Bayesian inference for complex models

Lecture 2

Aretha Teckentrup

School of Mathematics, University of Edinburgh

SFB1294 Data assimilation Spring school - March 21, 2022

THE UNIVERSITY *of* EDINBURGH
School of Mathematics

Outline of today's lecture

1. Brief recap
2. Markov chain Monte Carlo methods
3. Ratio estimators

Brief recap

Bayesian approach to inverse problems [Kaipio, Somersalo '04] [Stuart '10]

- We are interested in the following inverse problem: given observational data $y \in \mathbb{R}^{d_y}$, determine unknown parameter $u \in \mathbb{R}^{d_u}$ such that

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

where $\eta \sim \mathcal{N}(0, \gamma^2 I)$ represents observational noise.

Brief recap

Bayesian approach to inverse problems [Kaipio, Somersalo '04] [Stuart '10]

- We are interested in the following inverse problem: given observational data $y \in \mathbb{R}^{d_y}$, determine unknown parameter $u \in \mathbb{R}^{d_u}$ such that

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

where $\eta \sim \mathcal{N}(0, \gamma^2 I)$ represents observational noise.

- In the Bayesian approach, the solution to the inverse problem is the posterior density π^y on $\mathbb{R}^{d_u}$, given by

$$\underbrace{\pi^y(u)}_{p(u|y)} = \underbrace{\frac{1}{Z}}_{1/p(y)} \overbrace{\exp\big(-\Phi(u;y)\big)}^{p(y|u)} \underbrace{\pi_0(u)}_{p(u)},$$

where $Z = \int_{\mathbb{R}^{d_u}} \exp\big(-\Phi(u;y)\big)\pi_0(u)du = \mathbb{E}_{\pi_0}\Big(\exp\big(-\Phi(\cdot;y)\big)\Big)$ and $\Phi(u;y) = \frac{1}{2\gamma^2}\|y - \mathcal{G}(u)\|_2^2$.

Brief recap

Metropolis Hastings algorithm [Robert, Casella '99]

- We are interested in computing the posterior mean $\mathbb{E}_{\pi^y}[u]$, or more generally a quantity of interest $\mathbb{E}_{\pi^y}[\phi(u)]$.
- We can use Metropolis Hastings: $\mathbb{E}_{\pi^y}[\phi(u)] \approx \frac{1}{N}\sum_{i=1}^{N}\phi(u^{(i)})$.

ALGORITHM 1. (Metropolis Hastings)

- Choose $u^{(1)}$ with $\pi^y(u^{(1)}) > 0$.
- At state $u^{(i)}$, sample a proposal u' from density $q(u' \,|\, u^{(i)})$.
- Accept sample u' with probability

$$\alpha(u' \,|\, u^{(i)}) = \min\left(1, \frac{\pi^y(u')\, q(u^{(i)} \,|\, u')}{\pi^y(u^{(i)})\, q(u' \,|\, u^{(i)})}\right),$$

i.e. $u^{(i+1)} = u'$ with probability $\alpha(u' \,|\, u^{(i)})$; otherwise stay at $u^{(i+1)} = u^{(i)}$.

Brief recap

Metropolis Hastings algorithm [Robert, Casella '99]

- We are interested in computing the posterior mean $\mathbb{E}_{\pi^y}[u]$, or more generally a quantity of interest $\mathbb{E}_{\pi^y}[\phi(u)]$.
- We can use Metropolis Hastings: $\mathbb{E}_{\pi^y}[\phi(u)] \approx \frac{1}{N}\sum_{i=1}^{N}\phi(u^{(i)})$.

ALGORITHM 1. (Metropolis Hastings)

- Choose $u^{(1)}$ with $\pi^y(u^{(1)}) > 0$.
- At state $u^{(i)}$, sample a proposal u' from density $q(u' \,|\, u^{(i)})$.
- Accept sample u' with probability

$$\alpha(u' \,|\, u^{(i)}) = \min\left(1, \frac{\pi^y(u')\, q(u^{(i)} \,|\, u')}{\pi^y(u^{(i)})\, q(u' \,|\, u^{(i)})}\right),$$

  i.e. $u^{(i+1)} = u'$ with probability $\alpha(u' \,|\, u^{(i)})$; otherwise stay at $u^{(i+1)} = u^{(i)}$.

- A central limit theorem gives $\mathbb{E}[\sum_{i=1}^{N}\phi(u^{(i)})] \approx \mathbb{E}_{\pi^y}[\phi(u)]$ and $\mathbb{V}[\sum_{i=1}^{N}\phi(u^{(i)})] \approx \frac{\sigma_\phi^2}{N}$ for large N.

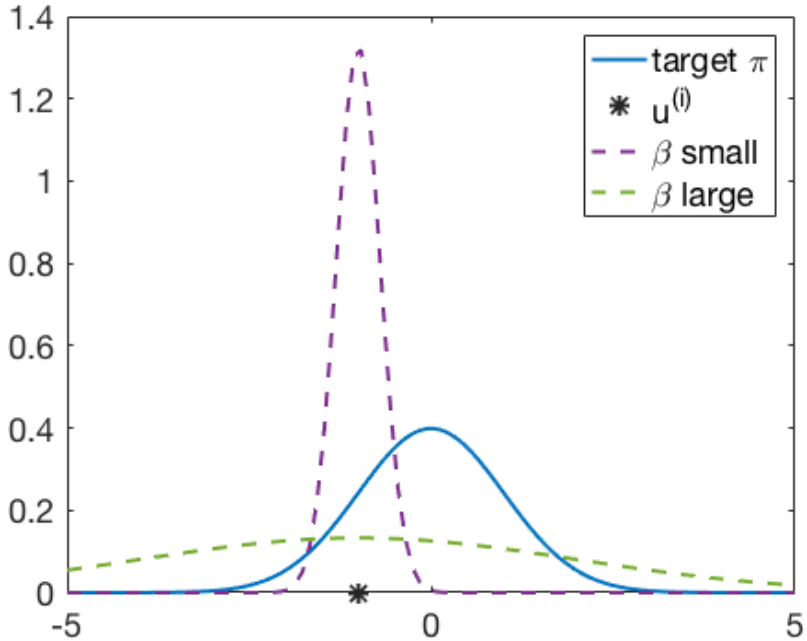
Brief recap

Random walk proposal [Roberts et al, '97]

The random walk proposal is given by $q(u' \mid u^{(i)}) = \mathcal{N}(u^{(i)}, \beta^2 \mathrm{I})$, for some $\beta > 0$, i.e.

$$u' = u^{(i)} + \beta \xi_i, \qquad \text{where} \qquad \xi_i \sim \mathcal{N}(0, \mathrm{I}), \quad \beta > 0,$$

- Here, β is a step size that needs to be tuned:
  - if β is too small, you don't explore the state space.
  - if β is too large, you reject too often.
  - both scenarios lead to large asymptotic variance σ_ϕ^2.



- A general rule of thumb is to tune β such that the average acceptance probability is $\alpha \approx 0.234$, to achieve small asymptotic variance.

