An adaptive scheme for MCMC in infinite dimensions

Jonas Wallin [Lund U.]
&
Sreekar Vadlamani [TIFR CAM]

Advances in Applied Probability, ICTS
August 05–17, 2019

Motivation: Bayesian inverse problems

Motivation: Bayesian inverse problems

- Forward model:

$$y_\star = \mathcal{G}(u)$$

- Noisy observations:

$$y = \mathcal{G}(u) + \eta$$

- Goal: estimate u given observation y (statistical inverse problem)
- Bayesian approach: assume a prior on u, and propose the posterior distribution $u|y$ as “solution” to the inverse problem

An example: Inverse problem for diffusion coefficient

[Calderón problem]

Consider solution to the following two–point boundary value problem on unit interval (find p, given κ):

$$-\frac{d}{dx}\left(e^{\kappa(x)}\frac{dp(x)}{dx}\right) = 0$$

such that $p(0) = p_0 < p(1) = p_1$.

Solution:

$$p(x) = (p_1 - p_0)\,\frac{J_x(\kappa(x))}{J_1(\kappa(x))} + p_0,$$

where $J_x(f(x)) = \int_0^x \exp(-f(y))\,dy$

An example: Inverse problem for diffusion coefficient

[Calderón problem]

Consider solution to the following two–point boundary value problem on unit interval (find p, given κ):

$$-\frac{d}{dx}\left(e^{\kappa(x)}\frac{dp(x)}{dx}\right) = 0$$

such that $p(0) = p_0 < p(1) = p_1$.

Solution:

$$p(x) = (p_1 - p_0)\,\frac{J_x(\kappa(x))}{J_1(\kappa(x))} + p_0,$$

where $J_x(f(x)) = \int_0^x \exp(-f(y))\,dy$

Forward model: $\kappa \mapsto \mathcal{G}(\kappa) = (p(x_1), \ldots, p(x_k))$

An example: Inverse problem for diffusion coefficient

[Calderón problem]

Consider solution to the following two–point boundary value problem on unit interval (find p, given κ):

$$-\frac{d}{dx}\left(e^{\kappa(x)}\frac{dp(x)}{dx}\right) = 0$$

such that $p(0) = p_0 < p(1) = p_1$.

Solution:

$$p(x) = (p_1 - p_0)\,\frac{J_x(\kappa(x))}{J_1(\kappa(x))} + p_0,$$

where $J_x(f(x)) = \int_0^x \exp(-f(y))\,dy$

Forward model: $\kappa \mapsto \mathcal{G}(\kappa) = (p(x_1), \ldots, p(x_k))$

Inverse problem: Estimate κ from noisy measurements of $\{p(x_k)\}_{k=1}^{N}$.

In this example:

- $\mathcal{G} : C[0,1] \to \mathbb{R}^N$ is clearly not invertible, but satisfies certain regularity properties – growth and locally Lipschitz [Stuart 2010]

In this example:

- $\mathcal{G} : C[0,1] \to \mathbb{R}^N$ is clearly not invertible, but satisfies certain regularity properties – growth and locally Lipschitz [Stuart 2010]
- Writing $y = (y_1, \ldots, y_N)$ for the noisy observations of $\{p(x_k)\}_{k=1}^N$

$$p(y|\kappa) \propto \exp\left(-\frac{1}{2}(y - \mathcal{G}(\kappa))^t \Gamma^{-1}(y - \mathcal{G}(x))\right)$$

where we assumed noise to be distributed as mean zero Gaussian with covariance Γ.

In this example:

- $\mathcal{G} : C[0,1] \to \mathbb{R}^N$ is clearly not invertible, but satisfies certain regularity properties – growth and locally Lipschitz [Stuart 2010]

- Writing $y = (y_1, \ldots, y_N)$ for the noisy observations of $\{p(x_k)\}_{k=1}^N$

$$p(y|\kappa) \propto \exp\left(-\frac{1}{2}(y - \mathcal{G}(\kappa))^t \Gamma^{-1}(y - \mathcal{G}(x))\right)$$

  where we assumed noise to be distributed as mean zero Gaussian with covariance Γ.

