Adaptive Markov Chain Monte Carlo

Scott C. Schmidler

Stat 863: Advanced Statistical Computing
Duke University
Fall 2018

Overview

- Basic ideas of adaptive MCMC and examples
- Theory for AMCMC: asymptotics vs mixing times
- Combining adaptive strategies: exploration/exploitation

Metropolis Algorithm

General case: $\pi(dx) = \pi(x)\mu(dx)$ for some σ-finite μ on $\mathcal{X}$.

To draw samples from $\pi(x)$:

Choose *proposal* kernel $q(x, x')$.

Metropolis-Hastings

- Draw $x^* \sim q(x^{(t)}, \cdot)$
- Set $x^{(t+1)} = \begin{cases} x^* & \text{w/ prob } \alpha = \min\left(1, \frac{\pi(x^*)q(x^*,x)}{\pi(x)q(x,x^*)}\right) \\ x^{(t)} & \text{otherwise} \end{cases}$

Result: reversible MC with stationary distribution $\pi(x)$.

Random-walk Metropolis

How to choose proposal q? Common choices:

- Random walk: $x^* = x + \epsilon$
  - e.g. if $\mathcal{X} = \mathbb{R}^d$, take $\epsilon \sim N(0, \sigma^2 I_d)$.
- Independent: $q(x, x') \equiv q(x')$ (MIS)

Works under simple conditions (support). May not be efficient.

Example: Suppose $\pi(x) = N_2(0, \Sigma)$

Consider $\Sigma = \begin{bmatrix} \sigma_1 & \rho \\ \rho & \sigma_2 \end{bmatrix}$, with $\sigma_1 = 2$, $\sigma_2 = 1$ and $\rho = .95$

Efficiency

Statistical Efficiency: $var(\hat{f})$

Under reasonably weak conditions*, for any function f with $var_\pi(f) \leq \infty$, we obtain a CLT:

$$\sqrt{n}(\bar{f}_n - \mu_f) \to N(0, \sigma^2_{\bar{f}_n})$$

where

$$\sigma^2_{\bar{f}_n} = \sigma^2_f(1 + 2\sum_{j=1}^{n}(1 - \frac{j}{n})\rho_j)$$

and ρ_j is lag-j autocorrelation:

$$\rho_j = \frac{1}{\sigma^2_f} E\left((f(X^{(n)}) - \mu_f)(f(X^{(n+j)}) - \mu_f)\right)$$

Adaptive Metropolis

$q(x, x'; \theta)$ some parametric family. Can we "tune" θ automatically?

$$q^{(t)}(\cdot, \cdot) = q(\cdot, \cdot; \theta^{(t)}) \quad \text{for} \quad \theta^{(t)} = \theta(X_1, \ldots, X_{t-1})$$

Obvious, old idea. But ... no longer a Markov chain.

- Does it converge to π?
- Does it converge at all?

Old solutions:

- Stop adapting at some finite time t^*: for $t > t^*$ run a Markov chain; discard $X_{t \leq t^*}$. (But how to choose t^*?)
- Adapt only at *regeneration times*. (But difficult to identify.)

Adaptive Metropolis: Simple example

Haario *et al* (2001): For $n > 2d$, take

$$q^{(t)}(x, \cdot) = (1-\beta)N(x, (2.38)^2\hat{\Sigma}^{(t)}/d) + \beta N(x, .1^2 I_d/d)$$

Note: $(2.38)^2\Sigma/d$ “optimal” under Langevin diffusion approximation argument of RR01.

Key: *proof* of convergence (WLLN; uses “mixingales”).

I. Adaptive Metropolis kernels

Two approaches developed by various authors

Adaptive random-walk proposals

$$q_{n+1}(x, \cdot) = (1-\alpha)N(x, \hat{\Sigma}_n) + \alpha N(x, \Sigma_0)$$

e.g. Haario *et al*, Roberts & Rosenthal

Adaptive independence proposals (AMIS)

$$q_{n+1}(x, \cdot) = g(\cdot; \hat{\theta}_n) \quad \hat{\theta}_n = \theta(X_1, \ldots, X_n)$$

e.g. Andrieu & Moulines, Ji & Schmidler, etc.

Adaptive Metropolis algorithm: example

Convergence theorems

$X_1, \ldots, X_n$ no longer a Markov chain.

Under what conditions does $\hat{f}_n = \frac{1}{n}\sum_{i=1}^n f(X_i)$ converge?

- Haario *et al* 2001: WLLN, using “mixingales”
- Andrieu & Robert (2001): SA interpretation of Haario algorithm
- Andrieu & Moulines (2005), Atchadé& Rosenthal (2005): generalizations to other algorithms (and a CLT)
- Roberts & Rosenthal (2007): Simplified conditions, coupling

General setup: (Roberts & Rosenthal, 2007)

π target distribution on $\mathcal{X}$ with σ-algebra $\mathcal{F}$

$\{P_\gamma\}_{\gamma\in\mathcal{Y}}$ collection of π-invariant Markov kernels on $\mathcal{X}$

$X_n \in \mathcal{X}$: State of algorithm

$\Gamma_n \in \mathcal{Y}$: Choice of kernel for $Q_{n,n+1}$

$\mathcal{G}_n = \sigma(X_0, \ldots, X_n, \Gamma_0, \ldots, \Gamma_n)$ filtration generated by $\{(X_n, \Gamma_n)\}$.

