Adaptive Incremental Mixture MCMC

Florian Maire, UCD & Insight

joint work with: Nial Friel (UCD & Insight)
Adrian Raftery (University of Washington)
Antonietta Mira (University of Lugano)

Trinity Statistics Seminar, 21th of November 2014

Outlines

General context

Let π be a probability distribution defined on $(\mathsf{X}, \mathcal{X})$, $\mathsf{X} \subseteq \mathbb{R}^d$

Aim: getting samples $(X_1, \ldots, X_n) \sim \pi$

to: estimate any expectation $\mathbb{E}_\pi[h(X)]$

Given that π might be:

- complicated – not belonging to usual families
- high-dimensional – Bayesian inference with $X \equiv \theta$, $d \gg 1$
- multi-modal – clustering problems
- ...

but we assume that π is known (possibly up to a constant)...

Terminology

Two main *universal* approaches sharing a common philosophy

Particle based methods

A set of particles sampled from an instrumental density:

$$(X_1, \ldots, X_n) \sim_{i.i.d.} Q$$

weighted by an importance function

$$W_k \propto W(X_k) = \frac{\pi(X_k)}{Q(X_k)}$$

concern: weight degeneration

Markov chain based methods

A Markov chain $\{X_k,\ k \in \mathbb{N}\}$ with proposal

$$\tilde{X} \sim P(X_k, \cdot)$$

accepted / rejected with probability

$$\alpha(X_k, \tilde{X}) = 1 \wedge \frac{\pi(\tilde{X})P(\tilde{X}, X_k)}{\pi(X_k)P(X_k, \tilde{X})}$$

concern: bad mixing

Asymptotic estimate

To be useful, these methods must come with asymptotic guarantees *e.g* CLT:

$$\sqrt{n}\left(\hat{\mu}_h(X_1,\ldots,X_n) - \mathbb{E}_\pi\left[h(X)\right]\right) \xrightarrow{P} \mathcal{N}(0, \hat{\sigma}_h^2(X_1,\ldots,X_n))$$

Particle based methods

$$\hat{\mu}_h(X_1,\ldots,X_n) = \frac{\sum_{k=1}^n W(X_k)h(X_k)}{\sum_{k=1}^n W(X_k)}$$

and the variance write:

$$\hat{\sigma}_h^2(X_1,\ldots,X_n) = \mathbb{E}_\pi\left[W(X)(h(X) - \mathbb{E}_\pi\left[h(X)\right])^2\right]$$

Markov chain based methods

$$\hat{\mu}_h(X_1,\ldots,X_n) = \frac{1}{n}\sum_{k=1}^n h(X_k)$$

$$\hat{\sigma}_h^2(X_1,\ldots,X_n) = \mathrm{Var}_\pi[h(X)]\left(1 + 2\sum_{k=1}^\infty \rho_k\right)$$

Independence sampler: an "hybrid" case

A Markov chain $\{X_k,\, k \in \mathbb{N}\}$ with proposal distribution

$$\tilde{X} \sim Q \qquad \text{(independent of } X_k\text{!)}$$

accepted / rejected w.p.

$$\alpha(X_k, \tilde{X}) = 1 \wedge \frac{\pi(\tilde{X})Q(X_k)}{\pi(X_k)Q(\tilde{X})} = 1 \wedge \frac{W(\tilde{X})}{W(X_k)}$$

$\Rightarrow$ Apparently of limited interest as we lose the local exploration offered by the markovian proposal...

But, if ever $Q \approx \pi$, $\tilde{X}$ will be accepted w.p. $\to 1$

$\Longrightarrow \mathrm{Cov}(h(X_k), h(X_{k+1})) \to 0\,(X_k \perp\!\!\!\perp X_{k+1}) \Longrightarrow \hat{\sigma}_h^2(X_1, \ldots, X_n) \to \mathrm{Var}_\pi[h(X)]$

Outlines

How to get $Q \approx \pi$?

Adaptive algorithms:

- Adaptive Importance sampling: AMIS, Adaptive Multiple Importance Sampling (Cornuet et al, 2012), IMIS, Incremental Mixture of Importance Sampling (Raftery and Le Bao, 2010), ...
- Adaptive Markov chain Monte Carlo: Adaptive Proposal (Haario et al, 1999), Adaptive Metropolis (Haario et al, 2001), Delayed Rejection Adaptive Metropolis (Haario et al, 2006), ...

Idea: use the knowledge of the past particles / states to define a sequence of instrumental kernels $\{Q_k,\, k \in \mathbb{N}\}$ such that $Q_k \to \pi$...

Bad aspect: Adaptive MCMC methods lose most of the *nice* theory that MCMC rely on

$\Rightarrow$ new arguments required to prove stationarity, reversibility, ergodicity...

Adaptive MCMC

- Most frequent approach:
  - the proposal distribution Q_k belongs to a parameterized family:

$$Q_k \equiv Q_{\theta_k}\,, \qquad (\theta_k \in \Theta \subseteq \mathbb{R}^m)$$

  - samples of the chain are recursively used to build a sequence of parameter $\{\hat{\theta}_k,\, k \in \mathbb{N}\}$ *optimizing* a criterion (acceptance rate, moment matching with π, Kullback-Leibler minimization w.r.t. π...)
  - also connected with the EM literature

$\Rightarrow$ issue: needs some *apriori* knowledge of π to chose a reasonable Q_θ & *constrain* a bit the adaption

- Some non-parametric approaches have also been proposed
  - Interpolation of a set of support point to match π

$\Rightarrow$ issue: seems to struggle in dimension higher than 1

Adaptive IS: Incremental Mixture of Importance Sampling, Raftery and Le Bao, 2010–motivations

IMIS aims at building a collection of Gaussian kernels $\{\phi_1, \ldots, \phi_m\}$ s.t.

$$R_m = (1/m) \sum_{\ell=1}^{m} \phi_\ell \approx \pi$$

- A particle X_k with an high importance weight W_k highlights a region of the support lacking of particles ($W_k = W(X_k)/\sum_{j=1}^{n} W(X_j)$)
- By construction, IMIS recursively populates with new batch of particles these regions
- This is achieved by specifying a Gaussian distribution ϕ_m *covering* this part of the support (and then populating by sampling from ϕ_m)

IMIS – initiation

IMIS starts with a *naive* / *defensive* / *flat* distribution Q, very uninformative w.r.t. π...
First steps:

(i) samples $(X_1, \ldots, X_{N_0}) \sim Q$ (an instrumental kernel)

(ii) set $I = \arg\max_{j \in \{1,\ldots,N_0\}} W_j$, where W_j is the j-th particle IS weight

(iii) set $\phi_1 = \mathcal{N}(\mu_1, \Sigma_1)$ where

$$\mu_1 = X_I, \qquad \Sigma_1 = \frac{1}{|\mathfrak{N}(X_I)| - 1} \sum_{x \in \mathfrak{N}(X_I)} (x - \mu_1)(x - \mu_1)^\mathsf{T}$$

$\mathfrak{N}(X_I)$ denoting a neighborhood of X_I

IMIS – key point

The key point is to resample N new particles through

$$(X_{N_0+1}, \ldots, X_{N_0+N}) \sim \phi_1$$

and to regard the $N_0 + N$ particles $\{X_1, \ldots, X_{N_0+N}\}$ as being iid realizations of the proposal mixture

$$Q_1 = (1/2)Q + (1/2)\phi_1$$

the following reweighting step for all $j \in \{1, \ldots, N_0 + N\}$

$$W_1(X_j) \propto \frac{\pi(X_j)}{\omega_1 Q(X_j) + (1-\omega_1)\phi_1(X_j)}, \qquad \omega_k = \frac{N_0}{N_0 + kN}$$

keeps the weighted particles target π.

IMIS – iterations

At iteration k, we have the particles $\{X_1, \ldots, X_{(k-1)N+N_0}\}$ and the collection of Gaussian kernels $\{\phi_1, \ldots, \phi_k\}$

(i) simulate N new particles $\{X_{(k-1)N+N_0+1}, \ldots, X_{kN+N_0}\} \sim \phi_k$

(ii) reweight all the particles for all $j \in \{1, \ldots, kN + N_0\}$

$$W_k(X_j) \propto \frac{\pi(X_j)}{N_0 Q(X_j) + N\left(\phi_1(X_j) + \ldots + \phi_k(X_j)\right)}$$

(iii) get the next Gaussian kernel $\phi_{k+1} = \mathcal{N}(\mu_{k+1}, \Sigma_{k+1})$ where $I = \arg\max_{j \in \{1,\ldots,kN+N_0\}} W_k(X_j)$

$$\mu_{k+1} = X_I, \qquad \Sigma_{k+1} = \frac{1}{|\mathfrak{N}(X_I)| - 1} \sum_{x \in \mathfrak{N}(X_I)} (x - \mu_{k+1})(x - \mu_{k+1})^{\mathsf{T}}$$

$\mathfrak{N}(X_I)$ denoting a neighborhood of X_I

Perspectives

- IMIS is "Fully adaptive" in a sense that it follows the generation of particles (limited apriori knowledge required)
- Main issue: an ever increasing set of particles $X_1, \ldots, X_{kN+N_0}$
- when $k \gg 1$ the reweighting step starts to be prohibitively slow...
- (In practice, IMIS is combined with an optimization step which maps the state space beforehand – this fastens convergence and a reasonable limited number of iterations is then achievable)

Our question:

- Is it possible to derive an "MCMC" equivalent to the IMIS Methodology?

