Learn from Thy Neighbour: Parallel-Chain Adaptive MCMC

Radu Craiu

Department of Statistics
University of Toronto

Collaborators: Jeffrey Rosenthal (Statistics, Toronto)
Chao Yang (Mathematics, Toronto)

UBC, April 2008

Outline

1. Brief Review
   - Super-short Intro to MCMC
   - Adaptive Metropolis
2. Some Theoretical Tools
   - Some (NOT ALL!) Theory for Adaptive MCMC (AMCMC)
3. Can't Learn whAt We don't See (CLAWS)
   - The Problem
   - INter-Chain Adaptation (INCA)
   - Tempered INCA (TINCA)
4. ANTagonistic LEaRning (ANTLER)
   - The Problem
   - Regional AdaPTation (RAPT)
5. Conclusions
   - Discussion

Intro to Markov Chain Monte Carlo

- We wish to sample from some distribution for $X \in \mathcal{S}$ that has density π. Obtaining independent draws is too hard.
- We construct and run a Markov chain with transition $T(x_{old}, x_{new})$ that leaves π invariant

$$\int_{\mathcal{S}} \pi(x) T(x, y) dx = \pi(y).$$

- A number of initial realisations from the chain are discarded (burn-in) and the remaining are used to estimate expectations or quantiles of functions of X.

Metropolis algorithms

- The Metropolis sampler is one of the most used algorithms in MCMC. It operates as follows:

- Given the current state of the MC, x, a "proposed sample" y is drawn from a proposal distribution $P(y|x)$ that satisfies symmetry, i.e. $P(y|x) = P(x|y)$.
- The proposal y is accepted with probability $\min\{1, \pi(y)/\pi(x)\}$.
- If y is accepted, the next state is y, otherwise it is (still) x.

- The *random walk Metropolis* is obtained when $y = x + \epsilon$ with $\epsilon \sim f$, f symmetric, usually $N(0, V)$.
- If $P(y|x) = P(y)$ then we have the *independent Metropolis* sampler (acceptance ratio is modified).

Adapting the proposal

- How to determine what is a good proposal distribution? This is particularly difficult when $\mathcal{S}$ is a high dimensional space.
- Many MCMC algorithms are "adaptive" in some sense, e.g. adaptive directional sampling, multiple-try Metropolis with independent and dependent proposals, delayed rejection Metropolis ...
- Adaptive MCMC algorithms are designed to automatically find the "good" parameters of the proposal distribution (e.g. variance V).

Adaptive Metropolis

- Non-Markovian Adaptation (Haario, Saksman and Tamminen (HST); Bernoulli, 2001). Learn the geography of the stationary distribution "on the fly". Involves re-using the past realisations of the Markov chain to modify the proposal distribution of a random walk Metropolis (RWM) algorithm.
- Suppose the random-walk Metropolis sampler is used for the target π. The proposal distribution is $q(y|x) = N(x, \Sigma)$
- After an initialisation period, we choose at each time t the proposal $q_t(y|x_t) = N(x_t, \Sigma_t)$ where $\Sigma_t \propto \text{SamVar}(\tilde{X}_t)$ and $\tilde{X}_t = (X_1, \ldots, X_t)$.
- This choice is based on optimality results for the variance of a RWM in the case of Gaussian targets. (Roberts and Rosenthal, Stat. Sci., '01; Bedard, '07)

Adaptive Metropolis (cont'd)

- HST extend the idea to componentwise adaptation for MCMC (Metropolis within Gibbs) as a remedy for slow adaptation in large dimensional problems.
- Gåsemyr (Scand. J. Stat., 2005) introduces an independent adaptive Metropolis.
- Andrieu and Robert (2002) and Andrieu and Moulines (Ann. Appl. Prob., 2006) prove that the adaptation can be proved correct via theory for stochastic approximation algorithms.
- Roberts and Rosenthal (2005) introduce general conditions that validate an adaptive scheme. They also introduce scary examples where intuitively attractive adaptive schemes fail miserably.
- Giordani and Kohn (JCGS, 2006) use mixture of normals for adaptive independent Metropolis.

Theory for AMCMC

- Consider an adaptive MCMC procedure, i.e. a collection of chain kernels $\{T_\gamma\}_{\gamma \in \Gamma}$ each of which has π as a stationary distribution. One can think of γ as being the *adaption parameter*.
- Simultaneous Uniform Ergodicity: For all $\epsilon > 0$ there is $N = N(\epsilon)$ such that $||T_\gamma^N(x, \cdot) - \pi(\cdot)||_{TV} \leq \epsilon$ for all $x \in \mathcal{S}, \gamma \in \Gamma$.
- Let $D_n = \sup_{x \in \mathcal{S}} ||T_{\gamma_{n+1}}(x, \cdot) - T_{\gamma_n}(x, \cdot)||_{TV}$. Diminishing Adaptation: $\lim_{n \to \infty} D_n = 0$ in probability.

Theory of AMCMC - cont'd

Suppose that each T_γ is a Metropolis-Hastings algorithm with proposal distribution $P_\gamma(dy|x) = f_\gamma(y|x)\lambda(dy)$.

- If the adaptive MCMC algorithm satisfies Diminishing Adaptation
- and if λ is finite on $\mathcal{S}$
- and if $f_\gamma(y|x)$ is uniformly bounded
- and if for each fixed $y \in \mathcal{S}$ the mapping $(x, \gamma) \to f_\gamma(y|x)$ is continuous
- Then the adaptive algorithm is ergodic.

What's Next?

What Remains to Be Done

"Although more theoretical work can be expected, the existing body of results provides sufficient justification and guidelines to build adaptive MH samplers for challenging problems. The main theoretical obstacles having been solved, research is now needed to design efficient and reliable adaptive samplers for broad classes of problems." (Giordani and Kohn)

Two Practical Issues

- Multimodality is a never-ending source of headaches in MCMC.
- Adaptive algorithms are particularly vulnerable to this - quality of initial sample is central to the performance of the sampler.

- "Optimal" proposal may depend on the region of the current state.
- What to do if regions are not exactly known but they are approximated.

CLAWS: A simple example

- Consider sampling from a mixture of two 10-dimensional multivariate normals

$$\pi(x|\mu_1, \mu_2, \Sigma_1, \Sigma_2) = 0.5n(x; \mu_1, \Sigma_1) + 0.5n(x; \mu_2, \Sigma_2)$$

  with $\mu_1 - \mu_2 = 6$, $\Sigma_1 = I_{10}$ and $\Sigma_2 = 4I_{10}$.
- A RWM chain started in one of the modes needs to run for a very long time before it visits the other mode. Even longer if dimension is higher. Adaptive RWM cannot solve the problem unless the chain visits both modes.
- Idea: Handle Multimodality via Parallel Learning from Multiple Chains.

