Optimal Scaing and Adaptive Markov Chain Monte Carlo

Krzysztof Latuszynski
(University of Warwick, UK)

OxWaSP - module 1

the usual MCMC setting

- let π be a target probability distribution on $\mathcal{X}$, typically arising as a posterior distribution in Bayesian inference,
- the goal is to evaluate
$$I := \int_{\mathcal{X}} f(x)\pi(dx).$$
- direct sampling from π is not possible or inefficient for example π is known up to a normalising constant
- MCMC approach is to simulate $(X_n)_{n\geq 0}$, an ergodic Markov chain with **transition kernel** P and limiting distribution π, and take ergodic averages as an estimate of I.
- the usual estimate
$$\hat{I} := \frac{1}{n}\sum_{k=t}^{t+n} f(X_k)$$
- **SLLN** for Markov chains holds under very mild conditions
- **CLT** for Markov chains holds under some additional assumptions and is verifiable in many situations of interest

the usual MCMC setting

- let π be a target probability distribution on $\mathcal{X}$, typically arising as a posterior distribution in Bayesian inference,
- the goal is to evaluate
$$I := \int_{\mathcal{X}} f(x)\pi(dx).$$
- direct sampling from π is not possible or inefficient
for example π is known up to a normalising constant
- MCMC approach is to simulate $(X_n)_{n\geq 0}$, an ergodic Markov chain with **transition kernel** P and limiting distribution π, and take ergodic averages as an estimate of I.
- the usual estimate
$$\hat{I} := \frac{1}{n}\sum_{k=t}^{t+n} f(X_k)$$
- **SLLN** for Markov chains holds under very mild conditions
- **CLT** for Markov chains holds under some additional assumptions and is verifiable in many situations of interest

the usual MCMC setting

- let π be a target probability distribution on $\mathcal{X}$, typically arising as a posterior distribution in Bayesian inference,
- the goal is to evaluate

$$I := \int_{\mathcal{X}} f(x)\pi(dx).$$

- direct sampling from π is not possible or inefficient
  for example π is known up to a normalising constant
- MCMC approach is to simulate $(X_n)_{n\geq 0}$, an ergodic Markov chain with **transition kernel** P and limiting distribution π, and take ergodic averages as an estimate of I.
- the usual estimate

$$\hat{I} := \frac{1}{n}\sum_{k=t}^{t+n} f(X_k)$$

- **SLLN** for Markov chains holds under very mild conditions
- **CLT** for Markov chains holds under some additional assumptions and is verifiable in many situations of interest

the usual MCMC setting

- let π be a target probability distribution on $\mathcal{X}$, typically arising as a posterior distribution in Bayesian inference,
- the goal is to evaluate
$$I := \int_{\mathcal{X}} f(x)\pi(dx).$$
- direct sampling from π is not possible or inefficient
for example π is known up to a normalising constant
- MCMC approach is to simulate $(X_n)_{n\geq 0}$, an ergodic Markov chain with **transition kernel** P and limiting distribution π, and take ergodic averages as an estimate of I.
- the usual estimate
$$\hat{I} := \frac{1}{n}\sum_{k=t}^{t+n} f(X_k)$$
- **SLLN** for Markov chains holds under very mild conditions
- **CLT** for Markov chains holds under some additional assumptions and is verifiable in many situations of interest

the usual MCMC setting

- let π be a target probability distribution on $\mathcal{X}$, typically arising as a posterior distribution in Bayesian inference,
- the goal is to evaluate
$$I := \int_{\mathcal{X}} f(x)\pi(dx).$$
- direct sampling from π is not possible or inefficient
for example π is known up to a normalising constant
- MCMC approach is to simulate $(X_n)_{n\geq 0}$, an ergodic Markov chain with **transition kernel** P and limiting distribution π, and take ergodic averages as an estimate of I.
- the usual estimate
$$\hat{I} := \frac{1}{n}\sum_{k=t}^{t+n} f(X_k)$$
- **SLLN** for Markov chains holds under very mild conditions
- **CLT** for Markov chains holds under some additional assumptions and is verifiable in many situations of interest

the usual MCMC setting

- let π be a target probability distribution on $\mathcal{X}$, typically arising as a posterior distribution in Bayesian inference,
- the goal is to evaluate

$$I := \int_{\mathcal{X}} f(x)\pi(dx).$$

- direct sampling from π is not possible or inefficient
  for example π is known up to a normalising constant
- MCMC approach is to simulate $(X_n)_{n\geq 0}$, an ergodic Markov chain with **transition kernel** P and limiting distribution π, and take ergodic averages as an estimate of I.
- the usual estimate

$$\hat{I} := \frac{1}{n}\sum_{k=t}^{t+n} f(X_k)$$

- **SLLN** for Markov chains holds under very mild conditions
- **CLT** for Markov chains holds under some additional assumptions and is verifiable in many situations of interest