$\Pr(X_{n+1} \in A \mid X_n = x, \Gamma_n = \gamma, \mathcal{G}_{n-1}) = P_\gamma(x, A)$

Ergodicity

Marginal kernel:

$$K^{(n)}((x,\gamma), A) = \Pr(X_n \in A \mid X_0 = x, \Gamma_0 = \gamma) \neq \prod_{i=0}^{n-1} P_{\Gamma_i}$$

Say the algorithm is *ergodic* if

$$\lim_{n\to\infty} \|K^{(n)}((x,\gamma), \cdot) - \pi(\cdot)\| = 0 \qquad \forall x \in \mathcal{X}, \gamma \in \mathcal{Y}$$

Cautionary example (RR07)

$\mathcal{X} = \{1, \ldots, k \geq 4\}$

$\pi(1) = a > 0 \qquad \pi(2) = b > 0$ small $\qquad \pi(x) = \frac{1-a-b}{k-2} > 0$

$Q_\theta(x, \cdot) = \text{Unif}\{x - \theta, \ldots, x + \theta\} \qquad \Theta = \mathbb{N}$

Initialize $\theta_0 = 1$, and adapt according to:

- If *accept*, $\theta_{n+1} = \theta_n + 1$
- If *reject*, $\theta_{n+1} = \theta_n - 1$

Discrete analog to adaptive-scale random-walk.

See Jeff Rosenthal's applet:

`http://probability.ca/jeff/java/adapt.html`

Sufficient conditions for convergence

Theorem (Roberts & Rosenthal, 2007):

- *Diminishing adaptation*:
  $\lim_{t\to\infty} \sup_{x\in\mathcal{X}} \|P_{\Gamma_{n+1}}(x,\cdot) - P_{\Gamma_n}(x,\cdot)\| = 0$ in probability.
- *Bounded convergence*: $P_{\gamma\in\Gamma}$ are "simultaneously polynomially ergodic"

Then adaptive algorithms is *ergodic*:

$$\lim_{n\to\infty} \left| K^{(n)}((x,\theta),\cdot) - \pi(\cdot) \right| = 0 \qquad \forall x, \theta$$

where $K^{(n)}((x,\theta),B) = P(X_n \in B \mid X_0 = x, \theta_0 = \theta)$ involves *marginalization*.

Sufficient conditions for convergence

Theorem 5 (Roberts & Rosenthal, 2007):

- (a) *Simultaneous Uniform Ergodicity*: $\forall \epsilon > 0, \exists N(\epsilon) \in \mathbb{N}$ s.t.
  $\|P_\gamma^N(x,\cdot) - \pi(\cdot)\| \leq \epsilon \qquad \forall x \in \mathcal{X}, \gamma \in \Gamma$
- (b) *Diminishing adaptation*:
  $\lim_{t\to\infty} \sup_{x\in\mathcal{X}} \|P_{\Gamma_{n+1}}(x,\cdot) - P_{\Gamma_n}(x,\cdot)\| = 0$ in prob.

Then adaptive algorithm is ergodic.

Note: $D_n = \sup_{x\in\mathcal{X}} \|P_{\Gamma_{n+1}}(x,\cdot) - P_{\Gamma_n}(x,\cdot)\|$ a $\mathcal{G}_{n+1}$-meas. r.v.
Note: Infinite adaptation allowed (i.e. $\sum D_n = \infty$ or $\sum p_n = \infty$); Γ_n need not converge.

Adaptive Metropolis kernels

Recall two approaches:

Adaptive random-walk proposals

$$q_{n+1}(x,\cdot) = (1-\alpha)N(x,\hat{\Sigma}_n) + \alpha N(x,\Sigma_0)$$

e.g. Haario *et al*, Roberts & Rosenthal

Adaptive independence proposals (AMIS)

$$q_{n+1}(x,\cdot) = g(\cdot;\hat{\theta}_n) \quad \hat{\theta}_n = \theta(X_1,\ldots,X_n)$$

e.g. Andrieu & Moulines, Ji & Schmidler, etc.

Adaptive Metropolized independence sampler (AMIS) [Ji and Schmidler, 2013]

Finite mixture proposal distribution:

$$q(x) = \lambda N(x;\tilde{\mu},\tilde{\Sigma}) + (1-\lambda)\sum_{m=1}^{M} w_m N(x;\mu_m,\Sigma_m)$$

Wish to minimize

$$\mathcal{D}\left[\pi(x) \,\|\, q(x;\psi)\right] = \mathbb{E}_\pi\left[\log\frac{\pi(x)}{q(x;\psi)}\right]$$

wrt proposal parameters $\psi = \{(w_i,\mu_i,\Sigma_i)\}_{i=1}^M$.