- Assume a Gaussian prior μ_0 incorporating certain information about the model (shift invariance), and write μ^y for the posterior measure of κ given y, then

$$\frac{d\mu^y}{d\mu_0} \propto \exp\left(-\frac{1}{2}(y - \mathcal{G}(\kappa))^t \Gamma^{-1}(y - \mathcal{G}(x))\right)$$

Some takeaways

- Honestly an infinite dimensional inverse problem

- Similar problems arise in Lagrangian data assimilation; signal processing; geophysics; ...

- Ability to sample from such posterior distributions assumes the centerstage in such problems

Some takeaways

- Honestly an infinite dimensional inverse problem
- Similar problems arise in Lagrangian data assimilation; signal processing; geophysics; ...
- Ability to sample from such posterior distributions assumes the centerstage in such problems

MCMC to the rescue

MCMC

- Construct a Markov chain whose limiting distribution is the distribution of interest, called the target measure.

MCMC

- Construct a Markov chain whose limiting distribution is the distribution of interest, called the target measure.
- Generate samples of from this Markov chain which has run for long enough time.

MCMC

- Construct a Markov chain whose limiting distribution is the distribution of interest, called the target measure.
- Generate samples of from this Markov chain which has run for long enough time.
- Ergodicity says that the time average of such sample approximates the average w.r.t. the target measure.

MCMC

- Construct a Markov chain whose limiting distribution is the distribution of interest, called the target measure.
- Generate samples of from this Markov chain which has run for long enough time.
- Ergodicity says that the time average of such sample approximates the average w.r.t. the target measure.
- The sequence of samples generated by bay of Markov chain are correlated, and thus are slow to converge to the target distribution.

MCMC

- Construct a Markov chain whose limiting distribution is the distribution of interest, called the target measure.
- Generate samples of from this Markov chain which has run for long enough time.
- Ergodicity says that the time average of such sample approximates the average w.r.t. the target measure.
- The sequence of samples generated by bay of Markov chain are correlated, and thus are slow to converge to the target distribution.
- A way out: an additional Metropolis–Hastings step is introduced to improve the efficiency.

MCMC in infinite dimensions

(The issues)

- Step 0: How to characterise measures in infinite dimensions?
- Construct a Markov chain: Needs care! Not many well studied infinite dimensional dynamical models (existence and uniqueness).
- Density: What does a density mean in infinite dimensions?

Characterising measures on function spaces

- Defining distributions on function spaces is usually a tough task;

- Defining distributions on function spaces is usually a tough task;
- except when some regularity in terms of the basis functions is assumed,

- Defining distributions on function spaces is usually a tough task;
- except when some regularity in terms of the basis functions is assumed,
- in which case an i.i.d. collection of random variables can then be used to generate infinite dimensional random elements (using Mercer expansion; Karhunen–Loéve expansion).

- Defining distributions on function spaces is usually a tough task;
- except when some regularity in terms of the basis functions is assumed,
- in which case an i.i.d. collection of random variables can then be used to generate infinite dimensional random elements (using Mercer expansion; Karhunen–Loéve expansion).

★ Most common distributions on infinite dimensional function spaces relate to Gaussian processes / fields. Examples: (fractional) Brownian motion; Ornstein–Uhlenbeck process; Gaussian random fields with given mean and covariance function.

The state of art MCMC algorithms in infinite dimensional spaces primarily deal with target measure μ_1 which can be expressed as:

$$\frac{d\mu_1}{d\mu_0} \propto \exp\left(-\Phi(Y, u)\right), \tag{1}$$

where,

- μ_0 is a given centred Gaussian measure with covariance $\mathcal{C}$;
- Y can be regarded as a parameter

The state of art MCMC algorithms in infinite dimensional spaces primarily deal with target measure μ_1 which can be expressed as:

$$\frac{d\mu_1}{d\mu_0} \propto \exp\left(-\Phi(Y, u)\right), \tag{1}$$

where,

- μ_0 is a given centred Gaussian measure with covariance $\mathcal{C}$;
- Y can be regarded as a parameter (in the example, Y shall take the role of **observations**).

The state of art MCMC algorithms in infinite dimensional spaces primarily deal with target measure μ_1 which can be expressed as:

$$\frac{d\mu_1}{d\mu_0} \propto \exp\left(-\Phi(Y, u)\right), \tag{1}$$

where,

- μ_0 is a given centred Gaussian measure with covariance $\mathcal{C}$;
- Y can be regarded as a parameter (in the example, Y shall take the role of **observations**).
- $\Phi(\cdot, \cdot)$ is also called the potential (satisfying certain regularity assumptions).

