Exact Bayesian Inference (for Big Data)

Single- and Multi- Core Approaches

Murray Pollock

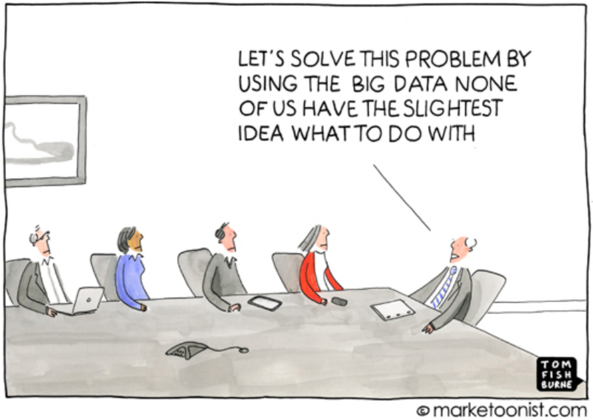
H Dai, P Fearnhead, AM Johansen, GO Roberts

m.pollock@warwick.ac.uk

www.warwick.ac.uk/mpollock

Warwick Statistics

Cartoon


LET'S SOLVE THIS PROBLEM BY USING THE BIG DATA NONE OF US HAVE THE SLIGHTEST IDEA WHAT TO DO WITH
TOM FISH BURNE
© marketoonist.com

Problem

- Big Data challenge?
  - Algorithmic 'Scalability'
- Target of interest:

$$\pi(x) \propto \prod_{i=0}^{N} f_i(x).$$

- Want to use [MCMC].
- Approaches:
  - Single-Core
  - Multi-Core

Problem

- Big Data challenge?
  - Algorithmic 'Scalability'
- Target of interest:

$$\pi(x) \propto \prod_{i=0}^{N} f_i(x).$$

- Want to use [MCMC].
- Approaches:
  - Single-Core
  - Multi-Core

Problem

- Big Data challenge?
  - Algorithmic 'Scalability'
- Target of interest:

$$\pi(x) \propto \prod_{i=0}^{N} f_i(x).$$

- Want to use [MCMC].
- Approaches:
  - Single-Core
  - Multi-Core

Problem

- Big Data challenge?
  - Algorithmic 'Scalability'
- Target of interest:

$$\pi(x) \propto \prod_{i=0}^{N} f_i(x).$$

- Want to use [MCMC].
- Approaches:
  - Single-Core
  - Multi-Core

Problem

- Big Data challenge?
  - Algorithmic 'Scalability'
- Target of interest:

$$\pi(x) \propto \prod_{i=0}^{N} f_i(x).$$

- Want to use [MCMC].
- Approaches:
  - Single-Core
  - Multi-Core

Problem

- Big Data challenge?
  - Algorithmic ‘Scalability’
- Target of interest:

$$\pi(x) \propto \prod_{i=0}^{N} f_i(x).$$

- Want to use [MCMC].
- Approaches:
  - Single-Core
  - Multi-Core

Single-Core

- Single-Core
  - Problem: Metropolis move from $\theta \to \phi$ is accepted w.p.,

$$\min\left\{1, \frac{\pi(\phi)}{\pi(\theta)}\right\}$$

  - Goal: Scalability of iterative cost.
  - Lots of work!: Pseudo-Marginal; Stochastic gradient schemes. . .
- *[1] The Scalable Langevin Exact Algorithm: Bayesian Inference for Big Data* available at: https://arxiv.org/abs/1609.03436

Single-Core

- Single-Core
  - Problem: Metropolis move from $\theta \rightarrow \phi$ is accepted w.p.,

$$\min\left\{1, \frac{\pi(\phi)}{\pi(\theta)}\right\}$$

  - Goal: Scalability of iterative cost.
  - Lots of work!: Pseudo-Marginal; Stochastic gradient schemes...
- *[1] The Scalable Langevin Exact Algorithm: Bayesian Inference for Big Data* available at: https://arxiv.org/abs/1609.03436

Single-Core

- Single-Core
  - Problem: Metropolis move from $\theta \to \phi$ is accepted w.p.,

$$\min\left\{1, \frac{\pi(\phi)}{\pi(\theta)}\right\}$$

  - Goal: Scalability of iterative cost.
  - Lots of work!: Pseudo-Marginal; Stochastic gradient schemes...
- *[1] The Scalable Langevin Exact Algorithm: Bayesian Inference for Big Data* available at: https://arxiv.org/abs/1609.03436

Single-Core

- Single-Core
  - Problem: Metropolis move from $\theta \to \phi$ is accepted w.p.,

$$\min\left\{1, \frac{\pi(\phi)}{\pi(\theta)}\right\}$$

  - Goal: Scalability of iterative cost.
  - Lots of work!: Pseudo-Marginal; Stochastic gradient schemes...
- *[1] The Scalable Langevin Exact Algorithm: Bayesian Inference for Big Data* available at: https://arxiv.org/abs/1609.03436

Single-Core

- Single-Core
  - Problem: Metropolis move from $\theta \to \phi$ is accepted w.p.,

$$\min\left\{1, \frac{\pi(\phi)}{\pi(\theta)}\right\}$$

  - Goal: Scalability of iterative cost.
  - Lots of work!: Pseudo-Marginal; Stochastic gradient schemes...
- *[1] The Scalable Langevin Exact Algorithm: Bayesian Inference for Big Data* available at: https://arxiv.org/abs/1609.03436

Multi-Core

- Multi-Core
  - Solution to Single-Core:
    1. Break data into S 'shards' (of size N/S)
    2. Separate inferences [MCMC]
    3. 'Recombine' on 'mother-core'
  - Problem: Recombining – How do you do it?
  - Lots of work!: Consensus; Averaging; Kernel methods. . .
  - Constraints / Assumptions
- *[2] Bayesian Fusion: An exact and parallelisable consensus approach to unifying distributed analyses*

Multi-Core

- Multi-Core
  - Solution to Single-Core:
    1. Break data into S 'shards' (of size N/S)
    2. Separate inferences [MCMC]
    3. 'Recombine' on 'mother-core'
  - Problem: Recombining – How do you do it?
  - Lots of work!: Consensus; Averaging; Kernel methods...
  - Constraints / Assumptions
- *[2] Bayesian Fusion: An exact and parallelisable consensus approach to unifying distributed analyses*

Multi-Core

- Multi-Core
  - Solution to Single-Core:
    1. Break data into S 'shards' (of size N/S)
    2. Separate inferences [MCMC]
    3. 'Recombine' on 'mother-core'
  - Problem: Recombining – How do you do it?
  - Lots of work!: Consensus; Averaging; Kernel methods. . .
  - Constraints / Assumptions
- *[2] Bayesian Fusion: An exact and parallelisable consensus approach to unifying distributed analyses*

Multi-Core

- Multi-Core
  - Solution to Single-Core:
    1. Break data into S 'shards' (of size N/S)
    2. Separate inferences [MCMC]
    3. 'Recombine' on 'mother-core'
  - Problem: Recombining – How do you do it?
  - Lots of work!: Consensus; Averaging; Kernel methods...
  - Constraints / Assumptions
- *[2] Bayesian Fusion: An exact and parallelisable consensus approach to unifying distributed analyses*

