Accelerating MCMC for imaging science by using an implicit Langevin algorithm

K.C Zygalakis

School of Mathematics, University of Edinburgh

Maxwell Institute for Mathematical Sciences

14th International Conference on Monte Carlo Methods and Applications
Mini-symposium on Recent Progress in Langevin MC
Paris

Collaborators

Yoann Altmann (HW)

Paul Dobson (UoE)

Tereza Klatzer (UoE)

Marcelo Pereyra (HW)

Jesus Maria Sanz-Serna (UC3M)

Engineering and Physical Sciences Research Council

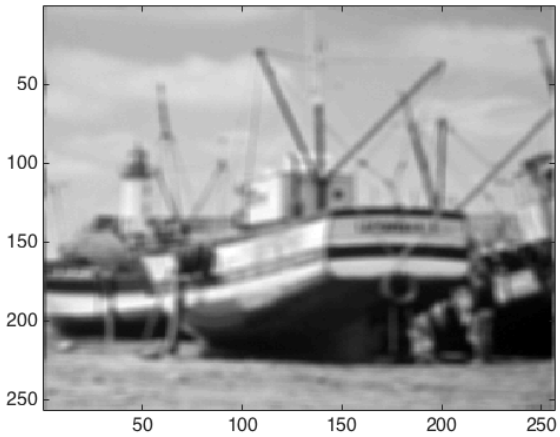
The Leverhulme Trust

Overview

Overview

THE UNIVERSITY of EDINBURGH

Imaging inverse problems



- We are interested in an unknown image $x \in \mathbb{R}^d$.
- We measure y, related to x by a statistical model $p(y|x)$.
- The recovery of x from y is ill-posed or ill-conditioned, resulting in significant uncertainty about x.
- For example, in many imaging problems

$$y = Ax + w,$$

for some operator A that is rank-deficient, and additive noise w.

The Bayesian framework

- We use priors to reduce uncertainty and deliver accurate results.
- Given the prior $p_r(x)$, the posterior distribution of x given y

$$\pi(x|y) = p(y|x)p_r(x)/p_r(y)$$

models our knowledge about x after observing y.

- Two main approaches in terms of incorporating prior information
  1. Give a functional form to $p_r(x)$; this normally results to $\pi(x|y)$ being log-concave; i.e.,
  $$\pi(x|y) = \exp\{-\phi(x)\}/Z\,,$$
  where $\phi(x)$ is a convex function and $Z = \int \exp\{-\phi(x)\}\mathrm{d}x$
  2. *Learn* the prior from the available data;

Quantities of interest

We are normally interested in calculating the following quantities of interest associated with our posterior distribution

1. $\text{argmax}_{x \in \mathbb{R}^d} \pi(x|y) = \text{argmin}_{x \in \mathbb{R}^d} \phi(x)$ (can be computed efficiently, even in very high dimensions, by (proximal) convex optimisation)
2. $\mathbb{E}_\pi(f) := \int_{\mathbb{R}^d} f(x)\pi(x|y)dx$ (one option is to use Markov Chain Monte Carlo for calculating this)