Inter-chain Adaptation (INCA)

- Run multiple chains started from a initialising distribution that is overdispersed w.r.t. π.
- Learn about the geography of the stationary distribution from all the chains simultaneously. Apply the changes to all the transition kernels simultaneously.
- At all times the parallel chains have the same transition kernels. The only difference is the region of the space explored by each chain.
- Use the past history from all the chains to adapt the kernel.
- This is different from using an independent chain for adaptation only (R & R, 2006).

INCA (cont'd)

- Suppose we run in parallel K chains. After m realisations $\{X_1^{(i)}, \ldots, X_m^{(i)} : 1 \leq i \leq K\}$ we assume that each chain runs independently of the others using transition kernel T_m.
- If we consider the K chains jointly, since the processes are independently coupled, the new process has transition kernel

$$\tilde{T}_m(\tilde{x}, \tilde{A}) = T_m(x_1, A_1) \otimes T_m(x_2, A_2) \otimes \ldots \otimes T_m(x_K, A_K),$$

where $\tilde{A} = A_1 \times \ldots \times A_K$ and $\tilde{x} = (x_1, \ldots, x_K)$.

INCA for RWM

- RWM with Gaussian proposal of variance $\mathcal{H}$.
- Suppose $K = 2$. After an initialisation period of length m_0 at each $m > m_0$ we update the proposal distribution's variance using $\mathcal{H}_m = Var(\mathbf{X_m^{(1)}}, \mathbf{X_m^{(2)}})$, where $\mathbf{X_m^{(i)}}$ are all the realisations obtained up to time m by the i-th process. The values for all chains are used to compute the sample variance $\mathcal{H}_m$.

INCA for RWM

- Show that if

$$D_m = \sup_{x \in \mathcal{S}} || T_{m+1}(x, \cdot) - T_m(x, \cdot)||,$$

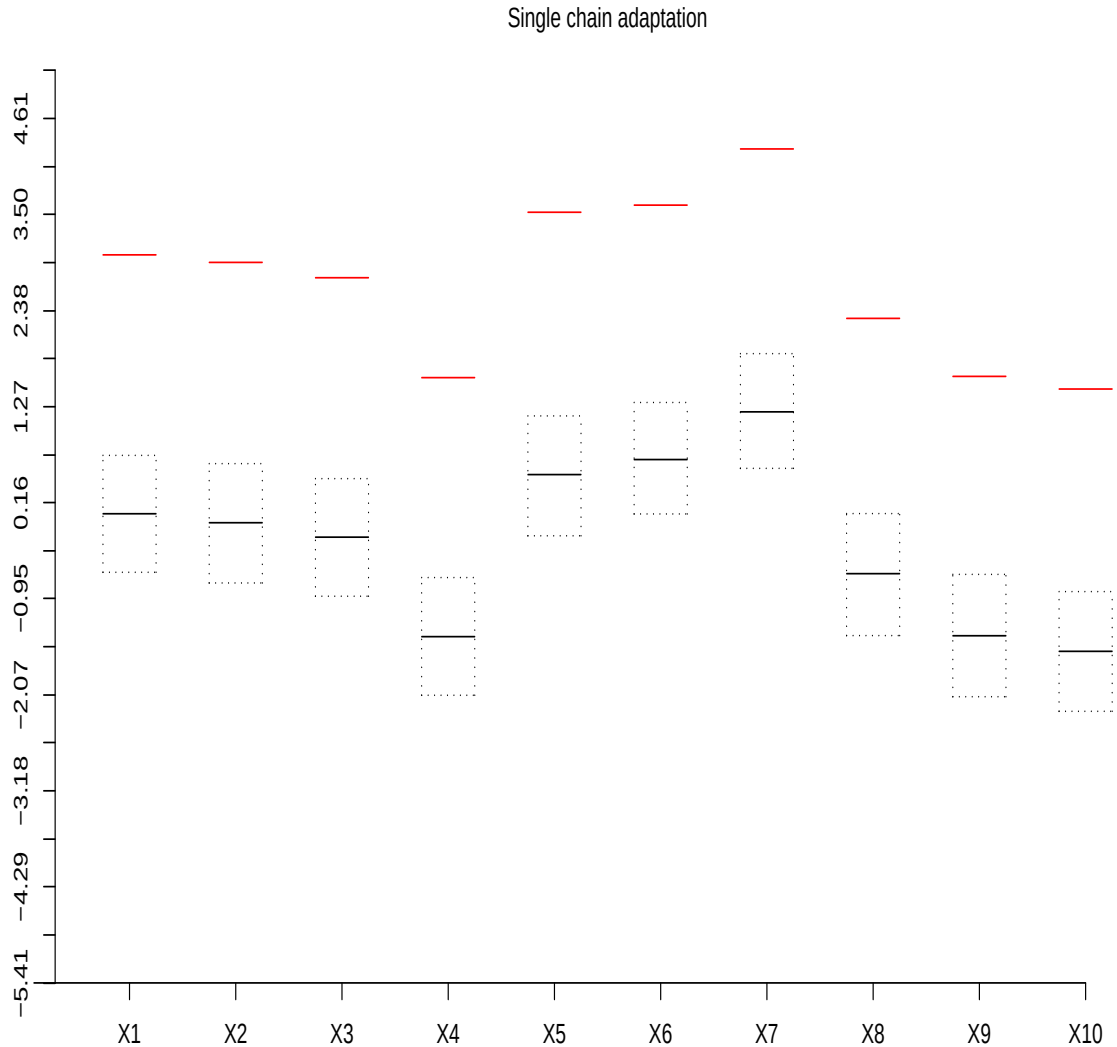
then $D_m \to 0$ using an argument similar to that used by HST or R&R since $\mathcal{H}_{m+1}$ differs from $\mathcal{H}_m$ by $O(m^{-1})$.

- Joint adaptive ergodicity is implied by marginal adaptive ergodicity since $\forall x, y \in \mathcal{S}$