Multi-Core

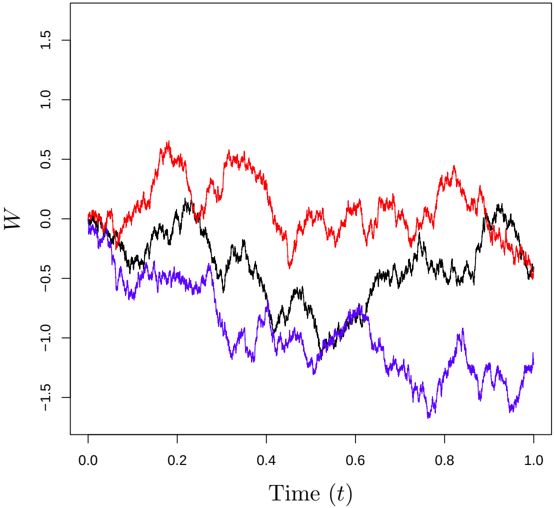
- Multi-Core
  - Solution to Single-Core:
    1. Break data into S 'shards' (of size N/S)
    2. Separate inferences [MCMC]
    3. 'Recombine' on 'mother-core'
  - Problem: Recombining – How do you do it?
  - Lots of work!: Consensus; Averaging; Kernel methods...
  - Constraints / Assumptions
- *[2] Bayesian Fusion: An exact and parallelisable consensus approach to unifying distributed analyses*

Multi-Core

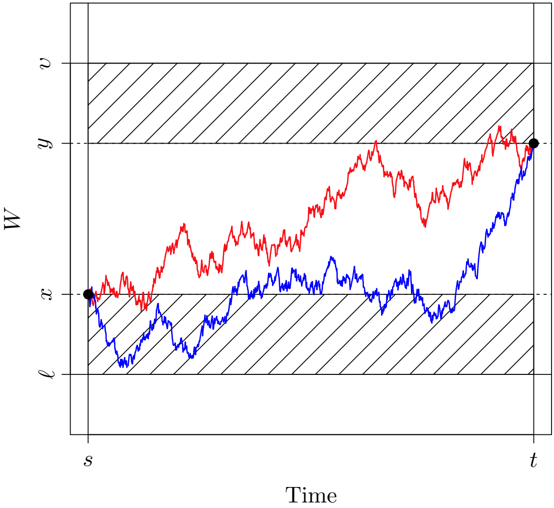
- Multi-Core
  - Solution to Single-Core:
    1. Break data into S 'shards' (of size N/S)
    2. Separate inferences [MCMC]
    3. 'Recombine' on 'mother-core'
  - Problem: Recombining – How do you do it?
  - Lots of work!: Consensus; Averaging; Kernel methods...
  - Constraints / Assumptions
- *[2] Bayesian Fusion: An exact and parallelisable consensus approach to unifying distributed analyses*

0 - Retrospective Trust / Tricks

Brownian Motion



Brownian Motion



Path-space Rejection Sampling

Path-space Rejection Sampling:

- We want $X \sim \mathbb{Q}$ where:

$$\mathbb{Q} : \mathrm{d}X_t = \alpha(X_t)\,\mathrm{d}t + \Lambda^{1/2}\,\mathrm{d}B_t, \quad X_0 = x \in \mathbb{R}^d, t \in [0, T]$$

- Discretisation Free Approach!: Path-space Rejection Sampler (PRS) (see arXiv 1302.6964 for details)
  1. $X_T \sim h_T(X_0)$
  2. $X^{\mathrm{fin}} \sim \mathbb{P}|X_T$* (eg $\mathbb{W}$ or $\mathbb{O}$)
  3. (Accept / Reject)** / Assign Weight**

Path-space Rejection Sampling

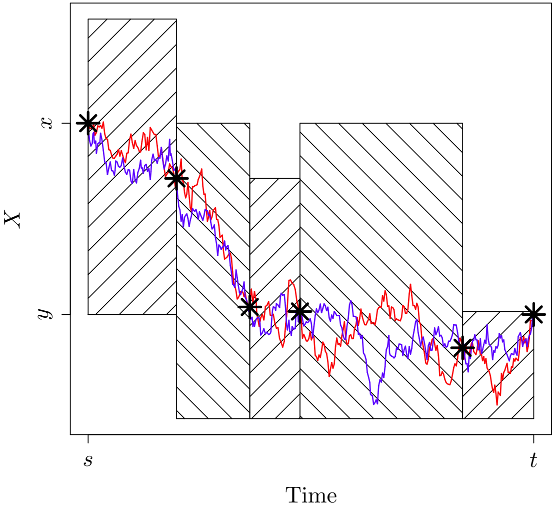
Path-space Rejection Sampling:

- We want $X \sim \mathbb{Q}$ where:

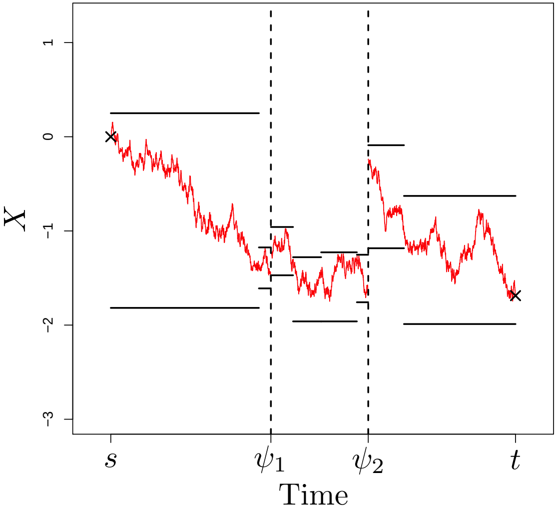
$$\mathbb{Q} : \mathrm{d}X_t = \alpha(X_t)\,\mathrm{d}t + \Lambda^{1/2}\,\mathrm{d}B_t, \quad X_0 = x \in \mathbb{R}^d, t \in [0, T]$$

- Discretisation Free Approach!: Path-space Rejection Sampler (PRS) (see arXiv 1302.6964 for details)
  1. $X_T \sim h_T(X_0)$
  2. $X^{\text{fin}} \sim \mathbb{P}|X_T$* (eg $\mathbb{W}$ or $\mathbb{O}$)
  3. (Accept / Reject)** / Assign Weight**

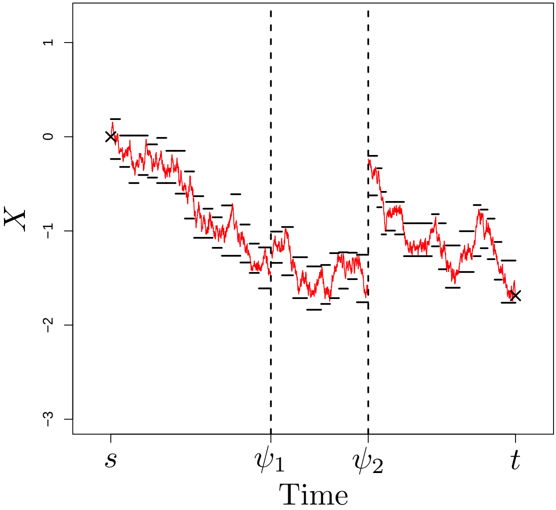
Path-space Rejection Sampling



Path-space Rejection Sampling



Path-space Rejection Sampling



Langevin Diffusion

Langevin Diffusion:

- In $\mathbb{Q}$ set $\alpha(X_t) := \frac{1}{2}\Lambda\nabla \log \nu(X_t)$
- Invariant distribution ν
  - Direct statistical exploitation... $\nu \equiv \pi$ ($\mathbb{L}$)
  - Langevin + $\nu \equiv \pi$ + Discretisation + Correction $\Longrightarrow$ MALA
- PRS Class (however [1] $y := X_T \sim h \equiv \nu^{1/2} \ldots$)
  - $\lim_{T\to\infty} p_T(x,y) = \underbrace{w_T(x,y)\cdot\nu^{1/2}(y)}_{\propto h} \cdot \underbrace{P(X)}_{\in[0,1]} \to \nu$
  - $\nu \equiv \pi^2$ ($\mathbb{DL}$)
- If $X_0 \sim \nu$, then $\forall t > 0$, $X_t \sim \nu$

Langevin Diffusion:

- In $\mathbb{Q}$ set $\alpha(X_t) := \frac{1}{2}\Lambda\nabla \log \nu(X_t)$
- Invariant distribution ν
  - Direct statistical exploitation... $\nu \equiv \pi$ ($\mathbb{L}$)
  - Langevin + $\nu \equiv \pi$ + Discretisation + Correction $\Longrightarrow$ MALA
- PRS Class (however [1] $y := X_T \sim h \equiv \nu^{1/2} \ldots$)
  - $\lim_{T\to\infty} p_T(x,y) = \underbrace{w_T(x,y)\cdot\nu^{1/2}(y)}_{\propto h} \cdot \underbrace{P(X)}_{\in[0,1]} \to \nu$
  - $\nu \equiv \pi^2$ ($\mathbb{D}\mathbb{L}$)
- If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$

Langevin Diffusion

Langevin Diffusion:

- In $\mathbb{Q}$ set $\alpha(X_t) := \frac{1}{2}\Lambda\nabla \log \nu(X_t)$
- Invariant distribution ν
  - Direct statistical exploitation... $\nu \equiv \pi$ ($\mathbb{L}$)
  - Langevin + $\nu \equiv \pi$ + Discretisation + Correction $\implies$ MALA
- PRS Class (however [1] $y := X_T \sim h \equiv \nu^{1/2} \ldots$)
  - $\lim_{T\to\infty} p_T(x,y) = \underbrace{w_T(x,y)\cdot\nu^{1/2}(y)}_{\propto h}\cdot\underbrace{P(X)}_{\in[0,1]} \to \nu$
  - $\nu \equiv \pi^2$ ($\mathbb{DL}$)
- If $X_0 \sim \nu$, then $\forall t > 0$, $X_t \sim \nu$

Langevin Diffusion

Langevin Diffusion:

- In $\mathbb{Q}$ set $\alpha(X_t) := \frac{1}{2}\Lambda\nabla \log \nu(X_t)$
- Invariant distribution ν
  - Direct statistical exploitation… $\nu \equiv \pi$ ($\mathbb{L}$)
  - Langevin + $\nu \equiv \pi$ + Discretisation + Correction $\Longrightarrow$ MALA
- PRS Class (however [1] $y := X_T \sim h \equiv \nu^{1/2} \ldots$)
  - $\lim_{T\to\infty} p_T(x,y) = \underbrace{w_T(x,y)\cdot\nu^{1/2}(y)}_{\propto h}\cdot\underbrace{P(X)}_{\in[0,1]} \to \nu$
  - $\nu \equiv \pi^2$ ($\mathbb{DL}$)
- If $X_0 \sim \nu$, then $\forall t > 0$, $X_t \sim \nu$

Langevin Diffusion:

- In $\mathbb{Q}$ set $\alpha(X_t) := \frac{1}{2}\Lambda\nabla \log \nu(X_t)$
- Invariant distribution ν
  - Direct statistical exploitation... $\nu \equiv \pi$ ($\mathbb{L}$)
  - Langevin + $\nu \equiv \pi$ + Discretisation + Correction $\Longrightarrow$ MALA
- PRS Class (however [1] $y := X_T \sim h \equiv \nu^{1/2} \ldots$)
  - $\lim_{T\to\infty} p_T(x,y) = \underbrace{w_T(x,y)\cdot\nu^{1/2}(y)}_{\propto h} \cdot \underbrace{P(X)}_{\in[0,1]} \to \nu$
  - $\nu \equiv \pi^2$ ($\mathbb{DL}$)
- If $X_0 \sim \nu$, then $\forall t > 0$, $X_t \sim \nu$

Langevin Diffusion

Langevin Diffusion:

- In $\mathbb{Q}$ set $\alpha(X_t) := \frac{1}{2}\Lambda\nabla \log \nu(X_t)$
- Invariant distribution ν
  - Direct statistical exploitation... $\nu \equiv \pi$ ($\mathbb{L}$)
  - Langevin + $\nu \equiv \pi$ + Discretisation + Correction $\Longrightarrow$ MALA
- PRS Class (however [1] $y := X_T \sim h \equiv \nu^{1/2} \ldots$)
  - $\lim_{T\to\infty} p_T(x,y) = \underbrace{w_T(x,y)\cdot \nu^{1/2}(y)}_{\propto h} \cdot \underbrace{P(X)}_{\in[0,1]} \to \nu$
  - $\nu \equiv \pi^2$ ($\mathbb{D}\mathbb{L}$)
- If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$

Langevin Diffusion

Langevin Diffusion:

- In $\mathbb{Q}$ set $\alpha(X_t) := \frac{1}{2}\Lambda\nabla \log \nu(X_t)$
- Invariant distribution ν
  - Direct statistical exploitation… $\nu \equiv \pi$ ($\mathbb{L}$)
  - Langevin + $\nu \equiv \pi$ + Discretisation + Correction $\Longrightarrow$ MALA
- PRS Class (however [1] $y := X_T \sim h \equiv \nu^{1/2} \ldots$)
  - $\lim_{T\to\infty} p_T(x,y) = \underbrace{w_T(x,y)\cdot \nu^{1/2}(y)}_{\propto h} \cdot \underbrace{P(X)}_{\in[0,1]} \to \nu$
  - $\nu \equiv \pi^2$ ($\mathbb{DL}$)
- If $X_0 \sim \nu$, then $\forall t > 0$, $X_t \sim \nu$

Langevin Diffusion

Langevin Diffusion:

- In $\mathbb{Q}$ set $\alpha(X_t) := \frac{1}{2}\Lambda\nabla \log \nu(X_t)$
- Invariant distribution ν
  - Direct statistical exploitation... $\nu \equiv \pi$ ($\mathbb{L}$)
  - Langevin + $\nu \equiv \pi$ + Discretisation + Correction $\Longrightarrow$ MALA
- PRS Class (however [1] $y := X_T \sim h \equiv \nu^{1/2}\ldots$)
  - $\lim_{T\to\infty} p_T(x,y) = \underbrace{w_T(x,y)\cdot\nu^{1/2}(y)}_{\propto h} \cdot \underbrace{P(X)}_{\in[0,1]} \to \nu$
  - $\nu \equiv \pi^2$ ($\mathbb{D}\mathbb{L}$)
- If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$

Langevin Diffusion

Langevin Diffusion:

- In $\mathbb{Q}$ set $\alpha(X_t) := \frac{1}{2}\Lambda\nabla \log \nu(X_t)$
- Invariant distribution ν
  - Direct statistical exploitation... $\nu \equiv \pi$ ($\mathbb{L}$)
  - Langevin + $\nu \equiv \pi$ + Discretisation + Correction $\Longrightarrow$ MALA
- PRS Class (however [1] $y := X_T \sim h \equiv \nu^{1/2} \ldots$)
  - $\lim_{T\to\infty} p_T(x,y) = \underbrace{w_T(x,y) \cdot \nu^{1/2}(y)}_{\propto h} \cdot \underbrace{P(X)}_{\in[0,1]} \to \nu$
  - $\nu \equiv \pi^2$ ($\mathbb{DL}$)
- If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$

1 - Single Core: Quasi-Stationary Monte Carlo

Quasi-Stationary Monte Carlo

- Consider Brownian motion, killed at τ with intensity

$$\kappa(x) = \frac{\|\nabla \log \pi(x)\|^2 + \Delta \log \pi(x)}{2} - \ell \in \mathbb{R}_{\geq 0},$$

and the quasi-limiting distribution

$$\lim_{t \to \infty} \mathcal{L}(X_t | \tau > t).$$

- Under weak regularity conditions has quasi-stationary distribution π.
- Statistical Interpretation:
  - Big Data? → Subsampling
  - Implementation? → ScaLE

Quasi-Stationary Monte Carlo

- Consider Brownian motion, killed at τ with intensity

$$\kappa(x) = \frac{\|\nabla \log \pi(x)\|^2 + \Delta \log \pi(x)}{2} - \ell \in \mathbb{R}_{\geq 0},$$

and the quasi-limiting distribution

$$\lim_{t \to \infty} \mathcal{L}(X_t | \tau > t).$$

- Under weak regularity conditions has quasi-stationary distribution π.
- Statistical Interpretation:
  - Big Data? → Subsampling
  - Implementation? → ScaLE