Reversibility and stationarity

- How to design P so that X_n converges in distribution to π ?
- **Definition.** P is reversible with respect to π if

$$\pi(x)P(x,y) = \pi(y)P(y,x)$$

  as measures on $\mathcal{X} \times \mathcal{X}$
- **Lemma.** If P is reversible with respect to π then $\pi P = \pi$, so it is also stationary.

Reversibility and stationarity

- How to design P so that X_n converges in distribution to π ?
- **Definition.** P is reversible with respect to π if

$$\pi(x)P(x,y) = \pi(y)P(y,x)$$

as measures on $\mathcal{X} \times \mathcal{X}$

- **Lemma.** If P is reversible with respect to π then $\pi P = \pi$, so it is also stationary.

Reversibility and stationarity

- How to design P so that X_n converges in distribution to π ?
- **Definition.** P is reversible with respect to π if

$$\pi(x)P(x,y) = \pi(y)P(y,x)$$

  as measures on $\mathcal{X} \times \mathcal{X}$
- **Lemma.** If P is reversible with respect to π then $\pi P = \pi$, so it is also stationary.

The Metropolis algorithm

- **Idea.** Take any transition kernel Q with transition densities $q(x, y)$ and make it reversible with respect to π
- **Algorithm.** Given X_n
  sample $Y_{n+1} \sim Q(X_n, \cdot)$
- with probability $\alpha(X_n, Y_{n+1})$ set $X_{n+1} = Y_{n+1}$, otherwise set $X_{n+1} = X_n$
- where

$$\alpha(x, y) = \min\{1, \frac{\pi(y)q(y, x)}{\pi(x)q(x, y)}\}.$$

- Under mild assumptions on Q the algorithm is ergodic.
- However it's performance depends heavily on Q
- is is **difficult** to design the proposal Q so that P has **good convergence properties**, especially if $\mathcal{X}$ is high dimensional

The Metropolis algorithm

- **Idea.** Take any transition kernel Q with transition densities $q(x, y)$ and make it reversible with respect to π
- **Algorithm.** Given X_n
  sample $Y_{n+1} \sim Q(X_n, \cdot)$
- with probability $\alpha(X_n, Y_{n+1})$ set $X_{n+1} = Y_{n+1}$, otherwise set $X_{n+1} = X_n$
- where

$$\alpha(x, y) = \min\{1, \frac{\pi(y)q(y, x)}{\pi(x)q(x, y)}\}.$$

- Under mild assumptions on Q the algorithm is ergodic.
- However it's performance depends heavily on Q
- is is **difficult** to design the proposal Q so that P has **good convergence properties**, especially if $\mathcal{X}$ is high dimensional

The Metropolis algorithm

- **Idea.** Take any transition kernel Q with transition densities $q(x,y)$ and make it reversible with respect to π
- **Algorithm.** Given X_n
  sample $Y_{n+1} \sim Q(X_n, \cdot)$
- with probability $\alpha(X_n, Y_{n+1})$ set $X_{n+1} = Y_{n+1}$, otherwise set $X_{n+1} = X_n$
- where
$$\alpha(x,y) = \min\{1, \frac{\pi(y)q(y,x)}{\pi(x)q(x,y)}\}.$$
- Under mild assumptions on Q the algorithm is ergodic.
- However it's performance depends heavily on Q
- is is **difficult** to design the proposal Q so that P has **good convergence properties**, especially if $\mathcal{X}$ is high dimensional

The Metropolis algorithm

- **Idea.** Take any transition kernel Q with transition densities $q(x, y)$ and make it reversible with respect to π
- **Algorithm.** Given X_n
  sample $Y_{n+1} \sim Q(X_n, \cdot)$
- with probability $\alpha(X_n, Y_{n+1})$ set $X_{n+1} = Y_{n+1}$, otherwise set $X_{n+1} = X_n$
- where

$$\alpha(x, y) = \min\{1, \frac{\pi(y)q(y, x)}{\pi(x)q(x, y)}\}.$$

- Under mild assumptions on Q the algorithm is ergodic.
- However it's performance depends heavily on Q
- is is **difficult** to design the proposal Q so that P has **good convergence properties**, especially if $\mathcal{X}$ is high dimensional