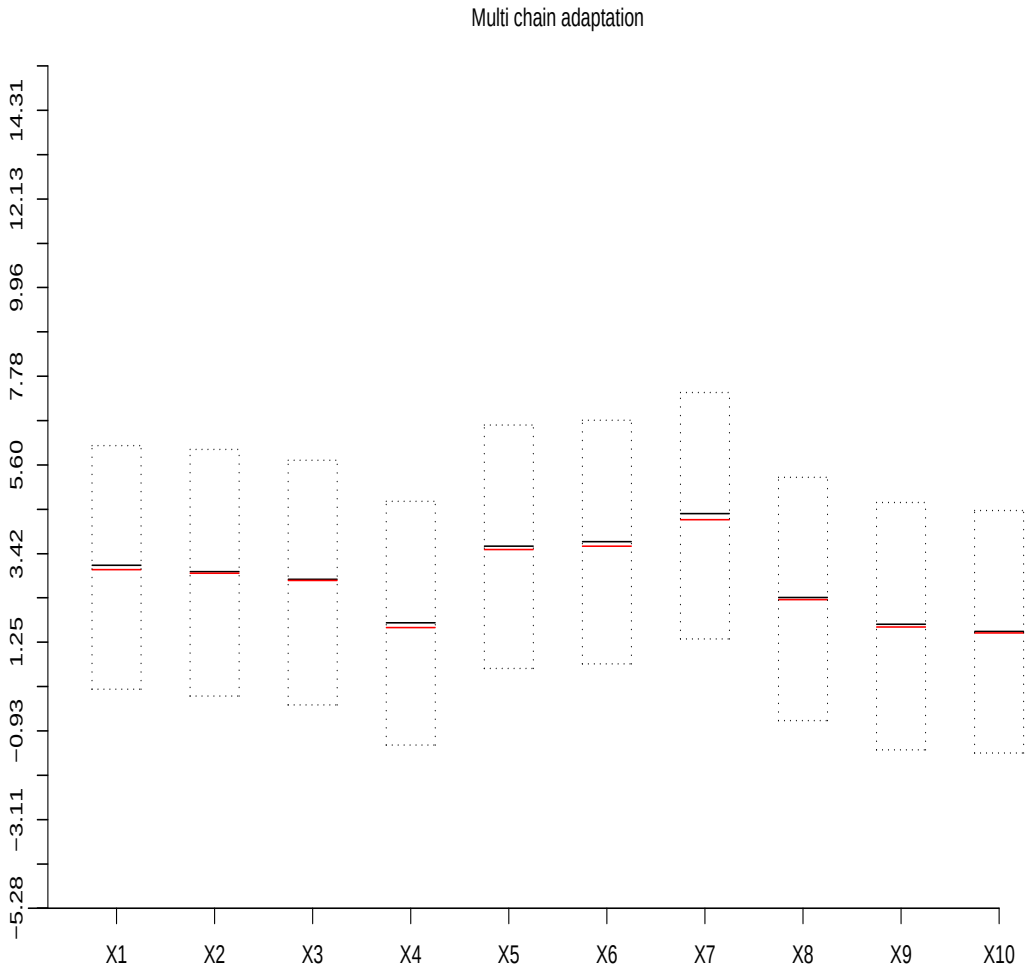
$$\begin{aligned} & \sup_{A,B} || \tilde{T}_m(x, y; A \times B) - \pi(A)\pi(B)|| = \\ = \quad & \sup_{A,B} || T_m(x, A) T_m(y, B) - \pi(A)\pi(B)|| \leq \\ \leq \quad & \sup_{A} || T_m(x, A) - \pi(A)|| + \sup_{B} || T_m(y, B) - \pi(B)|| \to 0. \end{aligned}$$

Example Revisited

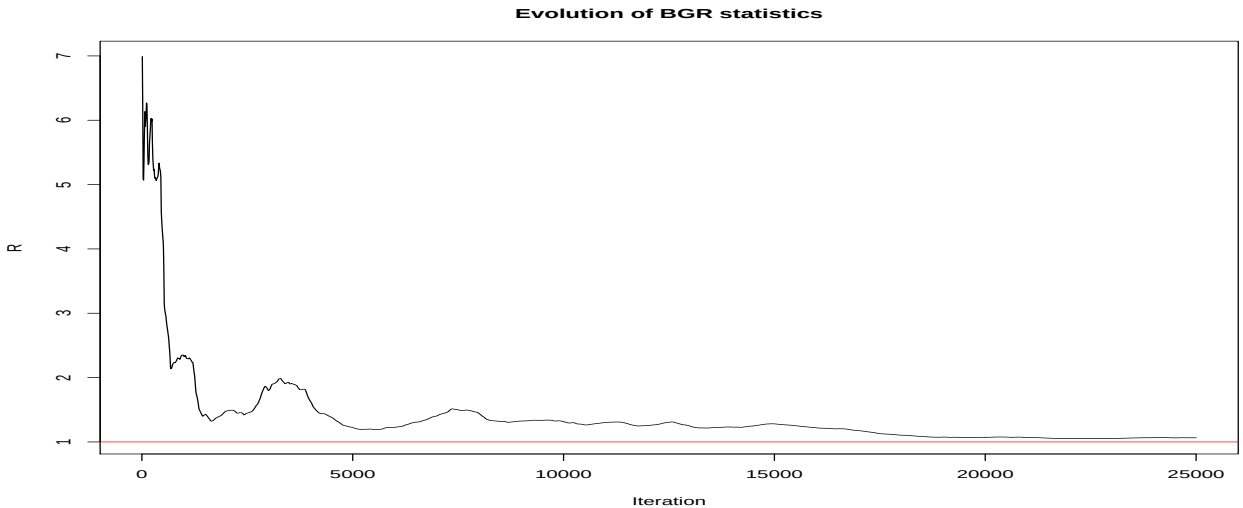
$$\pi(x) = 0.5n_{10}(x; \mu_1, \Sigma_1) + 0.5n_{10}(x; \mu_2, \Sigma_2),\ N = 250,000.$$



Example Revisited



Alternative Implementation: Collapsible INCA

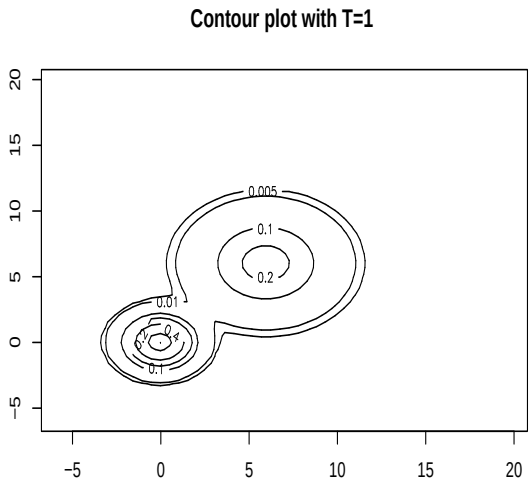


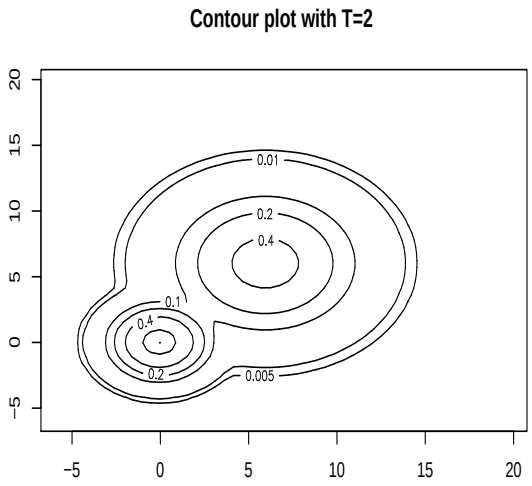
- Once the BGR diagnostic statistic has stabilised around 1, we could collapse all chains into one.
- Only one of the K chains continues to run but its past history is enriched using the K past histories.
- BGR is not used here as a convergence indicator but rather as a measure of the amount of information exchanged between the parallel chains.