Quasi-Stationary Monte Carlo

- Consider Brownian motion, killed at τ with intensity

$$\kappa(x) = \frac{\|\nabla \log \pi(x)\|^2 + \Delta \log \pi(x)}{2} - \ell \in \mathbb{R}_{\geq 0},$$

  and the quasi-limiting distribution

$$\lim_{t\to\infty} \mathcal{L}(X_t | \tau > t).$$

- Under weak regularity conditions has quasi-stationary distribution π.
- Statistical Interpretation:
  - Big Data? → Subsampling
  - Implementation? → ScaLE

Quasi-Stationary Monte Carlo

- Consider Brownian motion, killed at τ with intensity

$$\kappa(x) = \frac{\|\nabla \log \pi(x)\|^2 + \Delta \log \pi(x)}{2} - \ell \in \mathbb{R}_{\geq 0},$$

and the quasi-limiting distribution

$$\lim_{t \to \infty} \mathcal{L}(X_t | \tau > t).$$

- Under weak regularity conditions has quasi-stationary distribution π.
- Statistical Interpretation:
  - Big Data? → Subsampling
  - Implementation? → ScaLE

Quasi-Stationary Monte Carlo

- Consider Brownian motion, killed at τ with intensity

$$\kappa(x) = \frac{\|\nabla \log \pi(x)\|^2 + \Delta \log \pi(x)}{2} - \ell \in \mathbb{R}_{\geq 0},$$

and the quasi-limiting distribution

$$\lim_{t \to \infty} \mathcal{L}(X_t | \tau > t).$$

- Under weak regularity conditions has quasi-stationary distribution π.
- Statistical Interpretation:
  - Big Data? → Subsampling
  - Implementation? → ScaLE

1.2 - Subsampling

- QSMC $\equiv$ Simulating BM + inhomogeneous Poisson Process κ
- Evaluating κ is $\mathcal{O}(N)$.
- * If $\forall x$, $\kappa(x) \leq K$ (requires localisation argument) then:
  - Simulating $PP(\kappa(x)) \equiv$ Simulating $PP(K)$ and accepting w.p. $\kappa(X_t)/K$.
- We can make our algorithm worse (!) by choosing $\tilde{K} \geq K \ldots$
- Remark on coins
- Suppose $\exists A \sim \mathcal{A}$, $\tilde{\kappa_A}(\cdot) \in [0, \tilde{K}]$ such that $\mathbb{E}_{\mathcal{A}}[\tilde{\kappa_A}(x)/\tilde{K}] = \kappa(x)/\tilde{K}$ then:
  - Simulating $A \sim \mathcal{A}$ $PP(\tilde{K})$ and accepting w.p. $\tilde{\kappa}_A(X_t)/\tilde{K} \equiv$ *.

Subsampling

- QSMC ≡ Simulating BM + inhomogeneous Poisson Process κ
- Evaluating κ is $\mathcal{O}(N)$.
- * If $\forall x$, $\kappa(x) \leq K$ (requires localisation argument) then:
  - Simulating $PP(\kappa(x)) \equiv$ Simulating $PP(K)$ and accepting w.p. $\kappa(X_t)/K$.
- We can make our algorithm worse (!) by choosing $\tilde{K} \geq K \ldots$
- Remark on coins
- Suppose $\exists A \sim \mathcal{A}$, $\tilde{\kappa_A}(\cdot) \in [0, \tilde{K}]$ such that $\mathbb{E}_{\mathcal{A}}[\tilde{\kappa_A}(x)/\tilde{K}] = \kappa(x)/\tilde{K}$ then:
  - Simulating $A \sim \mathcal{A}$ $PP(\tilde{K})$ and accepting w.p. $\tilde{\kappa}_A(X_t)/\tilde{K} \equiv$ *.

Subsampling

- QSMC $\equiv$ Simulating BM + inhomogeneous Poisson Process κ
- Evaluating κ is $\mathcal{O}(N)$.
- * If $\forall x, \kappa(x) \leq K$ (requires localisation argument) then:
  - Simulating $PP(\kappa(x)) \equiv$ Simulating $PP(K)$ and accepting w.p. $\kappa(X_t)/K$.
- We can make our algorithm worse (!) by choosing $\tilde{K} \geq K \ldots$
- Remark on coins
- Suppose $\exists A \sim \mathcal{A}, \tilde{\kappa_A}(\cdot) \in [0, \tilde{K}]$ such that $\mathbb{E}_{\mathcal{A}}[\tilde{\kappa_A}(x)/\tilde{K}] = \kappa(x)/\tilde{K}$ then:
  - Simulating $A \sim \mathcal{A}$ $PP(\tilde{K})$ and accepting w.p. $\tilde{\kappa}_A(X_t)/\tilde{K} \equiv$ *.

Subsampling

- QSMC ≡ Simulating BM + inhomogeneous Poisson Process κ
- Evaluating κ is $\mathcal{O}(N)$.
- * If $\forall x$, $\kappa(x) \leq K$ (requires localisation argument) then:
  - Simulating $PP(\kappa(x)) \equiv$ Simulating $PP(K)$ and accepting w.p. $\kappa(X_t)/K$.
- We can make our algorithm worse (!) by choosing $\tilde{K} \geq K \ldots$
- Remark on coins
- Suppose $\exists A \sim \mathcal{A}$, $\tilde{\kappa_A}(\cdot) \in [0, \tilde{K}]$ such that $\mathbb{E}_{\mathcal{A}}[\tilde{\kappa_A}(x)/\tilde{K}] = \kappa(x)/\tilde{K}$ then:
  - Simulating $A \sim \mathcal{A}$ $PP(\tilde{K})$ and accepting w.p. $\tilde{\kappa}_A(X_t)/\tilde{K} \equiv$ *.

Subsampling

- QSMC $\equiv$ Simulating BM + inhomogeneous Poisson Process κ
- Evaluating κ is $\mathcal{O}(N)$.
- * If $\forall x$, $\kappa(x) \leq K$ (requires localisation argument) then:
  - Simulating $PP(\kappa(x)) \equiv$ Simulating $PP(K)$ and accepting w.p. $\kappa(X_t)/K$.
- We can make our algorithm worse (!) by choosing $\tilde{K} \geq K \ldots$
- Remark on coins
- Suppose $\exists A \sim \mathcal{A}$, $\tilde{\kappa_A}(\cdot) \in [0, \tilde{K}]$ such that $\mathbb{E}_{\mathcal{A}}[\tilde{\kappa_A}(x)/\tilde{K}] = \kappa(x)/\tilde{K}$ then:
  - Simulating $A \sim \mathcal{A}$ $PP(\tilde{K})$ and accepting w.p. $\tilde{\kappa}_A(X_t)/\tilde{K} \equiv$ *.

Subsampling

- QSMC $\equiv$ Simulating BM + inhomogeneous Poisson Process κ
- Evaluating κ is $\mathcal{O}(N)$.
- * If $\forall x,\ \kappa(x) \leq K$ (requires localisation argument) then:
  - Simulating $PP(\kappa(x)) \equiv$ Simulating $PP(K)$ and accepting w.p. $\kappa(X_t)/K$.
- We can make our algorithm worse (!) by choosing $\tilde{K} \geq K\ldots$
- Remark on coins
- Suppose $\exists A \sim \mathcal{A},\ \tilde{\kappa_A}(\cdot) \in [0, \tilde{K}]$ such that $\mathbb{E}_{\mathcal{A}}[\tilde{\kappa_A}(x)/\tilde{K}] = \kappa(x)/\tilde{K}$ then:
  - Simulating $A \sim \mathcal{A}\ PP(\tilde{K})$ and accepting w.p. $\tilde{\kappa}_A(X_t)/\tilde{K} \equiv *$.