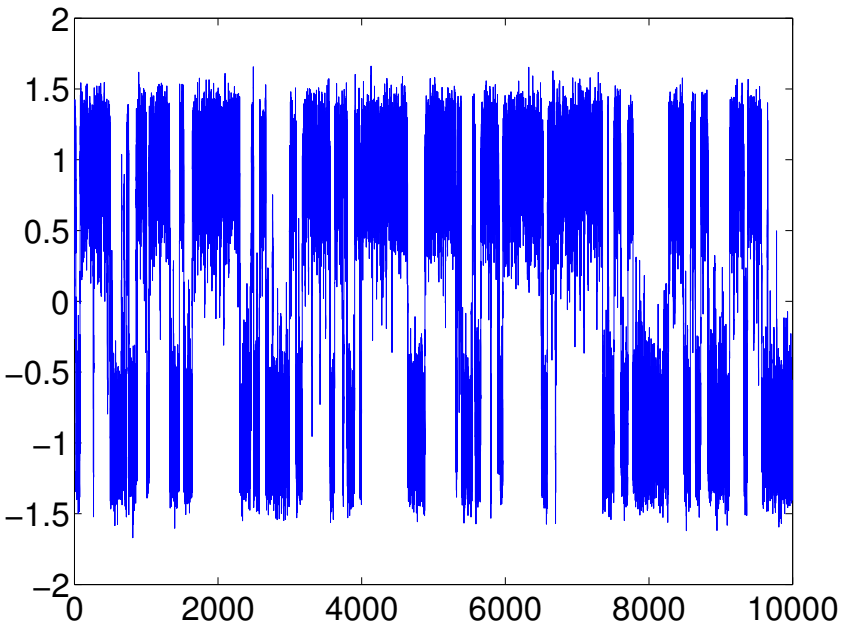
Langevin dynamics

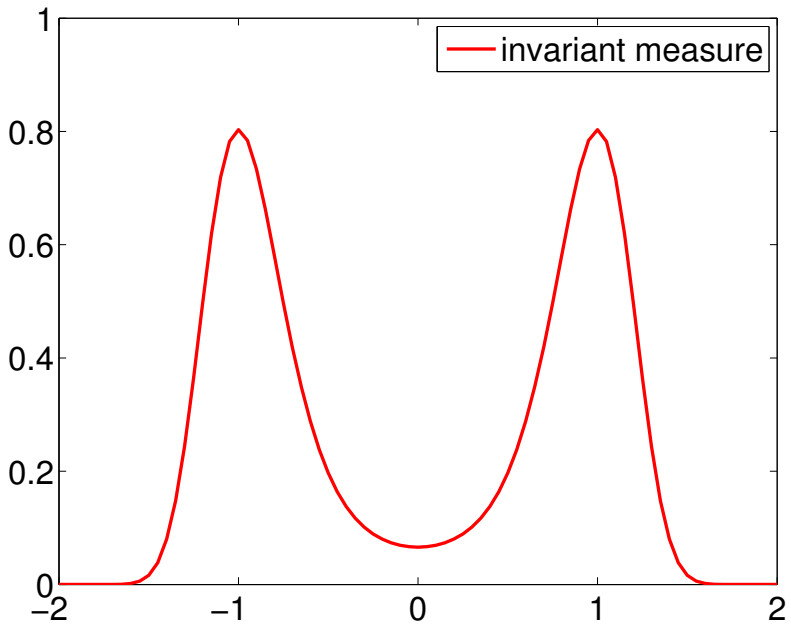
Consider the stochastic differential equation

$$dX_t = \nabla \log \pi(X_t)dt + \sqrt{2}dW_t.$$

Under appropriate assumptions on $\nabla \log \pi(x)$ one can show that its dynamics are ergodic with respect to $\pi(x) : \mathbb{R}^n \mapsto \mathbb{R}$ i.e

$$\lim_{T\to\infty} \frac{1}{T} \int_0^T f(X_s)ds = \mathbb{E}_\pi[f] := \int_{\mathbb{R}^n} f(x)\pi(x)dx.$$





In an ideal world!!!

Sampling

Go to infinity as quickly as possible (in terms of function evaluations). Once there produce samples that are i.i.d.

In real life...

In practice there are a few issues that stop you from the ideal approach

- Cannot take time-step arbitrary large (*stability*)
- The numerical invariant measure is not the same as the posterior (*asymptotic bias*)

A very simple algorithm

Euler Maryuama: $X_{n+1} = X_n + \Delta t \nabla \log \pi(X_n) + \sqrt{2\Delta t}\xi_n,\ \xi_n \sim \mathcal{N}(0, I_d)$

Priors in computational imaging

$$\pi(x|y) \propto \exp\{-g_1(x) - g_2(x)\},$$

where $g_1(x), g_2(x)$ are lower semicontinuous convex functions from $\mathbb{R}^n \mapsto (-\infty, \infty]$. Typically g_1 is L-Lipschitz differentiable, *e.g*

$$g_1(x) = \frac{1}{2\sigma^2}||y - Ax||_2^2,$$

for some observation $y \in \mathbb{R}^p$ and linear operator $A \in \mathbb{R}^{p \times n}$ and

$$g_2(x) = \alpha||Bx||_\dagger + 1_S(x),$$

for some norm $||\cdot||_\dagger$, dictionary $B \in \mathbb{R}^{n \times n}$, and convex set S. Often $g_2 \notin \mathcal{C}^1$

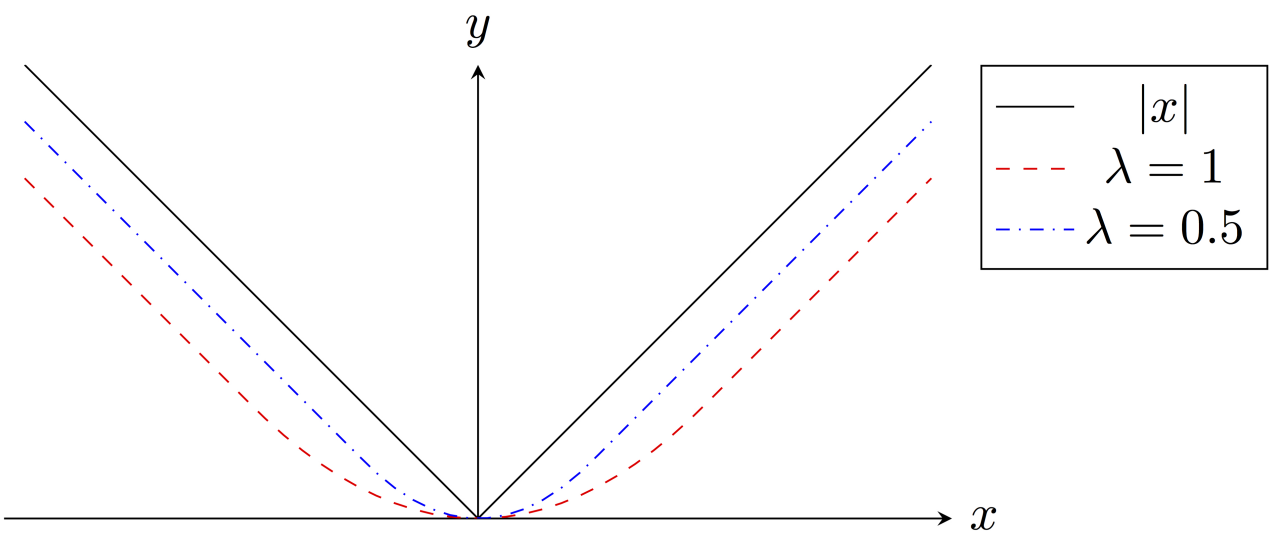
Question

How can we make sense of gradient algorithms if the gradient doesn't exist?

Poor applied mathematician's approach...

Create a smooth approximation to the prior. Take for example $g_2(x) = |x|$. Then

$$g_2^{\lambda}(x) = \begin{cases} \frac{x^2}{2\lambda}, & \text{if } |x| \leq \lambda \\ \\ |x| - \frac{\lambda}{2}, & \text{otherwise} \end{cases}$$



Convex analysis to the rescue!