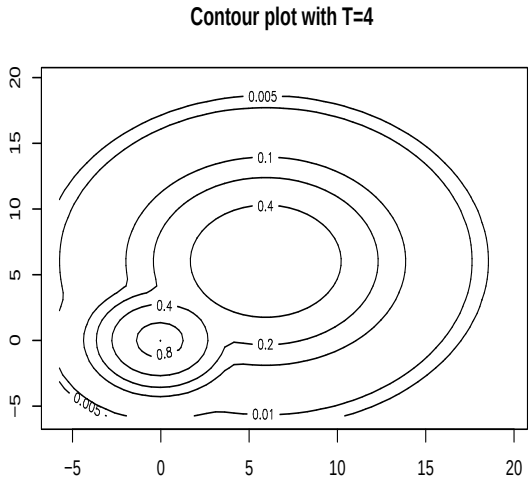
Tempered INCA (TINCA)

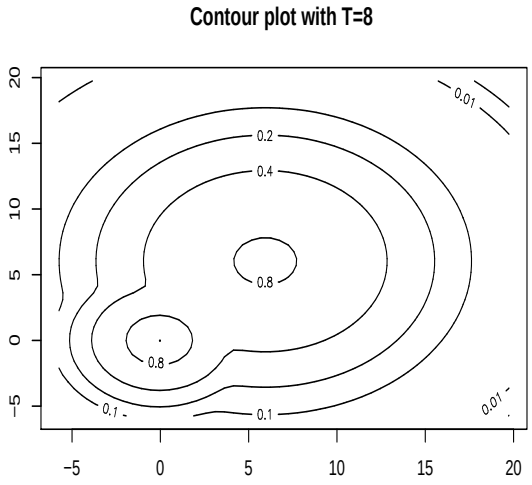
- Natural extension of the INCA idea: run parallel chains at higher temperature T.
- Perform Adaptation while simultaneously "cooling off".
- Gradual learning can be sped-up at higher-than-normal temperatures.

Tempered distributions









Tempered dist'ns: $\pi_T(x) = \pi(x)^{1/T}$

TINCA cont'd

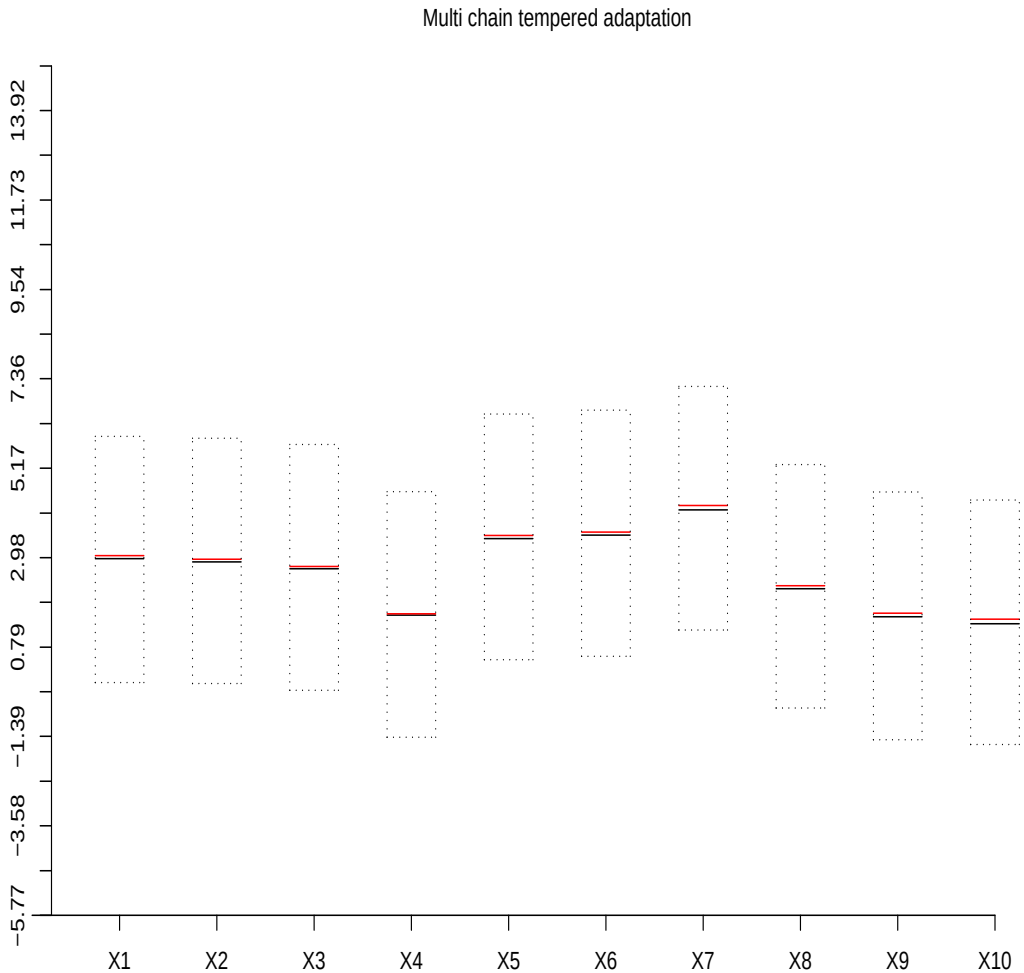
- Suppose we use tempered target distribution $\pi_\eta = \pi^{1/\eta}$, $\eta \in \{1, \ldots, T\}$.

Step I For each temperature η we run INCA until the BGR diagnostic stabilises around 1.

Step II We decrease η to $\eta - 1$ and redo Step I.