Subsampling

- QSMC $\equiv$ Simulating BM + inhomogeneous Poisson Process κ
- Evaluating κ is $\mathcal{O}(N)$.
- * If $\forall x$, $\kappa(x) \leq K$ (requires localisation argument) then:
  - Simulating $PP(\kappa(x)) \equiv$ Simulating $PP(K)$ and accepting w.p. $\kappa(X_t)/K$.
- We can make our algorithm worse (!) by choosing $\tilde{K} \geq K \ldots$
- Remark on coins
- Suppose $\exists A \sim \mathcal{A}$, $\tilde{\kappa_A}(\cdot) \in [0, \tilde{K}]$ such that $\mathbb{E}_{\mathcal{A}}[\tilde{\kappa_A}(x)/\tilde{K}] = \kappa(x)/\tilde{K}$ then:
  - Simulating $A \sim \mathcal{A}$ $PP(\tilde{K})$ and accepting w.p. $\tilde{\kappa}_A(X_t)/\tilde{K} \equiv$ *.

Subsampling

- QSMC $\equiv$ Simulating BM + inhomogeneous Poisson Process κ
- Evaluating κ is $\mathcal{O}(N)$.
- * If $\forall x$, $\kappa(x) \leq K$ (requires localisation argument) then:
  - Simulating $PP(\kappa(x)) \equiv$ Simulating $PP(K)$ and accepting w.p. $\kappa(X_t)/K$.
- We can make our algorithm worse (!) by choosing $\tilde{K} \geq K\ldots$
- Remark on coins
- Suppose $\exists A \sim \mathcal{A}$, $\tilde{\kappa_A}(\cdot) \in [0, \tilde{K}]$ such that $\mathbb{E}_{\mathcal{A}}[\tilde{\kappa_A}(x)/\tilde{K}] = \kappa(x)/\tilde{K}$ then:
  - Simulating $A \sim \mathcal{A}$ $PP(\tilde{K})$ and accepting w.p. $\tilde{\kappa}_A(X_t)/\tilde{K} \equiv$ *.

Subsampling

- Scalability $\equiv$ Finding $A \sim \mathcal{A}$ and $\tilde{\kappa}_A(\cdot)$ which are $\mathcal{O}(1)$ (trivial), such that $\tilde{K}/K \geq 1$ scales well...
- Intuition is the diffusion drift is a sum:

$$\nabla \log \pi(x) = \sum_{i=0}^{N} \nabla \log f_i(x)$$

- We require control variates for good scaling of $\tilde{K}/K$... (omitted)

Subsampling

- Scalability $\equiv$ Finding $A \sim \mathcal{A}$ and $\tilde{\kappa}_A(\cdot)$ which are $\mathcal{O}(1)$ (trivial), such that $\tilde{K}/K \geq 1$ scales well…
- Intuition is the diffusion drift is a sum:

$$\nabla \log \pi(x) = \sum_{i=0}^{N} \nabla \log f_i(x)$$

- We require control variates for good scaling of $\tilde{K}/K$… (omitted)

Subsampling

- Scalability $\equiv$ Finding $A \sim \mathcal{A}$ and $\tilde{\kappa}_A(\cdot)$ which are $\mathcal{O}(1)$ (trivial), such that $\tilde{K}/K \geq 1$ scales well...
- Intuition is the diffusion drift is a sum:

$$\nabla \log \pi(x) = \sum_{i=0}^{N} \nabla \log f_i(x)$$

- We require control variates for good scaling of $\tilde{K}/K \ldots$ (omitted)

1.3 - Single-Core: ScaLE

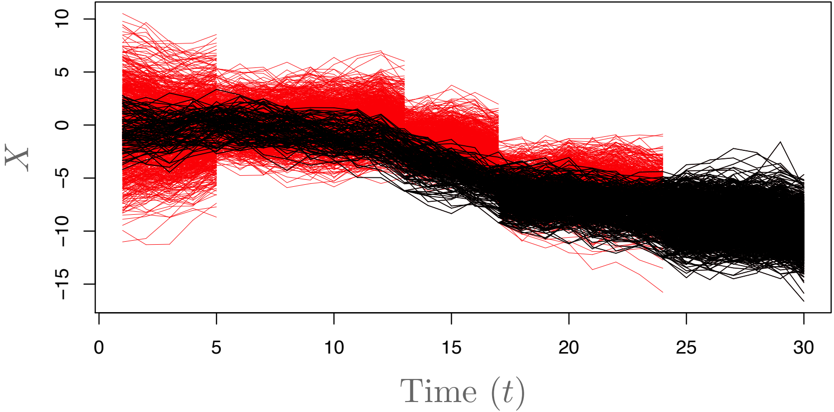
ScaLE

- Implementational Problem: Trajectory death!
- First Approach: Scalable Langevin Exact Algorithm (ScaLE)
  - Continuous time multi-level splitting / Importance sampling QSMC + SMC + Resampling

ScaLE

- Implementational Problem: Trajectory death!
- First Approach: Scalable Langevin Exact Algorithm (ScaLE)
  - Continuous time multi-level splitting / Importance sampling QSMC + SMC + Resampling

- Implementational Problem: Trajectory death!
- First Approach: Scalable Langevin Exact Algorithm (ScaLE)
  - Continuous time multi-level splitting / Importance sampling QSMC + SMC + Resampling


X
10
5
0
−5
−10
−15
0
5
10
15
20
25
30
Time (t)

1.4 - Summary

Summary

- Summary. . . :
  - QSMC: ‘Exact’ Bayesian Inference
  - No intrinsic cost for exactness.
  - ScaLE’s well!
- Missing Bits. . . :
  - Localisation
  - Theory: QSMC; (SMC-) ScaLE; Re-ScaLE.
  - Scaling: Dimensionality; Control-Variate. . .
  - Implementational Details

Summary

- Summary. . . :
  - QSMC: 'Exact' Bayesian Inference
  - No intrinsic cost for exactness.
  - ScaLE's well!
- Missing Bits. . . :
  - Localisation
  - Theory: QSMC; (SMC-) ScaLE; Re-ScaLE.
  - Scaling: Dimensionality; Control-Variate. . .
  - Implementational Details

Summary

- Summary. . . :
  - QSMC: ‘Exact’ Bayesian Inference
  - No intrinsic cost for exactness.
  - ScaLE’s well!
- Missing Bits. . . :
  - Localisation
  - Theory: QSMC; (SMC-) ScaLE; Re-ScaLE.
  - Scaling: Dimensionality; Control-Variate. . .
  - Implementational Details

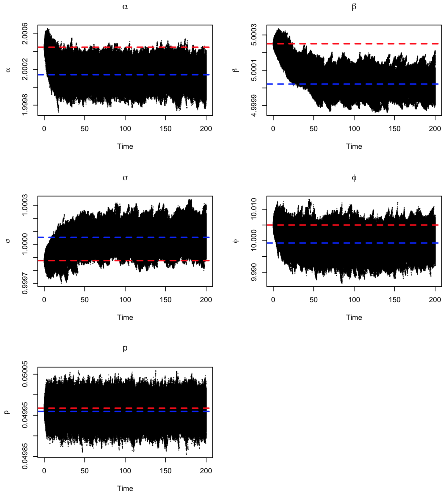
Summary

- Summary. . . :
  - QSMC: ‘Exact’ Bayesian Inference
  - No intrinsic cost for exactness.
  - ScaLE’s well!
- Missing Bits. . . :
  - Localisation
  - Theory: QSMC; (SMC-) ScaLE; Re-ScaLE.
  - Scaling: Dimensionality; Control-Variate. . .
  - Implementational Details

Summary

- Summary. . . :
  - QSMC: 'Exact' Bayesian Inference
  - No intrinsic cost for exactness.
  - ScaLE's well!
- Missing Bits. . . :
  - Localisation
  - Theory: QSMC; (SMC-) ScaLE; Re-ScaLE.
  - Scaling: Dimensionality; Control-Variate. . .
  - Implementational Details