The Metropolis algorithm

- **Idea.** Take any transition kernel Q with transition densities $q(x, y)$ and make it reversible with respect to π
- **Algorithm.** Given X_n
  sample $Y_{n+1} \sim Q(X_n, \cdot)$
- with probability $\alpha(X_n, Y_{n+1})$ set $X_{n+1} = Y_{n+1}$, otherwise set $X_{n+1} = X_n$
- where
$$\alpha(x, y) = \min\{1, \frac{\pi(y)q(y, x)}{\pi(x)q(x, y)}\}.$$
- Under mild assumptions on Q the algorithm is ergodic.
- However it's performance depends heavily on Q
- is is **difficult** to design the proposal Q so that P has **good convergence properties**, especially if $\mathcal{X}$ is high dimensional

The Metropolis algorithm

- **Idea.** Take any transition kernel Q with transition densities $q(x, y)$ and make it reversible with respect to π
- **Algorithm.** Given X_n
  sample $Y_{n+1} \sim Q(X_n, \cdot)$
- with probability $\alpha(X_n, Y_{n+1})$ set $X_{n+1} = Y_{n+1}$, otherwise set $X_{n+1} = X_n$
- where
$$\alpha(x, y) = \min\{1, \frac{\pi(y)q(y, x)}{\pi(x)q(x, y)}\}.$$
- Under mild assumptions on Q the algorithm is ergodic.
- However it's performance depends heavily on Q
- is is **difficult** to design the proposal Q so that P has **good convergence properties**, especially if $\mathcal{X}$ is high dimensional

the scaling problem

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- what happens if σ is small?

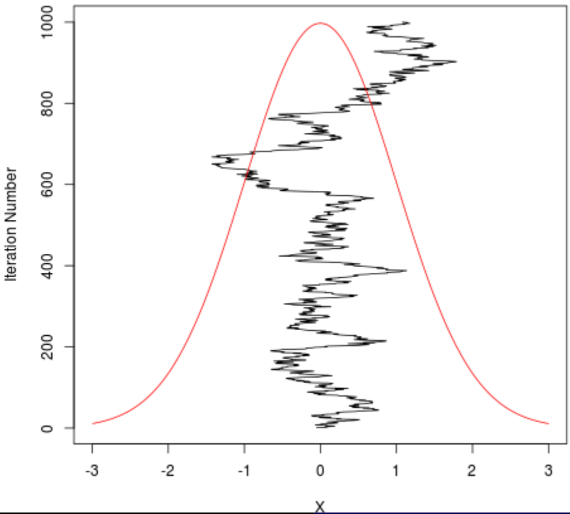
the scaling problem

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- what happens if σ is small?

the scaling problem

- take Random Walk Metropolis with proposal increments
-
$$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- what happens if σ is small?

small sigma...



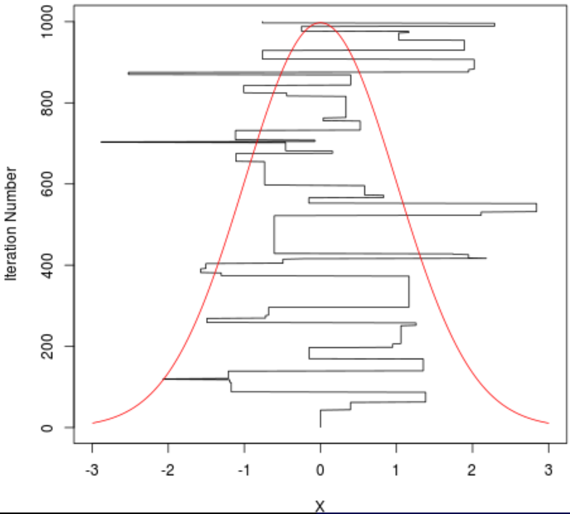
the scaling problem

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- what happens if σ is small?
- what happens if σ is large?

the scaling problem

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- what happens if σ is small?
- what happens if σ is large?

large sigma...