For convex functions there is a principled way of dealing with the non-differentiability of a function using Moreu-Yoshida envelopes

$$g^{\lambda}(x) = \min_{y \in \mathbb{R}^n} \{g(y) + (2\lambda)^{-1} \|x - y\|^2\}$$

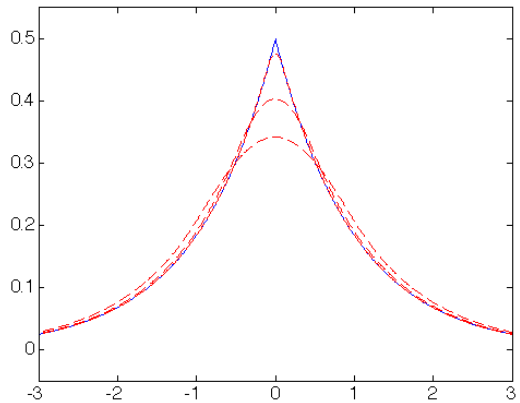
The reguralised function now inherits the convexity properties of the original one and also has a Lipschitz gradient

$$\nabla g^{\lambda}(x) = \lambda^{-1}(x - \text{prox}_g^{\lambda}(x)), \quad ||\nabla g^{\lambda}(x) - \nabla g^{\lambda}(y)|| \leq \lambda^{-1} \|x - y\|$$
$$\text{prox}_g^{\lambda}(x) = \underset{y \in \mathbb{R}^n}{\text{argmin}} \{g(y) + (2\lambda)^{-1} \|x - y\|^2\}$$

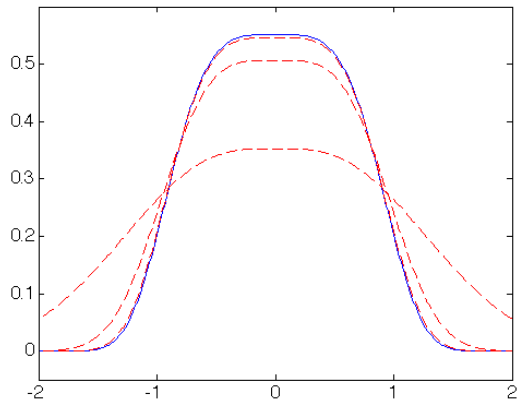
THE UNIVERSITY of EDINBURGH

Smoothed posterior distribution

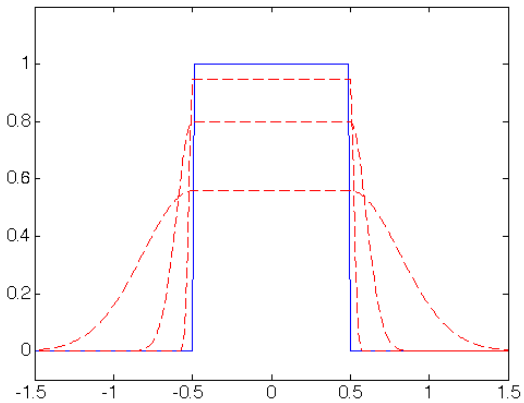
$$\pi_\lambda(x|y) = \frac{\exp[-g_1(x) - g_2^\lambda(x)]}{\int_{\mathbb{R}^d} \exp[-g_1(x) - g_2^\lambda(x)]\mathrm{d}x},$$



$\pi(x) \propto \exp(-|x|)$



$\pi(x) \propto \exp(-x^4)$

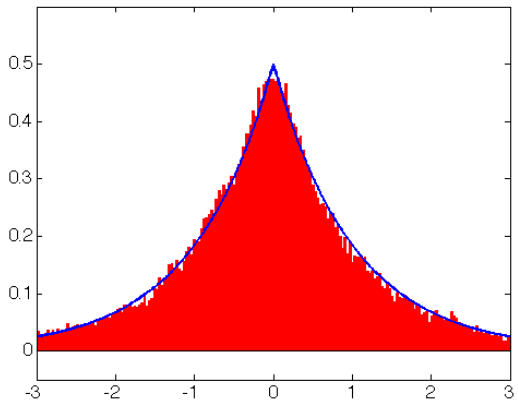


$\pi(x) \propto \mathbf{1}_{[-0.5,0.5]}(x)$

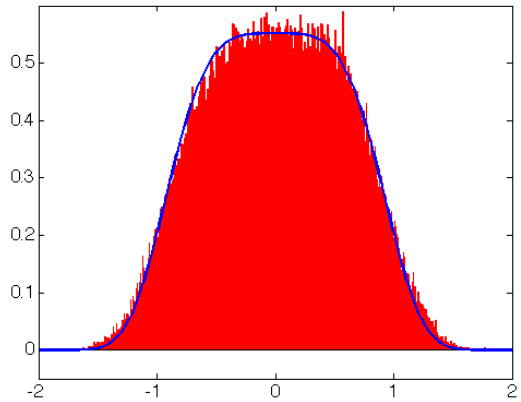
Figure: True densities (solid blue) and approximations (dashed red).

MYULA

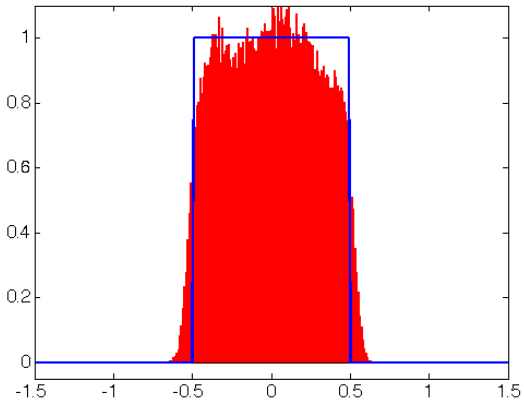
$$X_{n+1} = X_n - \Delta t \nabla g_1(X_n) - \frac{\Delta t}{\lambda}\left(X_n - \text{prox}_{g_2}^{\lambda}(X_n)\right) + \sqrt{2\Delta t}\xi_n.$$



$p(x) \propto \exp(-|x|)$



$p(x) \propto \exp(-x^4)$



$p(x) \propto \mathbf{1}_{[-0.5,0.5]}(x)$

Figure: True densities (blue) and MC approximations (red histogram).

