A Random Walk Through the Big Metropolis (Couples Welcome)

Jeffrey S. Rosenthal
University of Toronto
jeff@math.toronto.edu
http://probability.ca/jeff/

(CRM-SSC Prize talk, London, Ontario, May 31, 2006)

Random Processes

Random processes . . . stochastic processes . . . Markov chains . . . random walks . . . what are they?

- Probabilistic rules for "what to do next".
- Rules are re-applied over and over again.
- In the long run, even simple rules lead to interesting behaviour.
- Applications to gambling (e.g. "Gambler's Ruin"), sampling algorithms ("Markov chain Monte Carlo"), and more.

First Example: Simple Random Walk

Repeatedly make \$1 bets. Each time, win \$1 with prob p, or lose \$1 with prob $1-p$. $(0 < p < 1)$ [APPLET]

More formally:

Start at some integer X_0 (initial fortune).

Then iteratively, for $n = 1, 2, \ldots$, X_n is either $X_{n-1} + 1$ (prob p) or $X_{n-1} - 1$ (prob $1-p$).

Equivalently, $X_n = X_0 + Z_1 + Z_2 + \ldots + Z_n$, where $\{Z_i\}$ are i.i.d. with $\mathbf{P}[Z_i = +1] = p = 1 - \mathbf{P}[Z_i = -1]$.

Simple Random Walk (cont'd)

Even this simple example has many interesting properties:

- Distribution: $\frac{1}{2}(X_n - X_0 + n) \sim \text{Binomial}(n, p)$
- Limiting Distribution: $\frac{1}{\sqrt{n}}(X_n - X_0 - n(2p-1)) \approx \text{Normal}(0, 1)$ (n large) (CLT)
- Recurrence: $\mathbf{P}[\,\exists\; n \geq 1 : X_n = X_0] = 1$ iff symmetric, i.e. $p = 1/2$ (also true in dim $= 2$, but not in dim ≥ 3)
- Fluctuations: if $p = 1/2$, the process will eventually hit any sequence $a_1, a_2, \ldots, a_\ell$.
- Martingale: if $p = 1/2$, then $\mathbf{E}(X_n \,|\, X_0, \ldots, X_{n-1}) = X_{n-1}$, i.e. the process stays the same on average.
If $p \neq 1/2$, then true of $\{((1-p)/p)^{X_n}\}$.

Gambler's Ruin

What is prob of e.g. doubling your initial fortune (I) before going broke, say with $p = 0.492929$ as in craps? [APPLET]

No "direct computation" solution (since time unbounded).

Instead, can solve using difference equations, or martingales:

Game:	Symmetric	Craps	Roulette
I = 1	$p = 50\%$	$p = 244/495 = 49.29\%$	$p = 18/38 = 47.7\%$
I = 10	50%	42.98%	25.85%
I = 100	50%	5.58% (1 in 18)	0.0027% (1 in 37,000)
I = 500	50%	1 in 1.4 million	1 in 10^{23}
I = 1,000	50%	1 in 10^{16}	1 in 10^{48}

Law of Large Numbers at work!

Distributional Convergence

Consider again simple symmetric ($p = 1/2$) random walk, but restricted to a finite state space (say, $\mathcal{X} = \{0, 1, \ldots, 6\}$) by simply "ignoring" moves off of $\mathcal{X}$.

That is: if process tries to jump off $\mathcal{X}$, then the move is rejected and instead we simply set $X_n = X_{n-1}$.

What happens in the long run? [APPLET]

The chain's empirical distribution (black bars) converges to the "target" Uniform($\mathcal{X}$) distribution (blue bars).

Interesting! Useful??

Other Target Distributions

To converge to other distributions, $\pi(\cdot)$, besides Uniform($\mathcal{X}$):

From X_{n-1}, if trying to move to Y_n, then accept this only with probability $\min[1,\ \pi(Y_n)/\pi(X_{n-1})]$, otherwise reject it and set $X_n = X_{n-1}$. ("Metropolis Algorithm") [APPLET]

Then for large enough B ("burn-in time"), X_B, X_{B+1}, ... are approximate samples from $\pi(\cdot)$. So e.g. for large m:

$$\mathbf{E}_\pi(h) \approx \frac{1}{m} \sum_{i=B}^{B+m-1} h(X_i) \, .$$

"Markov Chain Monte Carlo" (MCMC).

Extremely popular in statistics, physics, computer science, finance, and more: 661,000 Google hits.

Evaluating MCMC Algorithms

e.g. Java applet example, with $\pi\{2\} = 0.0001$. [APPLET]

Still converges, but very slowly: difficult crossing state 2.

Alternately, from $X_{n-1} = x$, could select proposed next state by:

$$Y_n \sim \text{Uniform}\{x-\gamma, \ldots, x-1, x+1, \ldots, x+\gamma\},$$

for other $\gamma \in \mathbf{N}$ (besides $\gamma = 1$). [APPLET]

Research Questions:

1. How long until convergence? (i.e., how large should B be?)
2. How to select γ? (i.e., which MCMC algorithm is best?)

Easy enough in this simple example, but what about a . . .

Typical Statistical Application

Might wish to sample from e.g. this density on $\mathbf{R}^{K+3}$:

$$f(\sigma_\theta^2, \sigma_e^2, \mu, \theta_1, \ldots, \theta_K) \;=\; C\; e^{-b_1/\sigma_\theta^2} {\sigma_\theta^2}^{-a_1-1} e^{-b_2/\sigma_e^2} {\sigma_e^2}^{-a_2-1} e^{-(\mu-\mu_0)^2/2\sigma_0^2} \times \prod_{i=1}^{K} [e^{-(\theta_i-\mu)^2/2\sigma_\theta^2}/\sigma_\theta] \times \prod_{i=1}^{K} \prod_{j=1}^{J} [e^{-(Y_{ij}-\theta_i)^2/2\sigma_e^2}/\sigma_e]\,,$$

where K, J large, $\{Y_{ij}\}$ data (given), $a_1, a_2, b_1, b_2, \mu_0, \sigma_0^2$ are fixed prior parameters (given), and $C > 0$ is normalizing constant.

