= \frac{md(\theta)}{d}E_\theta\{f(X)\} \end{aligned}$$

where m is number of h_i (must be finite) and $d(\theta)$ is the normalizing constant for θ.

Umbrella Sampling (cont.)

No one knows which of Method I or Method II is better.

Choice of Distributions

How does one chose the distributions h_1, ..., h_m?

Acceptance rates can help here too. Geyer and Thompson (1995) recommend that they be chosen so that acceptance rates for "update I" proposals be about 20%. They also caution that this rule of thumb may be wrong and exhibit a toy problem in which any attempt to get acceptance rates below 60% makes the sampler fail to work.

The same Goldilocks idea we saw in choosing scale for Metropolis proposals. We don't want the steps to be too small or too large.

Small and large here refer to distance between distributions, which is very hard to visualize. Acceptance rates seem to be the only natural guide.

Does Tempering Always Work?

No. There is no magic in MCMC. Not PT or ST, not anything else.

Geyer and Thompson (1995) give a real application where ST apparently works and PT does not.

ST always works or diagnoses its own failure to work if the tuning constants c_1, ..., c_m are chosen correctly (by trial and error).

But ST can appear to work when the tuning constants are erroneous and it is in fact not working.

Perfect Sampling

An idea for producing IID samples from the stationary distribution of a Markov chain (Propp and Wilson, 1996).

First we consider toy version, useless in practice.

Consider finite state space Markov chain that we start at a large negative time X_{-T}, X_{-T+1}, X_{-T+2}, $\ldots$.

Also consider the IID sequence U_{-T}, U_{-T+1}, U_{-T+2}, $\ldots$ where U_i contains all of the pseudorandom variates needed to move from X_i to X_{i+1}.

Perfect Sampling (cont.)

Each X_{i+1} is a deterministic function of X_i and U_i, hence given $\omega = (U_{-T}, U_{-T+1}, \ldots)$ and $X_{-T} = x$, the entire future history $X_{-T+1}(\omega, x)$, $X_{-T+2}(\omega, x)$ is determined.

Now consider future histories for one fixed ω and all possible initial states x.

Suppose it happens that $X_0(\omega, x)$ does not depend on x. No matter where you start at time $-T$, for this particular sequence of pseudorandom variates, you are always at the same place at time zero.

Say chain has *coupled* by time zero if this happens.

Perfect Sampling (cont.)

If chain has coupled by time zero, then $X_0(\omega)$ is a realization of the stationary distribution of the chain!

Why? The chain can be made stationary by choosing the initial distribution at time $-T$ to be the stationary distribution. Then the state at time zero also has the stationary distribution. But the state at time zero is $X_0(\omega)$ regardless of the initial distribution.

Toy problem because to verify coupling have to see what happens for chain started at every possible state x at time $-T$.

Perfect Sampling (cont.)

Now suppose state space is also a complete lattice where we denote the partial order $\precsim$ and we denote the top and bottom elements $\top$ and $\perp$, respectively.

The only partial order actually used is the coordinatewise one if the state is a vector $x = (x_1, \ldots, x_d)$ then

$$x \precsim y \text{ if and only if } x_i \le y_i \text{ for all } i$$

(Caution: subscripts are coordinates not time).

This partial order gives rise to a complete lattice if and only if each coordinate has an upper and lower bound that is a possible value. Then the top element $\top$ is the state that has all coordinates at the maximum and the bottom element $\perp$ is the state that has all coordinates at the minimum.

Perfect Sampling (cont.)

And suppose update preserves the partial order

$$X_t(\omega, x) \lesssim X_t(\omega, y) \text{ implies } X_{t+1}(\omega, x) \lesssim X_{t+1}(\omega, y)$$

This is easily accomplished if the partial order is the coordinatewise one, the Gibbs sampler is used, and the one-dimensional conditionals are sampled using the inversion method.

Now to check coupling at time zero only need to check that chains started at top and bottom elements have coupled. Chains started at all other states are sandwiched in between, thus have coupled too.

Perfect Sampling (cont.)