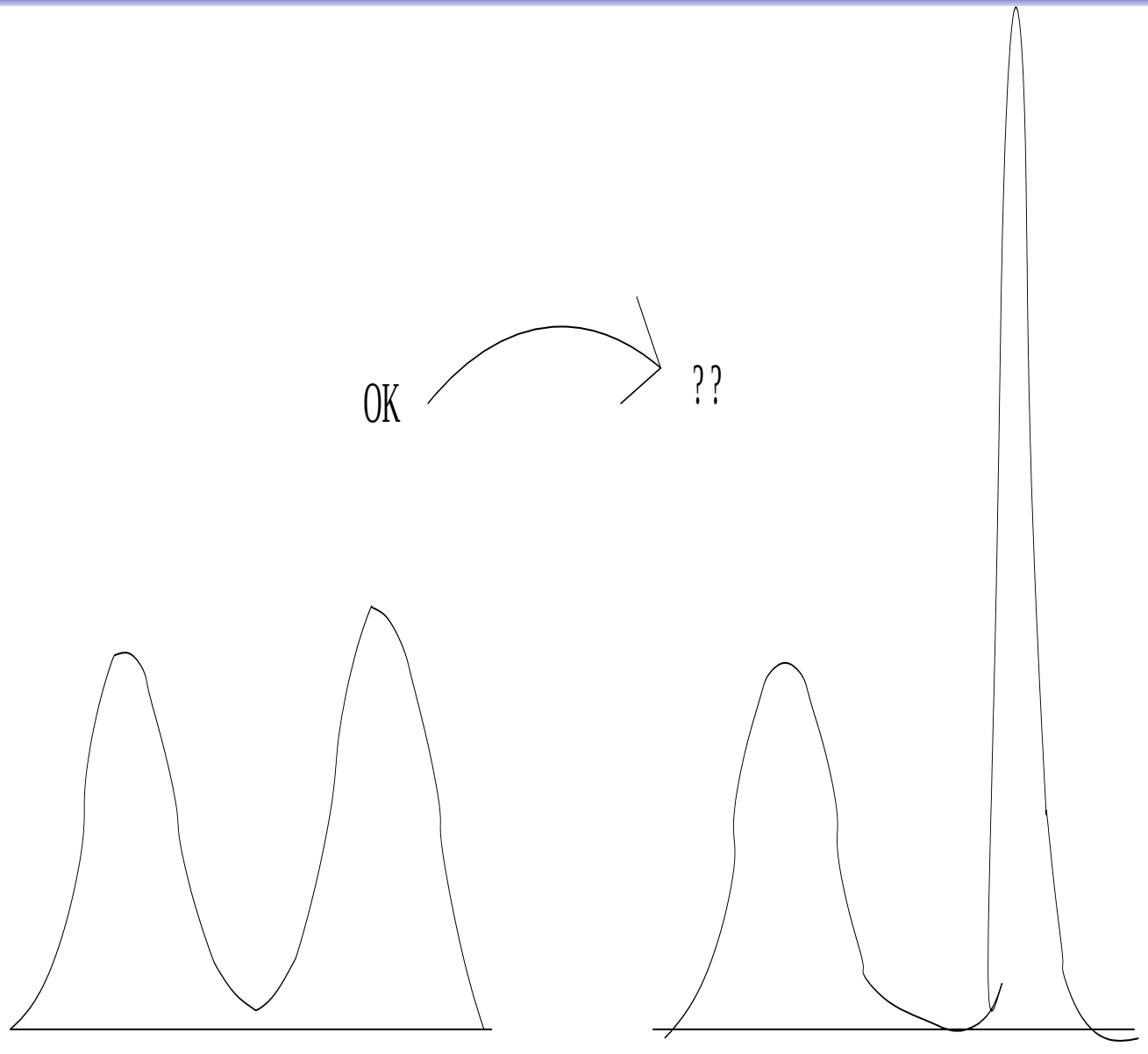
- The kernel learned at temperature η is assumed to be "reasonable" for initial sampling at temperature $\eta - 1$.
- This can be particularly effective when it is difficult to concoct a reasonably good initial proposal.

TINCA: Example Revisited



We used 5 chains and $T \in \{16, 8, 4, 2, 1\}$. Number of iterates needed for $R \leq 1.1$: $1200 + 420 + 1000 + 880 + 6300 \approx 10000$.

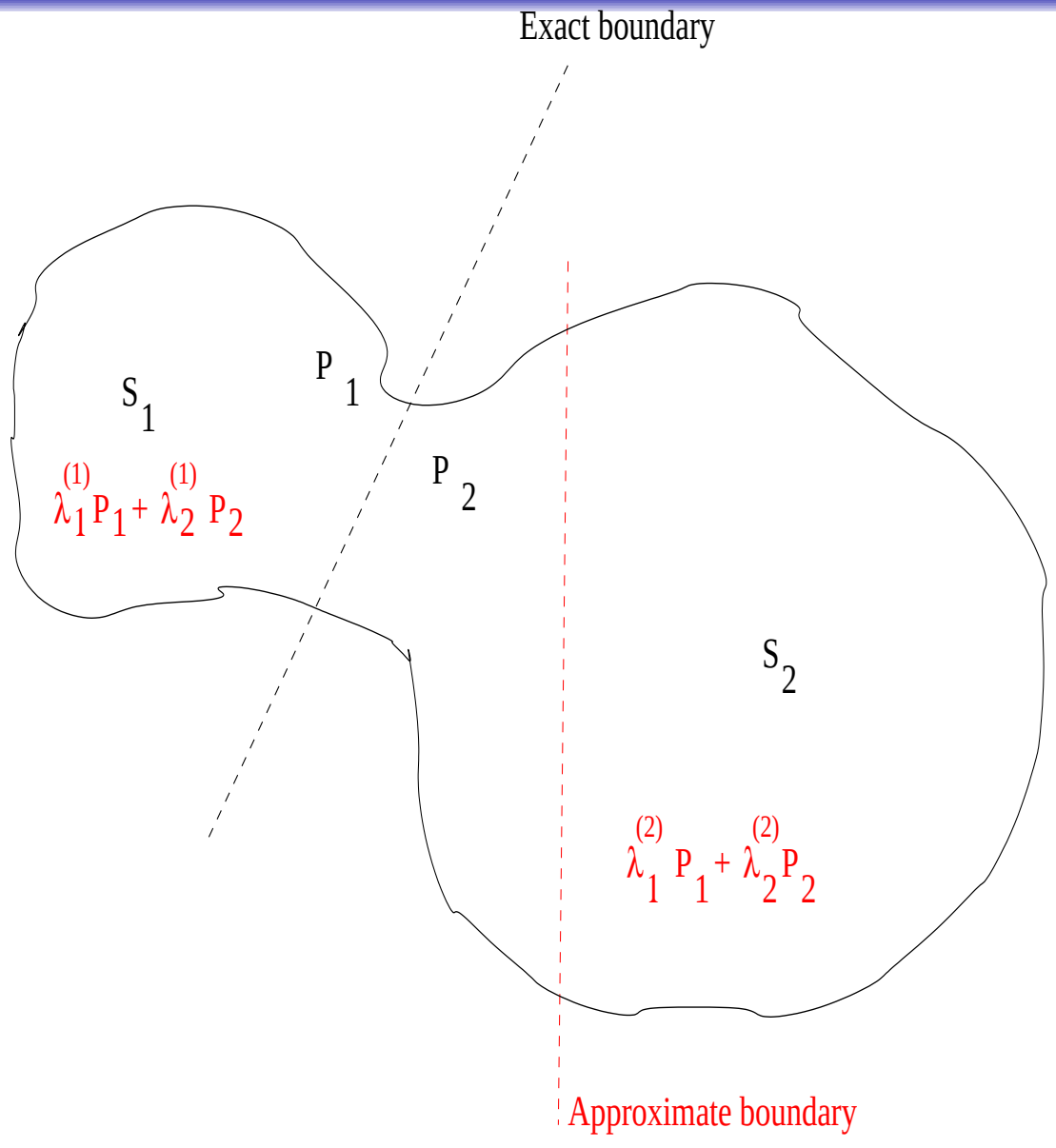
Misleading?