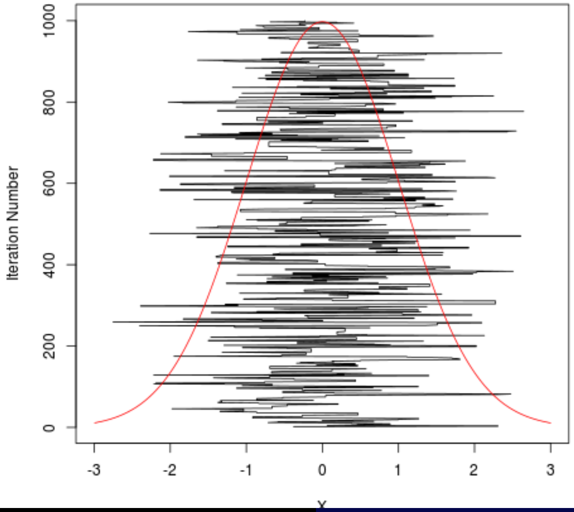
the scaling problem

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- what happens if σ is small?
- what happens if σ is large?
- so σ should be neither too small, nor too large (known as Goldilocks principle)

the scaling problem

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- what happens if σ is small?
- what happens if σ is large?
- so σ should be neither too small, nor too large (known as Goldilocks principle)

not too small and not too large...



diffusion limit [RGG97]

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- σ should be neither too small, nor too large (known as Goldilocks principle)
- but how to choose it?
- if the dimension of $\mathcal{X}$ goes to ∞, e.g. $\mathcal{X} = \mathbb{R}^d$, and $d \to \infty$,
- if the proposal is set as $Q = N(x, \frac{l^2}{d} I_d)$ for fixed $l > 0$,
- if we consider
$$Z_t = d^{-1/2} X^{(1)}_{\lfloor dt \rfloor}$$
- then Z_t converges to the Langevin diffusion
$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

diffusion limit [RGG97]

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- σ should be neither too small, nor too large (known as Goldilocks principle)
- but how to choose it?
- if the dimension of $\mathcal{X}$ goes to ∞, e.g. $\mathcal{X} = \mathbb{R}^d$, and $d \to \infty$,
- if the proposal is set as $Q = N(x, \frac{l^2}{d} I_d)$ for fixed $l > 0$,
- if we consider
$$Z_t = d^{-1/2} X^{(1)}_{\lfloor dt \rfloor}$$
- then Z_t converges to the Langevin diffusion
$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

diffusion limit [RGG97]

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- σ should be neither too small, nor too large (known as Goldilocks principle)
- but how to choose it?
- if the dimension of $\mathcal{X}$ goes to ∞, e.g. $\mathcal{X} = \mathbb{R}^d$, and $d \to \infty$,
- if the proposal is set as $Q = N(x, \frac{l^2}{d} I_d)$ for fixed $l > 0$,
- if we consider
$$Z_t = d^{-1/2} X^{(1)}_{\lfloor dt \rfloor}$$
- then Z_t converges to the Langevin diffusion
$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

diffusion limit [RGG97]

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- σ should be neither too small, nor too large (known as Goldilocks principle)
- but how to choose it?
- if the dimension of $\mathcal{X}$ goes to ∞, e.g. $\mathcal{X} = \mathbb{R}^d$, and $d \to \infty$,
- if the proposal is set as $Q = N(x, \frac{l^2}{d} I_d)$ for fixed $l > 0$,
- if we consider
$$Z_t = d^{-1/2} X^{(1)}_{\lfloor dt \rfloor}$$
- then Z_t converges to the Langevin diffusion
$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

diffusion limit [RGG97]

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- σ should be neither too small, nor too large (known as Goldilocks principle)
- but how to choose it?
- if the dimension of $\mathcal{X}$ goes to ∞, e.g. $\mathcal{X} = \mathbb{R}^d$, and $d \to \infty$,
- if the proposal is set as $Q = N(x, \frac{l^2}{d} I_d)$ for fixed $l > 0$,
- if we consider $$Z_t = d^{-1/2} X^{(1)}_{\lfloor dt \rfloor}$$
- then Z_t converges to the Langevin diffusion

$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

diffusion limit [RGG97]

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- σ should be neither too small, nor too large (known as Goldilocks principle)
- but how to choose it?
- if the dimension of $\mathcal{X}$ goes to ∞, e.g. $\mathcal{X} = \mathbb{R}^d$, and $d \to \infty$,
- if the proposal is set as $Q = N(x, \frac{l^2}{d} I_d)$ for fixed $l > 0$,
- if we consider
$$Z_t = d^{-1/2} X^{(1)}_{\lfloor dt \rfloor}$$
- then Z_t converges to the Langevin diffusion

$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

diffusion limit [RGG97]