As $q(x) \to \pi(x)$

- Acceptance rate increases
- Samples become approximately *iid*

Adaptive Metropolized independence sampler (AMIS)

Adaptive strategy: Minimize $\mathcal{D}[\pi(x) \,\|\, q(x;\psi)] = \mathbb{E}_\pi\left[\log\frac{\pi(x)}{q(x;\psi)}\right]$

ψ^* obtained as a root of derivative:

$$h(\psi) = -\int \frac{\pi(x)}{q(x;\psi)} \frac{\partial}{\partial\psi} q(x;\psi) = 0$$

Approximate $h(\psi)$ by Monte Carlo integration:

$$h(\psi) \approx \frac{1}{K}\sum_{k=1}^{K} f(X^{(k)},\psi) \quad \text{for} \quad f(x,\psi) = \frac{\partial}{\partial\psi}[\log\frac{\pi(x)}{q(x;\psi)}]$$

where $X^{(k)} \sim \pi(x)$.

$\hat{h}(X^{(1:K)}; \psi)$: estimate of $h(\psi)$ based on sample path $X^{(1:k)}$

Stochastic Approximation algorithm [Robbins and Monro, 1951].

$$\begin{aligned}\psi_{n+1} &= \psi_n + r_{n+1}(h(\psi_n) + \xi_{n+1}) \\ &= \psi_n + r_{n+1}\,\hat{h}(X_n^{(1:K)}; \psi_n)\end{aligned}$$

$\{r_n\}$ decreasing step-sizes satisfying $\sum_n r_n = \infty$ and $\sum_n r_n^2 < \infty$

Resulting chain is non-Markovian, but can be shown to satisfy a WLLN using results of [Roberts and Rosenthal, 2007]

Example: Logistic regression

Bayesian logistic regression model,

$$y_i \mid x_i, \beta \sim \text{Bernoulli}\left(g^{-1}(x_i\beta)\right) \qquad \beta \sim \pi_0(\beta)$$

$y_i \in \{0, 1\}$; $g(u)$ logistic link

Simulated data set:

- 200 observations
- $r = 10$ covariates
- $\beta_{1:10} = [-.01, -1.5, .15, .5, -.15, -.2, -.6, .25, 1.5, -.05]$

Bayesian logistic regression

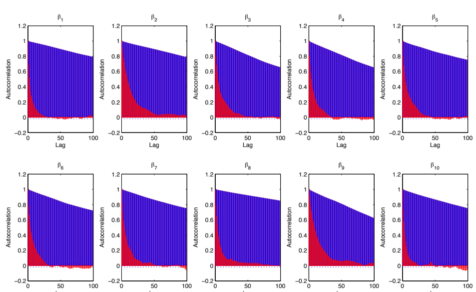


Figure: Autocorrelation of $\beta_{1:10}$ under data-augmentation Gibbs sampler [Holmes and Held, 2006] (blue), and adaptive MCMC algorithm (red).

Bayesian Variable Selection

GLM: $g(\mu_y) = \alpha + \sum_{i=1}^{p} \beta_i x_i$

g is *link function*, e.g. $g(x) = x$ or $g(x) = \text{logit}(x)$

x_i's are covariates (or *predictors*, or *features*)

Often many possible x_i's available: genes, SNPs, pixels, frequencies, QSAR, etc. Wish to retain the important ones.

(one) Bayesian approach: Let $\gamma = (\gamma_1, \ldots, \gamma_p) \in \{0, 1\}^p$ denote inclusion. Infer γ given data (X, Y):

$$\pi(\gamma, \beta \mid X, Y) \propto L(Y; X, \beta)\pi_0(\beta \mid \gamma)\pi_0(\gamma)$$

Variable selection priors

An alternative is the use of *point mass* variable selection priors:

$$\pi(\beta_i) = (1-p)\delta_0(\beta_i) + p\mathcal{N}(\beta_i|0, \sigma)$$

often call *spike-and-slab* priors.

Linear case $\Rightarrow$ closed-form Gibbs updates.
GLM case $\Rightarrow$ commonly assumed reversible-jump needed, but MH possible.

However ... resulting posterior often *multi-modal* in each variable giving combinatorial number of modes.

Random-walk Metropolis-Hastings will generally fail to mix well on such target distributions with multiple well-separated modes.

Adaptive Metropolized independence sampler (AMIS) [Ji and Schmidler, 2013]

Finite mixture proposal distribution:

$$q(x) = \lambda N(x; \tilde{\mu}, \tilde{\Sigma}) + (1-\lambda)\sum_{m=1}^{M} w_m N(x; \mu_m, \Sigma_m)$$

(see also Andrieu & Moulines 2005, others)

Point-mass mixture proposal for variable selection:

$$q(x) = (1-\lambda)\left[w_0\delta(x) + \sum_{m=1}^{M} w_m N(\mu_m, \Sigma_m)\right] + \lambda N(x; \tilde{\mu}, \tilde{\Sigma})$$

Adapt parameters $\psi = \{w_m, \mu_m, \Sigma_m\}_{m=0}^{M}$ to approximate $\pi(x)$.

Bayesian variable selection logistic regression

200 data points, $\beta = [1, 4, 2, -2, 0, 0, 0, 0, 0, 0]$.
Prior: $\pi_0(\beta_i) = 0.5\,\delta(\beta_i) + 0.5\,N(\beta_i \mid 0, \sigma^2)$

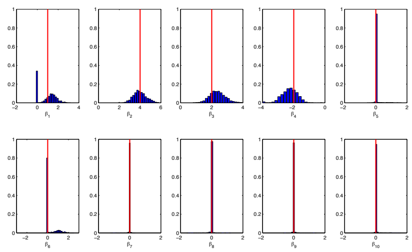


Figure: Posterior histograms of logistic coefficients $\beta_{1:10}$ obtained by adaptive MCMC (red: true values).