Antagonistic Learning (ANTLER)

- Consider sampling using RWM from a distribution π with support in $\mathcal{S} \subset R^d$.
- Suppose $\mathcal{S} = \mathcal{S}_1 \uplus \mathcal{S}_2$ is such that depending on whether the current value of the chain is in $\mathcal{S}_1$ or in $\mathcal{S}_2$ the optimal variance of the RWM proposal is different.
- We can construct examples where by collapsing the samples from the two regions we adaptively evolve towards a variance that is unsuitable for both $\mathcal{S}_1$ and $\mathcal{S}_2$.
- What type of adaptive algorithms can we design to remedy this problem?

RAPT


Exact boundary
S 1
P 1
P 2
S 2
$\lambda_1^{(1)} P_1 + \lambda_2^{(1)} P_2$
$\lambda_1^{(2)} P_1 + \lambda_2^{(2)} P_2$
Approximate boundary

Regional Adaptation (RAPT)

- Simplest scenario: assume that the regions $\mathcal{S}_1, \mathcal{S}_2$ are known. Then we can perform regional adaptation within each region (see also R & R).
- Requires some care in designing the adaptive algorithm. If we use proposal P_1 in $\mathcal{S}_1$ and P_2 in $\mathcal{S}_2$, respectively, then the acceptance ratio is (see also R & R's regional adaptation).

$$r(x, x_{new}) = \begin{cases} \frac{\pi(x_{new})}{\pi(x)} & \text{, if } x, x_{new} \in \mathcal{S}_i \\ \frac{\pi(x_{new}) p_1(x|x_{new})}{\pi(x) p_2(x_{new}|x)} & \text{, if } x \in \mathcal{S}_2, x_{new} \in \mathcal{S}_1 \\ \frac{\pi(x_{new}) p_2(x|x_{new})}{\pi(x) p_1(x_{new}|x)} & \text{, if } x \in \mathcal{S}_1, x_{new} \in \mathcal{S}_2. \end{cases}$$

Regional Adaptation (RAPT)

- Hard: find a way to determine the partition $\mathcal{S} = \mathcal{S}_1 \uplus \mathcal{S}_2$ (or $\mathcal{S} = \mathcal{S}_1 \uplus \mathcal{S}_2 \uplus \mathcal{S}_3 \uplus \ldots$)
- Short of that, we can allow for some uncertainty regarding the distribution to be used in each $\mathcal{S}_i$. i.e. we sample from a mixture of proposals.
- The mixture proportions are allowed to vary between regions and are adaptively adjusted based on the past realisations.
- In addition, the distributions entering the mixture are also adapted based on past realisations.

RAPT- Regions are approximated

- In each region $\mathcal{S}_j$ we sample using the proposal

$$\tilde{P}(X_t, \cdot) = \sum_{i=1}^{2} \lambda_i^{(j)} P_i(X_t, \cdot), \; j = 1, 2.$$

- Each P_i is adapted using samples from $\mathcal{S}_i$.
- The mixture weights $\lambda_i^{(j)}(t)$ are also adapted.

RAPT (con'd)

- For instance, $\lambda_i^{(j)} = \frac{n_i^{(j)}(t)}{\sum_{h=1}^{K} n_h^{(j)}(t)}$ and

$$n_i^{(j)}(t) = \#\{ \text{accepted moves up to time } t \text{ when the proposal dist'n is } P_i \text{ and the state of the chain is in } \mathcal{S}_j \}.$$

- Will tend to favor proposals with high acceptance rates; these are usually the ones creating "small jumps" and thus not necessarily the best for our purpose.

RAPT (con'd)

- Alternatively,

$$\lambda_i^{(j)} = \frac{d_i^{(j)}(t) n_i^{(j)}(t)}{\sum_{h=1}^{K} d_h^{(j)}(t) n_h^{(j)}(t)}$$

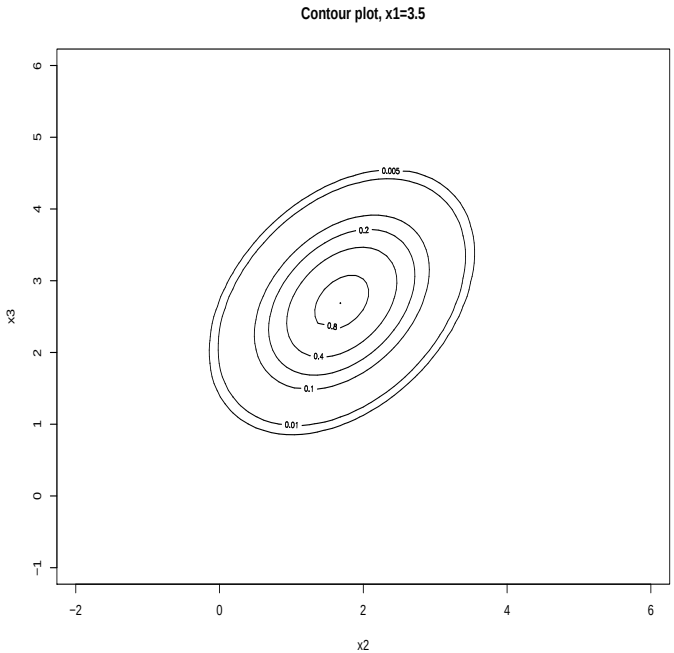
where,

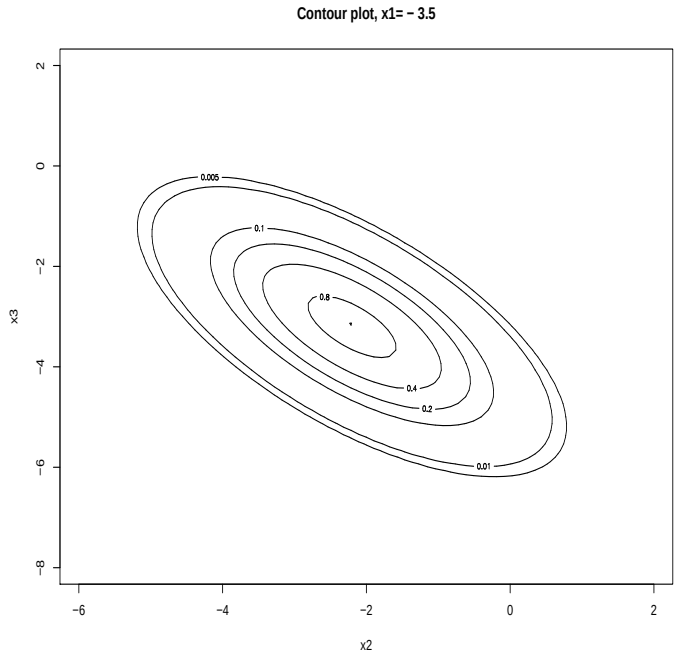
$d_i^{(j)}(t)$ = average square root jump distance up to time t when the proposal dist'n is P_i and the state of the chain is in $\mathcal{S}_j$.