Stiffness through smoothing

Conundrum

In order to get better approximation to true posterior ($\lambda = 0$) one might need to take λ very small, however this leads to severe time-step restriction since $\Delta t \sim \lambda$.

Possible solutions

- Consider explicit numerical schemes that allow for larger (effective) time-steps ($\Delta t \sim s^2 \lambda$ with s gradient evaluations)
- Use an implicit method (Δt can be *arbitrarily large*)

THE UNIVERSITY of EDINBURGH

Overview

THE UNIVERSITY of EDINBURGH

Proposed algorithm

$$X_{n+1} = X_n + \delta \nabla \log \pi \left(\theta X_{n+1} + (1-\theta) X_n\right) + \sqrt{2\delta}\xi_{n+1}$$

Set up

- We will now consider

$$\pi(x) \propto \exp\left(-\frac{1}{2}x^T\Sigma^{-1}x\right), \quad \Sigma = \text{diag}(\sigma_1^2, \cdots, \sigma_d^2),$$

- In this case the algorithm becomes

$$X_{n+1}^i = R_1(z_i)X_n^i + \sqrt{2\delta}R_2(z_i)\xi_n^i, \quad \xi_n^i \sim \mathcal{N}(0,1),$$

where $z_i = -\Delta t/\sigma_i^2$ and $X_0 = (X_0^1, \cdots, X_0^d)$ is a deterministic initial condition, while

$$R_1(z) = \frac{1+(1-\theta)z}{1-\theta z}, \quad R_2(z) = \frac{1}{1-\theta z}.$$

Analysis Gaussian I

Proposition I

Let $\pi(x) \propto \exp\left(-0.5x^T\Sigma^{-1}x\right)$ with $\Sigma = \text{diag}(\sigma_1^2, ..., \sigma_d^2)$, and let Q_n be the probability measure associated with n iterations of the generic Markov kernel. Then the following bound holds

$$W_2(\pi; Q_n) \leq W_2(\pi; \tilde{\pi}) + C^n W_2(\tilde{\pi}, Q_0)$$

where

$$\tilde{\pi} = \mathcal{N}\left(0, 2\delta(R_2(z))^2\left[\frac{1}{1 - R_1^2(z)}\right]\right)$$

is the numerical invariant measure and

$$C = \sqrt{\max_{1 \leq i \leq d} R_1(z_i)^2}.$$

Analysis Gaussian II

Proposition II

Let Q_n be the probability measure associated with the n-th iteration of the method starting at X_0. Then the number of steps n required such that $W_2(\pi, Q_n)^2 < \epsilon^2$ is given by

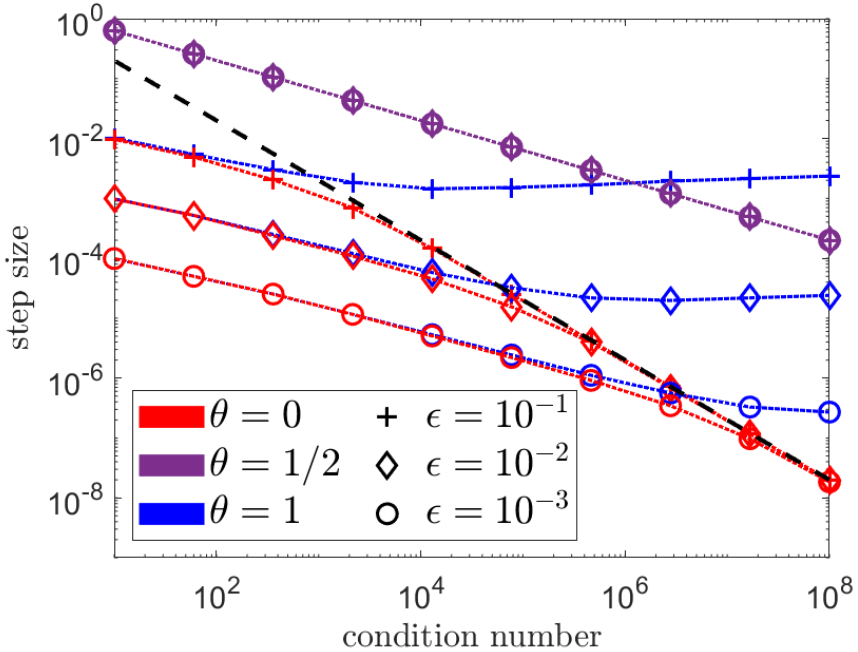
$$n \approx \begin{cases} \frac{\sqrt{\kappa}}{2}\left[\log\left(W_2(\pi, Q_0)\right) - \log(\epsilon)\right], & \theta = \frac{1}{2} \\ \min\left(\frac{d\sigma_{\max}^2}{\epsilon^2}, \frac{\sqrt{d\kappa}\sigma_{\max}}{2\epsilon}\right)\left[\log(W_2(\pi, Q_0)) - \log(2^{-1}\epsilon)\right], & \theta = 1 \end{cases}$$