Not easy to find perfect sampling scheme in complicated problem. Only examples in literature are either toy problems or have high degree of symmetry. No general methodology for “perfectizing” an arbitrary problem.

Perfect Sampling Algorithm

In order to choose $-T$ sufficiently large for coupling at time zero we may have to try many different $-T$. Thus we change notation.

Fix backwards infinite sequence

$$\omega = \{\ldots, U_{-2}, U_1, U_0\}$$

Let $X_0(\omega, x, t)$ denote the state at time zero when the chain is started in state x and time t and pseudorandom variates are taken from ω.

Perfect Sampling Algorithm (cont.)

Repeat the following.

- Fix backwards infinite sequence $\omega = \{\ldots, U_{-2}, U_1, U_0\}$.
- Find $-T$ such that $X_0(\omega, \top, -T) = X_0(\omega, \bot, -T)$.
- Output $X_0(\omega, \top, -T)$.

Produces IID samples from stationary distribution.

Perfect Sampling Algorithm (cont.)

Crucial point hidden in notation.

Try time $-T_1$. Start at $x_{-T_1} = \top$. Save seed of random number generator (RNG). Run chain from time $-T_1$ to time zero. Start at $x_{-T_1} = \perp$. Use same RNG seed. Run chain from time $-T_1$ to time zero.

If coupled by time zero, done. Otherwise, try time $-T_2 < -T_1$.

Start at $x_{-T_2} = \top$. Save RNG seed. Run chain from time $-T_2$ to time $-T_1$. Switch RNG seed to one used in first try. Run chain from time $-T_1$ to time zero. Do same starting at $x_{-T_2} = \perp$.

If coupled by time zero, done. Otherwise, try time $-T_3 < -T_2$.

Perfect Sampling Algorithm (cont.)

Every time an update is done from time t to time $t + 1$ the same U_t must be used! Otherwise not doing perfect sampling or anything else having a justification.

Eventually find some starting point sufficiently far back in the past so that the chains started at $\top$ and $\bot$ have coupled by time zero.

Then done. Output the state at time zero.

If never find such a point sufficiently far back in the past before giving up for lack of patience, then method fails.

Perfect Sampling as MCMC Diagnostic

Perfect sampling is the only MCMC diagnostic actually guaranteed to diagnose non-convergence.

It diagnoses non-convergence by failing to produce the requested number of IID samples in the time one has patience to wait!

Otherwise, of no use. Perfectizing an MCMC sampler only slows it down.

If the sampler works, then ordinary MCMC is more efficient than same sampler perfectized.

Perfect sampling only valuable when it doesn't work.

Measure Theory

Despite strenuous efforts to avoid measure theory, we have finally come to the point where a little bit seems necessary. At least I can't figure out how to avoid it.

In measure theory, probability distributions are represented by set functions (functions whose arguments are sets). Events are subsets of the state space, and a *probability measure* P maps events A to real numbers $P(A)$ which are between zero and one.

$P(A)$ is the probability of the event A.

Abstract Integration

If X is a random element of whose distribution is P, then we write

$$E\{g(X)\} = \int g(x) P(dx) \tag{4}$$

You are assumed to know what expectation means. The right-hand side is just another notation for it.

If X is a continuous random variable with probability density function f, then

$$E\{g(X)\} = \int g(x) f(x)\, dx$$

so the right-hand side of (4) is just ordinary integration in this case.

Abstract Integration (cont.)

If $X = (X_1, X_2, X_3)$ is a continuous random vector with probability density function f, then

$$E\{g(X)\} = \iiint g(x_1, x_2, x_3) f(x_1, x_2, x_3) \, dx_1 \, dx_2 \, dx_3$$

so the right-hand side of (4) is a triple integral in this case.

If X is a discrete random element having state space S and probability mass function f, then

$$E\{g(X)\} = \sum_{x \in S} g(x) f(x)$$

so the right-hand side of (4) is a sum in this case.

Abstract Integration (cont.)

So abstract integration provides one unifying notation for the disparate special cases discussed in master's level theory.