2^{27} dataset, contaminated regression model



2 - Multi-Core: Bayesian Fusion

Problem

- Recall Target:

$$\pi(x) \propto \prod_{c=1}^{C} f_c(x).$$

- C - Number of cores / experts / 'views' ...; f_c - Sub-posterior.
- Simple Approach... [Think (A)BC]
  1. Simulate $X^{(1)} \sim f_1, X^{(2)} \sim f_2, \ldots, X^{(C)} \sim f_C$.
  2. Accept if $X^{(1)} = X^{(2)} = \ldots = X^{(C)}$, else go to 1/.
  3. Return $X := X^{(1)}$ $(\sim \prod_{i=0}^{C} f_i \propto \pi)$.
- Recall Langevin: If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$:
  - $\mathbb{L}_1, \ldots, \mathbb{L}_C, \mathbb{D}\mathbb{L}_1, \ldots, \mathbb{D}\mathbb{L}_C \ldots$

Problem

- Recall Target:

$$\pi(x) \propto \prod_{c=1}^{C} f_c(x).$$

- C - Number of cores / experts / ‘views’ ...; f_c - Sub-posterior.

- Simple Approach... [Think (A)BC]
  1. Simulate $X^{(1)} \sim f_1, X^{(2)} \sim f_2, \ldots, X^{(C)} \sim f_C$.
  2. Accept if $X^{(1)} = X^{(2)} = \ldots = X^{(C)}$, else go to 1/.
  3. Return $X := X^{(1)}$ $(\sim \prod_{i=0}^{C} f_i \propto \pi)$.

- Recall Langevin: If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$:
  - $\mathbb{L}_1, \ldots, \mathbb{L}_C, \mathbb{D}\mathbb{L}_1, \ldots, \mathbb{D}\mathbb{L}_C \ldots$

Problem

- Recall Target:

$$\pi(x) \propto \prod_{c=1}^{C} f_c(x).$$

- C - Number of cores / experts / 'views' ...; f_c - Sub-posterior.
- Simple Approach... [Think (A)BC]
  1. Simulate $X^{(1)} \sim f_1, X^{(2)} \sim f_2, \ldots, X^{(C)} \sim f_C$.
  2. Accept if $X^{(1)} = X^{(2)} = \ldots = X^{(C)}$, else go to 1/.
  3. Return $X := X^{(1)}$ $(\sim \prod_{i=0}^{C} f_i \propto \pi)$.
- Recall Langevin: If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$:
  - $\mathbb{L}_1, \ldots, \mathbb{L}_C, \mathbb{DL}_1, \ldots, \mathbb{DL}_C \ldots$

Problem

- Recall Target:

$$\pi(x) \propto \prod_{c=1}^{C} f_c(x).$$

- C - Number of cores / experts / 'views' …; f_c - Sub-posterior.
- Simple Approach… [Think (A)BC]
  1. Simulate $X^{(1)} \sim f_1, X^{(2)} \sim f_2, \ldots, X^{(C)} \sim f_C$.
  2. Accept if $X^{(1)} = X^{(2)} = \ldots = X^{(C)}$, else go to 1/.
  3. Return $X := X^{(1)}$ $(\sim \prod_{i=0}^{C} f_i \propto \pi)$.
- Recall Langevin: If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$:
  - $\mathbb{L}_1,\ldots,\mathbb{L}_C,\mathbb{D}\mathbb{L}_1,\ldots,\mathbb{D}\mathbb{L}_C\ldots$

Problem

- Recall Target:

$$\pi(x) \propto \prod_{c=1}^{C} f_c(x).$$

- C - Number of cores / experts / 'views' . . . ; f_c - Sub-posterior.
- Simple Approach. . . [Think (A)BC]
  1. Simulate $X^{(1)} \sim f_1, X^{(2)} \sim f_2, \ldots, X^{(C)} \sim f_C$.
  2. Accept if $X^{(1)} = X^{(2)} = \ldots = X^{(C)}$, else go to 1/.
  3. Return $X := X^{(1)}$ $(\sim \prod_{i=0}^{C} f_i \propto \pi)$.
- Recall Langevin: If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$:
  - $\mathbb{L}_1, \ldots, \mathbb{L}_C, \mathbb{D}\mathbb{L}_1, \ldots, \mathbb{D}\mathbb{L}_C \ldots$

Problem

- Recall Target:

$$\pi(x) \propto \prod_{c=1}^{C} f_c(x).$$

- C - Number of cores / experts / 'views' ... ; f_c - Sub-posterior.
- Simple Approach... [Think (A)BC]
  1. Simulate $X^{(1)} \sim f_1, X^{(2)} \sim f_2, \ldots, X^{(C)} \sim f_C$.
  2. Accept if $X^{(1)} = X^{(2)} = \ldots = X^{(C)}$, else go to 1/.
  3. Return $X := X^{(1)}$ $(\sim \prod_{i=0}^{C} f_i \propto \pi)$.
- Recall Langevin: If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$:
  - $\mathbb{L}_1, \ldots, \mathbb{L}_C, \mathbb{D}\mathbb{L}_1, \ldots, \mathbb{D}\mathbb{L}_C \ldots$

Problem

- Recall Target:

$$\pi(x) \propto \prod_{c=1}^{C} f_c(x).$$

- C - Number of cores / experts / 'views' …; f_c - Sub-posterior.
- Simple Approach… [Think (A)BC]
  1. Simulate $X^{(1)} \sim f_1, X^{(2)} \sim f_2, \ldots, X^{(C)} \sim f_C$.
  2. Accept if $X^{(1)} = X^{(2)} = \ldots = X^{(C)}$, else go to 1/.
  3. Return $X := X^{(1)}$ $(\sim \prod_{i=0}^{C} f_i \propto \pi)$.
- Recall Langevin: If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$:
  - $\mathbb{L}_1, \ldots, \mathbb{L}_C, \mathbb{D}\mathbb{L}_1, \ldots, \mathbb{D}\mathbb{L}_C \ldots$

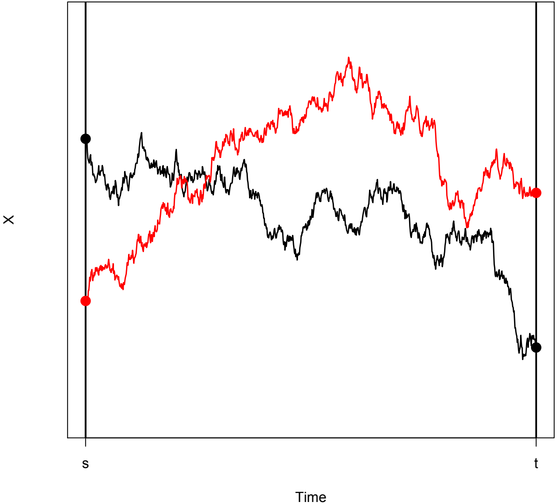
Problem

- Recall Target:

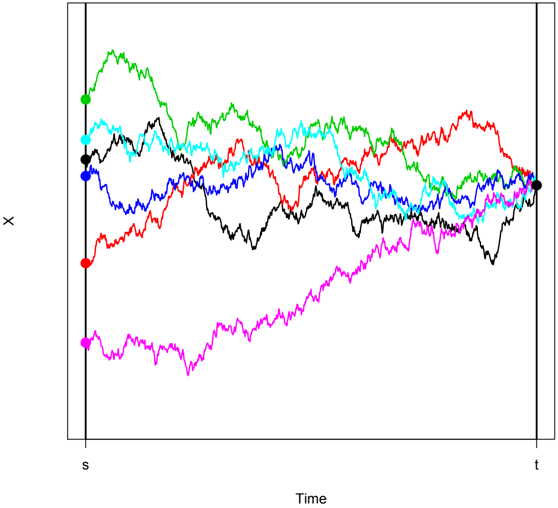
$$\pi(x) \propto \prod_{c=1}^{C} f_c(x).$$