Markov chain Monte Carlo methods

Challenge 1: High dimensional u

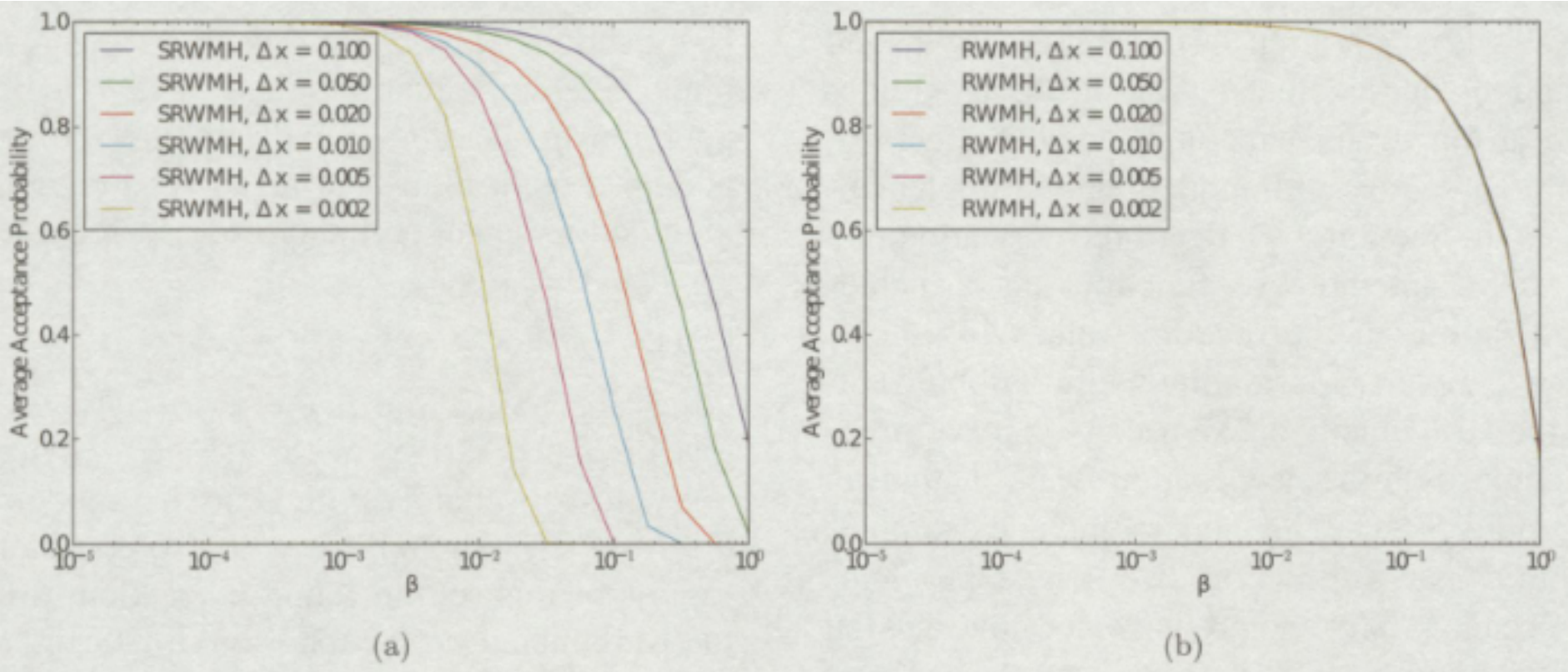
In many modern applications, the unknown u is very high-dimensional:

- In imaging applications, the dimension of u is the number of pixels. For a 640×480 image, this gives $u \in \mathbb{R}^{307200}$.
- In PDE applications, the parameters u are often used to model the diffusion coefficient $k(x)$.
  - When using a finite element method with step size Δx to solve the PDE, $k(x)$ if often modelled as piece-wise constant on each element. In two spatial dimensions, we have $d_u = (\Delta x)^{-2} = 256^2 = 65536$.
  - $k(x)$ often has low spatial regularity and large variations, which means that a large number of terms are required in Fourier-type expansions.

Markov chain Monte Carlo methods

Challenge 1: High dimensional u [Cotter et al, '13]

- Challenge in high dimensions: for any β, the average acceptance rate $\alpha \to 0$ as $d \to \infty$. $\Rightarrow \sigma_\phi^2 \to \infty$ as $d \to \infty$
- Example in discretised PDE, with $d_u = (\Delta x)^{-2}$:



Markov chain Monte Carlo methods

Pre-conditioned Crank-Nicolson (pCN) [Cotter et al '13]

- The pre-conditioned Crank-Nicolson (pCN) proposal is well-defined in the infinite-dimensional setting d_u. $\Rightarrow \sigma_\phi^2$ independent of d_u

- The specific form of the pCN proposal depends on the prior π_0. If π_0 is $\mathcal{N}(0, C_0)$, then $q(u' \,|\, u^{(i)})$ is defined by

$$u' = \sqrt{1-\beta^2}\, u^{(i)} + \beta \xi_i, \qquad \text{where } \xi_i \sim \mathcal{N}(0, C_0), \quad \beta \in [0,1].$$

β is a step size parameter that needs to be tuned.

Markov chain Monte Carlo methods

Pre-conditioned Crank-Nicolson (pCN) [Cotter et al '13]

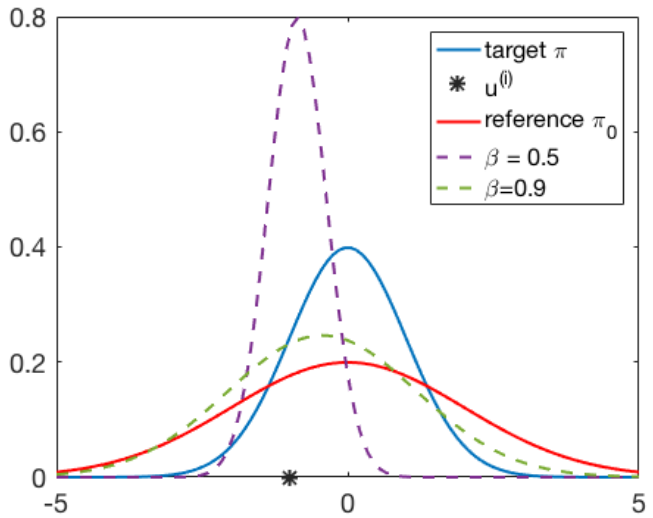
- The pre-conditioned Crank-Nicolson (pCN) proposal is well-defined in the infinite-dimensional setting d_u. $\Rightarrow \sigma_\phi^2$ independent of d_u

- The specific form of the pCN proposal depends on the prior π_0. If π_0 is $\mathcal{N}(0, C_0)$, then $q(u' \mid u^{(i)})$ is defined by

$$u' = \sqrt{1-\beta^2}\, u^{(i)} + \beta \xi_i, \qquad \text{where } \xi_i \sim \mathcal{N}(0, C_0), \quad \beta \in [0,1].$$

  β is a step size parameter that needs to be tuned.

- The same heuristic to tune β such that the average acceptance probability is $\alpha \approx 0.234$ is often used.
  $\beta \sim 1$



Markov chain Monte Carlo methods

Pre-conditioned Crank-Nicolson (pCN) [Cotter et al '13]

- The pCN proposal is π_0-reversible, i.e.

$$\pi_0(u^{(i)})\, q(u' \,|\, u^{(i)}) = \pi_0(u')\, q(u^{(i)} \,|\, u').$$

- The acceptance probability then becomes

$$\alpha(u'|u^{(i)}) = \min\left(1, \frac{L(y|u')\, \pi_0(u')\, q(u^{(i)} \,|\, u')}{L(y|u^{(i)})\, \pi_0(u^{(i)})\, q(u' \,|\, u^{(i)})}\right),$$

which depends on u' only through its likelihood $L(y|u')$.

Markov chain Monte Carlo methods

Pre-conditioned Crank-Nicolson (pCN) [Cotter et al '13]

- The pCN proposal is π_0-reversible, i.e.

$$\pi_0(u^{(i)})\, q(u' \,|\, u^{(i)}) = \pi_0(u')\, q(u^{(i)} \,|\, u').$$

- The acceptance probability then becomes

$$\alpha(u'|u^{(i)}) = \min\left(1, \frac{L(y|u')\, \pi_0(u')\, q(u^{(i)} \,|\, u')}{L(y|u^{(i)})\, \pi_0(u^{(i)})\, q(u' \,|\, u^{(i)})}\right),$$

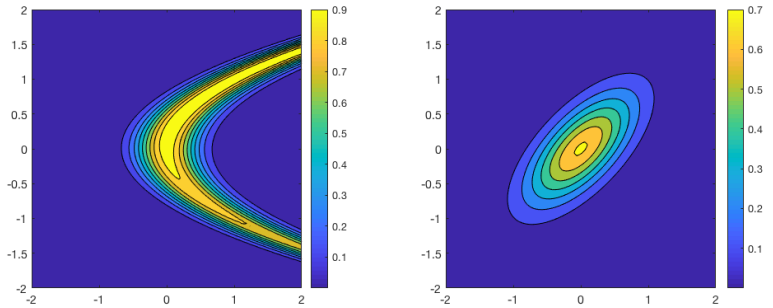
  which depends on u' only through its likelihood $L(y|u')$.

- The prior density $\pi_0(u) \propto e^{-u^{\mathrm{T}} C_0^{-1} u}$ becomes ill-defined in infinite dimensions.

Markov chain Monte Carlo methods

How to incorporate gradient information?