But it does more. If $X = (X_1, X_2)$ is a random vector having one continuous component X_1 and one discrete component X_2, then

$$E\{g(X)\} = \sum_{x_2 \in S} \int g(x_1, x_2) f(x_1, x_2) \, dx_1$$

when f is defined appropriately, although we don't have a master's level name for it (probability mass-density function?)

If X is a random object such that $E\{g(X)\}$ makes sense, then we use the right-hand side of (4) as another notation for it.

Signed Measures

If P and Q are probability measures and a and b are real numbers, then

$$m(A) = aP(A) + bQ(A)$$

defines a set function that is not necessarily a probability measure.

If S is the state space, then $P(S) = Q(S) = 1$, but $m(S) = a + b$.

m is called a *signed measure*, and

$$\int g(x)m(dx) = a\int g(x)P(dx) + b\int g(x)Q(dx)$$

Positive Measures, Subprobability Measures

A signed measure m is a *positive measure* if

$$m(A) \geq 0, \qquad \text{for all events } A.$$

A signed measure m is a *subprobability measure* if

$$0 \leq m(A) \leq 1, \qquad \text{for all events } A.$$

Conditional Probability Measures, Kernels

A conditional probability measure is just a probability measure that varies in accordance with some condition.

It is traditional in Markov chain theory to write $P(x, A)$ to mean for each fixed x, the function $A \mapsto P(x, A)$ is a probability measure.

The relation to conditional expectation is

$$E\{g(X_{n+1}) \mid X_n = x\} = \int g(y) P(x, dy)$$

The general notion is called a *kernel*. Write $K(x, A)$ to mean for each fixed x, the function $A \mapsto K(x, A)$ is a signed measure.

Reversibility and Preservation Revisited

A kernel K is *reversible* with respect to a signed measure m if

$$\iint g(x)h(y)m(dx)K(x,dy) = \iint h(x)g(y)m(dx)K(x,dy)$$

for all bounded functions g and h.

A kernel K is *Markov* if $A \mapsto K(x,A)$ is a probability measure for each fixed x.

A Markov kernel P *preserves* a probability measure π if

$$\iint g(y)\pi(dx)P(x,dy) = \int g(x)\pi(dx)$$

for every bounded function g.

Reversibility with respect to π implies preservation of π.

State-Dependent Mixing

Markov update mechanisms are represented by Markov kernels.

Have family of updates P_i, $i \in I$, choose one at random with probability $c_i(x)$ that depends on the current state x.

Mixture kernel is

$$P(x, A) = \sum_{i \in I} c_i(x) P_i(x, A)$$

Not a theorem that each P_i preserves π implies P preserves π.

State-Dependent Mixing (cont.)

A kernel K is *sub-Markov* if $A \mapsto K(x, A)$ is a subprobability measure for each fixed x.

Recall: choose update kernel P_i at random with probability $c_i(x)$

Define

$$K_i(x, A) = c_i(x) P_i(x, A)$$

If each K_i is reversible with respect to π, then mixture kernel

$$P(x, A) = \sum_{i \in I} c_i(x) P_i(x, A) = \sum_{i \in I} K_i(x, A)$$

is reversible with respect to π and hence preserves π.

Identity Kernel

$I(x, A)$ denotes identity kernel, which is Markov, corresponds to update that does nothing, so $X_{n+1} = X_n$ almost surely

$$I(x, A) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases}$$

The "kernel" way to write indicator functions: $I(x, A) = I_A(x)$.

Identity kernel is reversible with respect to and preserves every signed measure.

State-Dependent Mixing (cont.)

Why? Consider Bayesian model selection with Hasse diagram

$$\begin{tikzcd}
& \{1,2,3\} & \\
\{1,2\} \arrow[ur] & \{1,3\} \arrow[u] & \{2,3\} \arrow[ul] \\
\{1\} \arrow[u] \arrow[ur] & \{2\} \arrow[ul] \arrow[ur] & \{3\} \arrow[ul] \arrow[u] \\
& \varnothing \arrow[ul] \arrow[u] \arrow[ur] &
\end{tikzcd}$$

Sets indicate variables in model and number of parameters.