[Posterior for Variance Components Model.]

Can't do numerical integration ... nor even compute C.

Can use Metropolis, with e.g. $Y_n \sim \text{Normal}(X_{n-1}, \sigma^2)$.

But for what σ^2? And what burn-in B??

Bounding Convergence Through Coupling

Suppose that together with $\{X_n\}$, have a second process $\{X'_n\}$ with $X'_n \sim \pi(\cdot)$ for all n.

Then coupling inequality says

$$|\mathbf{P}(X_n \in A) - \pi(A)| \; \leq \; \mathbf{P}(X_n \neq X'_n)\,.$$

So, if can force $X'_n = X_n$ with high probability, then can bound convergence.

Simplest case: $\{X'_n\}$ independent of $\{X_n\}$ until the first time T with $X'_T = X_T$. After that the two processes proceed together, i.e. $X'_n = X_n$ for all $n \geq T$, so $\mathbf{P}(X_n \neq X'_n) = \mathbf{P}(T > n)$.

Problem: T may be very large, or even infinite. Bad!

Coupling via Minorisation Conditions

Suppose can find a "minorisation" (overlap) decomposition:

$$\mathcal{L}(X_n \mid X_{n-1} = x) \;=\; \epsilon\, \nu(\cdot) + (1-\epsilon)\, R_x(\cdot)\,,$$

$$\mathcal{L}(X'_n \mid X'_{n-1} = x') \;=\; \epsilon\, \nu(\cdot) + (1-\epsilon)\, R_{x'}(\cdot)\,.$$

Then given $X_{n-1} = x$ and $X'_{n-1} = x'$, can construct (X_n, X'_n) by:
(a) with probability ϵ, $X_n = X'_n \sim \nu(\cdot)$; or
(b) with probability $1-\epsilon$, $X_n \sim R_x(\cdot)$ and $X'_n \sim R_{x'}(\cdot)$.

This increases $\mathbf{P}(X_n = X'_n)$, and thus reduces convergence bound.

Can sometimes be applied to complicated statistical examples.

But not easy ... best years of my life ...

Another Approach: Adaptive MCMC

Consider again the Java applet example with $\mathcal{X} = \{1, 2, \ldots, 6\}$.

For each $\gamma \in \mathbf{N}$, have a Metropolis algorithm P_γ.

Which one is best? converges fastest? How to tell?? [APPLET]

Idea: Get the computer to modify the chain adaptively, i.e. choose a sequence $\{\Gamma_n\}$ of values for γ "on the fly".

Hopefully, computer can "learn" good MCMC algorithms for us.

But easier said than done ...

Adaptive MCMC (cont'd)

Helpful observations about Java applet example (and beyond):

- If γ too small (say, $\gamma = 1$), then usually accept, but don't move very far – bad!
- If γ too large (say, $\gamma = 50$), then hardly ever accept – bad!
- Best is a "moderate" value of γ, like 3 or 4, so step sizes and acceptance probs are both non-small. ["Goldilocks principle"]

<u>Conclude</u>: If the chain almost always accepts, then γ may be too <u>small</u> and should be <u>increased</u>.

But if the chain almost always rejects, then γ may be too <u>large</u> and should be <u>reduced</u>.

(Optimal acceptance rate?!?)

Adaptive MCMC (cont'd)

Then let computer search for "moderate" values of γ:

- Start with γ set to $\Gamma_0 = 2$ (say).
- Each time proposed move is accepted, set $\Gamma_n = \Gamma_{n-1} + 1$ (so γ increases, and acceptance rate decreases).
- Each time proposed move is rejected, set $\Gamma_n = \max(\Gamma_{n-1} - 1,\ 1)$ (so γ decreases, and acceptance rate increases).

Logical, natural adaptive scheme, which uses the computer to perform a "search" for a good γ, on the fly.

But does it work?? [APPLET]

NO IT DOESN'T!!

The chain eventually gets stuck with $X_n = \Gamma_n = 1$ for long stretches of time. [Asymmetric: entering $\{X_n = \Gamma_n = 1\}$ much easier than leaving it.]

Chain doesn't converge to $\pi(\cdot)$ at all.

The adaption has RUINED the algorithm.

Disaster!!

When Does Adaptive MCMC Preserve Convergence?

Various theorems (joint with G.O. Roberts) ensure that Adaptive MCMC will converge under certain conditions.

In Java example, suffices that $\mathbf{P}[\Gamma_n \neq \Gamma_{n-1}] \to 0$, i.e. probability of modifying γ goes to 0. ("Diminishing Adaptation")

We have applied these theorems to e.g.

- The "Adaptive Metropolis" (AM) algorithm, which attempts to adapt Metropolis algorithm proposal distributions to target.
- Metropolis-Hastings algorithms in which the proposal distribution from x is Normal(x, σ_x^2), where σ_x^2 is some function of x.

Seems promising; more examples coming soon!

Summary

Random processes / Markov chains are interesting and powerful.

- Complicated behaviour arises from repeating simple rules.
- Distributions, limits, recurrence, fluctuations, martingales, gambler's ruin, ...
- MCMC (Metropolis etc.): approximate samples (after convergence).
- Can bound convergence time using coupling & minorisations.
- Which algorithm? Can get computer to choose, if careful.

Lots of difficult research problems to keep us all busy!