The state of art MCMC algorithms in infinite dimensional spaces primarily deal with target measure μ_1 which can be expressed as:

$$\frac{d\mu_1}{d\mu_0} \propto \exp\left(-\Phi(Y, u)\right), \tag{1}$$

where,

- μ_0 is a given centred Gaussian measure with covariance $\mathcal{C}$;
- Y can be regarded as a parameter (in the example, Y shall take the role of **observations**).
- $\Phi(\cdot, \cdot)$ is also called the potential (satisfying certain regularity assumptions). EXAMPLE: when μ_1 is $\mathcal{N}(m, \mathcal{C})$, then

$$\Phi(u) = -\langle \mathcal{C}^{-1} u, m\rangle + \frac{1}{2}\langle \mathcal{C}^{-1} u, u\rangle$$

The state of art MCMC algorithms in infinite dimensional spaces primarily deal with target measure μ_1 which can be expressed as:

$$\frac{d\mu_1}{d\mu_0} \propto \exp\left(-\Phi(Y, u)\right), \tag{1}$$

where,

- μ_0 is a given centred Gaussian measure with covariance $\mathcal{C}$;
- Y can be regarded as a parameter (in the example, Y shall take the role of **observations**).
- $\Phi(\cdot, \cdot)$ is also called the potential (satisfying certain regularity assumptions). EXAMPLE: when μ_1 is $\mathcal{N}(m, \mathcal{C})$, then

$$\Phi(u) = -\langle \mathcal{C}^{-1}u, m\rangle + \frac{1}{2}\langle \mathcal{C}^{-1}u, u\rangle$$

Advantages:

$\hookrightarrow$ Gaussian measure μ_0 takes over the role of Lebesgue measure in finite dimensions.

$\hookrightarrow$ A well established theory of Gaussian measures in infinite dimensions comes in handy

Our focus: Gaussian measures on Hilbert spaces

*to avoid messy technical details involved with defining Gaussian measures in Banach spaces

The MC of MCMC

Metropolis–Hastings

Let μ_1 be the target measure of interest, and $q(\cdot,\cdot)$ be the proposal distribution (related to a Markov chain)

- Set $j = 0$. Pick x_0 from the state space of the Markov chain.
- Given x_j, propose $y \sim q(x_j, \cdot)$.
- Compute
$$\alpha(x_j, y) = \min\left(\frac{d\nu^\perp}{d\nu}, 1\right)$$
where $\nu(dv, du) = q(u, dv)\mu_1(du)$, and $\nu^\perp$ is ν with the roles of u and v switched.
- Set $x_{j+1} = y$ with probability $\alpha(x_j, y)$, else set $x_{j+1} = x_j$.
- Set $j = j + 1$, and return to step of proposal generation.

Note that we purposefully stayed clear of using densities.

A random walk model

$$y|x \overset{d}{=} x + \beta\,\xi$$

where $\xi \sim \mathcal{N}(0, \mathcal{C})$, and $\beta \in [0, 1]$ controls the step size

A random walk model

$$y|x \overset{d}{=} x + \beta\,\xi$$

where $\xi \sim \mathcal{N}(0, \mathcal{C})$, and $\beta \in [0, 1]$ controls the step size

Meaning, given the current state x_j, a proposal is generated from $\mathcal{N}(x_j, \mathcal{C})$.

A random walk model

$$y|x \stackrel{d}{=} x + \beta\,\xi$$

where $\xi \sim \mathcal{N}(0, \mathcal{C})$, and $\beta \in [0, 1]$ controls the step size

Meaning, given the current state x_j, a proposal is generated from $\mathcal{N}(x_j, \mathcal{C})$.

Technical problem: computing the Radon–Nikodym derivative involved in the acceptance probability α (**absolute continuity of Gaussian measures in infinite dimensions**)

A modified random walk model

Preconditioned Crank–Nicolson (pCN)

$$y|x \overset{d}{=} x\,\sqrt{1-\beta^2} + \beta\,\xi$$

where $\xi \sim \mathcal{N}(0, \mathcal{C})$, and $\beta \in [0, 1]$ controls the step size

A modified random walk model

Preconditioned Crank–Nicolson (pCN)

$$y|x \stackrel{d}{=} x\,\sqrt{1-\beta^2} + \beta\,\xi$$

where $\xi \sim \mathcal{N}(0, \mathcal{C})$, and $\beta \in [0, 1]$ controls the step size

Meaning, given the current state x_j, a proposal is generated from $\mathcal{N}(x_j\,\sqrt{1-\beta^2}, \mathcal{C})$.