State-Dependent Mixing (cont.)

Simplest idea is to have one elementary update for each arrow in diagram.

Updates need to be reversible so can go both ways, for example if in model $\varnothing$ propose point in model $\{2\}$ and vice versa.

This particular update makes no sense when current state x is not in either model $\varnothing$ or model $\{2\}$. Might as well choose with probability zero in that case.

Hence state-dependent mixing. Only choose elementary updates in states where they make sense.

Abstract Densities

If m and n are signed measures, and

$$m(A) = \int_A f(x) n(dx), \qquad \text{for all events } A$$

we say f is the *density* of m with respect to n.

Sometimes write

$$f = \frac{dm}{dn}$$

and call f the Radon-Nikodym derivative of m with respect to n.

Ordinary probability density function is abstract density with respect to Lebesgue measure (length in $\mathbb{R}$, volume in $\mathbb{R}^d$).

Ordinary probability mass function is abstract density with respect to counting measure (number of points in event).

Abstract Densities (cont.)

If m has density f with respect to n, then

$$n(A) = 0 \text{ implies } m(A) = 0, \qquad \text{for all events } A \tag{5}$$

When (5) holds, we say m is *dominated by* n. The Radon-Nikodym theorem says, that m dominated by n implies m has a density with respect to n, hence the name.

For us, main point is that abstract densities don't always exist.

Generalized Abstract Densities

For any signed measures m and n, let λ be any measure that dominates both (for example, $m + n$), define

$$f = \frac{dm}{d\lambda} \quad \text{and} \quad g = \frac{dn}{d\lambda}$$

which always exist by Radon-Nikodym theorem and

$$h(x) = \begin{cases} f(x)/g(x), & g(x) > 0 \\ \infty, & g(x) = 0 \end{cases} \tag{6}$$

Then write

$$\frac{dm}{dn} = h$$

regardless of whether ordinary Radon-Nikodym derivative exists.

Generalized Abstract Densities (cont.)

With these definitions

$$f = \frac{dm}{d\lambda} \quad \text{and} \quad g = \frac{dn}{d\lambda} \quad \text{and} \quad h = \frac{dm}{dn} = \frac{f}{g}$$

define

$$C = \{\, x : g(x) > 0 \,\} = \{\, x : h(x) < \infty \,\}$$

then

$$\begin{aligned} m(A \cap C) &= \int_{A \cap C} f(x) \lambda(dx) \\ &= \int_A h(x) g(x) \lambda(dx) \\ &= \int_A h(x) n(dx) \end{aligned}$$

So h is density of the part of m that is on C, which supports n.

Metropolis-Hastings-Green Elementary Update

Have proposal kernel $Q_i(x, A)$ chosen with probability $c_i(x)$.

Current state is x. Generate proposal y from $Q_i(x, \cdot\,)$.

Unnormalized measure to preserve is η. Define measures

$$\begin{aligned} m(B) &= \iint I_B(x, y) \eta(dx) c_i(x) Q_i(x, dy) \\ m_{\text{rev}}(B) &= \iint I_B(y, x) \eta(dx) c_i(x) Q_i(x, dy) \end{aligned}$$

and define

$$r = \frac{d m_{\text{rev}}}{d m}$$

then accept proposal with probability $\min\bigl(1, r(x, y)\bigr)$

Metropolis-Hastings-Green Elementary Update (cont.)

Can write

$$r(x, y) = \frac{c_i(y) \eta(dy) Q_i(y, dx)}{c_i(x) \eta(dx) Q_i(x, dy)}$$

(sloppy shorthand for actual definition).

Note proposal can be anything, arbitrary kernel $Q_i(x, A)$.

Only way to generalize would be to allow state-dependent mixing over continuum rather than countable set of $Q_i(x, A)$.

Metropolis-Hastings-Green Elementary Update (cont.)

If $\eta(N) = 0$, then no proposal in N can be accepted because $m_{\text{rev}}(A \times N) = 0$ for any set A, hence $r(x, y) = 0$ almost surely for $y \in N$.