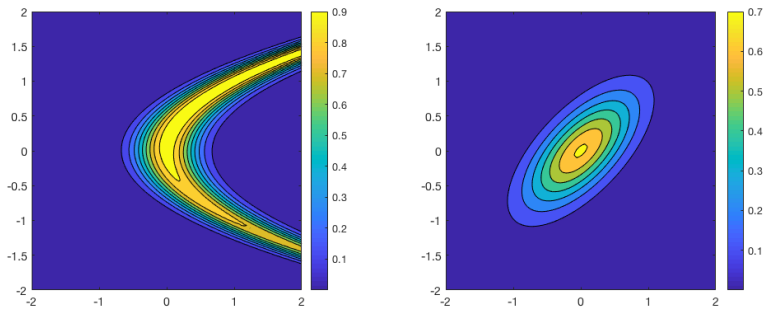
- The proposals we have seen so far are agnostic about which parts of state space are more probable.
- Ideally we would like proposals that take this into account ($\Rightarrow$ make it more probable to move to areas where π is large).



Markov chain Monte Carlo methods

How to incorporate gradient information?

- The proposals we have seen so far are agnostic about which parts of state space are more probable.
- Ideally we would like proposals that take this into account ($\Rightarrow$ make it more probable to move to areas where π is large).



- Connecting to optimisation, a possible way to do this is to use gradient information and propose the next move in the following way

$$u' = u^{(i)} + \beta \nabla \pi(u^{(i)})$$

1. This is deterministic move (we are losing randomness, and the ability to explore the state space, as we would converge to a local maximum).
2. How can we do this properly?

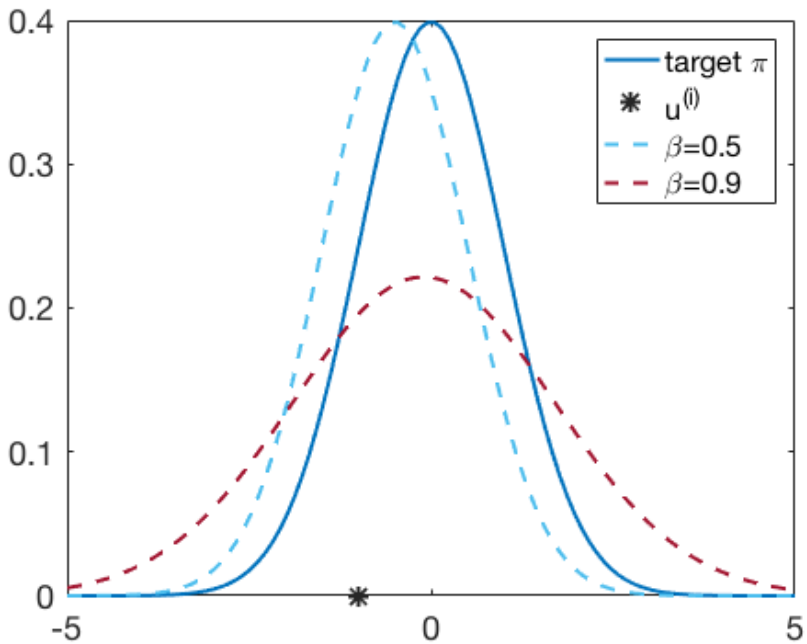
Markov chain Monte Carlo methods

Metropolis adjusted Langevin algorithm (MALA) [Pillai, Stuart, Thiery '12]

- MALA: $q(u'|u^{(i)}) = \mathcal{N}(u^{(i)} + \beta\nabla \log \pi^y(u^{(i)}), 2\beta \mathrm{I})$, for some $\beta > 0$, i.e.

$$u' = u^{(i)} + \beta\nabla \log \pi^y(u^{(i)}) + \sqrt{2\beta}\xi_i, \qquad \text{where} \qquad \xi_i \sim \mathcal{N}(0, \mathrm{I})$$

- For optimal efficiency, the step size β should be tuned to obtain an average acceptance rate of $\alpha \approx 0.574$. $\Rightarrow \beta \sim d_u^{-1/3}$



Markov chain Monte Carlo methods

Metropolis adjusted Langevin algorithm (MALA)

- MALA: $u' = u^{(i)} + \beta \nabla \log \pi^y(u^{(i)}) + \sqrt{2\beta}\xi_i$, with $\xi_i \sim \mathcal{N}(0, \mathrm{I})$.
- Note that this is one Euler-Maruyama step of the Langevin SDE

$$dX = \nabla \log \pi^y(X)dt + \sqrt{2}dW$$

$$X_{n+1} = X_n + \Delta t \nabla \log \pi(X_n) + \sqrt{2\Delta t}\xi_n$$

- The stationary distribution of the Langevin SDE is π^y.
  - If $u^{(i)} \sim \pi^y$ and the Euler-Maruyama method is exact, then $u' \sim \pi^y$.

Markov chain Monte Carlo methods

Metropolis adjusted Langevin algorithm (MALA)

- MALA: $u' = u^{(i)} + \beta \nabla \log \pi^y(u^{(i)}) + \sqrt{2\beta}\xi_i$, with $\xi_i \sim \mathcal{N}(0, \mathrm{I})$.
- Note that this is one Euler-Maruyama step of the Langevin SDE

$$dX = \nabla \log \pi^y(X)dt + \sqrt{2}dW$$

$$X_{n+1} = X_n + \Delta t \nabla \log \pi(X_n) + \sqrt{2\Delta t}\xi_n$$

- The stationary distribution of the Langevin SDE is π^y.
  - If $u^{(i)} \sim \pi^y$ and the Euler-Maruyama method is exact, then $u' \sim \pi^y$.
- See e.g. [Girolami, Calderhead '11] and [Cui, Law, Marzouk '16] for further work on including second order information.

Markov chain Monte Carlo methods

Non-reversible methods [Ottobre '16]

- The Markov chain $\{u^{(i)}\}_{i=1}^{N}$ produced by Metropolis-Hastings is reversible, since it satisfies detailed balance:

$$\pi^y(u^{(i)})\, q(u' \,|\, u^{(i)}) = \pi^y(u')\, q(u^{(i)} \,|\, u').$$

- Non-reversible Markov chains have been recently shown potential to converge to equilibrium faster, and to reduce the asymptotic variance.

- A non-reversible version of Metropolis-Hastings can be constructed by introducing a preferred direction into the proposal.

Markov chain Monte Carlo methods

I-Jump sampler [Ma et al '19]

ALGORITHM 2. (I-Jump)

- Choose $u^{(1)}$ with $\pi^y(u^{(1)}) > 0$, and pick $z^{(1)} \in \{-1, 1\}$ uniformly.
- At state $u^{(i)}$, if $z^{(i)} > 0$:
  - sample a proposal u' from density $q_1(u' \,|\, u^{(i)})$, and set

$$\alpha(u' \,|\, u^{(i)}) = \min\left(1, \frac{\pi^y(u')\, q_2(u^{(i)} \,|\, u^*)}{\pi^y(u^{(i)})\, q_1(u' \,|\, u^{(i)})}\right),$$

- At state $u^{(i)}$, if $z^{(i)} < 0$:
  - sample a proposal u' from density $q_2(u' \,|\, u^{(i)})$, and set

$$\alpha_2(u' \,|\, u^{(i)}) = \min\left(1, \frac{\pi^y(u')\, q_1(u^{(i)} \,|\, u^*)}{\pi^y(u^{(i)})\, q_2(u' \,|\, u^{(i)})}\right),$$

- With probability $\alpha(u'|u^{(i)})$, set $u^{(i+1)} = u'$; otherwise stay at $u^{(i+1)} = u^{(i)}$ and update $z^{(i+1)} = -z^{(i)}$.