A modified random walk model

Preconditioned Crank–Nicolson (pCN)

$$y|x \overset{d}{=} x\,\sqrt{1-\beta^2} + \beta\,\xi$$

where $\xi \sim \mathcal{N}(0, \mathcal{C})$, and $\beta \in [0, 1]$ controls the step size

Meaning, given the current state x_j, a proposal is generated from $\mathcal{N}(x_j\,\sqrt{1-\beta^2}, \mathcal{C})$.

Setting $\mathcal{C}$ as the covariance operator of μ_0 $\left(\frac{d\mu_1}{d\mu_0} \propto \exp(-\Phi(u))\right)$, the acceptance probability is given by:

$$\alpha(u, v) = \min\left(1, \exp\left(\Phi(u) - \Phi(v)\right)\right)$$

Stochastic Langevin dynamics

Consider the following SPDE defined on an appropriate space

$$\frac{\partial u}{\partial s} = -\mathcal{K}\left(\mathcal{L}u + \gamma \nabla \Phi(u)\right) + \sqrt{2\mathcal{K}}\frac{dB}{ds} \tag{2}$$

where $\mathcal{K}$ and $\mathcal{L}$ are elliptic operators, and B is Hilbert space valued standard Brownian motion with covariance given by $\mathcal{I}$.

Stochastic Langevin dynamics

Consider the following SPDE defined on an appropriate space

$$\frac{\partial u}{\partial s} = -\mathcal{K}\left(\mathcal{L}u + \gamma \nabla \Phi(u)\right) + \sqrt{2\mathcal{K}}\frac{dB}{ds} \tag{2}$$

where $\mathcal{K}$ and $\mathcal{L}$ are elliptic operators, and B is Hilbert space valued standard Brownian motion with covariance given by $\mathcal{I}$.

- ★ We usually set $\mathcal{K} = \mathcal{I}$ or $\mathcal{L}^{-1}$.
- ★ Setting $\gamma = 0$, the above dynamics has $\mu_0^\star \equiv \mathcal{N}(0, \mathcal{L}^{-1})$ as its stationary distribution, and setting $\gamma = 1$, it has $\tilde{\mu}$ as its stationary distribution, which is given by

$$\frac{d\tilde{\mu}}{d\mu_0^\star} \propto \exp\left(-\Phi(u)\right)$$

Generating proposals from the Markov process

A Crank–Nicolson discretisation (of time parameter) of the SPDE is used to generate proposals v given the current state u by.

$$v = \left(\mathcal{I} + \frac{\delta}{2}\mathcal{K}\mathcal{L}\right)^{-1} \left[\left(\mathcal{I} - \frac{\delta}{2}\mathcal{K}\mathcal{L}\right) u - \gamma\delta\mathcal{K}\nabla\Phi(u) + \sqrt{2\mathcal{K}\,\delta}\,\xi\right] \quad (3)$$

where $\xi \sim \mathcal{N}(0, \mathcal{I})$, and δ represents the measure of finite dimensional approximation.

Generating proposals from the Markov process

A Crank–Nicolson discretisation (of time parameter) of the SPDE is used to generate proposals v given the current state u by.

$$v = \left(\mathcal{I} + \frac{\delta}{2}\mathcal{K}\mathcal{L}\right)^{-1} \left[\left(\mathcal{I} - \frac{\delta}{2}\mathcal{K}\mathcal{L}\right) u - \gamma\delta\mathcal{K}\nabla\Phi(u) + \sqrt{2\mathcal{K}\,\delta}\,\xi\right] \quad (3)$$

where $\xi \sim \mathcal{N}(0, \mathcal{I})$, and δ represents the measure of finite dimensional approximation.

Two known resultant MCMC based algorithms for drawing a sample from μ_1 are:

- pCN: preconditioned Crank–Nicolson, by setting $\gamma = 0$ in the above discretisation.
- pCNL: preconditioned Crank–Nicolson Langevin, by setting $\gamma = 1$ in the above discretisation.