- C - Number of cores / experts / 'views' ...; f_c - Sub-posterior.
- Simple Approach... [Think (A)BC]
  1. Simulate $X^{(1)} \sim f_1, X^{(2)} \sim f_2, \ldots, X^{(C)} \sim f_C$.
  2. Accept if $X^{(1)} = X^{(2)} = \ldots = X^{(C)}$, else go to 1/.
  3. Return $X := X^{(1)}$ ($\sim \prod_{i=0}^{C} f_i \propto \pi$).
- Recall Langevin: If $X_0 \sim \nu$, then $\forall t > 0, X_t \sim \nu$:
  - $\mathbb{L}_1, \ldots, \mathbb{L}_C, \mathbb{D}\mathbb{L}_1, \ldots, \mathbb{D}\mathbb{L}_C \ldots$

Fusion Idea



Fusion Actual



Some Details

- Fusion Measure ($\mathfrak{X} \in \mathbf{\Omega_0}$)

$$\mathrm{d}\mathbb{F}(\mathfrak{X}) \propto \mathrm{d}\left(\times_{c=1}^{C} \mathbb{DL}_c^{\boldsymbol{x}_0^{(c)}, \boldsymbol{y}_T}\right)(\mathfrak{X}) \cdot \prod_{c=1}^{C}\left[f_c^2\left(\boldsymbol{x}_0^{(c)}\right) p_{T,c}^{\mathrm{dl}}\left(\boldsymbol{y}_T \,\middle|\, \boldsymbol{x}_0^{(c)}\right) \cdot \frac{1}{f_c(\boldsymbol{y}_T)}\right],$$

- Key Idea: If $\mathfrak{X} \sim \mathbb{F}$, then $\mathfrak{X}_T \sim \prod_{c=1}^{C} f_c \propto \pi$ (!)

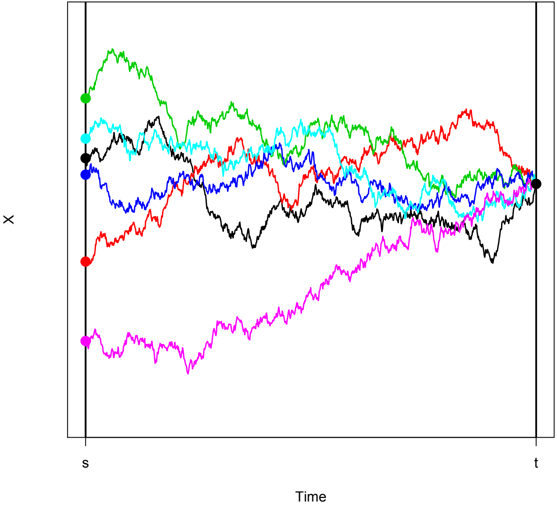
Some Details

- Fusion Measure ($\mathfrak{X} \in \boldsymbol{\Omega_0}$)

$$\mathrm{d}\mathbb{F}(\mathfrak{X}) \propto \mathrm{d}\left(\times_{c=1}^{C} \mathbb{DL}_c^{\boldsymbol{x}_0^{(c)},\boldsymbol{y}_T}\right)(\mathfrak{X}) \cdot \prod_{c=1}^{C}\left[f_c^2\left(\boldsymbol{x}_0^{(c)}\right) p_{T,c}^{\mathrm{dl}}\left(\boldsymbol{y}_T \,\middle|\, \boldsymbol{x}_0^{(c)}\right) \cdot \frac{1}{f_c(\boldsymbol{y}_T)}\right],$$

- Key Idea: If $\mathfrak{X} \sim \mathbb{F}$, then $\mathfrak{X}_T \sim \prod_{c=1}^{C} f_c \propto \pi$ (!)

Some Details



Some Details

- ‘Standard’ Multi-Core Problem ‘≡’ $\mathfrak{X} \sim \mathbb{F}$ (with practical constraints)
  - Rejection Sampling! Possible proposals $\mathfrak{X} \sim \mathbb{P}^{\cdot}$, w.p. $P(\mathfrak{X})$:
  - ‘Brownian’:

$$\mathrm{d}\mathbb{P}^{\text{bm}}(\mathfrak{X}) \propto \mathrm{d}\left(\times_{c=1}^{C} \mathbb{W}_c^{\boldsymbol{x}_0^{(c)}, \boldsymbol{y}_T}\right)(\mathfrak{X}) \cdot h_T^{\text{bm}}\left(\boldsymbol{X}_0^{(1:C)}, \boldsymbol{y}_T\right), \quad \mathfrak{X} \in \boldsymbol{\Omega_0}$$

Some Details

- 'Standard' Multi-Core Problem '$\equiv$' $\mathfrak{X} \sim \mathbb{F}$ (with practical constraints)
  - Rejection Sampling! Possible proposals $\mathfrak{X} \sim \mathbb{P}^{\cdot}$, w.p. $P(\mathfrak{X})$:
  - 'Brownian':

$$\mathrm{d}\mathbb{P}^{\mathrm{bm}}(\mathfrak{X}) \propto \mathrm{d}\left(\times_{c=1}^{C} \mathbb{W}_c^{\boldsymbol{x}_0^{(c)}, \boldsymbol{y}_T}\right)(\mathfrak{X}) \cdot h_T^{\mathrm{bm}}\left(\boldsymbol{x}_0^{(1:C)}, \boldsymbol{y}_T\right), \quad \mathfrak{X} \in \boldsymbol{\Omega_0}$$

Some Details

- 'Standard' Multi-Core Problem '≡' $\mathfrak{X} \sim \mathbb{F}$ (with practical constraints)
  - Rejection Sampling! Possible proposals $\mathfrak{X} \sim \mathbb{P}^{\cdot}$, w.p. $P(\mathfrak{X})$:
  - 'Brownian':

$$\mathrm{d}\mathbb{P}^{\mathrm{bm}}(\mathfrak{X}) \propto \mathrm{d}\left(\times_{c=1}^{C} \mathbb{W}_{c}^{\boldsymbol{X}_0^{(c)}, \boldsymbol{y}_T}\right)(\mathfrak{X}) \cdot h_T^{\mathrm{bm}}\left(\boldsymbol{X}_0^{(1:C)}, \boldsymbol{y}_T\right), \quad \mathfrak{X} \in \boldsymbol{\Omega_0}$$

Some Details

- Simple 'Brownian' Case:
  - 'Optimal' $h_T^{\text{bm}}(\cdot,\cdot)$:

$$h_T^{\text{bm}}\left(\boldsymbol{x}_0^{(1:C)},\boldsymbol{y}_T\right) \propto \underbrace{\left[\prod_{c=1}^{C} f_c\left(\boldsymbol{x}_0^{(c)}\right)\right]}_{\text{initial core draws}} \underbrace{\exp\left(-\frac{C\cdot\|\boldsymbol{y}_T-\bar{\boldsymbol{x}}_0\|^2}{2T}\right)\cdot\exp\left(-\frac{C\sigma^2}{2T}\right)}_{\text{end point draw}}$$

  - Need RS for $h_T^{\text{bm}}(\cdot,\cdot)$ end point.
  - Accept with probability

$$P(\mathfrak{X}) := \exp\left[-\sum_{c=1}^{C}\int_0^T \kappa_c(\boldsymbol{X}_t^{(c)})\,\mathrm{d}t\right] \in [0,1]$$

  - Exact ('Talking') vs. Approximate ('Silent' / 'Lecture') Fusion
- Remark: 'Ornstein-Uhlenbeck' special case