with δ given by

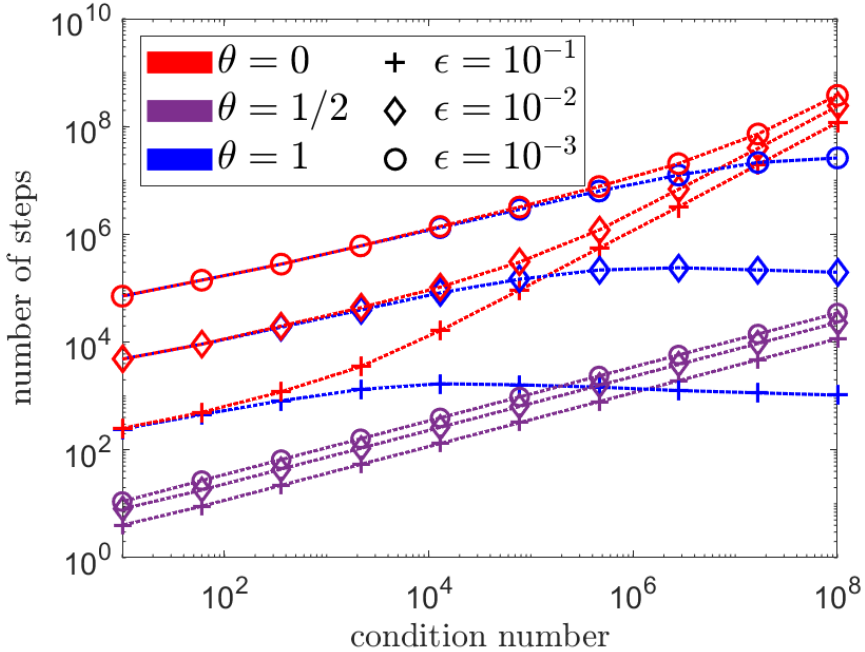
$$\delta = \begin{cases} \delta_*, & \theta = \frac{1}{2} \\ \max\left(\frac{\epsilon^2}{d}, \frac{2\epsilon\sigma_{\min}}{\sqrt{d}}\right) & \theta = 1. \end{cases}$$

where $\delta_* = \frac{2}{\sqrt{Lm}} = 2\sigma_{\min}\sigma_{\max}$.

Analysis Gaussian II



(a) δ against κ



(b) n against κ

THE UNIVERSITY of EDINBURGH

Non-linear case

A theorem

Let $U = -\log \pi$ and suppose that $U \in C^2$, m-strongly convex and has gradient which is L-Lipschitz. For any probability distribution, Q_0, let Q_n denote the probability distribution of X_n where $X_0 \sim Q_0$ and X_k is given by

$$X_{n+1} = \operatorname{argmin}_{x \in \mathbb{R}^d} F(x; X_n; \xi_{n+1}),$$

$$F(x; u, z) := \theta^{-1} U(\theta x + (1-\theta)u) + \frac{1}{2\delta}\|x - u - \sqrt{2\delta} z\|^2.$$

We assume that each step solved to a tolerance of ε, i.e. that $\|\nabla F(X_{n+1}; X_n, \xi_n)\| \leq \varepsilon$ for every n. Then for any initial probability distribution Q_0 with finite second moments we have

$$W_2(Q_n, \pi) \leq C^n W_2(Q_0, \pi) + \frac{1 - C^{n+1}}{1 - C} \frac{\frac{1}{2}\delta^2 L^{\frac{3}{2}}\sqrt{d} + \frac{2}{3} L \delta^{\frac{3}{2}} \sqrt{2d} + \varepsilon\delta}{1 + \theta\delta m}$$