What is known? – just a sample

- Hairer–Stuart–Voss (2007); Beskos–Stuart (2009); Beskos–Roberts–Stuart–Voss (2008): sampling in function spaces arising in different scenarios
- Law (2014): improving the efficiency of MCMC in function spaces
- Hairer–Stuart–Vollmer (2014): spectral gap of pCN
- Vollmer (2014): non–Gaussian priors
- Durmus–Fort–Moulines (2015): Wasserstein subgeometric ergodicity of pCN under some relaxations

Our contribution: adaptive MCMC schemes

The heuristic of adaptive MCMC

$\hookrightarrow$ The basic premise of adaptation: tuning the involved parameters, like mean and covariance to achieve efficient mixing.

The heuristic of adaptive MCMC

$\hookrightarrow$ The basic premise of adaptation: tuning the involved parameters, like mean and covariance to achieve efficient mixing.

- The choice of $\mathcal{K}$ in Lagevin dynamics is not really arbitrary, and there's some room for optimisation.

The heuristic of adaptive MCMC

$\hookrightarrow$ The basic premise of adaptation: tuning the involved parameters, like mean and covariance to achieve efficient mixing.

- The choice of $\mathcal{K}$ in Lagevin dynamics is not really arbitrary, and there's some room for optimisation.

- Often, in infinite dimensions, $\nabla\Phi$ could possibly be unavailable, and expensive to compute. Thus an estimate for $\nabla\Phi$ is a handy way to improve the proposals.

Our proposals

Our proposals

Recall: our goal is to sample from

$$\mu_1(du) \propto \exp\left(-\Phi(u, Y)\right)\ \mu_0(du)$$

where $\mu_0 \equiv \mathcal{N}(0, \mathcal{C})$, with eigen values $\{\sigma_i\}_{i \geq 1}$

In what follows, we set

$$\beta^2 = 8\delta/(2+\delta)^2$$

$$m = \text{ posterior mean}$$

$$\Lambda_\star = \text{ diag}(\{d_{\star,i}\})$$

$$d_{\star,i} = \frac{1}{\sigma_i} \times \quad i\text{-th directional variance of posterior}$$

Our proposals

Recall: our goal is to sample from

$$\mu_1(du) \propto \exp\left(-\Phi(u, Y)\right) \, \mu_0(du)$$

where $\mu_0 \equiv \mathcal{N}(0, \mathcal{C})$, with eigen values $\{\sigma_i\}_{i \geq 1}$

In what follows, we set

$$\beta^2 = 8\delta/(2+\delta)^2$$

$$m = \text{ posterior mean}$$

$$\Lambda_\star = \text{ diag}(\{d_{\star,i}\})$$

$$d_{\star,i} = \frac{1}{\sigma_i} \times \quad i\text{-th directional variance of posterior}$$

m and $\Lambda_\star$ have to be learnt from data.

Tweaking the dynamics

Inspired by "KL approximations for probability measures in infinite dimensions", by Pinski, Simpson, Stuart and Weber.

- Recall, the Langevin dynamics has two invariant measures for $\gamma = 0$ and $\gamma = 1$, respectively.
- One of them is the Gaussian measure μ with covariance $\mathcal{L}^{-1}$, and the other measure is $\tilde{\mu} \propto e^{-\Phi(u)}\,\mu$
- Thus, the true target measure can be expressed as

$$\frac{d\mu_1}{d\mu} = \frac{d\mu_1}{d\mu_0} \cdot \frac{d\mu_0}{d\mu} = e^{-\Phi(u)}\,\frac{d\mu_0}{d\mu}$$

Tweaking the dynamics

Inspired by “KL approximations for probability measures in infinite dimensions”, by Pinski, Simpson, Stuart and Weber.

- Recall, the Langevin dynamics has two invariant measures for $\gamma = 0$ and $\gamma = 1$, respectively.
- One of them is the Gaussian measure μ with covariance $\mathcal{L}^{-1}$, and the other measure is $\tilde{\mu} \propto e^{-\Phi(u)}\,\mu$
- Thus, the true target measure can be expressed as

$$\frac{d\mu_1}{d\mu} = \frac{d\mu_1}{d\mu_0} \cdot \frac{d\mu_0}{d\mu} = e^{-\Phi(u)}\,\frac{d\mu_0}{d\mu}$$

In what follows, we'll set

$$\mathcal{K} = \mathcal{C}^{1/2}\Lambda_\star \left(\mathcal{C}^{1/2}\right)^T = \mathcal{L}^{-1} \Longrightarrow \mu_0^\star \equiv \mathcal{N}\left(0, \mathcal{C}^{1/2}\Lambda_\star \left(\mathcal{C}^{1/2}\right)^T\right).$$