Conclusion: if initial state of chain in not in N, then chain forever avoids N.

Metropolis-Hastings-Green Elementary Update (cont.)

All MCMC ideas discussed above in applications are special cases of Metropolis-Hastings-Green (MHG).

One-variable-at-a-time Metropolis-Hastings updates are special cases: proposal only changes one coordinate.

Gibbs updates are special cases: when proposal is Gibbs, then MHG ratio is always one, and proposal is always accepted.

Metropolis-Hastings-Green Theorem

Define

$$\begin{aligned} a(x, y) &= \text{min}\bigl(1, r(x, y)\bigr) \\ b(x) &= 1 - \int a(x, y) Q_i(x, dy) \end{aligned}$$

Kernel describing MHG elementary update is

$$P_i(x, A) = b(x) I(x, A) + \int_A a(x, y) Q_i(x, dy)$$

Kernel we must verify is reversible with respect to η is

$$K_i(x, A) = c_i(x) P_i(x, A)$$

that is

$$\iint g(x) h(y) \eta(dx) c_i(x) P_i(x, dy)$$

is unchanged when g and h are swapped.

Metropolis-Hastings-Green Theorem (cont.)

$$
\begin{aligned}
&\iint g(x)h(y)c_i(x)\eta(dx)P_i(x,dy) \\
&\qquad\qquad = \int g(x)h(x)b(x)c_i(x)\eta(dx) \\
&\qquad\qquad\qquad + \iint g(x)h(y)a(x,y)c_i(x)\eta(dx)Q_i(x,dy)
\end{aligned}
$$

Clearly enough to show last term is unchanged when g and h are swapped.

Metropolis-Hastings-Green Theorem (cont.)

$$
\begin{aligned}
&\iint g(x)h(y)a(x,y)c_i(x)\eta(dx)Q_i(x,dy) \\
&\quad = \iint g(y)h(x)a(y,x)c_i(y)\eta(dy)Q_i(y,dx) \\
&\quad = \iint g(y)h(x)a(y,x)m_{\text{rev}}(dx,dy) \\
&\quad = \iint_{r(x,y)<\infty} g(y)h(x)a(y,x)m_{\text{rev}}(dx,dy) \\
&\quad = \iint_{r(x,y)<\infty} g(y)h(x)a(y,x)r(x,y)m(dx,dy) \\
&\quad = \iint g(y)h(x)a(y,x)r(x,y)m(dx,dy) \\
&\quad = \iint g(y)h(x)a(y,x)r(x,y)c_i(x)\eta(dx)Q_i(x,dy)
\end{aligned}
$$

Metropolis-Hastings-Green Theorem (cont.)

Enough to show

$$a(y, x) r(x, y) = a(x, y) \quad \text{whenever} \quad r(x, y) < \infty \tag{7}$$

Case I: $r(x, y) \leq 1$. Implies $a(x, y) = r(x, y)$ and $a(y, x) = 1$, in which case (7) holds.

Case II: $1 < r(x, y) < \infty$. Implies $a(x, y) = 1$ and

$$a(y, x) = r(y, x) = \frac{1}{r(x, y)}$$

in which case (7) holds again.

Spatial Point Processes

Geyer and Møller (1994) predates Green (1995).

Spatial point process is random number of points in region A with finite measure (length, area, volume, ...), each point having random position.

A homogeneous Poisson process has a Poisson distributed number of points and the locations of the points are independent and identically and uniformly distributed conditional on the number.

Non-Poisson Spatial Point Processes

We consider processes having unnormalized densities h_θ with respect to the Poisson processes.

Normalizing constant is

$$c(\theta) = \sum_{n=0}^{\infty} \frac{\mu^n e^{-\mu}}{n!} \int h_\theta(x) \lambda^n(dx)$$

where λ^n is measure on A^n (area, length, volume, ...).

Strauss Process

Exponential family with two natural statistics $t_1(x)$ is number of points in x and $t_2(x)$ is number of pairs of points whose distance apart is less than d, which is treated as known constant, not parameter to estimate.