Relative efficiency

Comparison with random-walk Metropolis:

	Metropolis		Adaptive		Eff. sample size
	$\hat{\beta}_i$	std error	$\hat{\beta}_i$	std error	$\sigma^2_{\text{MCMC}}/\sigma^2_{\text{AMCMC}}$
β_1	1.59	1.31	0.95	0.108	147.1
β_2	6.55	0.59	3.97	0.052	127.5
β_3	2.82	0.76	2.37	0.063	146.5
β_4	-3.70	0.05	-2.27	0.007	50.8

Table: Logistic coefficients estimated via Bayesian variable selection. Adaptive MCMC yields effective sample sizes 50-150$\times$ larger than Metropolis.

Example: Kernel regression

Kernel regression model:

$$\mu_i = w_0 + \sum_{j=1}^{n} K(x_i, x_j) w_j \qquad \text{for} \quad i = 1, \ldots, n$$

$K(x, x^*)$ some *Mercer kernel* (pos semidef inner product), commonly a *radial basis function* $\exp\{-\sum_{k=1}^{p} \rho_k (x_k - x_k^*)^2\}$ or linear kernel $\sum_{k=1}^{p} \rho_k x_k x_k^*$

Kernel classification using probit model with latent variables $z_i > 0$ *iff* $y_i = 1$, so $P(y_i = 1) = \Phi(\mu_i)$

Usually K fixed, but when $p \geq n$ we want to infer parameters of the kernel (ρ's) to do simultaneous *feature selection*.

Bayesian model selection for kernel scale parameters:

$$\rho_k \sim (1-\gamma)\,\delta + \gamma\,\text{Gamma}(a_\rho, a_\rho s) \qquad k = 1, \cdots, p$$
$$s \sim \text{Exp}(a_s) \qquad \gamma \sim \text{Beta}(a_\gamma, b_\gamma)$$

West *et al* developed MH algorithm, but mixes slowly.

We apply AMIS algorithm:

Adaptive mixture-of-Gammas proposal

$$q(\rho) = (1-\lambda)\Big[w_0 \delta(\rho) + \sum_{m=1}^{4} w_m \mathcal{G}(\rho; \alpha_m, \beta_m)\Big] + \lambda \mathcal{G}(\rho; 1, 10).$$

Example: Kernel regression

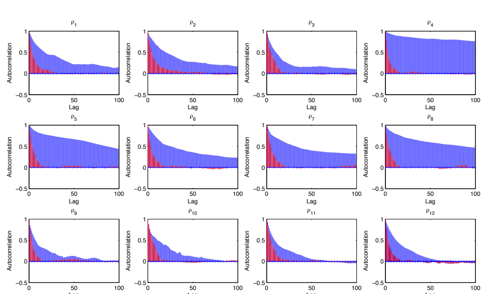


Figure: Autocorrelation of ρ_k's under MCMC algorithm of [Liang et al., 2006] (blue) and adaptive MCMC (red).

Example: Helix-coil model (Gibbs random field)

Biophysical (stat mech) model for predicting equilibrium conformation of short peptides [Schmidler et al., 2007].

Described by Gibbs distribution

$$P(X \in \mathcal{X} \mid R) = Z^{-1} e^{-\frac{1}{kT} U(X,R)}$$

with interaction potential

$$U(X, R) = \sum_{i=1}^{l} x_i \alpha_{R_i} + \sum_{i=1}^{l-3} x_{i:i+3} \beta_{R_i R_{i+3}} + \sum_{i=1}^{l-4} x_{i:i+4} \gamma_{R_i R_{i+4}}$$

Sort of like 1D Potts model with 4-nn interactions, 20 colors.

Many additional parameters. (Note interactions are 20$\times$20.) Select out important ones.

Adaptive MIS algorithm

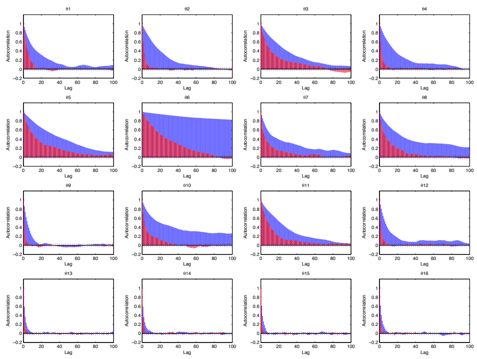


Figure: Autocorrelation of helix-coil parameters under MCMC algorithm

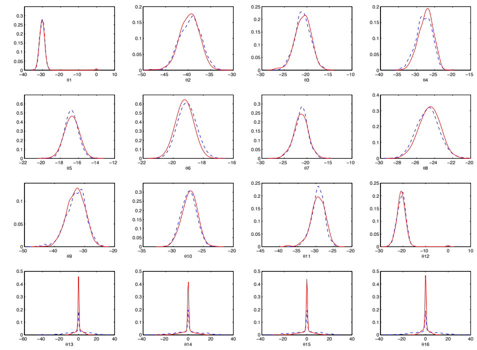


Figure: Posterior distributions of helix-coil model parameters obtained by MCMC algorithm of [Lucas, 2006] (dashed blue) and adaptive MCMC (red) are the same.