- Z_t converges to the Langevin diffusion

$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

- where $h(l) = 2l^2 \Phi(-Cl/2)$ is the speed of the diffusion and $A(l) = 2\Phi(Cl/2)$ is the asymptotic acceptance rate.
- maximising the speed $h(l)$ yields the optimal acceptance rate

$$A(l) = 0.234$$

which is independent of the target distribution π
- it is a remarkable result since it gives a simple criterion (and the same for all target distributions π) to assess how well the Random Walk Metropolis is performing.

diffusion limit [RGG97]

- Z_t converges to the Langevin diffusion

$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

- where $h(l) = 2l^2 \Phi(-Cl/2)$ is the speed of the diffusion and $A(l) = 2\Phi(Cl/2)$ is the asymptotic acceptance rate.
- maximising the speed $h(l)$ yields the optimal acceptance rate

$$A(l) = 0.234$$

which is independent of the target distribution π
- it is a remarkable result since it gives a simple criterion (and the same for all target distributions π) to assess how well the Random Walk Metropolis is performing.

diffusion limit [RGG97]

- Z_t converges to the Langevin diffusion

$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

- where $h(l) = 2l^2 \Phi(-Cl/2)$ is the speed of the diffusion and $A(l) = 2\Phi(Cl/2)$ is the asymptotic acceptance rate.
- maximising the speed $h(l)$ yields the optimal acceptance rate

$$A(l) = 0.234$$

  which is independent of the target distribution π
- it is a remarkable result since it gives a simple criterion (and the same for all target distributions π) to assess how well the Random Walk Metropolis is performing.

diffusion limit [RGG97]

- Z_t converges to the Langevin diffusion

$$dZ_t = h(l)^{1/2} dB_t + \frac{1}{2} h(l) \nabla \log \pi(Z_t) dt$$

- where $h(l) = 2l^2 \Phi(-Cl/2)$ is the speed of the diffusion and $A(l) = 2\Phi(Cl/2)$ is the asymptotic acceptance rate.
- maximising the speed $h(l)$ yields the optimal acceptance rate

$$A(l) = 0.234$$

  which is independent of the target distribution π
- it is a remarkable result since it gives a simple criterion (and the same for all target distributions π) to assess how well the Random Walk Metropolis is performing.

the scaling problem cd

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_\sigma(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- so the theory says the optimal average acceptance rate
$$\bar{\alpha} := \int \int \alpha(x,y) q_\sigma(x, dy) \pi(dx)$$
should be approximately $\alpha^* = 0.234$
- however it is not possible to compute σ^* for which $\bar{\alpha} = \alpha^*$.
- It is very tempting to adjust σ on the fly while simulation progress
- some reasons:
  - when to stop estimating $\bar{\alpha}$? (to increase or decrease σ)
  - we may be in a Metropolis within Gibbs setting of dimension 10000

the scaling problem cd

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- so the theory says the optimal average acceptance rate
$$\bar{\alpha} := \int\int \alpha(x,y) q_{\sigma}(x, dy)\pi(dx)$$
should be approximately $\alpha^* = 0.234$
- however it is not possible to compute σ^* for which $\bar{\alpha} = \alpha^*$.
- It is very tempting to adjust σ on the fly while simulation progress
- some reasons:
  - when to stop estimating $\bar{\alpha}$? (to increase or decrease σ)
  - we may be in a Metropolis within Gibbs setting of dimension 10000

the scaling problem cd

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- so the theory says the optimal average acceptance rate

$$\bar{\alpha} := \int \int \alpha(x,y) q_{\sigma}(x, dy) \pi(dx)$$

should be approximately $\alpha^* = 0.234$

- however it is not possible to compute σ^* for which $\bar{\alpha} = \alpha^*$.
- It is very tempting to adjust σ on the fly while simulation progress
- some reasons:
  - when to stop estimating $\bar{\alpha}$? (to increase or decrease σ)
  - we may be in a Metropolis within Gibbs setting of dimension 10000

the scaling problem cd

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_\sigma(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- so the theory says the optimal average acceptance rate
$$\bar{\alpha} := \int\int \alpha(x,y) q_\sigma(x,dy)\pi(dx)$$
should be approximately $\alpha^* = 0.234$
- however it is not possible to compute σ^* for which $\bar{\alpha} = \alpha^*$.
- It is very tempting to adjust σ on the fly while simulation progress
- some reasons:
  - when to stop estimating $\bar{\alpha}$? (to increase or decrease σ)
  - we may be in a Metropolis within Gibbs setting of dimension 10000