Unnormalized densities

$$h_\theta(x) = e^{t_1(x)\theta_1 + t_2(x)\theta_2}$$

Strauss Process Normalizing Constant

If $\theta_2 \leq 0$, then

$$\begin{aligned} c(\theta) &= \sum_{n=0}^{\infty} \frac{\mu^n e^{-\mu}}{n!} \int h_\theta(x) \lambda^n(dx) \\ &\leq \sum_{n=0}^{\infty} \frac{\mu^n e^{-\mu}}{n!} \cdot e^{n\theta_1} \int \lambda^n(dx) \\ &= \sum_{n=0}^{\infty} \frac{\mu^n e^{-\mu}}{n!} \cdot e^{n\theta_1} \lambda(A)^n \\ &\leq \exp\Bigl[\mu + e^{\theta_1} + \lambda(A)\Bigr] \end{aligned}$$

Strauss Process Normalizing Constant (cont.)

If $\theta_2 > 0$, consider region $B \subset A$ so small that any pair of points in B has distance apart less than d, then

$$\begin{aligned}
c(\theta) &= \sum_{n=0}^{\infty} \frac{\mu^n e^{-\mu}}{n!} \int h_\theta(x) \lambda^n(dx) \\
&\ge \sum_{n=0}^{\infty} \frac{\mu^n e^{-\mu}}{n!} \cdot \exp\left[n\theta_1 + \frac{n(n-1)}{2} \theta_2 \right] \int_B \lambda^n(dx) \\
&= \sum_{n=0}^{\infty} \frac{\mu^n e^{-\mu}}{n!} \cdot \exp\left[n\theta_1 + \frac{n(n-1)}{2} \theta_2 \right] \lambda(B)^n \\
&= \infty
\end{aligned}$$

Strauss Process (cont.)

So Strauss process only exists when $\theta_2 \leq 0$.

Similar sorts of checks have to be made for all models specified by unnormalized densities. Similar situation in Bayesian inference with improper priors.

Must check using calculus. Cannot simulate what does not exist. MCMC does not do calculus.

Geyer-Møller Update

Let $n(x)$ denote number of points in x.

i-th update only valid when $n(x) = i$, in which case propose to add a point ξ uniformly distributed in A, or when $n(x) = i + 1$, in which case propose to delete the last point ξ.

State dependent mixing

$$c_i(x) = \begin{cases} 1/2, & n(x) = i \\ 1/2, & n(x) = i + 1 \\ 0, & \text{otherwise} \end{cases}$$

For fixed x have $\sum_i c_i(x) = 1$ except when $n(x) = 0$ (empty point pattern) unless we define $c_{-1}(x) = 1/2$ in this case and let the "-1" update be the identity update (which does nothing).

Geyer-Møller Update (cont.)

Let $x \cup \xi$ denote pattern when point ξ added to pattern x.

For move from x to $y = x \cup \xi$ when $n(x) = k$ the MHG ratio is

$$\begin{aligned} r(x, y) &= \frac{\frac{1}{2} h_\theta(y) \mu^{k+1} e^{-\mu} / (k+1)!}{\frac{1}{2} h_\theta(x) \mu^k e^{-\mu} / k! \lambda(A)} \\ &= \frac{h_\theta(y) \cdot \mu \cdot \lambda(A)}{h_\theta(x) \cdot (k+1)} \end{aligned}$$

For move other way have

$$r(y, x) = \frac{1}{r(x, y)}$$

Non-Poisson Spatial Point Processes (cont.)

Traditional to use $\mu = 1$ to describe density.

Also traditional to use $h_\theta(x)$ that is symmetric under exchange of points in pattern. In this case, the update that re-orders the points randomly also preserves the stationary distribution.

Equivalent to picking random point to delete rather than last point.

Geyer-Møller Update (cont.)

With probability $\frac{1}{2}$ propose upstep and with probability $\frac{1}{2}$ propose downstep except when at empty state, in which case propose identity step.