Motivation

- the sequential nature of the chain well suited to a *long* exploration

Outlines

Incremental Mixture MCMC: unfolding IMIS

- A sequence of random variable $\{X_n, n \in \mathbb{N}\}$

$$\begin{cases} X_0 \sim Q \\ X_{n+1} \sim K_n(\cdot) \end{cases} \tag{1}$$

where K_n is a M–H kernel with independent proposal dist.

$$Q_n = \omega_n Q + (1-\omega_n)\frac{1}{m_n}\sum_{\ell=1}^{m_n}\phi_\ell$$

and acceptance $\alpha_n(X_n, \tilde{X}) = 1 \wedge \frac{W_n(\tilde{X})}{W_n(X_n)}$

- is meant to emulate the population $\{X_n, n \in \mathbb{N}\}$ (obtained after k generations of IMIS),

$$(X_1, \ldots, X_n) \sim \omega_k Q + (1-\omega_k)\frac{1}{k}\sum_{\ell=1}^{k}\phi_\ell, \qquad (n = kN + N_0)$$

weighted by the function $W_k(x)$

Building on the analogy

- 1 iteration of IMIS $\Rightarrow$ 1 new component ϕ_ℓ
- But, we want 1 iteration IMMCMC $\not\Rightarrow$ 1 new component ϕ_ℓ!
- for a better match, a new kernel should be designed after N iterations

But

- Adding a component every N iteration, deterministically sounds odd:

$\Rightarrow$ What if X_N lies in an area well supported by Q_N?

- We rather suggest letting the incremental kernel develop stochastically:
- At iteration n, increase the mixture if

$$\{W_n(\tilde{X}) > W_n^*\}, \qquad \tilde{X} \sim Q_n$$

$\Rightarrow$ "add a component when it worths it"

Main challenge: How to choose W_n^*?

Two alternatives:

(1) constant $W_n^* = c$

- needs off line calibration
- too much dependency on the sample path
- instable

(2) bounded in probability by a sequence of parameter $\{\epsilon_n,\ n \in \mathbb{N}\}$

$$\mathbb{P}_n[\{W_n(\tilde{X}) > W_n^*\}] \leq \epsilon_n$$

- more control
- still how to define a suitable W_n^* achieving the bound in probability?

$$\mathbb{P}_n[\{W_n(\tilde{X}) > W_n^*\}] = \int_{\mathsf{X}} \mathbb{1}_{\{W_n(x) > W_n^*\}}(x) Q_n(\mathrm{d}x)$$

$$\approx R^{-1} \sum_{k=1}^{R} \mathbb{1}_{\{W_n(X_k) > W^*\}}(X_k) \quad (2)$$

Example with a challenging target

We consider $\mathsf{X} = \mathbb{R}^2$

$$\pi(\mathrm{d}x) = \Phi(f_b(x), \mu, \Sigma)\mathrm{d}x\,,$$

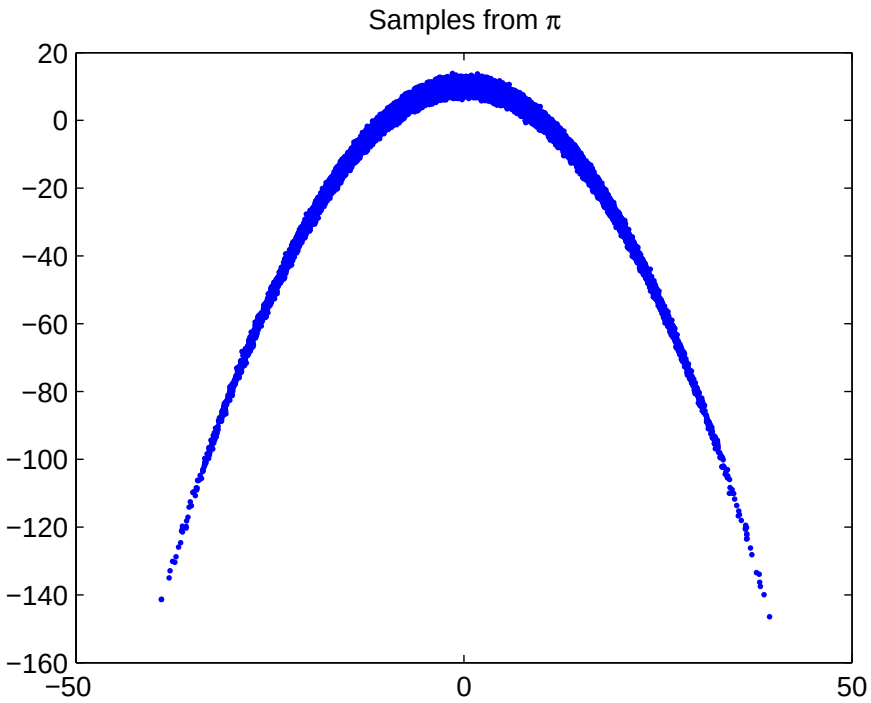
where

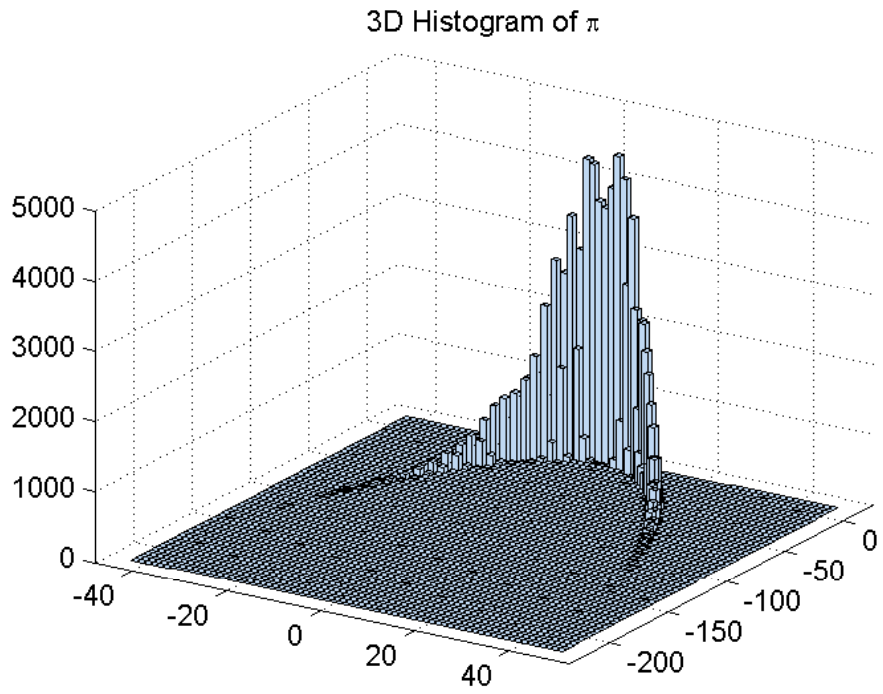
- $x \to \Phi(x, \mu, \Sigma)$ is the two-dimensional Gaussian density function with mean μ and covariance matrix Σ
- $f_b : \mathbb{R}^2 \to \mathbb{R}^2$ is the mapping defined by

$$f_b : \begin{pmatrix} x \\ y \end{pmatrix} \to \begin{pmatrix} x \\ y + bx^2 - 100b \end{pmatrix}$$

We have used $b = 0.1$, $\mu = [0, 0]$ and $\Sigma = \mathrm{diag}([100, 1])$

Banane shape distribution (or Twisted Gaussian)