Markov chain Monte Carlo methods

I-Jump sampler [Ma et al '19]

- You are free to choose q_1 and q_2, but a typical choice is directed random walks. In one dimension:

$$
\begin{aligned}
q_1: &\quad u' = u^{(i)} + \gamma_i, \quad \gamma_i \sim \Gamma(\alpha, \beta), \\
q_2: &\quad u' = u^{(i)} - \gamma_i, \quad \gamma_i \sim \Gamma(\alpha, \beta),
\end{aligned}
$$

Markov chain Monte Carlo methods

I-Jump sampler [Ma et al '19]

- You are free to choose q_1 and q_2, but a typical choice is directed random walks. In one dimension:

$$\begin{aligned} q_1: &\quad u' = u^{(i)} + \gamma_i, \quad \gamma_i \sim \Gamma(\alpha, \beta), \\ q_2: &\quad u' = u^{(i)} - \gamma_i, \quad \gamma_i \sim \Gamma(\alpha, \beta), \end{aligned}$$

- In higher dimensions, the algorithm requires a direction vector $y^{(i)} \in \mathbb{R}^{d_u}$:

$$\begin{aligned} q_1: &\quad u' = u^{(i)} + \gamma_i y^{(i)}, \quad \gamma_i \sim \Gamma(\alpha, \beta), \\ q_2: &\quad u' = u^{(i)} - \gamma_i y^{(i)}, \quad \gamma_i \sim \Gamma(\alpha, \beta), \end{aligned}$$

- As part of the algorithm, you may want to periodically resample $y^{(i)}$.

Markov chain Monte Carlo methods

Challenge 2: Computationally expensive likelihood

In many modern applications, the likelihood $e^{-\frac{1}{2\gamma^2}\|y-\mathcal{G}(u)\|_2^2}$ is very expensive to evaluate for given u:

- If the data y is high-dimensional, evaluating $\|y - \mathcal{G}(u)\|_2$ becomes time-consuming.
- In PDE applications, the evaluation of $\mathcal{G}(u)$ requires the numerical solution of the PDE.

Markov chain Monte Carlo methods

Challenge 2: Computationally expensive likelihood

In many modern applications, the likelihood $e^{-\frac{1}{2\gamma^2}\|y-\mathcal{G}(u)\|_2^2}$ is very expensive to evaluate for given u:

- If the data y is high-dimensional, evaluating $\|y - \mathcal{G}(u)\|_2$ becomes time-consuming.
- In PDE applications, the evaluation of $\mathcal{G}(u)$ requires the numerical solution of the PDE.

The methods presented earlier help in this context since they reduce the asymptotic variance σ_ϕ^2, and hence the number of required samples N.

But it is further possible to reduce the cost per sample.

Markov chain Monte Carlo methods

Surrogate transition method [Liu '01]

- The surrogate transition method uses a surrogate posterior $\pi^*(u)$ to pre-screen proposals.

ALGORITHM 3. (Surrogate Transition method)

- At state $u^{(i)}$, sample a proposal u^* from density $q^*(u^* \,|\, u^{(i)})$.
- Set $u' = u^*$ with probability

$$\alpha_1(u^* \,|\, u^{(i)}) = \min\left(1, \frac{\pi^*(u')\, q^*(u^{(i)} \,|\, u^*)}{\pi^*(u^{(i)})\, q^*(u^* \,|\, u^{(i)})}\right),$$

  otherwise $u' = u^{(i)}$. Denote $u' \sim q(u'|u^{(i)})$.
- Accept u' with probability

$$\alpha_2(u' \,|\, u^{(i)}) = \min\left(1, \frac{\pi^y(u')\, q(u^{(i)} \,|\, u')}{\pi^y(u^{(i)})\, q(u' \,|\, u^{(i)})}\right) = \min\left(1, \frac{\pi^y(u')\, \pi^*(u^{(i)})}{\pi^y(u^{(i)})\, \pi^*(u')}\right),$$

  i.e. $u^{(i+1)} = u'$ with probability $\alpha_2(u' \,|\, u^{(i)})$; otherwise stay at $u^{(i+1)} = u^{(i)}$.

Markov chain Monte Carlo methods

Surrogate transition method [Liu '01]

- The surrogate transition method uses a surrogate posterior $\pi^*(u)$ to pre-screen proposals.

ALGORITHM 3. (Surrogate Transition method)

- At state $u^{(i)}$, sample a proposal u^* from density $q^*(u^* \,|\, u^{(i)})$.
- Set $u' = u^*$ with probability

$$\alpha_1(u^* \,|\, u^{(i)}) = \min\left(1, \frac{\pi^*(u')\, q^*(u^{(i)} \,|\, u^*)}{\pi^*(u^{(i)})\, q^*(u^* \,|\, u^{(i)})}\right),$$

  otherwise $u' = u^{(i)}$. Denote $u' \sim q(u'|u^{(i)})$.
- Accept u' with probability

$$\alpha_2(u' \,|\, u^{(i)}) = \min\left(1, \frac{\pi^y(u')\, q(u^{(i)} \,|\, u')}{\pi^y(u^{(i)})\, q(u' \,|\, u^{(i)})}\right) = \min\left(1, \frac{\pi^y(u')\, \pi^*(u^{(i)})}{\pi^y(u^{(i)})\, \pi^*(u')}\right),$$

  i.e. $u^{(i+1)} = u'$ with probability $\alpha_2(u' \,|\, u^{(i)})$; otherwise stay at $u^{(i+1)} = u^{(i)}$.

- We evaluate $L(y|u')$ only for proposals that were accepted for π^*.

Ratio estimators

Computing expectations

- We now focus on alternative methods to Markov chain Monte Carlo.

Ratio estimators

Computing expectations

- We now focus on alternative methods to Markov chain Monte Carlo.
- In most cases, we do not have a closed form expression for the posterior distribution π^y, since the normalising constant Z is not known explicitly.
- However, the prior distribution is known in closed form, and furthermore often has a simple structure (e.g. multivariate Gaussian or independent uniform).

Ratio estimators

Computing expectations

Using Bayes' Theorem, we can write $\mathbb{E}_{\pi^y}[\phi]$ as

$$\begin{aligned}
\mathbb{E}_{\pi^y}[\phi] &= \int_{\mathbb{R}^{d_u}} \phi(u)\, \pi^y(u)\, du \\
&= \int_{\mathbb{R}^{d_u}} \phi(u) \frac{\pi^y(u)}{\pi_0(u)}\, \pi_0(u)\, du \leftarrow \text{importance sampling} \\
&= \frac{1}{Z} \int_{\mathbb{R}^{d_u}} \phi(u) \exp[-\Phi(u; y)]\, \pi_0(u)\, du \\
&= \frac{\mathbb{E}_{\pi_0}[\phi \exp[-\Phi(\cdot; y)]]}{\mathbb{E}_{\pi_0}[\exp[-\Phi(\cdot; y)]]}.
\end{aligned}$$

Ratio estimators

Computing expectations

Using Bayes' Theorem, we can write $\mathbb{E}_{\pi^y}[\phi]$ as

$$\begin{aligned}
\mathbb{E}_{\pi^y}[\phi] &= \int_{\mathbb{R}^{d_u}} \phi(u)\, \pi^y(u)\, du \\
&= \int_{\mathbb{R}^{d_u}} \phi(u) \frac{\pi^y(u)}{\pi_0(u)}\, \pi_0(u)\, du \leftarrow \text{importance sampling} \\
&= \frac{1}{Z} \int_{\mathbb{R}^{d_u}} \phi(u) \exp[-\Phi(u;y)]\, \pi_0(u)\, du \\
&= \frac{\mathbb{E}_{\pi_0}[\phi \exp[-\Phi(\cdot;y)]]}{\mathbb{E}_{\pi_0}[\exp[-\Phi(\cdot;y)]]}.
\end{aligned}$$

We have rewritten the posterior expectation as a ratio of two prior expectations.

We can now use different methods to estimate the two prior expectations, e.g. Monte Carlo based methods.

Ratio estimators

Standard Monte Carlo method

- The standard Monte Carlo method is a *sampling method*.
- To estimate $\mathbb{E}_{\pi_0}[f]$, for some $f : \mathbb{R}^{d_u} \to \mathbb{R}$, sampling methods use a sample average:

$$\mathbb{E}_{\pi_0}[f] = \int_{\mathbb{R}^{d_u}} f(u)\, \pi_0(u) du \approx \sum_{i=1}^{N} w_i\, f(u^{(i)}),$$

where the choice of samples $\{u^{(i)}\}_{i=1}^N$ and weights $\{w_i\}_{i=1}^N$ determines the sampling method.