Some Details

- Simple ‘Brownian’ Case:
  - ‘Optimal’ $h_T^{\text{bm}}(\cdot,\cdot)$:

$$h_T^{\text{bm}}\left(\boldsymbol{X}_0^{(1:C)}, \boldsymbol{y}_T\right) \propto \underbrace{\left[\prod_{c=1}^{C} f_c\left(\boldsymbol{X}_0^{(c)}\right)\right]}_{\text{initial core draws}} \underbrace{\exp\left(-\frac{C \cdot \|\boldsymbol{y}_T - \bar{\boldsymbol{X}}_0\|^2}{2T}\right) \cdot \exp\left(-\frac{C\sigma^2}{2T}\right)}_{\text{end point draw}}$$

  - Need RS for $h_T^{\text{bm}}(\cdot,\cdot)$ end point.
  - Accept with probability

$$P(\mathfrak{X}) := \exp\left[-\sum_{c=1}^{C}\int_0^T \kappa_c\left(\boldsymbol{X}_t^{(c)}\right)\mathrm{d}t\right] \in [0,1]$$

  - Exact (‘Talking’) vs. Approximate (‘Silent’ / ‘Lecture’) Fusion
- Remark: ‘Ornstein-Uhlenbeck’ special case

Some Details

- Simple ‘Brownian’ Case:
  - ‘Optimal’ $h_T^{\mathrm{bm}}(\cdot,\cdot)$:

$$h_T^{\mathrm{bm}}\left(\boldsymbol{X}_0^{(1:C)}, \boldsymbol{y}_T\right) \propto \underbrace{\left[\prod_{c=1}^{C} f_c\left(\boldsymbol{X}_0^{(c)}\right)\right]}_{\text{initial core draws}} \underbrace{\exp\left(-\frac{C \cdot \|\boldsymbol{y}_T - \bar{\boldsymbol{X}}_0\|^2}{2T}\right) \cdot \exp\left(-\frac{C\sigma^2}{2T}\right)}_{\text{end point draw}}$$

  - Need RS for $h_T^{\mathrm{bm}}(\cdot,\cdot)$ end point.
  - Accept with probability

$$P(\mathfrak{X}) := \exp\left[-\sum_{c=1}^{C}\int_0^T \kappa_c(\boldsymbol{X}_t^{(c)})\,\mathrm{d}t\right] \in [0,1]$$

  - Exact (‘Talking’) vs. Approximate (‘Silent’ / ‘Lecture’) Fusion
- Remark: ‘Ornstein-Uhlenbeck’ special case

Some Details

- Simple 'Brownian' Case:
  - 'Optimal' $h_T^{\text{bm}}(\cdot,\cdot)$:

$$h_T^{\text{bm}}\left(\boldsymbol{X}_0^{(1:C)}, \boldsymbol{y}_T\right) \propto \underbrace{\left[\prod_{c=1}^{C} f_c\left(\boldsymbol{X}_0^{(c)}\right)\right]}_{\text{initial core draws}} \underbrace{\exp\left(-\frac{C \cdot \|\boldsymbol{y}_T - \bar{\boldsymbol{X}}_0\|^2}{2T}\right) \cdot \exp\left(-\frac{C\sigma^2}{2T}\right)}_{\text{end point draw}}$$

  - Need RS for $h_T^{\text{bm}}(\cdot,\cdot)$ end point.
  - Accept with probability

$$P(\mathfrak{X}) := \exp\left[-\sum_{c=1}^{C}\int_0^T \kappa_c(\boldsymbol{X}_t^{(c)})\,\mathrm{d}t\right] \in [0,1]$$

  - Exact ('Talking') vs. Approximate ('Silent' / 'Lecture') Fusion
- Remark: 'Ornstein-Uhlenbeck' special case

Some Details

- Simple 'Brownian' Case:
  - 'Optimal' $h_T^{\mathrm{bm}}(\cdot,\cdot)$:

$$h_T^{\mathrm{bm}}\left(\boldsymbol{X}_0^{(1:C)}, \boldsymbol{y}_T\right) \propto \underbrace{\left[\prod_{c=1}^{C} f_c\left(\boldsymbol{X}_0^{(c)}\right)\right]}_{\text{initial core draws}} \underbrace{\exp\left(-\frac{C\cdot\|\boldsymbol{y}_T - \bar{\boldsymbol{X}}_0\|^2}{2T}\right)\cdot\exp\left(-\frac{C\sigma^2}{2T}\right)}_{\text{end point draw}}$$

  - Need RS for $h_T^{\mathrm{bm}}(\cdot,\cdot)$ end point.
  - Accept with probability

$$P(\mathfrak{X}) := \exp\left[-\sum_{c=1}^{C}\int_0^T \kappa_c(\boldsymbol{X}_t^{(c)})\,\mathrm{d}t\right] \in [0,1]$$

  - Exact ('Talking') vs. Approximate ('Silent' / 'Lecture') Fusion

- Remark: 'Ornstein-Uhlenbeck' special case

Some Details

- Simple 'Brownian' Case:
  - 'Optimal' $h_T^{\mathrm{bm}}(\cdot,\cdot)$:

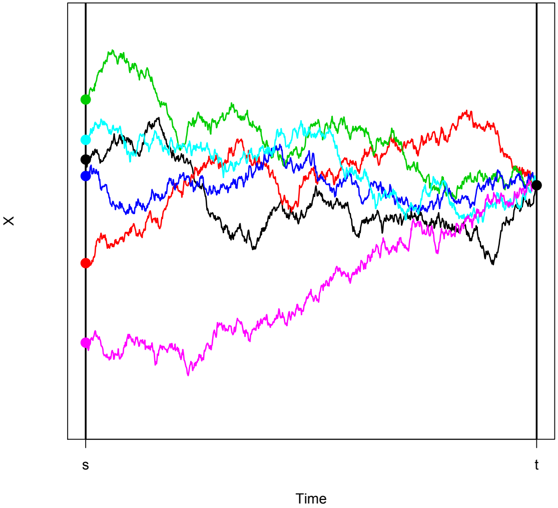
$$h_T^{\mathrm{bm}}\left(\boldsymbol{X}_0^{(1:C)}, \boldsymbol{y}_T\right) \propto \underbrace{\left[\prod_{c=1}^{C} f_c\left(\boldsymbol{X}_0^{(c)}\right)\right]}_{\text{initial core draws}} \underbrace{\exp\left(-\frac{C \cdot \|\boldsymbol{y}_T - \bar{\boldsymbol{X}}_0\|^2}{2T}\right) \cdot \exp\left(-\frac{C\sigma^2}{2T}\right)}_{\text{end point draw}}$$

  - Need RS for $h_T^{\mathrm{bm}}(\cdot,\cdot)$ end point.
  - Accept with probability

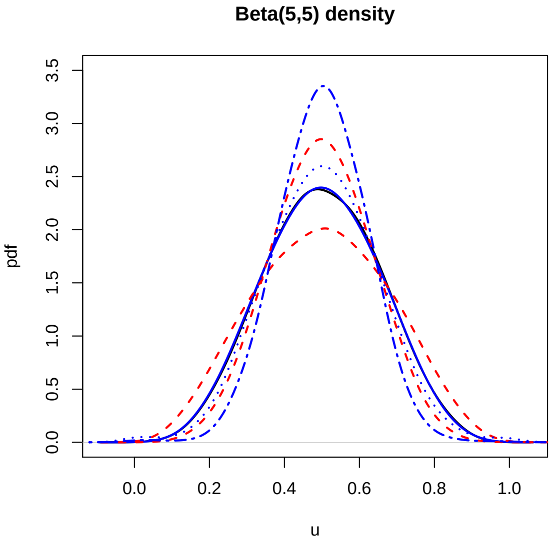
$$P(\mathfrak{X}) := \exp\left[-\sum_{c=1}^{C}\int_0^T \kappa_c(\boldsymbol{X}_t^{(c)})\,\mathrm{d}t\right] \in [0,1]$$

  - Exact ('Talking') vs. Approximate ('Silent' / 'Lecture') Fusion
- Remark: 'Ornstein-Uhlenbeck' special case

Some Details



Example



Questions?