$\Rightarrow$ Simulation

Samples of the two approaches

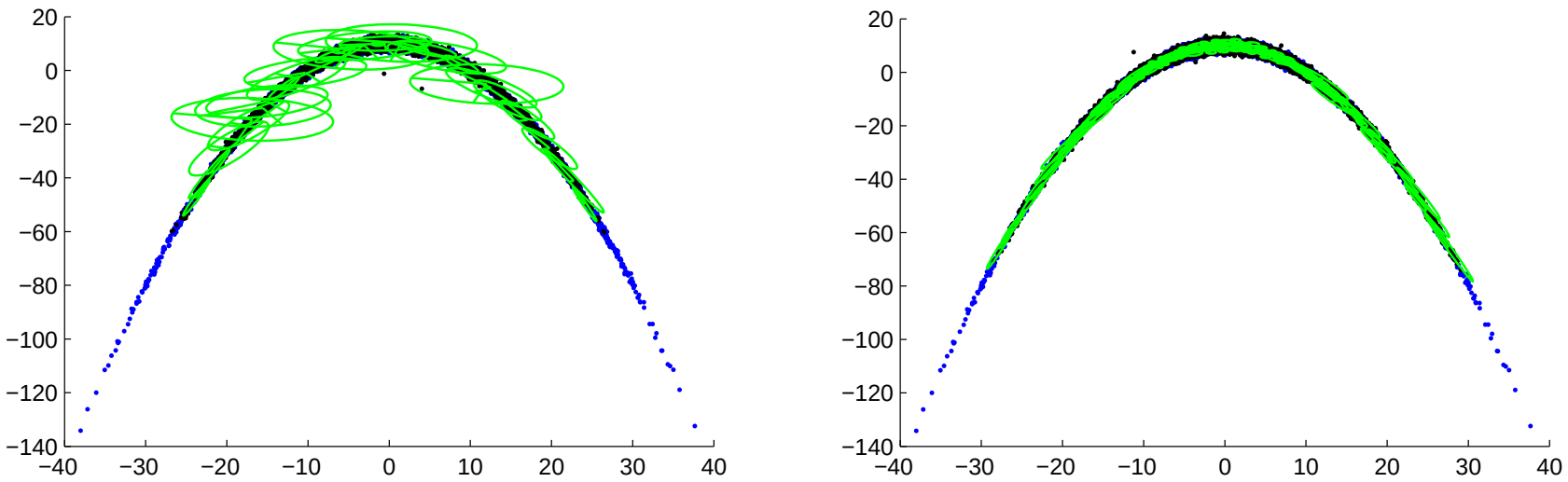
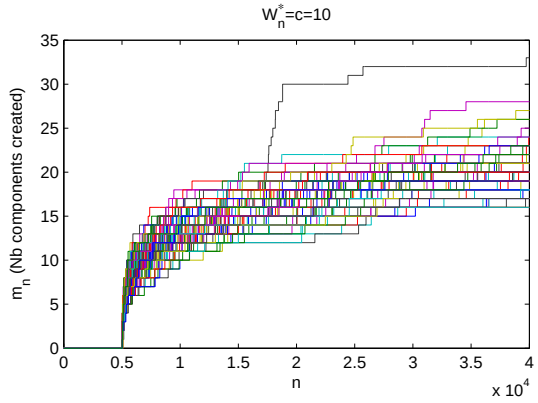


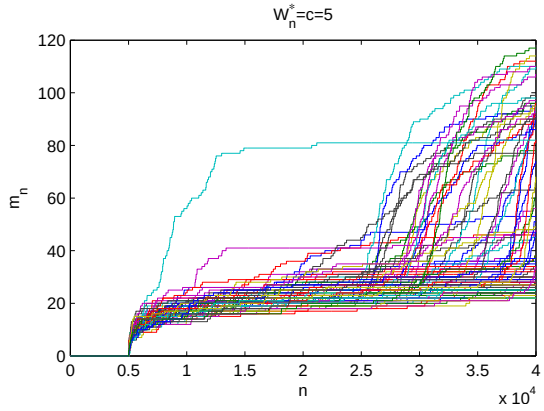
Figure: Support (blue), Sample paths (black), Confidence interval (0.5)
Incremental Components (green) :

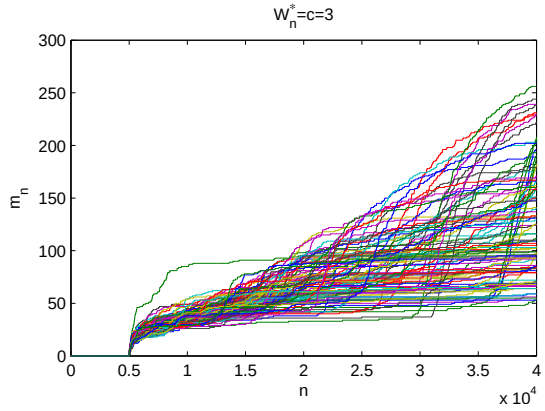
- Constant threshold (left)
- Bounded in proba. (right)

Incremental kernel design in simulation - constant threshold

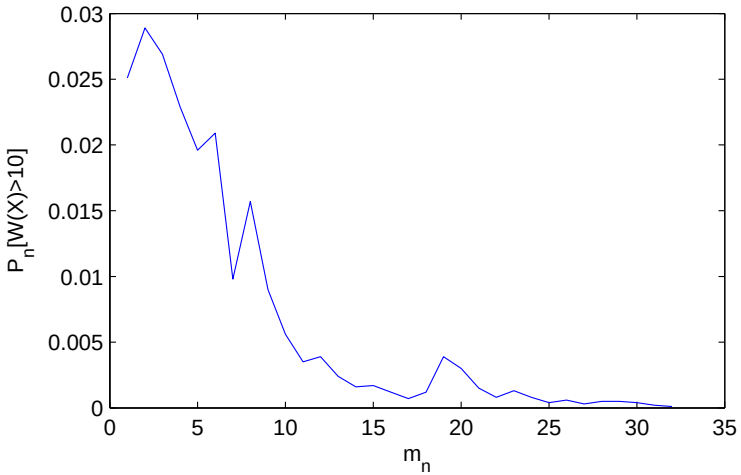
Constant thresholds $W_n^* = c$ (100 independent run of 40,000 it.)







Estimated probability of adding a kernel for 1 run $c = 10$



Incremental kernel design in simulation - bound in proba

$$\mathbb{P}_n[W_n(X) > W^*] \leq 0.005$$

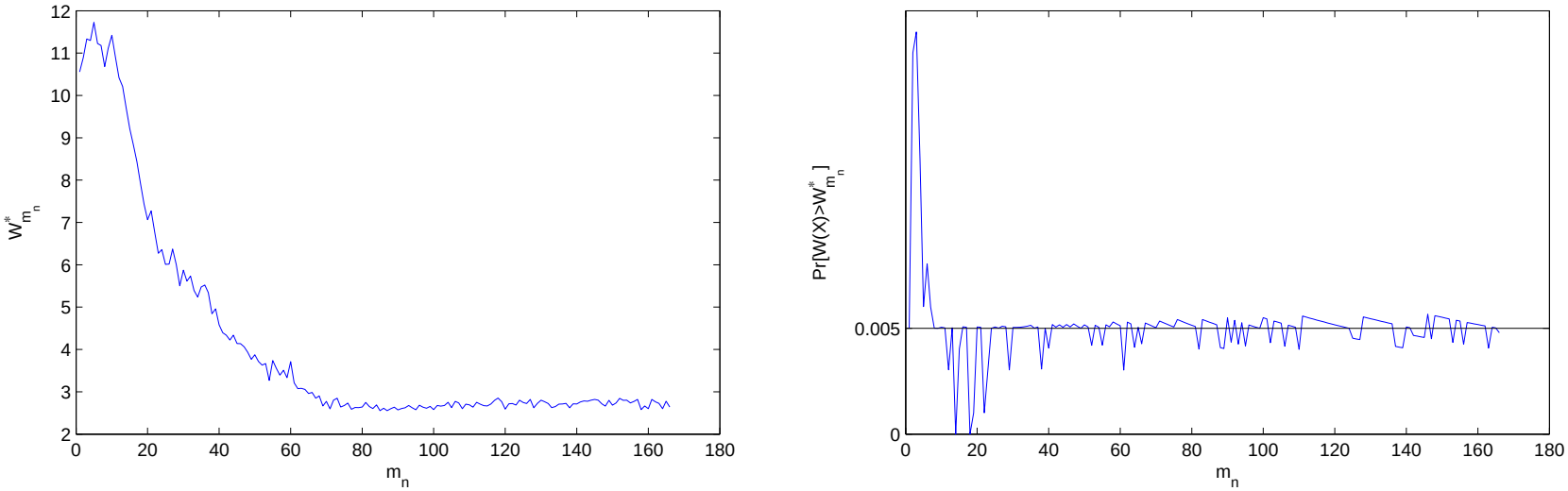


Figure: Threshold $W^*_{m_n}$ (left) and probability $\mathbb{P}_n[W_n(\tilde{X}_n) > W^*_n]$ (right)

- much more consistent than using Markov bound
- after $n = 40,000$ it. $m_n = 166$ kernels designed, $(40,000 \times 0.005 = 200)$