and

$$C = \sqrt{\max_{z \in [m\delta, L\delta]} R_1(-z)^2}.$$

Overview

THE UNIVERSITY of EDINBURGH

Gaussian mixture I

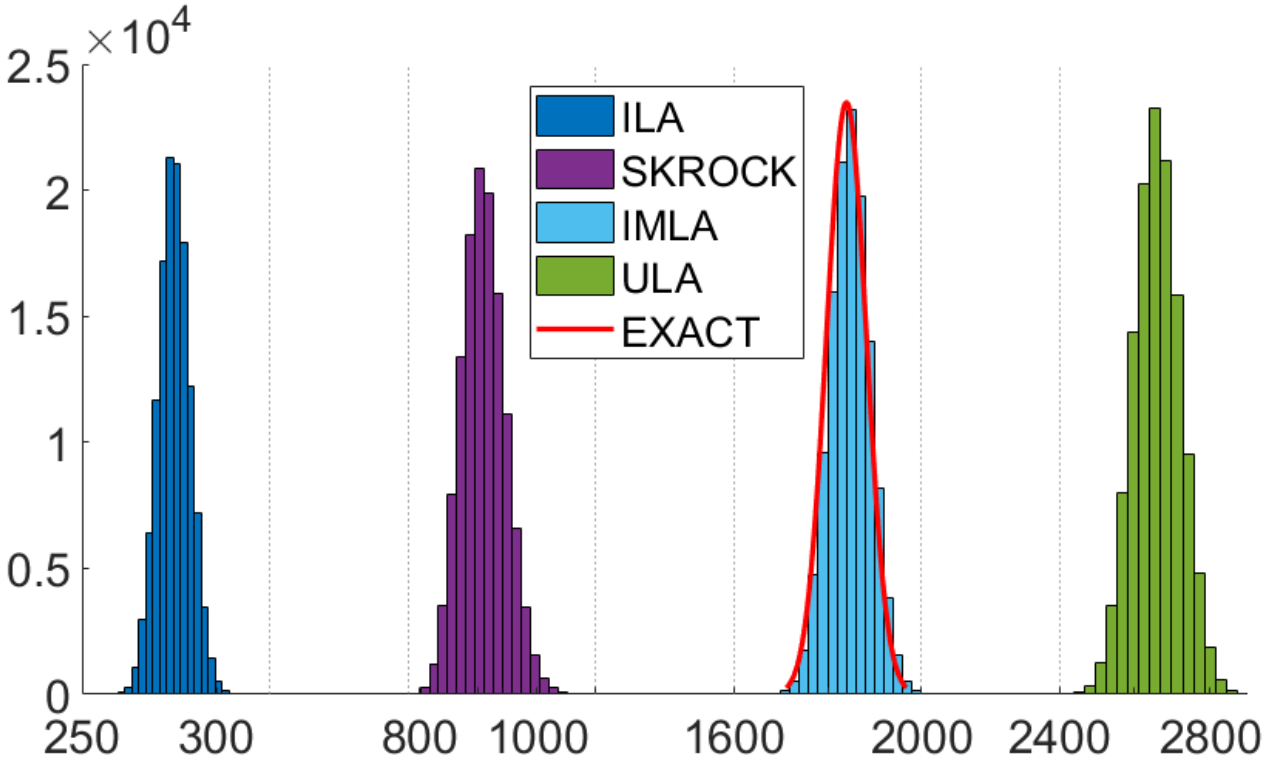


Figure: Comparison of $\log \pi$ for GMM experiment

Gaussian mixture II

Algorithm	Mean $\mathcal{W}_2(\pi, \tilde{\pi})$	Std. Dev.
EXACT	1.4123e-07	1.4448e-07
IMLA ($\theta = 1/2$)	1.4095e-07	1.5540e-07
ILA ($\theta = 1$)	6.2550e-04	1.7596e-06
ULA ($\theta = 0$)	6.6183e-05	7.7698e-07
SKROCK	2.0804e-04	1.3499e-06

Table: Summary of Wasserstein errors

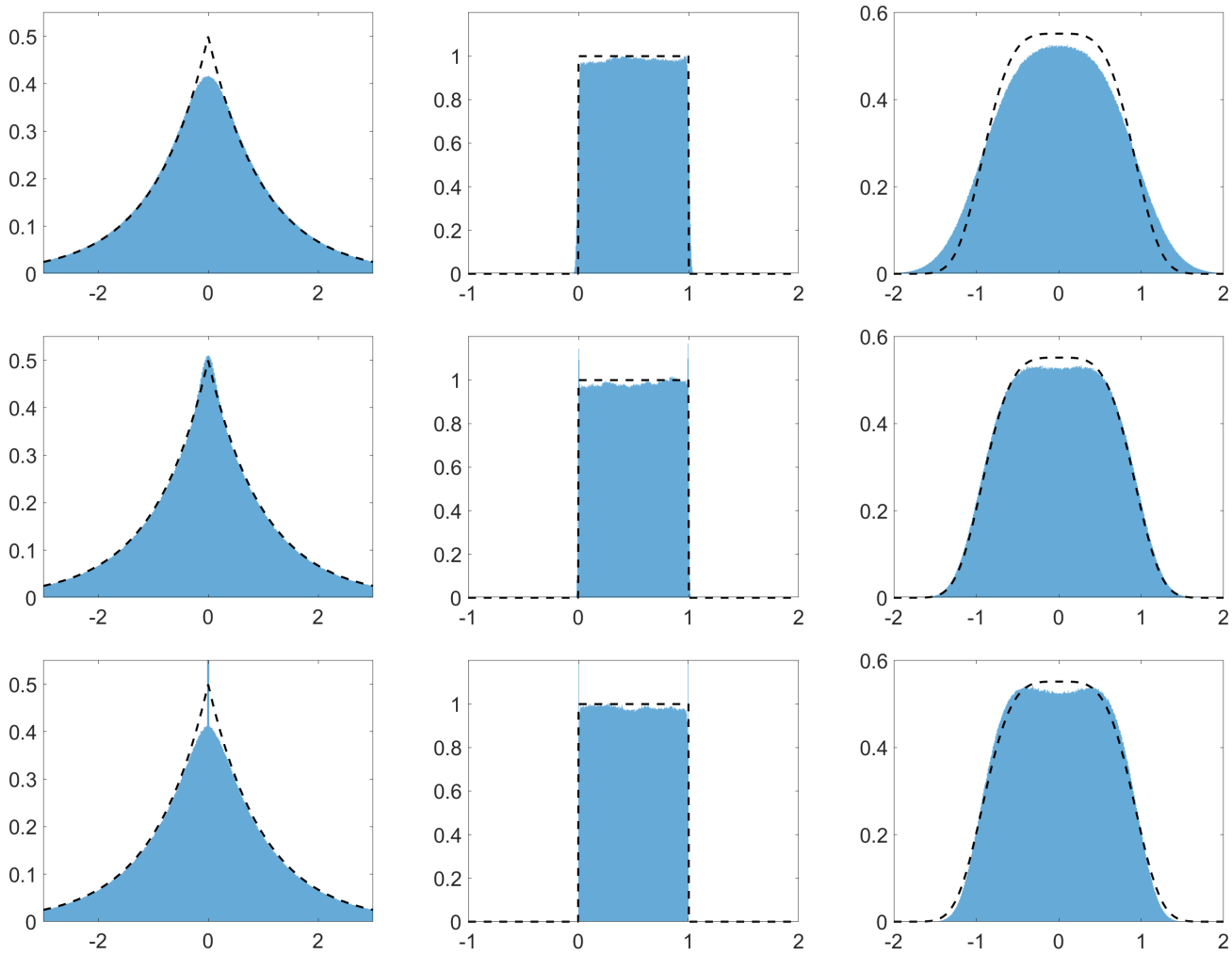
One dimensional distributions I

- Laplace $\pi(x) \propto e^{-|x|}$,
- Uniform $\pi(x) = e^{-\iota_{[0,1]}(x)}$,
- Light-tailed $\pi(x) \propto e^{-x^4}$.