Ratio estimators

Standard Monte Carlo method

- The standard Monte Carlo method is a *sampling method*.
- To estimate $\mathbb{E}_{\pi_0}[f]$, for some $f : \mathbb{R}^{d_u} \to \mathbb{R}$, sampling methods use a sample average:

$$\mathbb{E}_{\pi_0}[f] = \int_{\mathbb{R}^{d_u}} f(u)\, \pi_0(u) du \approx \sum_{i=1}^{N} w_i\, f(u^{(i)}),$$

  where the choice of samples $\{u^{(i)}\}_{i=1}^N$ and weights $\{w_i\}_{i=1}^N$ determines the sampling method.
- In standard Monte Carlo, $w_i = \frac{1}{N}$ and $\{u^{(i)}\}_{i=1}^N$ is a sequence of independent and identically distributed (i.i.d.) random variables: $\{u^{(i)}\}_{i=1}^N$ are mutually independent and $u^{(i)} \sim \pi_0$, for all $1 \leq i \leq N$.
- Since π_0 is fully known and simple, i.i.d. samples from π_0 can be generated on a computer using a (pseudo-)random number generator. For more details, see [Robert, Casella '99], [L'Ecuyer '11].

Ratio estimators

Definition of Monte Carlo ratio estimator

- In the Bayesian inverse problem, we want to compute

$$\mathbb{E}_{\pi^y}[\phi] = \frac{\mathbb{E}_{\pi_0}[\phi \exp[-\Phi(\cdot; y)]]}{\mathbb{E}_{\pi_0}[\exp[-\Phi(\cdot; y)]]}.$$

- Using Monte Carlo, we approximate this by

$$\mathbb{E}_{\pi_0}[\phi \exp[-\Phi(\cdot; y)]] \approx \frac{1}{N} \sum_{i=1}^{N} \phi(u^{(i)}) \exp[-\Phi(u^{(i)}; y)],$$

$$\mathbb{E}_{\pi_0}[\exp[-\Phi(\cdot; y)]] \approx \frac{1}{N} \sum_{i=1}^{N} \exp[-\Phi(u^{(i)}; y)],$$

where $\{u^{(i)}\}_{i=1}^{N}$ is an i.i.d. sequence distributed according to π_0.

(It is also possible to use different samples in the two estimators.)

Ratio estimators

Expected value and variance of Monte Carlo

Consider a general Monte Carlo estimator $\widehat{E}_N^{\mathrm{MC}} = \frac{1}{N}\sum_{i=1}^{N} f(u^{(i)})$, with $\{u^{(i)}\}_{i=1}^{N}$ an i.i.d. sequence distributed as π_0.

Ratio estimators

Expected value and variance of Monte Carlo

Consider a general Monte Carlo estimator $\widehat{E}_N^{\mathrm{MC}} = \frac{1}{N}\sum_{i=1}^N f(u^{(i)})$, with $\{u^{(i)}\}_{i=1}^N$ an i.i.d. sequence distributed as π_0.

Lemma (Expected Value and Variance)

$$\mathbb{E}[\widehat{E}_N^{\mathrm{MC}}] = \mathbb{E}_{\pi_0}[f], \qquad \mathbb{V}[\widehat{E}_N^{\mathrm{MC}}] = \frac{\mathbb{V}_{\pi_0}[f]}{N}.$$

Proof: Since $\{u^{(i)}\}_{i=1}^N$ is an i.i.d. sequence, we have

$$\mathbb{E}\Big[\frac{1}{N}\sum_{i=1}^N f(u^{(i)})\Big] = \frac{1}{N}\mathbb{E}\Big[\sum_{i=1}^N f(u^{(i)})\Big] = \frac{1}{N}\sum_{i=1}^N \mathbb{E}_{\pi_0}[f] = \mathbb{E}_{\pi_0}[f],$$

and

$$\mathbb{V}\Big[\frac{1}{N}\sum_{i=1}^N f(u^{(i)})\Big] = \frac{1}{N^2}\mathbb{V}\Big[\sum_{i=1}^N f(u^{(i)})\Big] = \frac{1}{N^2}\sum_{i=1}^N \mathbb{V}_{\pi_0}[f] = \frac{1}{N}\mathbb{V}_{\pi_0}[f].$$

Ratio estimators

Mean square error of Monte Carlo

A measure of accuracy of $\widehat{E}_N^{\mathrm{MC}} = \frac{1}{N}\sum_{i=1}^{N} f(u^{(i)})$ as an estimator of $\mathbb{E}_{\pi_0}[f]$ is given by the mean square error (MSE):

$$e(\widehat{E}_N^{\mathrm{MC}})^2 := \mathbb{E}[(\widehat{E}_N^{\mathrm{MC}} - \mathbb{E}_{\pi_0}[f])^2].$$

Ratio estimators

Mean square error of Monte Carlo

A measure of accuracy of $\widehat{E}_N^{\mathrm{MC}} = \frac{1}{N}\sum_{i=1}^N f(u^{(i)})$ as an estimator of $\mathbb{E}_{\pi_0}[f]$ is given by the mean square error (MSE):

$$e(\widehat{E}_N^{\mathrm{MC}})^2 := \mathbb{E}[(\widehat{E}_N^{\mathrm{MC}} - \mathbb{E}_{\pi_0}[f])^2].$$

Lemma (Mean Square Error)

$$e(\widehat{E}_N^{\mathrm{MC}})^2 = \frac{\mathbb{V}_{\pi_0}[f]}{N}.$$

Proof: Since $\mathbb{E}[\widehat{E}_N^{\mathrm{MC}}] = \mathbb{E}_{\pi_0}[f]$ and $\mathbb{V}[\widehat{E}_N^{\mathrm{MC}}] = \frac{\mathbb{V}_{\pi_0}[f]}{N}$, this follows by definition.

Note that the convergence rate does not depend on the dimension of u.

Ratio estimators

Mean square error of Monte Carlo ratio estimator [Scheichl, Stuart, ALT '17]

- Recall: $\mathbb{E}_{\pi^y}[\phi] = \frac{\mathbb{E}_{\pi_0}[\phi \exp[-\Phi(\cdot;y)]]}{\mathbb{E}_{\pi_0}[\exp[-\Phi(\cdot;y)]]} =: \frac{Q}{Z} \approx \frac{\widehat{Q}_N^{\mathrm{MC}}}{\widehat{Z}_N^{\mathrm{MC}}}$.
- We know how to bound the MSEs of the individual estimators $\widehat{Q}_N^{\mathrm{MC}}$ and $\widehat{Z}_N^{\mathrm{MC}}$. Can we bound the MSE of $\widehat{Q}_N^{\mathrm{MC}}/\widehat{Z}_N^{\mathrm{MC}}$?

Ratio estimators

Mean square error of Monte Carlo ratio estimator [Scheichl, Stuart, ALT '17]

- Recall: $\mathbb{E}_{\pi^y}[\phi] = \frac{\mathbb{E}_{\pi_0}[\phi \exp[-\Phi(\cdot;y)]]}{\mathbb{E}_{\pi_0}[\exp[-\Phi(\cdot;y)]]} =: \frac{Q}{Z} \approx \frac{\widehat{Q}_N^{\mathrm{MC}}}{\widehat{Z}_N^{\mathrm{MC}}}$.
- We know how to bound the MSEs of the individual estimators $\widehat{Q}_N^{\mathrm{MC}}$ and $\widehat{Z}_N^{\mathrm{MC}}$. Can we bound the MSE of $\widehat{Q}_N^{\mathrm{MC}}/\widehat{Z}_N^{\mathrm{MC}}$?
- Rearranging the MSE and applying the triangle inequality, we have

$$e\Big(\frac{\widehat{Q}_N^{\mathrm{MC}}}{\widehat{Z}_N^{\mathrm{MC}}}\Big)^2 = \mathbb{E}\Big[\Big(\frac{Q}{Z} - \frac{\widehat{Q}_N^{\mathrm{MC}}}{\widehat{Z}_N^{\mathrm{MC}}}\Big)^2\Big]$$
$$\leq \frac{2}{Z^2}\Big(\mathbb{E}\big[(Q - \widehat{Q}_N^{\mathrm{MC}})^2\big] + \mathbb{E}\big[(\widehat{Q}_N^{\mathrm{MC}}/\widehat{Z}_N^{\mathrm{MC}})^2 (Z - \widehat{Z}_N^{\mathrm{MC}})^2\big]\Big).$$

Ratio estimators

Mean square error of Monte Carlo ratio estimator [Scheichl, Stuart, ALT '17]

$$e\left(\frac{\widehat{Q}_N^{\mathrm{MC}}}{\widehat{Z}_N^{\mathrm{MC}}}\right)^2 \leq \frac{2}{Z^2}\left(\mathbb{E}\left[(Q-\widehat{Q}_N^{\mathrm{MC}})^2\right] + \mathbb{E}\left[(\widehat{Q}_N^{\mathrm{MC}}/\widehat{Z}_N^{\mathrm{MC}})^2(Z-\widehat{Z}_N^{\mathrm{MC}})^2\right]\right)$$

Theorem (Hölder's Inequality)

For any random variables X, Y *and* $p, q \in [1, \infty]$, *with* $p^{-1} + q^{-1} = 1$,

$$\mathbb{E}[|XY|] \leq \mathbb{E}[|X|^p]^{1/p}\, \mathbb{E}[|Y|^q]^{1/q}.$$

Here, $\mathbb{E}[|X|^\infty]^{1/\infty} := \operatorname{ess\,sup} X$.