Incremental kernel design in simulation - bound in proba

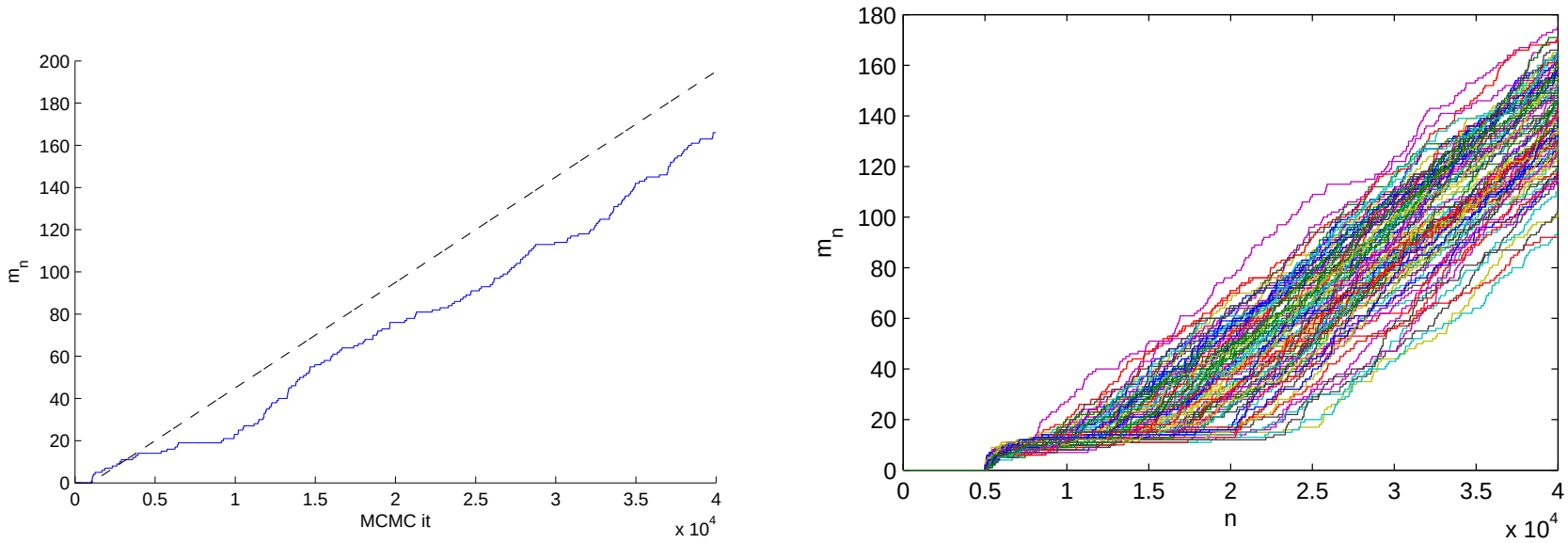
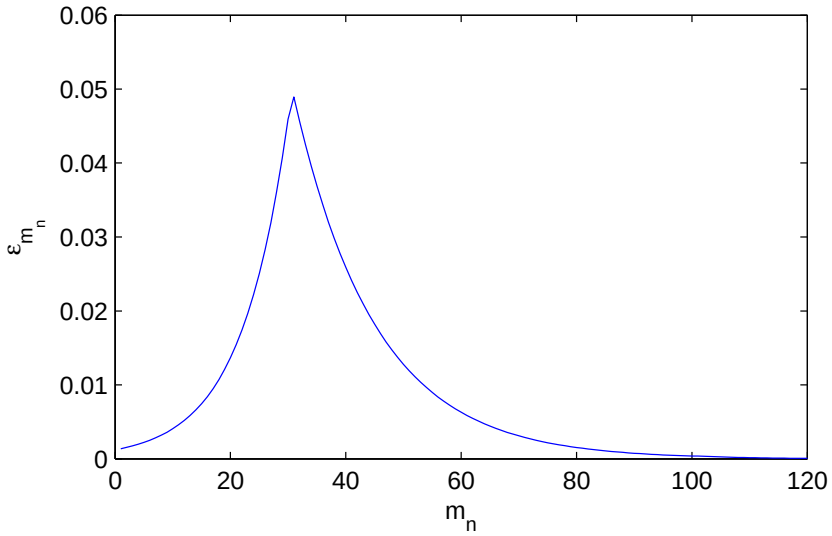


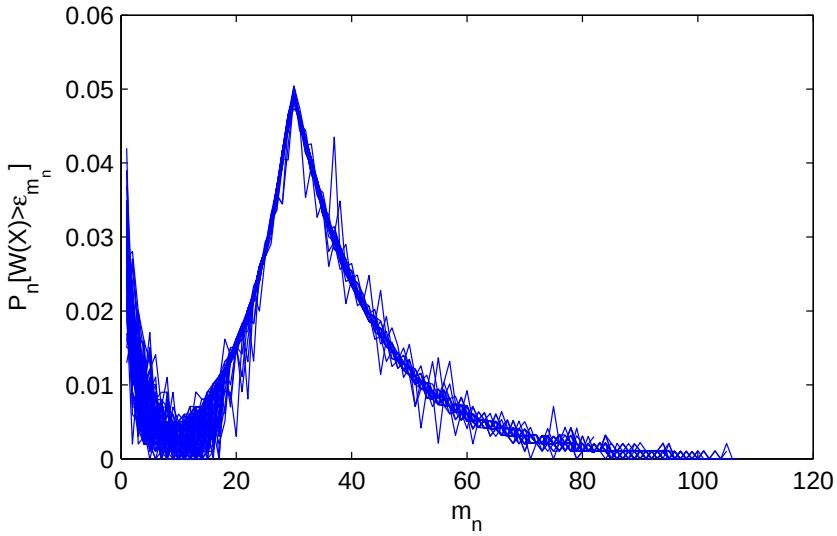
Figure: Right: evolution of the number of kernels in the proposal as MCMC progresses (blue) and "theoretic rate" (black) – left: for 100 runs

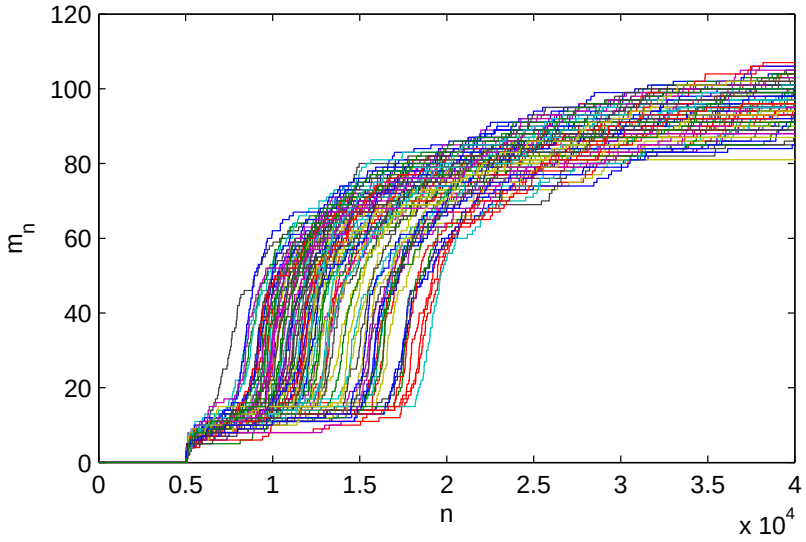
$\Rightarrow$ More consistency than the fixed threshold

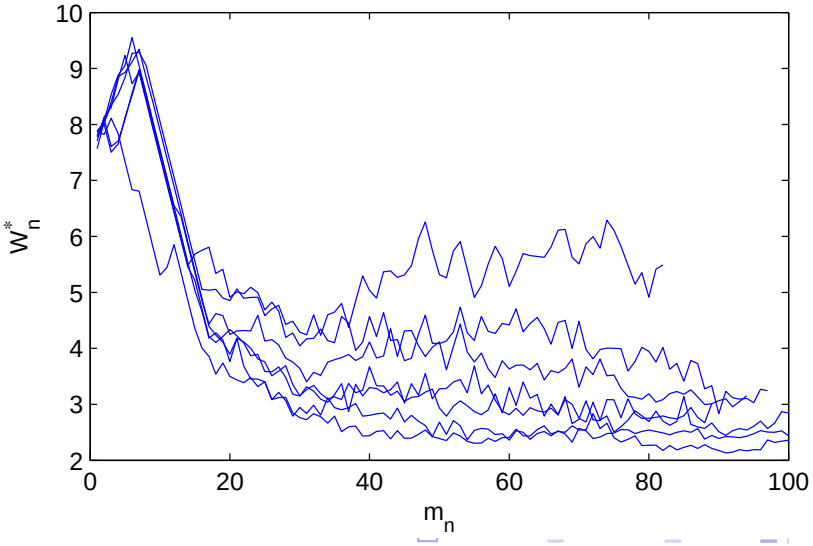
$\Rightarrow$ Obviously, we don't want to keep adding kernels infinitely

Simulation with a specific sequence of ϵ_n









A word about convergence of IM–MCMC

- Instead of Roberts and Rosenthal assumptions,
  - (i) diminishing adaption: transition kernels tend to become closer and closer in probability
  - (ii) containment: each transition kernel is a finite time step away from an ϵ-ball centered on the target (relaxation of the simultaneous ergodicity)
- Holden's proof of geometric convergence for adaptive chains with independent proposals, seems more straightforward in our case,
  - (i) main assumption: a strong Dœblin condition – it exists a function $\gamma_n : \mathsf{Y}^n \to]0, 1)$ such that for all $(x, y) \in \mathsf{X}^2$

$$\frac{\pi(z)}{Q_n(z)} \leq \frac{1}{\gamma_n(\tilde{y}^n)}$$

where $\tilde{y}^n \in \mathsf{Y}^n$ is a history dependent vector.

which holds (by construction) for IM-MCMC.