Alternative representation

$$X_{n+1} = \theta^{-1}\mathrm{prox}_U^{\delta\theta}(X_n + \theta\sqrt{2\delta}\xi_n) - \frac{1-\theta}{\theta}X_n.$$

One dimensional distributions II



(a) Laplace (b) Uniform (c) Light-tailed

One dimensional disributions III

Distribution	SD IMLA	SD ILA	SD MYULA	SD EXACT
Laplace	1.4046	1.4005	1.4356	1.4142
Uniform	0.2923	0.2936	0.2949	0.2887
$\exp(-x^4)$	0.5964	0.5777	0.6590	0.5813

Table: Summary of the estimated vs. the exact standard deviations (SD) for each method (IMLA, ILA and MYULA)

Image deconvolution using a CRR-NN prior

$$\pi(x|y) \propto \exp\left(-\frac{\|Ax-y\|^2}{2\sigma^2} - \frac{\lambda}{\mu} R_\Theta(\mu x)\right).$$



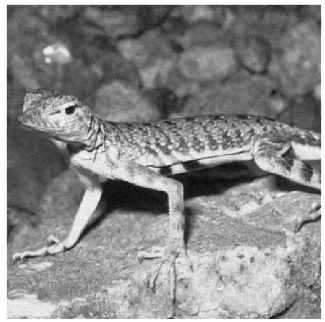



Posterior means

Mean crr-nn IMLA 29.63dB

Mean crr-nn IMLA 29.70dB

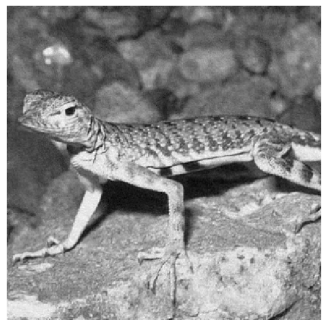
Mean crr-nn IMLA 30.04dB



Mean crr-nn SKROCK 29.21dB

Mean crr-nn SKROCK 28.86dB

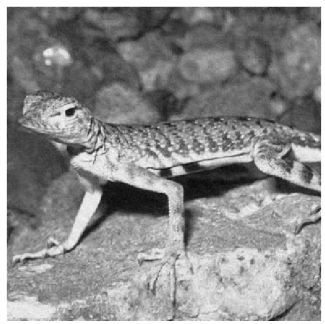
Mean crr-nn SKROCK 29.75dB