Proposal–1: adapting the base measure [pCNLAM]

Proposal–1: adapting the base measure [pCNLAM]

Proposal, v, given u is generated by

$$v = (1 - c_\beta)u + c_\beta m(u) + \beta \mathcal{C}^{1/2}\Lambda_\star^{1/2}\xi,$$

where $m(u) = u - \mathcal{C}^{1/2}\Lambda_\star((\mathcal{C}^{1/2})^T\nabla\Phi(u) + \mathcal{C}^{-1/2}u)$ and $c_\beta = 1 - (1-\beta^2)^{1/2}$.

Proposal–1: adapting the base measure [pCNLAM]

Proposal, v, given u is generated by

$$v = (1 - c_\beta)u + c_\beta m(u) + \beta \mathcal{C}^{1/2}\Lambda_\star^{1/2}\xi,$$

where $m(u) = u - \mathcal{C}^{1/2}\Lambda_\star((\mathcal{C}^{1/2})^T \nabla\Phi(u) + \mathcal{C}^{-1/2}u)$ and $c_\beta = 1 - (1 - \beta^2)^{1/2}$.

The Metropolis–Hastings ratio depends on

$$\begin{aligned} \mathcal{J}_{pCNL_{AM}}(v,u) \quad = \quad & \Phi(u) - \Phi(v) + \frac{1}{2}\langle v, (\Lambda_\star^{-1} - \mathcal{I})v\rangle_{\mathcal{C}} - \frac{1}{2}\langle u, (\Lambda_\star^{-1} - \mathcal{I})u\rangle_{\mathcal{C}} \\ & -\frac{c_\beta}{\beta^2}\langle m(u), v - (1 - c_\beta)u - \frac{1}{2}c_\beta m(u)\rangle_{\mathcal{K}} \\ & +\frac{c_\beta}{\beta^2}\langle m(v), u - (1 - c_\beta)v - \frac{1}{2}c_\beta m(v)\rangle_{\mathcal{K}}. \end{aligned}$$

Proposal–2: adapting the base measure and mean [pCNAM]

Set m as the posterior expectation. Then, the proposal is generated by:

$$v = (1 - c_\beta)u + c_\beta m + \beta \mathcal{C}^{1/2}\Lambda_\star^{1/2}\xi,$$

where $c_\beta = 1 - (1 - \beta^2)^{1/2}$.

The corresponding Metropolis–Hastings ratio is

$$\begin{aligned} \mathcal{J}_{pCN_{AM}}(v,u) &= \Phi(u) - \Phi(v) + \frac{1}{2}\langle v, (\Lambda_\star^{-1} - \mathcal{I})v\rangle_{\mathcal{C}} \\ &\quad - \frac{1}{2}\langle u, (\Lambda_\star^{-1} - \mathcal{I})u\rangle_{\mathcal{C}} - \langle (v-u), m\rangle_{\mathcal{K}}. \end{aligned}$$

All these can be simplified by using the eigen basis of $\mathcal{C}$

The adaptation

In the above proposals, we also invoke an adaptation step:

$$\begin{aligned} \widehat{m}_k^{(j)} &= w_j u_k + (1 - w_j)\widehat{m}_{k-1}^{(j)} \\ \widehat{d}_k^{(j)} &= \frac{w_j}{\sigma_j}(u_k - \widehat{m}^{(j)} k)^2 + (1 - w_j)\widehat{d}_{k-1}^{(j)} \end{aligned}$$

where j denotes the dimension, k denotes the time step, and $w_j = 1/j$

How good are these proposals? (ergodicity)

- Very few results available on ergodicity of MCMC samplers in function spaces.