Adaptive MCMC theory

Nearly all theory to date deals with *ergodicity* (LLN).
A few give conditions for CLTs (e.g Andrieu & Moulines (2005)).

Important and a significant advance. But all *asymptotic*.

We already knew how to construct ergodic MCMC algorithms.

Adaptation is only interesting if it improves *rates*!

MCMC Theory

- Ergodicity: SLLN under usual conditions (ϕ-irred, aper, π-invariant)
- Geometric: $\exists \lambda \in [0,1)$ and $M(x) < \infty$ ($\pi -$ a.e.$x \in \mathcal{X}$) s.t.

$$\|\mu K^n - \pi\| \leq M(x)\lambda^n$$

  Requires minorization, drift conditions. Implies CLT.
- Uniform: $M(x) \equiv M$
- Rapid mixing: λ grows at most polynomially in d
  (Note G.E. requires only $\lambda^* > 0$; e.g.holds for any $|\mathcal{X}| < \infty$)
- Quantitative: e.g. Rosenthal 1995

Efficiency revisited

Asymptotic efficiency:
Relies on CLT, asymptotic variance = integrated autocorrelation

Finite sample efficiency:
Convergence as well as autocorrelation

$$\mathrm{MSE}(\hat{\theta}) = \mathrm{Bias}^2(\hat{\theta}) + \mathrm{Var}(\hat{\theta})$$

For multimodal targets, bias can dominate in MCMC.
For good adaptive MCMC algorithms, bias *will* dominate.

Examples

Mixtures of normals

$$\pi(z) = \frac{1}{2}N_M(z; -1_M, \sigma_1^2 I_M) + \frac{1}{2}N_M(z; 1_M, \sigma_2^2 I_M)$$

Upper/lower bounds on spectral gap (WSH07a,b) yield:

Thm: RW-MH is torpidly mixing.
Thm: Tempering is rapidly mixing for $\sigma_1 = \sigma_2$.
Thm: Tempering is torpidly mixing for $\sigma_1 \neq \sigma_2$.

Lower bounds on hitting times obtained by (SW10) yield:

Thm: Equi-energy sampler torpidly mixing for $\sigma_1 \neq \sigma_2$.
Thm: Haario adaptive RW kernel torpidly mixing for $\sigma_1 \neq \sigma_2$.

Example

Mean-field Ising model

$$\pi(x) = \frac{1}{Z}\exp\left\{\frac{\alpha}{2M}\left(\sum_{i=1}^{M} x_i\right)^2\right\} \qquad \mathcal{X} = \{-1,+1\}^M$$

Thm: Gibbs sampler (Glauber dynamics) is slowly mixing. Thm (WSH07a): Parallel tempering is rapidly mixing (see also).

Mean-field Potts

With $k \geq 3$ colors.

Thm (WSH07b): Tempering is torpidly mixing (see also BR06).
Thm (SW10): Equi-energy sampler is torpidly mixing.

MRAM Processes (SW09)

Let $X^{(1)},\ldots,X^{(I)}$ discrete time stochastic processes on $\mathcal{X}$.
So $X^{(i)} = X_0^{(i)}, X_1^{(i)}, \ldots$

Generated by time-inhomogeneous sequences of transition kernels:

$$K_{i,n} = \alpha\, T_i + (1-\alpha)R_{i,n}$$

with $\alpha \in [0,1]$, T_i an ergodic time-homogeneous Markov $\pi^{(i)}$-reversible transition kernel, and $R_{i,n}$ is a *resampling* kernel with proposal:

$$Q_{i,n}(X_{n-1}^{(i)}, y) = \sum_{i'=1}^{I}\sum_{j=0}^{n-1} w_{i'j}\,\delta(y - X_j^{(i')})$$

(Proposes new state from the set of previous samples $X_{0:n-1}^{(1:I)}$.)

MRAM Algorithms

Mulitchain resampling adaptive Metropolis (MRAM):

- Equi-Energy Sampler
- Importance-Resampling from the Past (Atchadé)
- Gelfand-Sahu

Lower bounds on MRAM mixing

Theorem (SW09)

For any $\epsilon > 0$ and any $A \subset \mathcal{X}$ such that $0 < \pi^{(i)}(A) < 1$ for all i, the mixing time τ_ϵ^* of the MRAM satisfies:

$$\tau_\epsilon^* \geq (\pi(A) - \epsilon)\left[cI \max_i \gamma(A,i)\Phi_{T_i}(A)\right]^{-1}.$$

Note similarity to the bound obtained previously (WSH07b) for non-adaptive swapping:

$$\tau_\epsilon^* \geq 2^{-8}\ln(2\epsilon)^{-1}\left[\max_i \gamma(A,i)\Phi_{T_i}(A)\right]^{-1/2}.$$

Idea of proof:

- τ_ϵ is for worst-case π_0, so initialize $X^{(i)} \sim \pi_{|A^c}$
- Let Y the *restriction* of X to A^c; rejects any move leaving A^c.
- Then $Y_n^{(i)} \sim \pi_{|A^c}$ for all i, n, and $X = Y$ for all $n < H_A$
- $Z_n^{(i)}$ indicates a rejection in $Y^{(i)}$ due to restriction. Then:

$$\begin{aligned}\Pr(H_A \leq n) \leq \sum_{i=1}^{I}\sum_{j=1}^{n}\Pr(Z_j^{(i)}) &\leq \sum_{i=1}^{I}\sum_{j=1}^{n}\int_{A^c} T_i(y,A)\psi_{i,j-1}(dy)\\ &\leq c\sum_{i=1}^{I}\sum_{j=1}^{n}\int_{A^c} T_i(y,A)\pi^{(i)}|_{A^c}(dy)\\ &= cn\sum_{i=1}^{I}\pi^{(i)}(A)\Phi_{T_i}(A)\end{aligned}$$

Assumption

c arises because MRAM only asymptotically π-invariant; don't approach π monotonically.

Assumption

There exists a constant $1 \leq c < \infty$ such that $Y_0^{(i)} \overset{ind}{\sim} \pi^{(i)}|_A$ implies the marginal $\mathcal{L}(Y_n^{(i)})$ has a density with respect to $\pi^{(i)}|_A$ bounded by c.

(Holds for $c = 1$ for method of Atchade (2007) and when $\alpha = 1$.)

Single chain

Note appearance of the conductance:

Corollary

For any $0 < \epsilon < 1/4$, the mixing time τ_ϵ^* of an adaptive sampler based on T, with $I = 1$, satisfies:

$$\tau_\epsilon^* \geq \frac{1}{4\Phi_T}.$$

Corollary

Slow mixing of the Markov chain with transition kernel T implies slow mixing of any MRAM process based on T that has $I = 1$.

Efficiency revisited

Asymptotic efficiency:
Relies on CLT, asymptotic variance = integrated autocorrelation

Finite sample efficiency:
Convergence as well as autocorrelation

$$\mathrm{MSE}(\hat{\theta}) = \mathrm{Bias}^2(\hat{\theta}) + \mathrm{Var}(\hat{\theta})$$

$\Rightarrow$ MRAM and IAMC sampling can only improve autocorrelation piece!

Suggests considering alternative "adaptation" strategies.

Generalized Wang-Landau (Atchade & Liu, 2009)

Partition state space $\mathcal{X} = \mathcal{X}_0 \cup \ldots \cup \mathcal{X}_k$ according to predefined energy levels $-\infty \leq e_0 < e_1 < \cdots < e_k \leq \infty$.

Goal: Sample from $\tilde{\pi}(x) = \sum_{i=1}^k \frac{\pi(x)}{\pi(\mathcal{X}_i)} \mathbf{1}_{\mathcal{X}_i}(x)$ *uniform energy*

Algorithm: Adaptively estimate $\hat{\pi}_n(i) \approx \pi(\mathcal{X}_i)$ by SA:
$\{\gamma_n\}$ a sequence of decreasing positive numbers.
Initialize $\phi_0(i) > 0$ for $i = 1, \ldots, k$, and $\hat{\pi}_0(i) = \frac{\phi_0(i)}{\sum_j \phi_0(j)}$

(i) Sample $X_{n+1} \sim \sum_{i=1}^k \frac{\pi(x)}{\hat{\pi}_n(i)} \mathbf{1}_{\mathcal{X}_i}(x)$ by MH.

(ii) Set $\phi_{n+1}(i) = \phi_n(i)\left(1 + \gamma_{a_n} \mathbf{1}_{\{X_{n+1} \in \mathcal{X}_i\}}\right)$; $\hat{\pi}_{n+1}(i) = \frac{\phi_{n+1}(i)}{\sum_j \phi_{n+1}(j)}$.

(iii) If $\max_i \left| v_{\kappa,n+1}(i) - \frac{1}{k} \right| \leq \frac{c}{k}$ where $v_{\kappa,n}(i) = \frac{1}{n-\kappa} \sum_{j=\kappa+1}^{n} \mathbf{1}_{\{X_j \in \mathcal{X}_i\}}$ then set $\kappa = n+1$ and $a_{n+1} = a_n + 1$, otherwise $a_{n+1} = a_n$.

Exploration/Exploitation Algorithm (Wang & SS, 2010)

1. Run two chains in parallel: X^{WL} and $X^{\mathrm{AMIS+}}$
2. Every N_c iterations, update the proposal distribution for $X^{\mathrm{AMIS+}}$.
3. At iteration $n = m * N_c$, let E_n be the energy ring of $X_{n-1}^{\mathrm{AMIS+}}$. Form KDE $\hat{f}$ by adding the samples $\{X_1^{\mathrm{WL}}, \ldots, X_n^{\mathrm{WL}}\}$ to those in E_n.
4. Propose $X_n^{\mathrm{AMIS+}}$ from $\hat{f}_c$.
5. At other iterations, run the two chains independently.

Improving on (generalized) Wang-Landau

Performance of the WL algorithm depends heavily on a good choice of the energy rings $E_0, \ldots, E_k$: number, spacing, max.