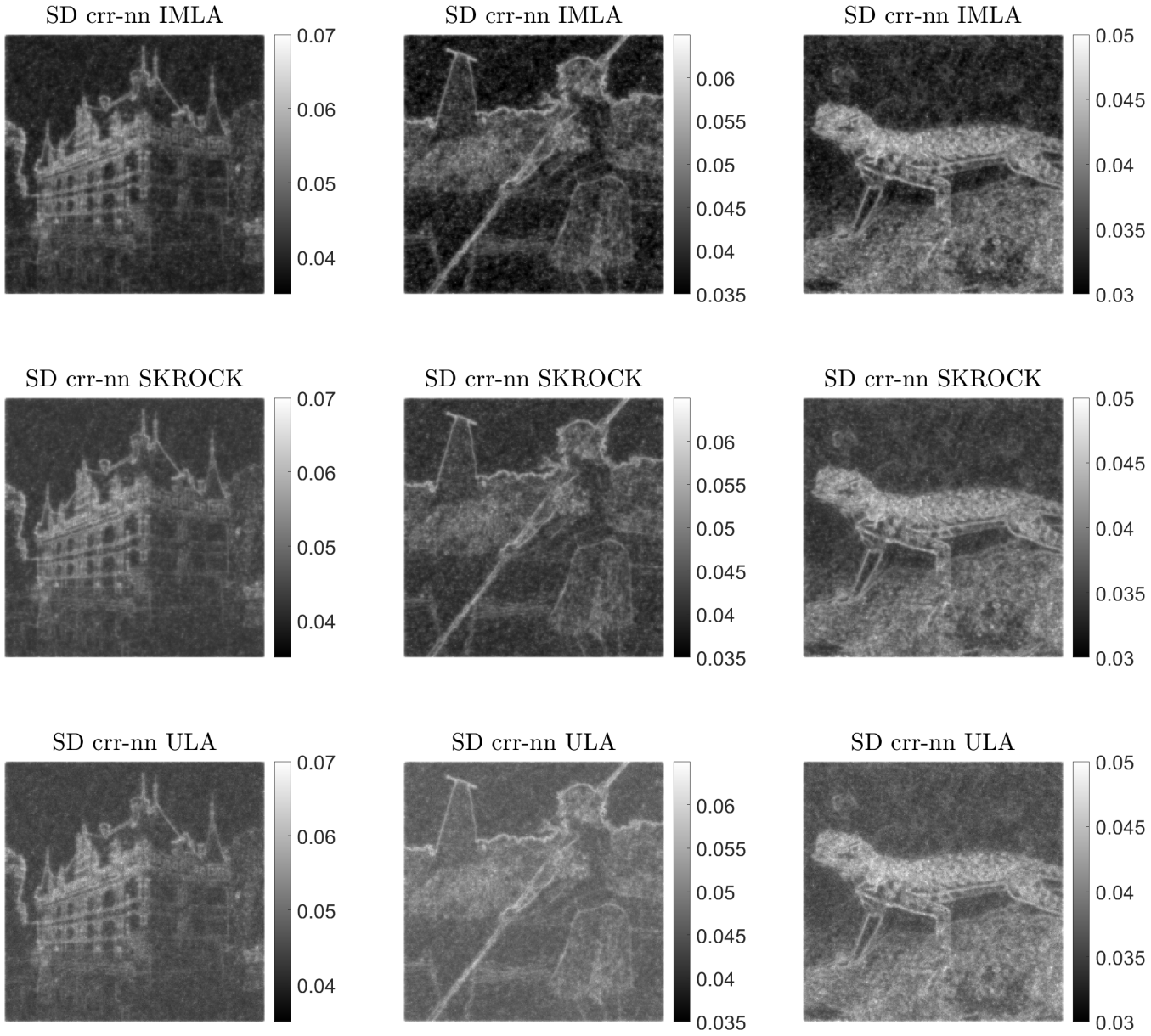
Mean crr-nn ULA 29.18dB

Mean crr-nn ULA 28.76dB

Mean crr-nn ULA 29.72dB



Pixel standard deviation



Convergence to equilibrium

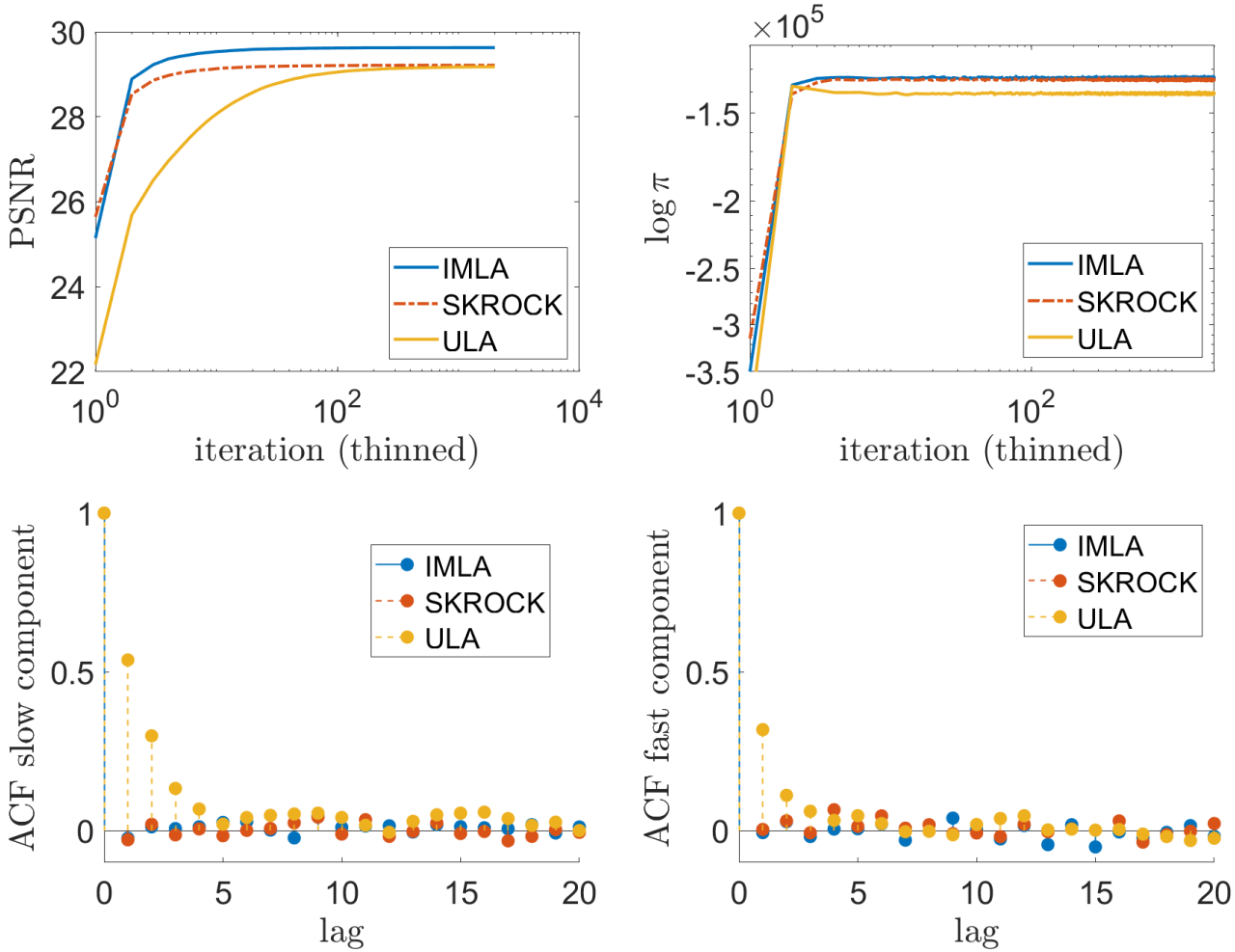


Figure: PSNR of the running mean (top left), $\log \pi$ traces (top right) and autocorrelation of the slowest and fastest component (bottom left and right) for IMLA, SKROCK, ULA for the castle image

Overview

Conclusions

1. Being able to take large steps is key when dealing with stiffness arising from regularisation
2. Proposed a new family of methods that for $\theta = 1/2$ provably accelerate the convergence to (numerical) equilibrium (similar behaviour to Nesterov method for optimization
3. In the case of non-smooth potentials the new methods are able to deal directly with regularisation
4. The method involves an implicit step (if this can be done fast the method is more computationally efficient that current state of the art SKROCK (and with proof in this case))