Outlines

Outlines

Adaptive Metropolis, AM (Haario,2001–Bernoulli,7(2))

A Metropolis algorithm with proposal

$$Q_n(X_n, \cdot) = Q(\cdot)\, \mathbb{1}_{n \leq N_0} + \psi_n(X_n, \cdot)\, \mathbb{1}_{n > N_0}$$

- Q is the "naive" proposal or prior
- ψ_n is a Gaussian with mean X_n and covariance matrix

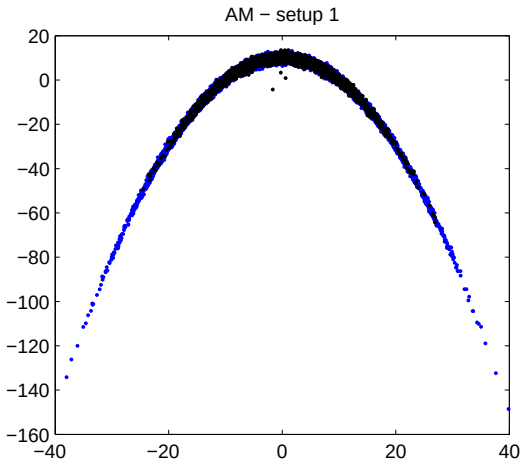
$$\Sigma_n = s_d\, \Gamma_n + s_d\, \epsilon I_d$$

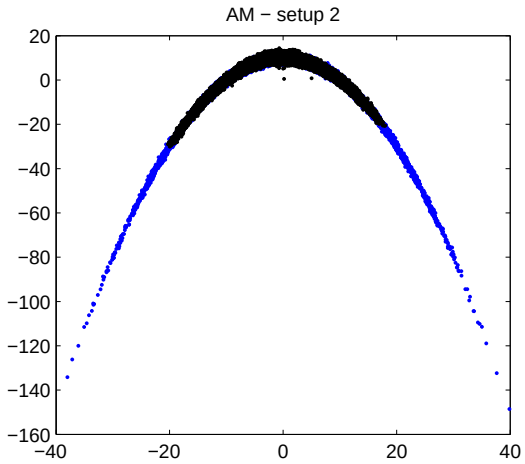
where $\Gamma_n = \mathrm{cov}(X_1, \ldots, X_{n-1})$, $s_d = (2.4)^2/d$ (d is the dimension of the state) and $\epsilon \ll d$ a constant parameter (allowing to have Σ_n positive definite)

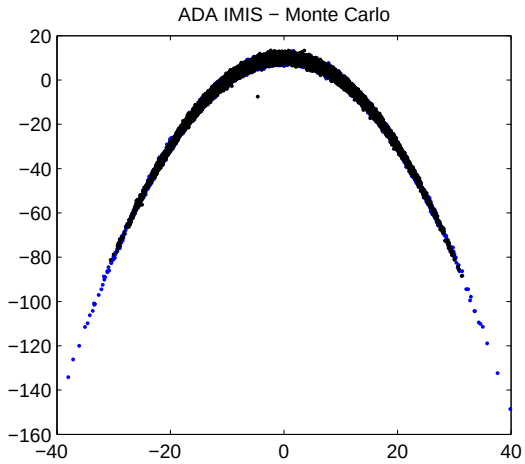
Time normalized comparison with IM-MCMC

method	(s_d, ϵ)	acc rate	m_n	Nb Iterations
AM	$(2.88, 0.001)$	0.10	–	90,000
AM	$(0.01, 1)$	0.52	–	90,000
IM–MCMC	–	0.52	96	40,000