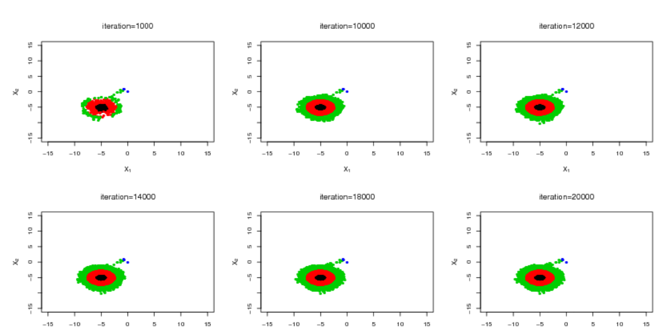
- *Adaptive-energy* GWL algorithm (AE-GWL), Wang & Schmidler (2011).

Monte-Carlo integration converges very slowly for WL

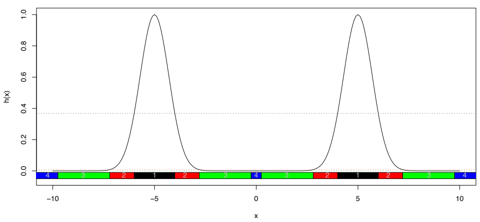
- Importance-resampling solution, Wang & Schmidler (2011).

Example

Figure: Example 2, modes at (-5,-5) and (5,5)



Slow mixing of generalized Wang-Landau



(b) $d = 4$, fixed energy levels

Theorem (SW11b): GWL slowly mixing for geometric energy-levels.

Energy level adaptation scheme

Performance of the WL algorithm depends heavily on a good choice of the energy rings $E_0, \ldots, E_k$.

We introduce an adaptive scheme to make updating energy levels fully automatic:

1. Initialize by a geometric progression:

   $e_0 = \inf_x E(x) = 0, \ e_1 = 1, \ e_2 = r_e, \ldots, E_{k-1} = r_e^{k-2}, E_k = infty.$

2. Every n_{split} iterations: if any $|\log(\phi_i) - \log(\phi_{i+1})| > E$, divide the i-th energy ring by adding a new $e_{i+1}^* = e_i \times \sqrt{\frac{e_{i+1}}{e_i}}$, again using geometric progression. Set $\log(\phi_{i+1}^*) = 0$.
3. Also update the second largest e_i;

   $$E_{k-1}^* = \frac{E_{k-1}^2}{E_k}$$

   Set $\log(\phi_k^*) = 0$.

Adaptive Energy Generalized Wang-Landau (AE-GWL)

Algorithm: Adaptively estimate $\hat{\pi}_n(i) \approx \pi(\mathcal{X}_i)$ by SA:
$\{\gamma_n\}$ a sequence of decreasing positive numbers.
Initialize $\phi_0(i) > 0$ for $i = 1, \ldots, k$, and $\hat{\pi}_0(i) = \frac{\phi_0(i)}{\sum_j \phi_0(j)}$

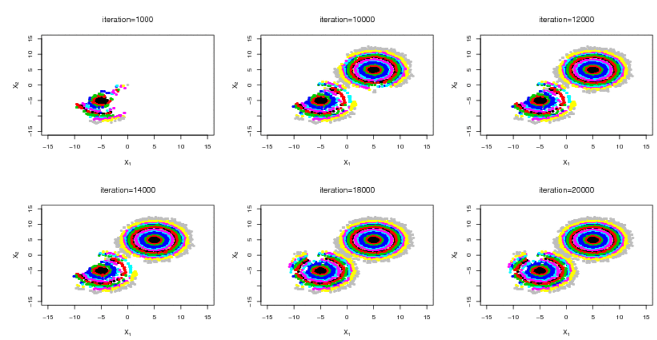
(i) Sample $X_{n+1} \sim \sum_{i=1}^k \frac{\pi(x)}{\hat{\pi}_n(i)} \mathbf{1}_{\mathcal{X}_i}(x)$ by MH.

(ii) Set $\phi_{n+1}(i) = \phi_n(i)\left(1 + \gamma_{a_n} \mathbf{1}_{\{X_{n+1} \in \mathcal{X}_i\}}\right)$ and $\hat{\pi}_{n+1}(i) = \frac{\phi_{n+1}(i)}{\sum_j \phi_{n+1}(j)}$.

(iii) If $\max_i \left| v_{\kappa,n+1}(i) - \frac{1}{k} \right| \le \frac{c}{k}$ where $v_{\kappa,n}(i) = \frac{1}{n-\kappa} \sum_{j=\kappa+1}^{n} \mathbf{1}_{\{X_j \in \mathcal{X}_i\}}$ then set $\kappa = n+1$ and $a_{n+1} = a_n + 1$, otherwise $a_{n+1} = a_n$.

(iv)* **For every n_{split} iterations, adaptively update $E = \{E_i\}$.**

Example



(c) $d = 4$, update internal energy levels

Types of MCMC adaptation

These ways of adapting address fundamentally different problems:

I & II: Improve mixing of chain among regions of target distribution *already visited*

- Improves autocorrelation of chain
- In general cannot help in exploring previously unseen regions

Call these *Exploitation* methods.

III: Tries to push chain away from points "like" those already seen.

- *Can* help in finding new regions; improve *mixing time*.
- May suffer from high autocorrelation.