Ratio estimators

Mean square error of Monte Carlo ratio estimator [Scheichl, Stuart, ALT '17]

$$e\left(\frac{\widehat{Q}_N^{\mathrm{MC}}}{\widehat{Z}_N^{\mathrm{MC}}}\right)^2 \leq \frac{2}{Z^2}\left(\mathbb{E}\left[(Q-\widehat{Q}_N^{\mathrm{MC}})^2\right] + \mathbb{E}\left[(\widehat{Q}_N^{\mathrm{MC}}/\widehat{Z}_N^{\mathrm{MC}})^2(Z-\widehat{Z}_N^{\mathrm{MC}})^2\right]\right)$$

Theorem (Hölder's Inequality)

For any random variables X, Y and $p, q \in [1, \infty]$, with $p^{-1} + q^{-1} = 1$,

$$\mathbb{E}[|XY|] \leq \mathbb{E}[|X|^p]^{1/p}\, \mathbb{E}[|Y|^q]^{1/q}.$$

Here, $\mathbb{E}[|X|^\infty]^{1/\infty} := \operatorname{ess\,sup} X$.

If $\operatorname{ess\,sup}_{\{u^{(i)}\}_{i=1}^N} (\widehat{Q}_N^{\mathrm{MC}}/\widehat{Z}_N^{\mathrm{MC}})^2 \leq C$, for a constant C independent of N, then the MSE of $\widehat{Q}_N^{\mathrm{MC}}/\widehat{Z}_N^{\mathrm{MC}}$ can be bounded in terms of the individual MSEs of $\widehat{Q}_N^{\mathrm{MC}}$ and $\widehat{Z}_N^{\mathrm{MC}}$.

In particular, the convergence rate in N carries over to the ratio estimator.

Ratio estimators

Higher order methods

- The convergence rate N^{-1} in the mean-square-error (MSE)

$$e(\widehat{E}_N^{\mathrm{MC}})^2 = \mathbb{E}[(\widehat{E}_N^{\mathrm{MC}} - \mathbb{E}_{\pi_0}[f])^2] = \frac{\mathbb{V}_{\pi_0}[f]}{N}$$

  of the MC estimator is quite slow.

- One way to improve the ratio estimator is to choose the sampling points $\{u^{(i)}\}_{i=1}^N$ in a more structured way than purely random to increase the convergence rate in N.

- This typically requires stronger assumptions on the function f that we want to estimate the expected value of.
  - Convergence of the MC estimator only requires $\mathbb{V}_{\pi_0}[f] < \infty$.

Ratio estimators

Quasi-Monte Carlo methods [Dick, Pillichshammer '10], [Leobacher, Pillichshammer '14]

- Suppose we want to compute the expected value of $f(u)$, where u is a random variable that is uniformly distributed on the unit cube $[0,1]^{d_u}$:

$$\mathbb{E}[f] = \int_{[0,1]^{d_u}} f(u)\mathrm{d}u.$$

- More general state spaces and distributions can be dealt with using a change of variables.

Ratio estimators

Quasi-Monte Carlo methods [Dick, Pillichshammer '10], [Leobacher, Pillichshammer '14]

- Suppose we want to compute the expected value of $f(u)$, where u is a random variable that is uniformly distributed on the unit cube $[0,1]^{d_u}$:

$$\mathbb{E}[f] = \int_{[0,1]^{d_u}} f(u)\mathrm{d}u.$$

- More general state spaces and distributions can be dealt with using a change of variables.

- Quasi-Monte Carlo methods approximate the expected value by an equal-weighted average:

$$\mathbb{E}[f] = \int_{[0,1]^{d_u}} f(u)\mathrm{d}u \approx \frac{1}{N}\sum_{i=1}^{N} f(u_i).$$

Ratio estimators

Low discrepancy sequence

- In Quasi-Monte Carlo (QMC) methods, we choose the set $\{u_i\}_{i=1}^N$ to be a low discrepancy point set.
- A low discrepancy set is "evenly distributed" over $[0,1]^{d_u}$.

Ratio estimators

Low discrepancy sequence

- In Quasi-Monte Carlo (QMC) methods, we choose the set $\{u_i\}_{i=1}^N$ to be a low discrepancy point set.
- A low discrepancy set is "evenly distributed" over $[0,1]^{d_u}$.
- The star discrepancy D_N^* of $\mathcal{P} = \{u_i\}_{i=1}^N$ is defined as

$$D_N^*(\mathcal{P}) := \sup_{\substack{\mathcal{B} \subseteq [0,1]^{d_u} \\ \mathcal{B} = \prod_{j=1}^{d_u} [0, u_j)}} \left| \text{vol}(\mathcal{B}) - A(\mathcal{B}, \{u_i\}_{i=1}^N) \right|,$$

  where $\text{vol}(\mathcal{B})$ denotes the volume of the box $\mathcal{B}$ and $A(\mathcal{B}, \{u_i\}_{i=1}^N)$ denotes the proportion of $\{u_i\}_{i=1}^N$ contained in $\mathcal{B}$.
- Roughly speaking, the discrepancy of a point set is low if the proportion of points inside a box $\mathcal{B}$ is close to the volume of $\mathcal{B}$.

Ratio estimators

Rank 1 Lattice Rules

- There are many ways of constructing low discrepancy point sets. We will focus on rank 1 lattice rules.
- Given a *generating vector* $z \in \mathbb{R}^{d_u}$, the N quadrature points are

$$u_i = \mathsf{frac}\left(\frac{i-1}{N} z\right) \quad i = 1, \ldots, N,$$

where "frac" denotes the fractional part of a number.

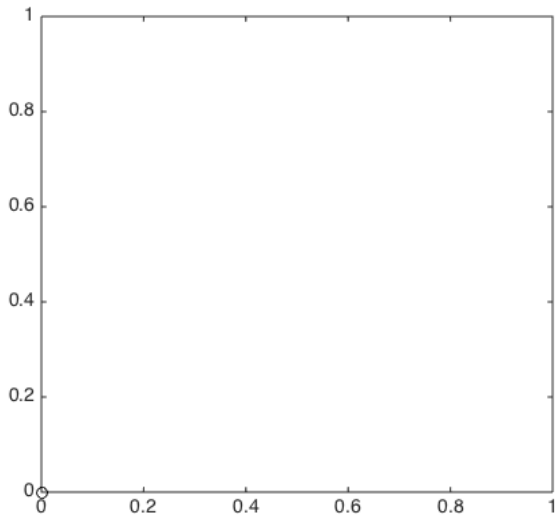
Ratio estimators

Rank 1 Lattice Rules

- There are many ways of constructing low discrepancy point sets. We will focus on rank 1 lattice rules.
- Given a *generating vector* $z \in \mathbb{R}^{d_u}$, the N quadrature points are

$$u_i = \text{frac}\left(\frac{i-1}{N} z\right) \quad i = 1, \ldots, N,$$

  where "frac" denotes the fractional part of a number.
- For example: $N = 21$, $z = [1, 13]$.