Table: Estimation over 100 runs

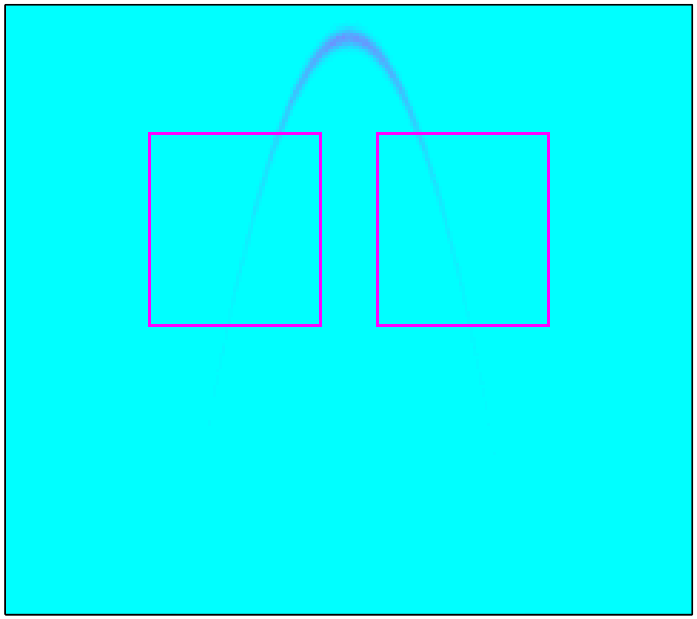




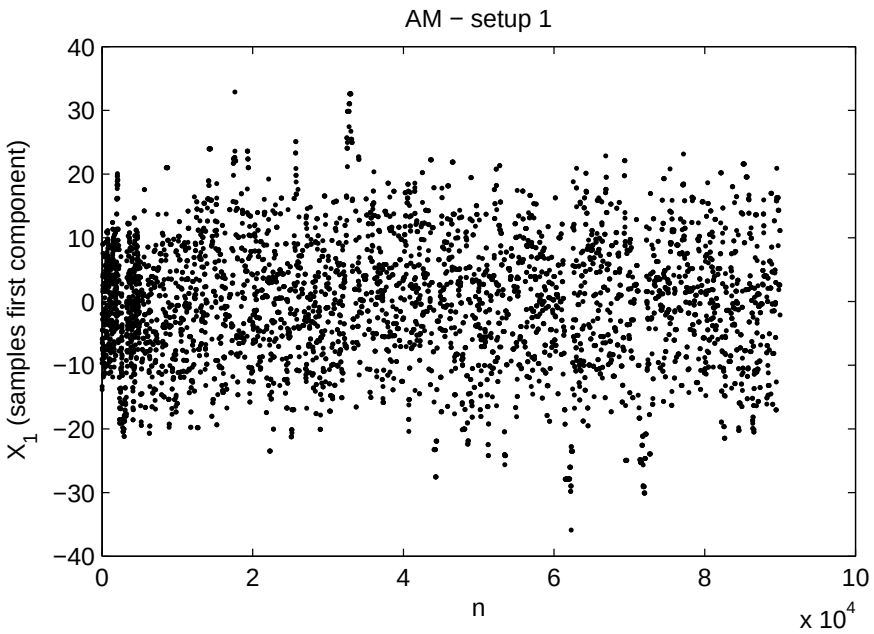


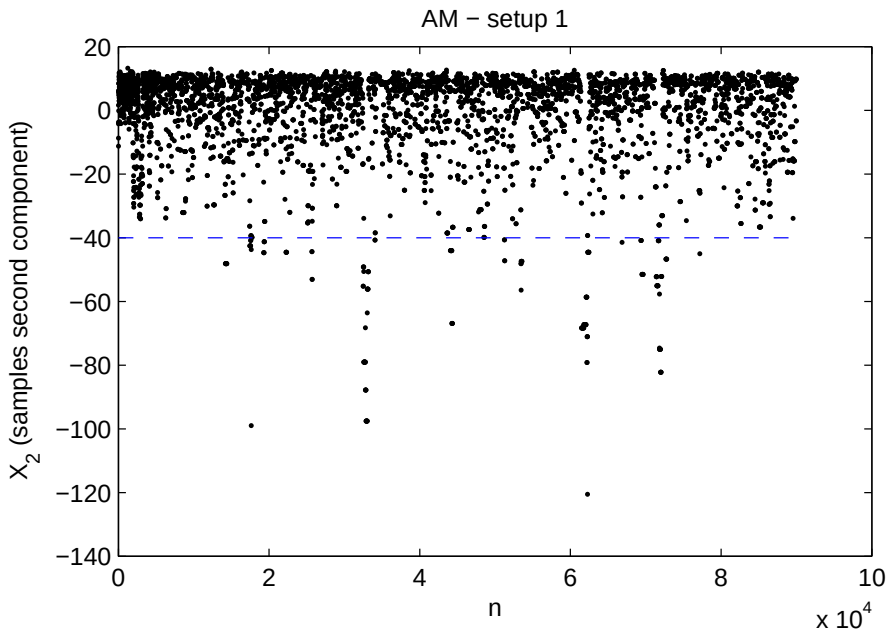
Tail exploration

We want to assess the sampling quality of the tail of π



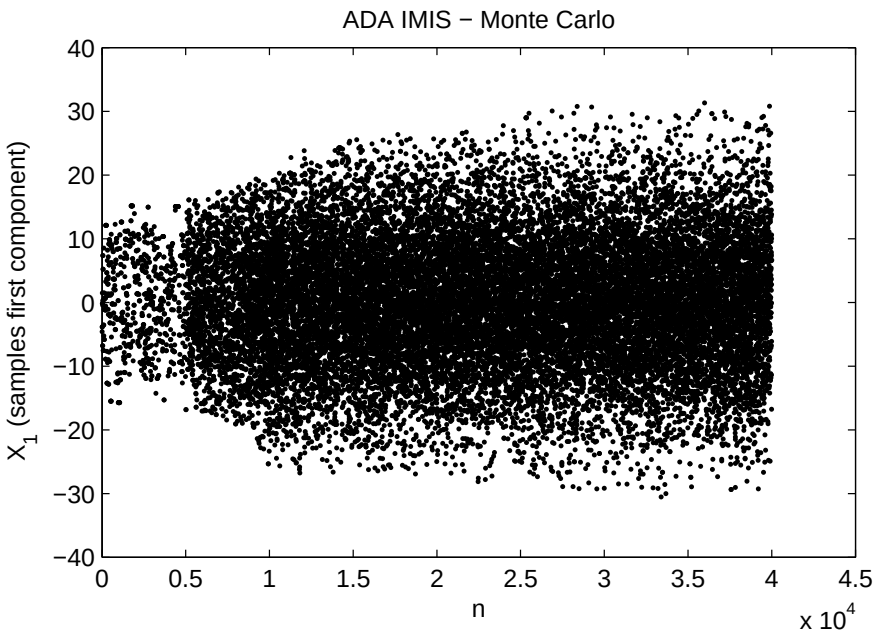
Adaptive Metropolis – setup 1

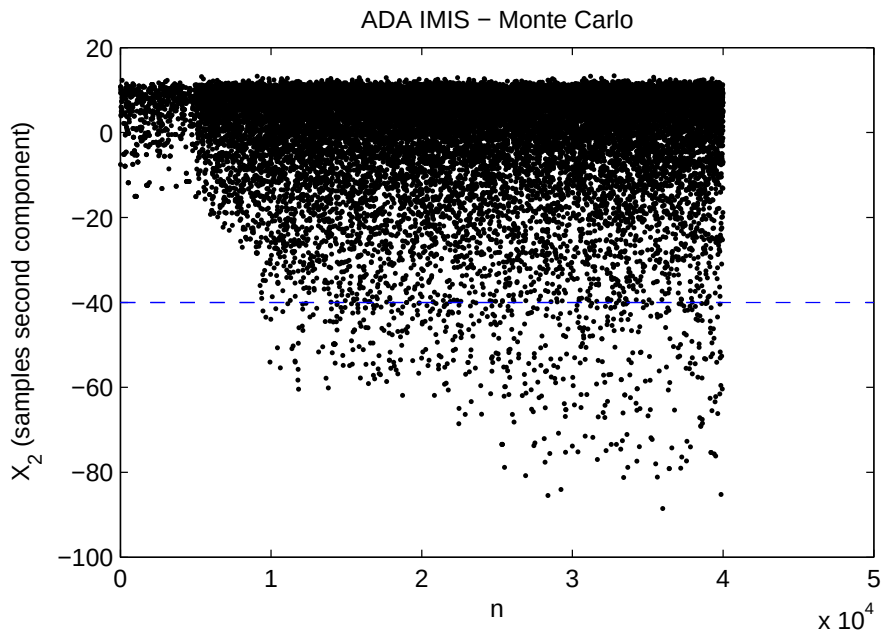




- 3754 states out of $90,000$ are such that $X_k^2 \leq 40$
- but only 48 unique points
- average waiting time in the tail ≈ 80 iterations!

What about ADA IMIS?





- 807 states out of $40,000$ are such that $X_k^2 \leq 40$
- and 467 unique points
- average waiting time in the tail ≈ 1.7 iterations!

Assessing the sampling efficiency

Kullback Leibler divergence between two measures on $(\mathsf{X}, \mathcal{X})$, say (μ, ν):

$$\mathrm{KL}\left(\mu \,\|\, \nu\right) = \mathbb{E}_{\mu}\left[\log\left(\frac{\mu(X)}{\nu(X)}\right)\right]$$

For an observed Markov chain $x_{1:n} \in \mathsf{X}^n$, let $\hat{f}_{x_{1:n}}$ be a kernel approximation of the empirical distribution $n^{-1}\sum_{k=1}^{n}\delta_{x_k}$

$$\mathrm{KL}\left(\pi \,\|\, \hat{f}_{x_{1:n}}\right) = \int_{\mathsf{X}} \log\left(\frac{\pi(x)}{\hat{f}_{x_{1:n}}(x)}\right)\pi(x)\lambda(\mathrm{d}x)$$

$$\rightsquigarrow \widetilde{\mathrm{KL}}\left(\pi \,\|\, \hat{f}_{x_{1:n}}\right) = \sum_{k=1}^{M} \log\left(\frac{\pi(\bar{x_m})}{\hat{f}_{x_{1:n}}(\bar{x_m})}\right)\pi(\bar{x_m})\mathrm{d}\lambda \qquad \text{(discretisation)}$$