- One could create more complicated weights based also on the "landing place" of the proposal, e.g. whether modes have been switched, how often, etc.

ANTLER: A simple example

$\pi(x) = 0.5n_3(x|\mu, \Sigma) + 0.5n_3(x|\mu', \Sigma')$ with $\mu = (4, 2, 3)^T$, $\mu' = (-4, -2, -3)^T$ and $\Sigma_{ii} = 0.5$, $\Sigma_{ij} = 0.3$, $\Sigma'_{ii} = 1$, $\Sigma'_{ij} = -0.4$.

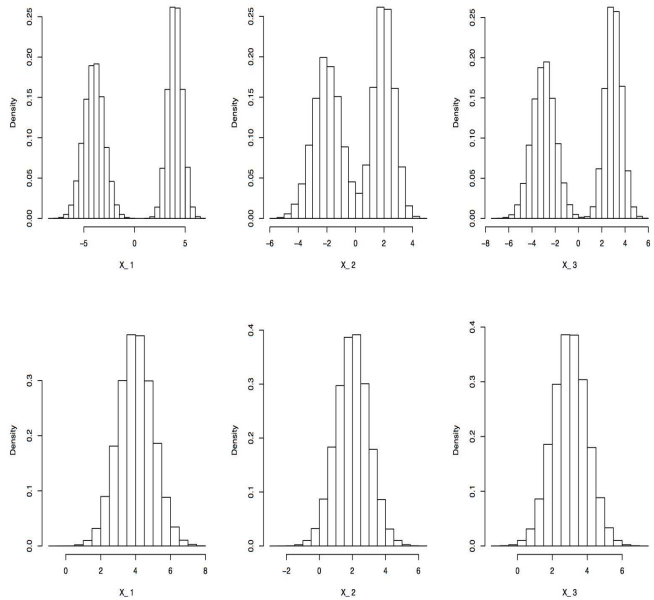




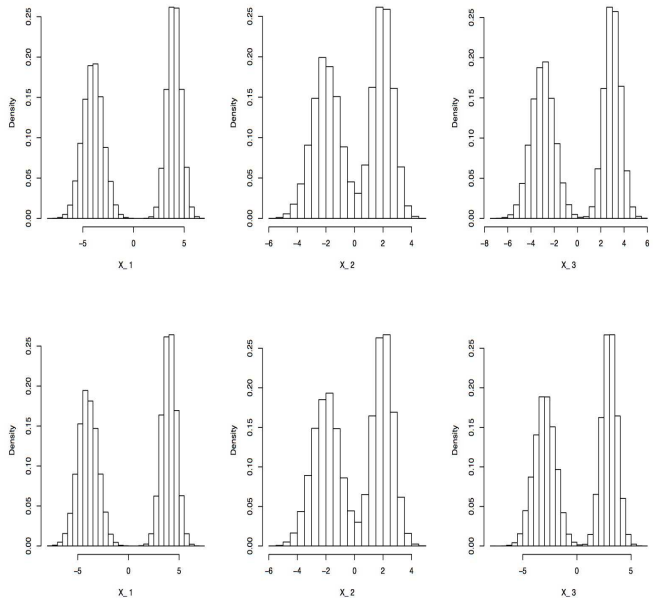
Example Revisited: Regions are known

Define $\mathcal{S}_1 = \{(x_1, x_2, x_3) : x_1 < 0\}$ and $\mathcal{S}_2 = \bar{\mathcal{S}}_1$.

Global Adaptation



Regional Adaptation



After 250,000 iterations the regional adaptation algorithm has produced

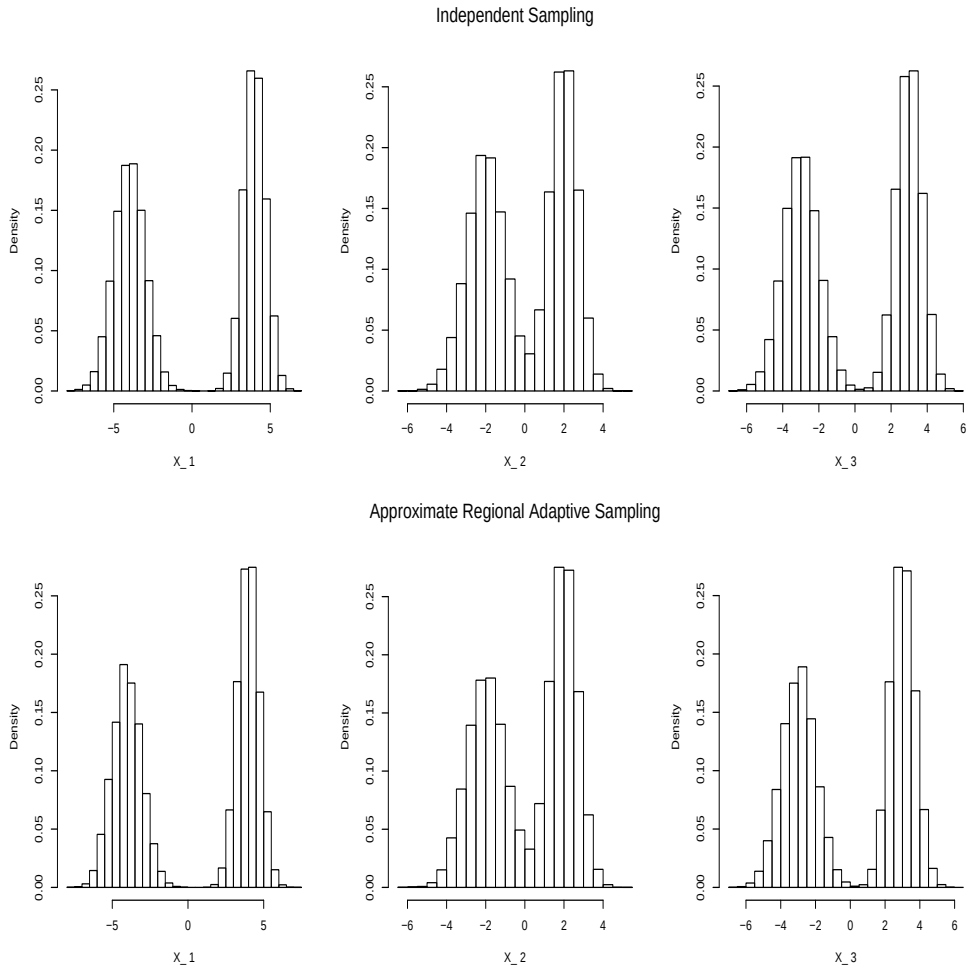
$$\hat{\Sigma} = \begin{pmatrix} 0.506 & 0.303 & 0.303 \\ 0.303 & 0.499 & 0.301 \\ 0.303 & 0.301 & 0.498 \end{pmatrix}, \hat{\Sigma}' = \begin{pmatrix} 0.974 & -0.389 & -0.392 \\ -0.389 & 0.982 & -0.395 \\ -0.392 & -0.395 & 0.986 \end{pmatrix}.$$