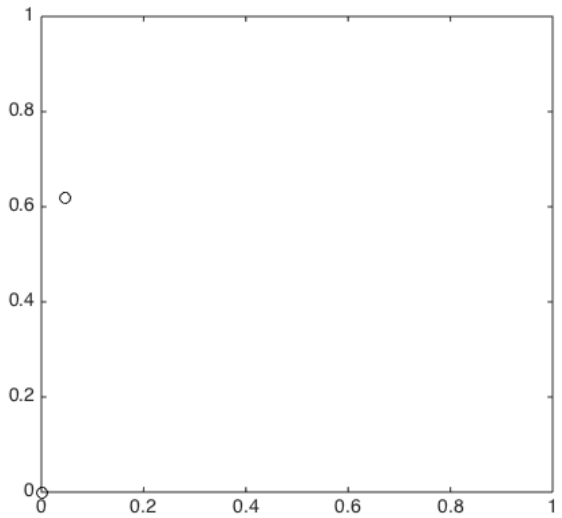
Ratio estimators

Rank 1 Lattice Rules

- There are many ways of constructing low discrepancy point sets. We will focus on rank 1 lattice rules.
- Given a *generating vector* $z \in \mathbb{R}^{d_u}$, the N quadrature points are

$$u_i = \text{frac}\left(\frac{i-1}{N} z\right) \quad i = 1, \ldots, N,$$

  where "frac" denotes the fractional part of a number.
- For example: $N = 21$, $z = [1, 13]$.



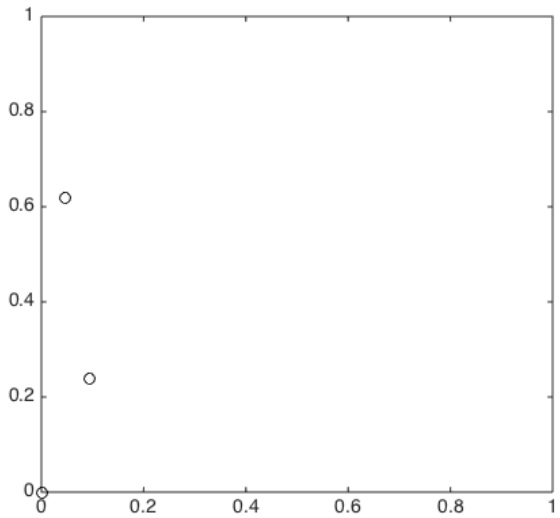
Ratio estimators

Rank 1 Lattice Rules

- There are many ways of constructing low discrepancy point sets. We will focus on rank 1 lattice rules.
- Given a *generating vector* $z \in \mathbb{R}^{d_u}$, the N quadrature points are

$$u_i = \text{frac}\left(\frac{i-1}{N} z\right) \quad i = 1, \ldots, N,$$

  where "frac" denotes the fractional part of a number.
- For example: $N = 21$, $z = [1, 13]$.



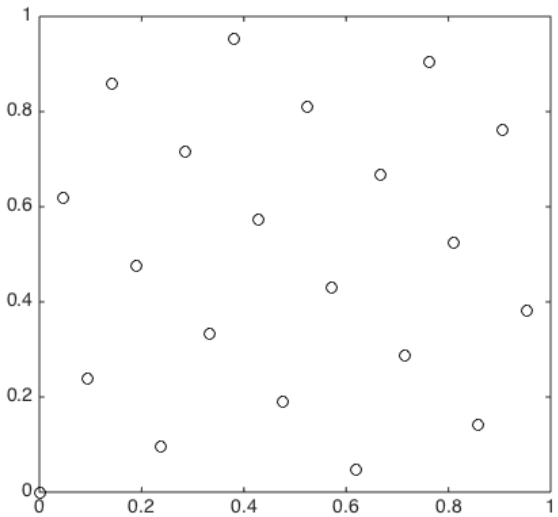
Ratio estimators

Rank 1 Lattice Rules

- There are many ways of constructing low discrepancy point sets. We will focus on rank 1 lattice rules.
- Given a *generating vector* $z \in \mathbb{R}^{d_u}$, the N quadrature points are

$$u_i = \text{frac}\left(\frac{i-1}{N} z\right) \quad i = 1, \ldots, N,$$

  where "frac" denotes the fractional part of a number.
- For example: $N = 21$, $z = [1, 13]$.



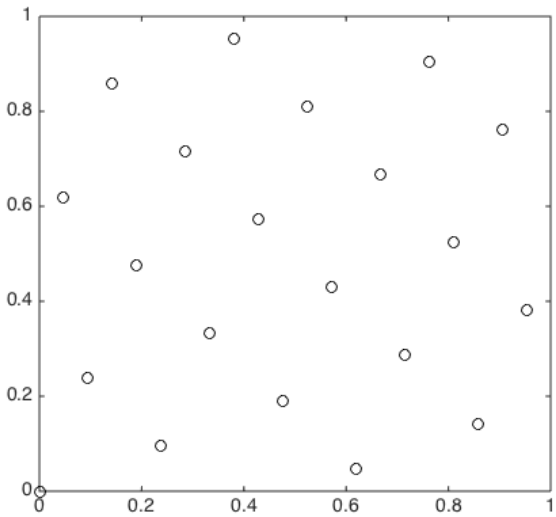
Ratio estimators

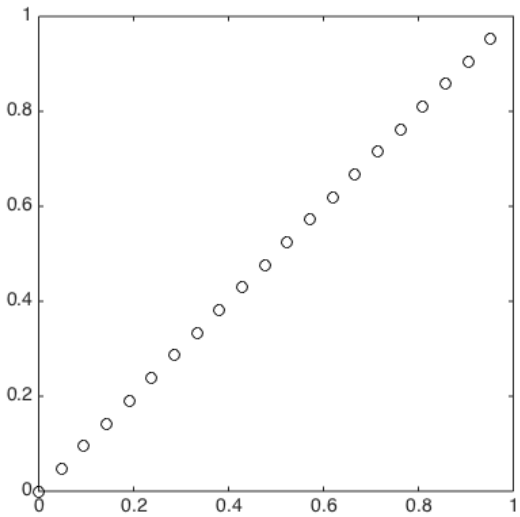
Rank 1 Lattice Rules

- There are many ways of constructing low discrepancy point sets. We will focus on rank 1 lattice rules.
- Given a *generating vector* $z \in \mathbb{R}^{d_u}$, the N quadrature points are

$$u_i = \text{frac}\left(\frac{i-1}{N} z\right) \quad i = 1, \ldots, N,$$

  where "frac" denotes the fractional part of a number.
- For example: $N = 21$, $z = [1, 13]$. $N = 21$, $z = [1, 1]$.





Ratio estimators

Randomised Quasi-Monte Carlo methods

- For Monte Carlo, we can easily compute an estimate of the sampling error $\mathbb{V}[f]N^{-1}$ from the computed samples $\{f(u_i)\}_{i=1}^N$ using the *sample variance*:

$$\mathbb{E}[(f - \mathbb{E}[f])^2] = \mathbb{V}[f] \approx \frac{1}{N-1} \sum_{i=1}^{N} \left(f(u_i) - \frac{1}{N} \sum_{i=1}^{N} f(u_i) \right)^2.$$

- Computing the QMC estimator gives an estimate of $\mathbb{E}[f]$, but does not give an estimate of the error in this approximation.

Ratio estimators

Randomised Quasi-Monte Carlo methods

- For Monte Carlo, we can easily compute an estimate of the sampling error $\mathbb{V}[f]N^{-1}$ from the computed samples $\{f(u_i)\}_{i=1}^N$ using the *sample variance*:

$$\mathbb{E}[(f - \mathbb{E}[f])^2] = \mathbb{V}[f] \approx \frac{1}{N-1}\sum_{i=1}^{N}\left(f(u_i) - \frac{1}{N}\sum_{i=1}^{N} f(u_i)\right)^2.$$

- Computing the QMC estimator gives an estimate of $\mathbb{E}[f]$, but does not give an estimate of the error in this approximation.
- To enable error estimation, it is common to randomise the low discrepancy point set. We have to be careful to keep the structure!
- For rank 1 lattice rules, we use a shift Δ that is uniformly distributed over $[0,1]^{d_u}$:

$$\widehat{E}_N^{\mathrm{rQMC}}(\Delta) = \frac{1}{N}\sum_{i=1}^{N} f(\mathsf{frac}\left(\frac{i-1}{N}z + \Delta\right)).$$

Ratio estimators

Randomised Quasi-Monte Carlo methods

- To estimate the error of $\widehat{E}_N^{\mathrm{rQMC}}$, we use M realisations of the random shift Δ, $\{\Delta_k\}_{k=1}^M$, and compute the sample variance:

$$\mathbb{V}[\widehat{E}_N^{\mathrm{rQMC}}] \approx \frac{1}{M-1}\sum_{k=1}^{M}\left(\widehat{E}_N^{\mathrm{rQMC}}(\Delta_k) - \frac{1}{M}\sum_{k=1}^{M}\widehat{E}_N^{\mathrm{rQMC}}(\Delta_k)\right)^2.$$