KL for the three previous runs

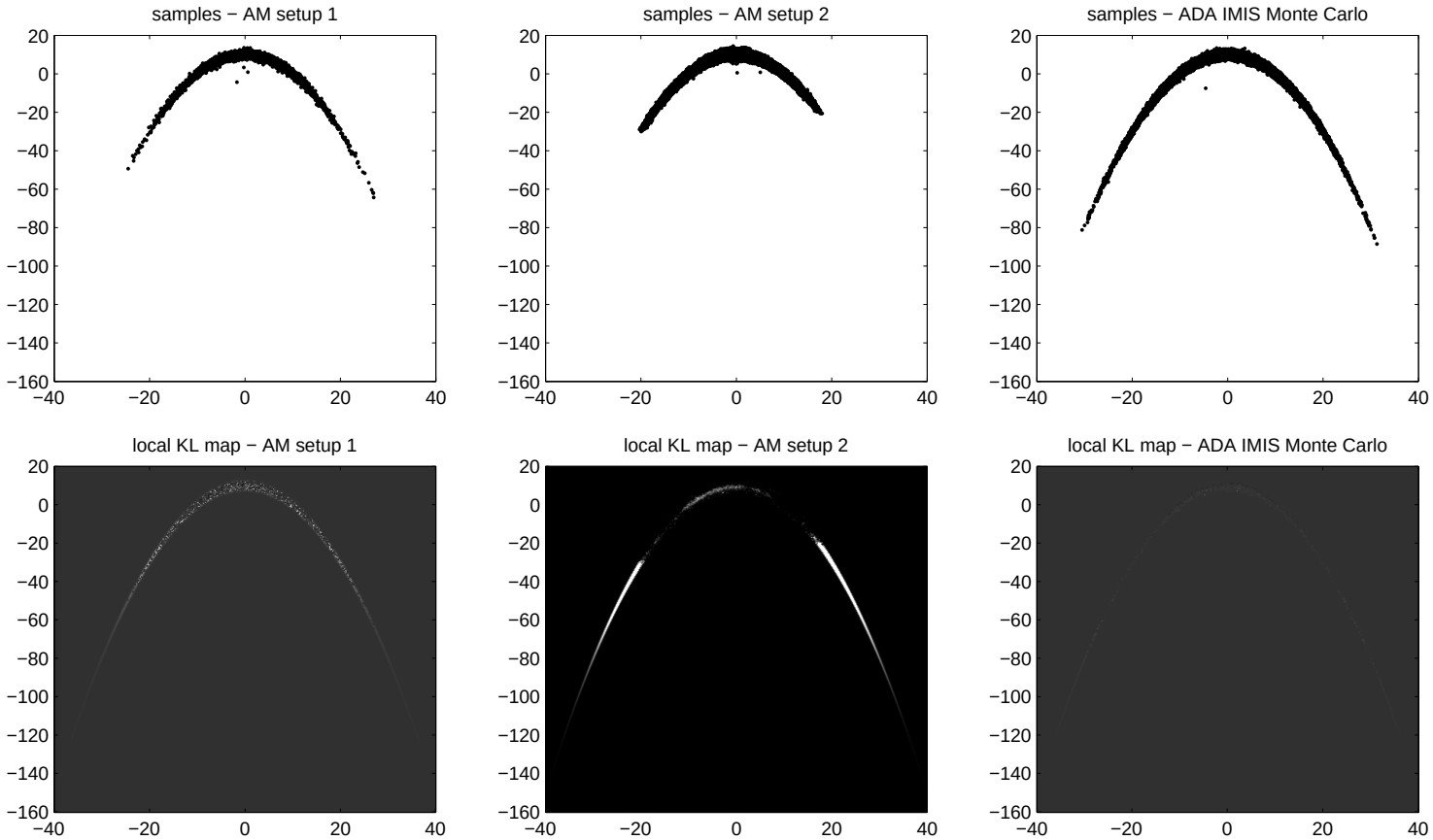
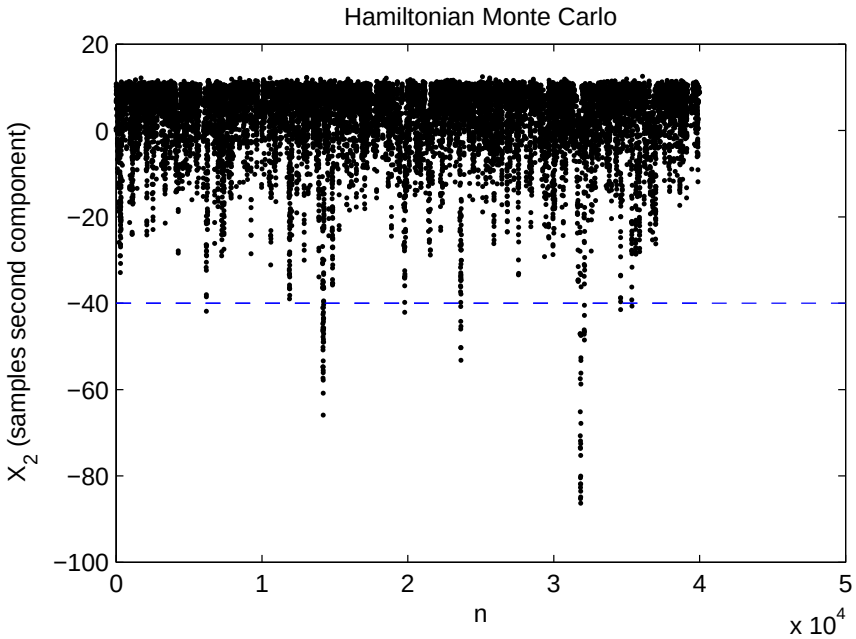


Figure: samples and discretised KL heat map (same scale) for 1 run of each setup

Hamiltonian Monte Carlo



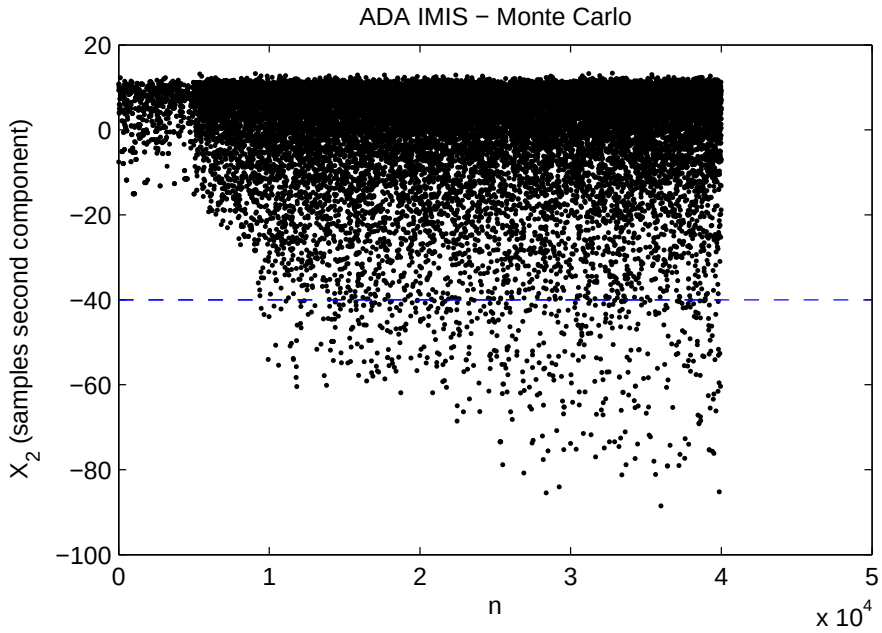
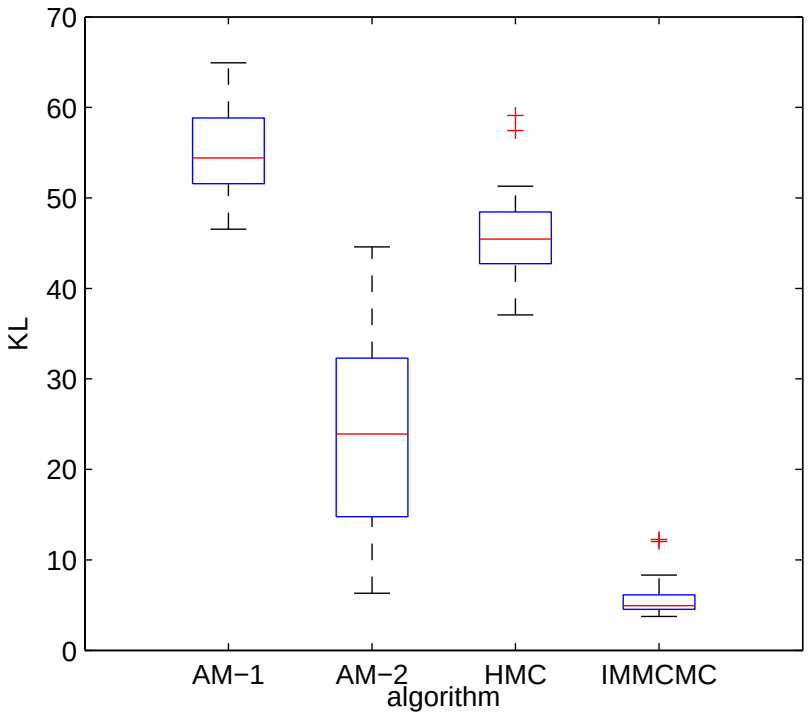


Figure: 1 run – acc rate HMC: 0.37 (ADAIMIS: 0.51) – time spent in tail HMC: 0.0033 (ADAIMIS: 0.02)

Variability over 100 independent runs



- account very well for mixing & exploration simultaneously
- ADA IMIS performs much better than AM on this example

Outlines

Comparison IM-MCMC vs IMIS

- Ridge-like simulated example in (Raftery et al)

$$\pi(\theta) \propto \underbrace{\mathrm{N}(\mu_i, D_i, \theta)}_{\text{prior}} \underbrace{\mathrm{N}(\mu_o, D_o, g(\theta))}_{\text{likelihood}}$$

  $\mu_i \in \mathbb{R}^6$ and $\mu_o \in \mathbb{R}^4$ - mapping g deterministic.
- Run IMIS until the "expected fraction of unique points in the resample is at least $1 - 1/e$"
- get the Effective sample size from IMIS

$$\mathrm{ESS}_{IMIS} = \frac{1/\sum_{k=1}^{N}(w_k^2)}{N}$$

  where N is the nb of particles when IMIS stops.

Comparison IM-MCMC vs IMIS

- for the same amount of wallclock time run IM MCMC and compare ESS_{IMIS} with

$$\text{ESS}_{IMMCMC} = \frac{N}{1 + 2\sum_{k=1}^{\infty} \rho_k}$$

where N is the nb of iterations of the Markov chain during the given time and ρ_k is the estimated lag k autocorrelation function of the chain.