For upstep, simulate ξ uniformly distributed in A. MHG ratio is

$$r(x, x \cup \xi) = \frac{\lambda(A)}{n(x) + 1} \cdot \frac{h_\theta(x \cup \xi)}{h_\theta(x)}$$

For downstep, pick ξ uniformly from among points in x. Let $x \setminus \xi$ denote pattern x with point ξ deleted. MHG ratio is

$$r(x, x \setminus \xi) = \frac{n(x) + 1}{\lambda(A)} \cdot \frac{h_\theta(x \setminus \xi)}{h_\theta(x)}$$

MHG with Jacobians and Augmented State Space

Green (1995) also proposed what is in some respects a special case of MHG and in other respects an extension.

So widely used that many users think MHGJ is the general version. This form of elementary update moves between parts of the state space that are Euclidean spaces of different dimension, hence often called "dimension jumping".

Suppose state space is disjoint union

$$S = \bigcup_{m \in M} S_m$$

where S_m is a Euclidean space of dimension d_m.

In Bayesian model averaging m indexes models and S_m is the the parameter space of model m.

MHGJ (cont.)

Have unnormalized density $h(x)$, a nonnegative function on S.

MHGJ elementary updates move from one S_m to another.

Say i-th elementary update moves from $S_{m(i)}$ to $S_{n(i)}$.

Only makes sense to have $c_i(x) > 0$ when $x \in S_{m(i)} \cup S_{n(i)}$.

MHGJ (cont.)

Let $U_{m(i)}$ and $U_{n(i)}$ be Euclidean spaces such that

$$S_{m(i)} \times U_{m(i)} \text{ is same dimension as } S_{n(i)} \times U_{n(i)}$$

Have proposal density $q_i(x, u)$, which describes conditional distribution of u given x such that

$$\begin{aligned} u \in U_{m(i)} &\text{ when } x \in S_{m(i)} \\ u \in U_{n(i)} &\text{ when } x \in S_{n(i)} \end{aligned}$$

Let g_i be a function that maps points in $S_{m(i)} \times U_{m(i)}$ to points in $S_{n(i)} \times U_{n(i)}$ and vice versa and that is its own inverse.

MHGJ (cont.)

MHGJ elementary update proposes u using $q_i(x, \cdot\,)$ and then move to $g_i(x, u) = (y, v)$.

MHG ratio is

$$r(x, u, y, v) = \frac{c_i(y) h(y) q_i(y, v)}{c_i(x) h(x) q_i(x, u)} \cdot \operatorname{det}\Bigl(\nabla g_i(x, u)\Bigr)$$

MHGJ Theorem

We verify

$$\iint s(x, u) t(y, v) c_i(x) h(x) q_i(x, u) \, dx \, du \tag{8}$$

is unchanged when s and t are interchanged, where it is understood that $(y, v) = g(x, u)$.

This is more than we need to verify the reversibility required for state-dependent mixing. That only requires (8) is unchanged when s and t are interchanged in the special case where s and t are functions of their first arguments only (unaugmented rather than augmented state).

MHGJ Theorem (cont.)

Could give direct proof that (8) is unchanged when s and t are interchanged, but it would follow earlier proof exactly, merely substituting augmented state for unaugmented state.

Hence we merely show that $r(x, u, y, v)$ is appropriate generalized abstract derivative. Tricky because (y, v) is deterministic function of (x, u).

Need to consider two distributions, with densities

$$f_{Y,V}(y, v) = c_i(y)h(y)q_i(y, v) \tag{9a}$$
$$f_{X,U}(x, u) = c_i(x)h(x)q_i(x, u) \tag{9b}$$

each considers first variable as current state having stationary distribution and second variable as proposed augmentation.

MHGJ Theorem (cont.)

MHG ratio should be ratio of (9a) and (9b), but have to express in terms of same variables first.

Do multivariate change of variable in (9a) changing variable from (y, v) to (x, u) obtaining

$$f_{Y,V}(x, u) = c_i(y) h(y) q_i(y, v) \cdot \operatorname{det}\Bigl(\nabla g_i(x, u)\Bigr)$$

where, as before, $(y, v) = g(x, u)$.

Now ratio is asserted MHG ratio.