How good are these proposals? (ergodicity)

- Very few results available on ergodicity of MCMC samplers in function spaces.
- Known ergodicity results for finite dimensional setup break down in function spaces [Hairer–Stuart–Voss (2014)]
- The set is empty for ergodicity of adaptive MCMC samplers in function spaces – (ongoing work)

How good are these proposals? (ergodicity)

- Very few results available on ergodicity of MCMC samplers in function spaces.
- Known ergodicity results for finite dimensional setup break down in function spaces [Hairer–Stuart–Voss (2014)]
- The set is empty for ergodicity of adaptive MCMC samplers in function spaces – (ongoing work)
- A bit of cheating: in practice, samplers are not run for infinite time, and thus adaptation can be stopped at a predefined time step – ergodicity can then be proven in certain cases

How good are these proposals? (ergodicity)

- Very few results available on ergodicity of MCMC samplers in function spaces.
- Known ergodicity results for finite dimensional setup break down in function spaces [Hairer–Stuart–Voss (2014)]
- The set is empty for ergodicity of adaptive MCMC samplers in function spaces – (ongoing work)
- A bit of cheating: in practice, samplers are not run for infinite time, and thus adaptation can be stopped at a predefined time step – ergodicity can then be proven in certain cases
- Proposals with extra drift term need extra care.

How good are these proposals? (example)

Binomial random field

Let

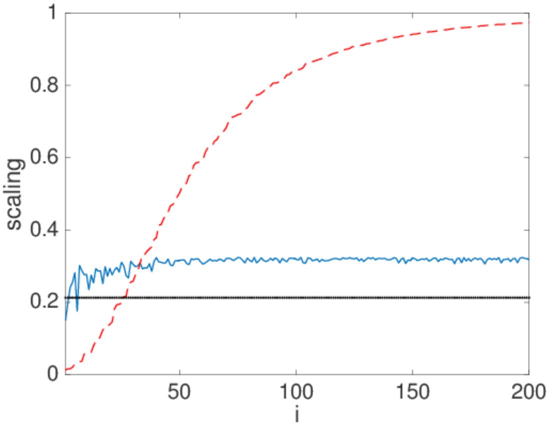
$$U \sim \mathcal{N}\left(0, \sigma^2(\kappa^2 - \Delta)^{\alpha}\right)$$

and the observations be

$$Y_i \sim \mathrm{Bin}\left(n_i,\ \mathrm{logit}(U(s_i))\right)$$

for $i = 1, \ldots, N$, and $(n_i - 1) \sim \mathrm{Poisson}(1)$.

Goal: To sample from the posterior: U given $\{Y_i\}$



The red dashed line is the scaling for mGrad, the blue full line is the scaling of the eigen basis for pCNLAM and the dotted black line is for the pCN. Both figures are generated by sampling 100k samples.

Performance

Method	min (ESS)/s	min (ESS)/iter	median(ESS)/s	median(ESS)/iter	β
pCN	0.5	0.0001	2.0	0.0005	0.10
pCNAM	1.5	0.004	4.5	0.0014	0.18
pCNLAM	5.4	0.0024	14.5	0.006	0.24
mGrad	0.3	0.0006	4.3	0.009	0.70
pCNLHM	5.2	0.024	6.0	0.028	0.49

As per the status quo, mGrad is known to be one of the best adaptive schemes (Titsias–Papaspiliopoulos, 2018).

WAIT!... Aren't all numerically implementable algorithms essentially finite dimensional? So, why talk about infinite dimensional samplers?

WAIT!... Aren't all numerically implementable algorithms essentially finite dimensional? So, why talk about infinite dimensional samplers?

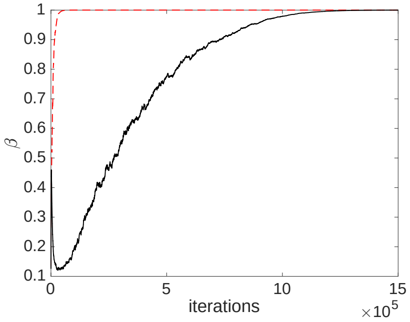
Good question!

WAIT!... Aren't all numerically implementable algorithms essentially finite dimensional? So, why talk about infinite dimensional samplers?

Good question! Of course, to implement our proposals, we truncate onto some finitely many Karhunen–Loéve basis elements. We show that for a fixed acceptance probability, our proposal pCNAM, shows faster convergence of β to 1, which is reflective of faster mixing.

WAIT!... Aren't all numerically implementable algorithms essentially finite dimensional? So, why talk about infinite dimensional samplers?

Good question! Of course, to implement our proposals, we truncate onto some finitely many Karhunen–Loéve basis elements. We show that for a fixed acceptance probability, our proposal pCNAM, shows faster convergence of β to 1, which is reflective of faster mixing.



The figure depicts the development of β at a fixed acceptance rate of 0.234, for the truncated estimation method (red dashed), and the untruncated estimation method (black full).

Thank you!!