Call these *Exploration* methods.

Hybrid adaptation strategies

Can we combine types to achieve best of both?
Yes but requires some care.

One approach: Mixture kernels

$$K_{\text{adapt}} = \alpha K_{\text{exploit}} + (1-\alpha) K_{\text{explore}}$$

Suffers problems in multimodal examples (Wiehe & Schmidler, 2010).

Alternative approach:

Run exploration chain independently in parallel, but use samples to augment AMIS approximation.

Exploration/Exploitation Algorithm (Wang & SS, 2011)

1. Run two chains in parallel: $X^{\text{AE-WL}}$ and $X^{\text{AMIS+}}$
2. Every N_c iterations, update the proposal distribution for $X^{\text{AMIS+}}$.
3. At iteration $n = m * N_c$, let E_n be the energy ring of $X_{n-1}^{\text{AMIS+}}$. Form KDE $\hat{f}$ by adding the samples $\{X_1^{\text{AE-WL}}, \ldots, X_n^{\text{AE-WL}}\}$ to those in E_n.
4. Propose $X_n^{\text{AMIS+}}$ from $\hat{f}_c$.
5. At other iterations, run the two chains independently.

Mixture Exponential regression [Kou et al., 2006]

$$y_i \sim \alpha \, \text{Exp}[\theta_1(x_i)] + (1-\alpha) \, \text{Exp}[\theta_2(x_i)]$$

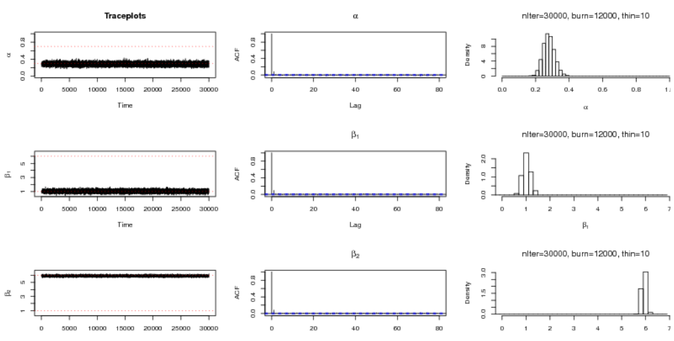
$$\theta_j(x_i) = \exp(\beta_j^T x_i), \quad \alpha = .3, \quad \beta_1 = 1, \, \beta_2 = 6, \quad x_i \equiv 1.$$

$$L(Y|\alpha, \beta_1, \beta_2) \propto \prod_{i=1}^{n} \left[\frac{\alpha}{\theta_1(x_i)} \exp\left(-\frac{y_i}{\theta_1(x_i)}\right) + \frac{1-\alpha}{\theta_2(x_i)} \exp\left(-\frac{y_i}{\theta_2(x_i)}\right) \right]$$

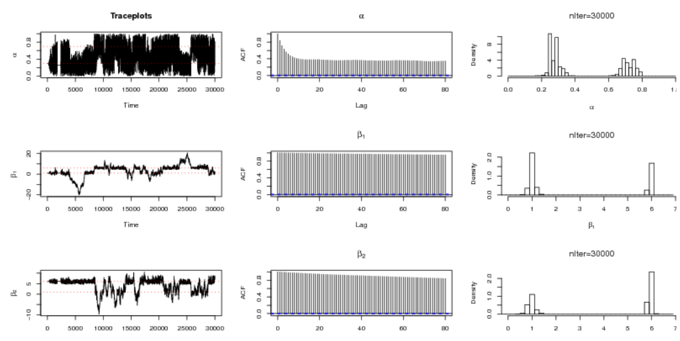
Priors: $\pi(\alpha) =$Beta$(1,1)$, $\pi(\beta_j) =$N$(0,100)$ for $j = 1,2$

$$E(\alpha, \beta_1, \beta_2) = -\log(\pi(\alpha, \beta_1, \beta_2|Y)) \propto -l(Y|\alpha, \beta_1, \beta_2) + \frac{\beta_1^2 + \beta_2^2}{2\sigma^2}$$

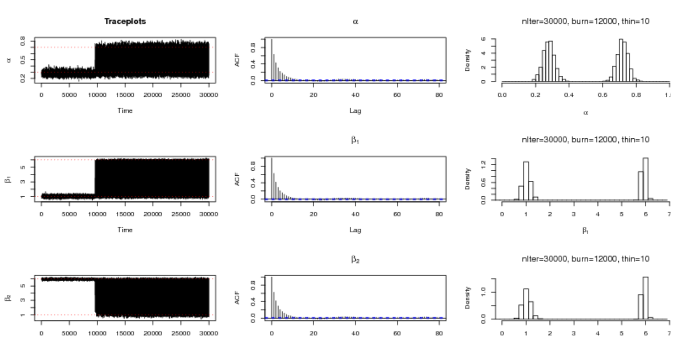
Mixture exponential regression: AMIS



Mixture exponential regression: WL



Mixture exponential regression: XX



Conclusions

Key ideas:

- Many ergodic adaptive MCMC methods may not improve *rate*.
- Convergence of MC estimators involves *both* bias *and* variance.
- Existing adaptation strategies improve one or the other.
- Improvements from algorithms which combining types of strategies.