The non-regional one:

$$\hat{\Sigma} = \begin{pmatrix} 8.454 & 3.990 & 5.982 \\ 3.990 & 2.529 & 3.018 \\ 5.982 & 3.018 & 5.044 \end{pmatrix}.$$

Simulation Results - Approximate Regional Adaptation

- Define $\mathcal{S}_1 = \{(x_1, x_2, x_3) : x_1 < 1.5\}$ and $\mathcal{S}_2 = \bar{\mathcal{S}}_1$.
- Use the acceptance ratio based mixture weights.



Simulation Results

Using the acceptance ratio-based weights after $506,000$ samples the two sample variances in the two regions are:

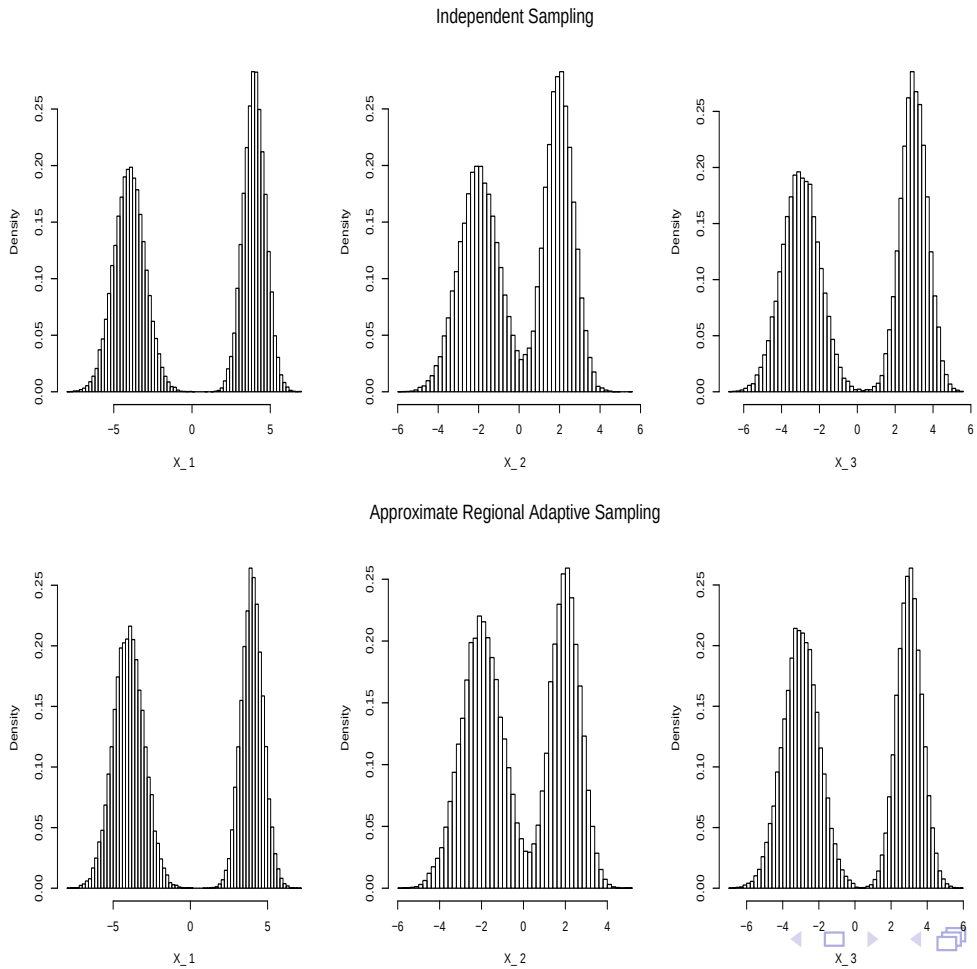
$$\hat{\Sigma} = \begin{pmatrix} 0.502 & 0.299 & 0.304 \\ 0.299 & 0.506 & 0.304 \\ 0.304 & 0.304 & 0.507 \end{pmatrix}, \hat{\Sigma}' = \begin{pmatrix} 0.956 & -0.393 & -0.354 \\ -0.393 & 1.017 & -0.417 \\ -0.354 & -0.417 & 0.971 \end{pmatrix}.$$

$$\lambda_1^{(1)} = 0.96,$$

$$\lambda_1^{(2)} = 0.76.$$

Simulation Results - Approximate Regional Adaptation

- Define $\mathcal{S}_1 = \{(x_1, x_2, x_3) : x_1 < 1.5\}$ and $\mathcal{S}_2 = \bar{\mathcal{S}}_1$.
- Use the acceptance ratio & square root mean jump distance mixture weights.



Simulation Results

Weights based on the acceptance proportion & square root mean square jump, $N = 506,000$, the two sample variances in the two regions are:

$$\hat{\Sigma} = \begin{pmatrix} 0.504 & 0.300 & 0.300 \\ 0.300 & 0.499 & 0.300 \\ 0.300 & 0.299 & 0.497 \end{pmatrix}, \hat{\Sigma}' = \begin{pmatrix} 1.010 & -0.405 & -0.402 \\ -0.405 & 1.003 & -0.398 \\ -0.402 & -0.398 & 1.012 \end{pmatrix}.$$

$$\lambda_1^{(1)} = 0.48,$$

$$\lambda_1^{(2)} = 0.12.$$

Conclusions & Discussion

- INCA can be used not only for Random Walk Metropolis but also for other MCMC algorithms, e.g. Independence Metropolis.
- INCA can also be implemented with other adaptive schemes, such as the kernel method of Giordani and Kohn (2006).
- RAPT should be used in combination with INCA as adaptation within each region does not ensure good traffic between regions.
- One possible approach is to augment the mixture to include an additional kernel who is adapted using realisations from all the regions.
- INCA and RAPT could and probably should be used together for improved efficiency.
- More complex examples need to be studied to better understand the strengths and weaknesses of the methods proposed here.