Effective Sample Size

- IMIS: $46,200$ particles (67 generations) and $m_n = 67$ components

$$\text{ESS}_{IMIS} = 0.04$$

- IM-IMIS: $2,500$ MCMC iterations and $m_n = 77$ components

component	$\text{ESS}_{IM-MCMC}$
θ_1	0.11
θ_2	0.05
θ_3	0.08
θ_4	0.03
θ_5	0.09
θ_6	0.07

Here we are...

- the IM-MCMC methodology aims at extending IMIS to a sequential context
- early simulation results are encouraging
- interesting work as it allows to see analogies between particle base method and MCMC

Other interesting things to derive

- how to compare properly particle based methods and MCMC
- refine convergence rate by using the non-ordinary IM-MCMC transition...

Thank you for your attention!

IM–MCMC with "bad proposal"

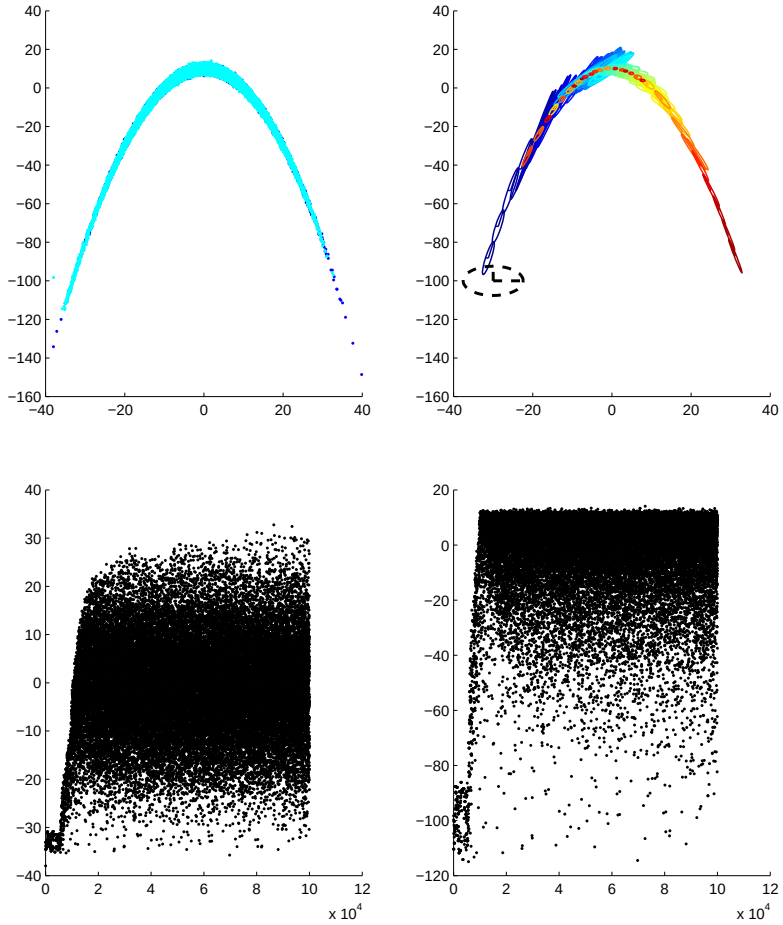


Figure: sample path of one Markov chain with "bad" initial proposal displayed by the thick ellipse (upper right hand side) – note the sequence of kernels created by IM-MCMC in rainbow style

IM–MCMC with "bad proposal"–even worst!

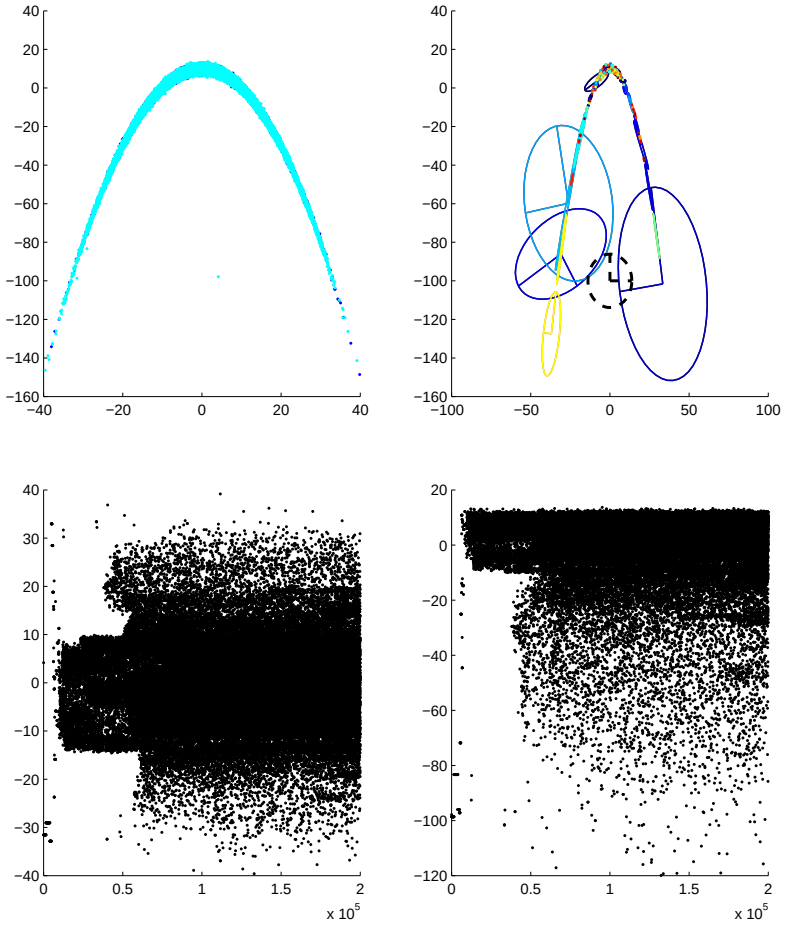


Figure: sample path of one Markov chain with "bad" initial proposal displayed by the thick ellipse (upper right hand side) – note the sequence of kernels created by IM-MCMC in rainbow style

Function evaluations

IMIS at generation k

- Target evaluation

$$\tau_\pi^{\text{IMIS}}(k) = N_0 + kN$$

- Gaussian evaluation

$$\tau_\phi^{\text{IMIS}}(k) = k(N_0 + (k-1)N)$$

IM–MCMC at iteration n

- Target evaluation

$$\tau_\pi^{\text{IM-MCMC}}(n) = n$$

- Gaussian evaluation

$$\tau_\phi^{\text{IM-MCMC}}(n) = \sum_{i=1}^{n}(m_i+1)+m_n$$

Gaussian function evaluation for IMIS

generation	$\phi_0 = Q$	ϕ_1	ϕ_2	ϕ_3	$\cdots$	ϕ_{k-1}
1	N_0	–	–	–	$\cdots$	–
2	N	$N_0 + N$	–	–	$\cdots$	–
3	N	N	$N_0 + 2N$	–	$\cdots$	–
4	N	N	N	$N_0 + 3N$	$\cdots$	–
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k	N	N	N	N	$\cdots$	$N_0 + (k-1)N$

Table: Gaussian pdf evaluation for each component of the incremental kernel per generation

At generation k:

- ϕ_{k-1} has just been created w.p. 1 thus the previous population consisting of $N_0 + (k-2)N$ particles needs to be evaluated for ϕ_{k-1}
- in addition to that, the N new particles need to be evaluated for $\phi_0, \ldots, \phi_{k-1}$
- the nb of Gaussian pdf eval. since the generation 0 is the sum of all integers in the table (constant over the columns...)

Gaussian function evaluation for IM–MCMC

At iteration n:

- either $m_n = m_{n-1}$, in which case, the proposed new state $\tilde{X}$ needs to be evaluated at $\phi_0 = Q, \phi_1, \ldots, \phi_{m_n}$, *i.e* $m_n + 1$ Gaussian pdf
- either $m_n = m_{n-1} + 1$ and in addition of the $m_n + 1$ evaluations caused by $\tilde{X}$, the current state X_n needs to be evaluated at the newly created kernel ϕ_{m_n}

This leaves with

$$\tau_{\phi}^{\text{IM-MCMC}}(n) = \sum_{i=1}^{n}(m_i + 1) + m_n$$

Gaussian pdf evaluation since starting with X_0

IMIS: $48,600$ particles (71 generations) and $m_n = 71$ components

normalisation	time	target	gaussian	function
it. completed n	9,264	51,600	40,367	40,441
samples retained	1,000	20,000	20,000	20,000
m_n	57	117	113	113
θ_1	.11	.15	.14	.15
θ_2	.11	.14	.15	.15
θ_3	.07	.13	.15	.14
θ_4	.11	.12	.16	.13
θ_5	.13	.15	.14	.14
θ_6	.06	.15	.15	.14

Table: $\text{ESS}_{IM-MCMC}$