Ratio estimators

Randomised Quasi-Monte Carlo methods

- To estimate the error of $\widehat{E}_N^{\mathrm{rQMC}}$, we use M realisations of the random shift Δ, $\{\Delta_k\}_{k=1}^M$, and compute the sample variance:

$$\mathbb{V}[\widehat{E}_N^{\mathrm{rQMC}}] \approx \frac{1}{M-1}\sum_{k=1}^{M}\left(\widehat{E}_N^{\mathrm{rQMC}}(\Delta_k) - \frac{1}{M}\sum_{k=1}^{M}\widehat{E}_N^{\mathrm{rQMC}}(\Delta_k)\right)^2.$$

- Since we need to compute $\widehat{E}_N^{\mathrm{rQMC}}(\Delta_k)$, for $k = 1, \ldots, M$ anyway, it is common to use the *shift-averaged* estimator

$$\widehat{E}_{N,M}^{\mathrm{rQMC}} = \frac{1}{M}\sum_{k=1}^{M}\frac{1}{N}\sum_{i=1}^{N} f\left(\mathrm{frac}\left(\frac{i-1}{N}z + \Delta_k\right)\right),$$

as an approximation to $\mathbb{E}[f]$.

Ratio estimators

Randomised Quasi-Monte Carlo methods

- Using weighted spaces of dominating mixed smoothness, it is possible to obtain an error bound

$$\mathbb{E}_\Delta[|\mathbb{E}_{\pi_0}[f] - \widehat{E}_{M,N}^{\mathrm{rQMC}}|] \leq C\|f\|_{1,\mathrm{weighted}}\, N^{-1+\delta}, \qquad \delta > 0.$$

$$\|f\|_{1,\mathrm{weighted}} := \left(\sum_{\nu \subseteq \{1:d_u\}} \gamma_\nu^{-1} \int_{[0,1]^{|\nu|}} \left(\int_{[0,1]^{d_u-|\nu|}} \frac{\partial^{|\nu|} f}{\partial u_\nu} \mathrm{d}u_{\{1:d_u\}\setminus\nu}\right)^2 \mathrm{d}u_\nu\right)^{1/2}$$

Ratio estimators

Randomised Quasi-Monte Carlo methods

- Using weighted spaces of dominating mixed smoothness, it is possible to obtain an error bound

$$\mathbb{E}_\Delta[|\mathbb{E}_{\pi_0}[f] - \widehat{E}_{M,N}^{\mathrm{rQMC}}|] \leq C\|f\|_{1,\mathrm{weighted}}\, N^{-1+\delta}, \qquad \delta > 0.$$

$$\|f\|_{1,\mathrm{weighted}} := \left(\sum_{\nu \subseteq \{1:d_u\}} \gamma_\nu^{-1} \int_{[0,1]^{|\nu|}} \left(\int_{[0,1]^{d_u-|\nu|}} \frac{\partial^{|\nu|} f}{\partial u_\nu} \mathrm{d}u_{\{1:d_u\}\setminus\nu}\right)^2 \mathrm{d}u_\nu\right)^{1/2}$$

- We want to keep M small (≈ 16) to retain the good convergence rate of QMC.
- The special case $N = 1$ corresponds to standard Monte Carlo.

Ratio Estimators

Related works

A number of works have recently considered the ratio estimator approach in the context of PDE constrained inverse problems, as it allows to reuse machinery developed for π_0:

- [Schillings, Schwab '13]: dimension-adaptive sparse grids
- [Dick, Gantner, Le Gia, Schwab '17]: (multilevel) higher order Quasi-Monte Carlo
- [Gantner, Peters '18]: higher order Quasi-Monte Carlo for PDEs on random domains

Ratio Estimators

Related works

A number of works have recently considered the ratio estimator approach in the context of PDE constrained inverse problems, as it allows to reuse machinery developed for π_0:

- [Schillings, Schwab '13]: dimension-adaptive sparse grids
- [Dick, Gantner, Le Gia, Schwab '17]: (multilevel) higher order Quasi-Monte Carlo
- [Gantner, Peters '18]: higher order Quasi-Monte Carlo for PDEs on random domains

When $\gamma^2 \ll 1$ or $d_y \gg 1$, the posterior density π^y may concentrate, and the prior evaluations become difficult to evaluate accurately.

- [Schillings, Schwab '16]: rescaling of parameter space around (unique) MAP point
- [Schillings, Sprungk, Wacker '20]: using Lapalace approximation of posterior as reference measure

References I

S. L. Cotter, G. O. Roberts, A. M. Stuart, and D. White, *MCMC methods for functions: modifying old algorithms to make them faster*, Statistical Science, (2013), pp. 424–446.

T. Cui, K. J. Law, and Y. M. Marzouk, *Dimension-independent likelihood-informed MCMC*, Journal of Computational Physics, 304 (2016), pp. 109–137.

J. Dick, R. N. Gantner, Q. T. Le Gia, and C. Schwab, *Multilevel higher-order quasi-Monte Carlo Bayesian estimation*, Mathematical Models and Methods in Applied Sciences, 27 (2017), pp. 953–995.

J. Dick and F. Pillichshammer, *Digital nets and sequences: discrepancy theory and quasi–Monte Carlo integration*, Cambridge University Press, 2010.

R. N. Gantner and M. D. Peters, *Higher-Order Quasi-Monte Carlo for Bayesian Shape Inversion*, SIAM/ASA Journal on Uncertainty Quantification, 6 (2018), pp. 707–736.

References II


M. Girolami and B. Calderhead, *Riemann manifold Langevin and Hamiltonian Monte Carlo methods*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 73 (2011), pp. 123–214.


B. Hosseini, *Two Metropolis–Hastings Algorithms for Posterior Measures with Non-Gaussian Priors in Infinite Dimensions*, SIAM/ASA Journal on Uncertainty Quantification, 7 (2019), pp. 1185–1223.


J. Kaipio and E. Somersalo, *Statistical and computational inverse problems*, Springer, 2004.


P. L'Ecuyer, *Random number generation*, in Handbook of Computational Statistics, Springer, 2011, pp. 35–71.


G. Leobacher and F. Pillichshammer, *Introduction to quasi-Monte Carlo integration and applications*, Springer, 2014.


J. S. Liu, *Monte Carlo strategies in scientific computing*, vol. 10, Springer, 2001.

References III


Y.-A. Ma, E. B. Fox, T. Chen, and L. Wu, *Irreversible samplers from jump and continuous Markov processes*, Statistics and Computing, 29 (2019), pp. 177–202.


N. S. Pillai, A. M. Stuart, and A. H. Thiéry, *Optimal scaling and diffusion limits for the Langevin algorithm in high dimensions*, The Annals of Applied Probability, 22 (2012), pp. 2320–2356.


C. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer, 1999.


G. O. Roberts, A. Gelman, and W. R. Gilks, *Weak convergence and optimal scaling of random walk Metropolis algorithms*, The annals of applied probability, 7 (1997), pp. 110–120.


R. Scheichl, A. Stuart, and A. Teckentrup, *Quasi-Monte Carlo and multilevel Monte Carlo methods for computing posterior expectations in elliptic inverse problems*, SIAM/ASA Journal on Uncertainty Quantification, 5 (2017), pp. 493–518.


C. Schillings and C. Schwab, *Sparse, adaptive Smolyak quadratures for Bayesian inverse problems*, Inverse Problems, 29 (2013), p. 065011.

References IV

———, *Scaling limits in computational Bayesian inversion*, ESAIM: Mathematical Modelling and Numerical Analysis, 50 (2016), pp. 1825–1856.

C. Schillings, B. Sprungk, and P. Wacker, *On the convergence of the Laplace approximation and noise-level-robustness of Laplace-based Monte Carlo methods for Bayesian inverse problems*, Numerische Mathematik, 145 (2020), pp. 915–971.

A. Stuart, *Inverse Problems: A Bayesian Perspective*, Acta Numerica, 19 (2010), pp. 451–559.

S. J. Vollmer, *Dimension-independent MCMC sampling for inverse problems with non-Gaussian priors*, SIAM/ASA Journal on Uncertainty Quantification, 3 (2015), pp. 535–561.