the scaling problem cd

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_{\sigma}(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- so the theory says the optimal average acceptance rate
$$\bar{\alpha} := \int\int \alpha(x,y) q_{\sigma}(x,dy)\pi(dx)$$
should be approximately $\alpha^* = 0.234$
- however it is not possible to compute σ^* for which $\bar{\alpha} = \alpha^*$.
- It is very tempting to adjust σ on the fly while simulation progress
- some reasons:
  - when to stop estimating $\bar{\alpha}$? (to increase or decrease σ)
  - we may be in a Metropolis within Gibbs setting of dimension 10000

the scaling problem cd

- take Random Walk Metropolis with proposal increments
- $$Y_{n+1} \sim q_\sigma(X_n, \cdot) = X_n + \sigma N(0, Id).$$
- so the theory says the optimal average acceptance rate
$$\bar{\alpha} := \int \int \alpha(x, y) q_\sigma(x, dy) \pi(dx)$$
should be approximately $\alpha^* = 0.234$
- however it is not possible to compute σ^* for which $\bar{\alpha} = \alpha^*$.
- It is very tempting to adjust σ on the fly while simulation progress
- some reasons:
  - when to stop estimating $\bar{\alpha}$? (to increase or decrease σ)
  - we may be in a Metropolis within Gibbs setting of dimension 10000

the Adaptive Scaling Algorithm

1. draw proposal
$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, Id),$$
2. Set X_{n+1} according to the usual Metropolis acceptance rate $\alpha(X_n, Y_{n+1})$.
3. Update scale by
$$\log \sigma_{n+1} = \log \sigma_n + \gamma_n(\alpha(X_n, Y_{n+1}) - \alpha^*)$$
where $\gamma_n \to 0$.

- Recall we follow a very precise mathematical advice from diffusion limit analysis [RGG97]
- The algorithm dates back to [GRS98]
(a slightly different version making use of regenerations)
- Exactly this version analyzed in [Vih09]

the Adaptive Scaling Algorithm

1. draw proposal
$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, Id),$$
2. Set X_{n+1} according to the usual Metropolis acceptance rate $\alpha(X_n, Y_{n+1})$.
3. Update scale by
$$\log \sigma_{n+1} = \log \sigma_n + \gamma_n(\alpha(X_n, Y_{n+1}) - \alpha^*)$$
where $\gamma_n \to 0$.

- Recall we follow a very precise mathematical advice from diffusion limit analysis [RGG97]
- The algorithm dates back to [GRS98]
(a slightly different version making use of regenerations)
- Exactly this version analyzed in [Vih09]

the Adaptive Scaling Algorithm

1. draw proposal
$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, Id),$$
2. Set X_{n+1} according to the usual Metropolis acceptance rate $\alpha(X_n, Y_{n+1})$.
3. Update scale by
$$\log \sigma_{n+1} = \log \sigma_n + \gamma_n(\alpha(X_n, Y_{n+1}) - \alpha^*)$$
where $\gamma_n \to 0$.

- Recall we follow a very precise mathematical advice from diffusion limit analysis [RGG97]
- The algorithm dates back to [GRS98]
(a slightly different version making use of regenerations)
- Exactly this version analyzed in [Vih09]

the Adaptive Scaling Algorithm

1. draw proposal
$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, Id),$$
2. Set X_{n+1} according to the usual Metropolis acceptance rate $\alpha(X_n, Y_{n+1})$.
3. Update scale by
$$\log \sigma_{n+1} = \log \sigma_n + \gamma_n(\alpha(X_n, Y_{n+1}) - \alpha^*)$$
where $\gamma_n \to 0$.

- Recall we follow a very precise mathematical advice from diffusion limit analysis [RGG97]
- The algorithm dates back to [GRS98]
(a slightly different version making use of regenerations)
- Exactly this version analyzed in [Vih09]

the Adaptive Scaling Algorithm

1. draw proposal

$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, Id),$$

2. Set X_{n+1} according to the usual Metropolis acceptance rate $\alpha(X_n, Y_{n+1})$.
3. Update scale by

$$\log \sigma_{n+1} = \log \sigma_n + \gamma_n(\alpha(X_n, Y_{n+1}) - \alpha^*)$$

where $\gamma_n \to 0$.

- Recall we follow a very precise mathematical advice from diffusion limit analysis [RGG97]
- The algorithm dates back to [GRS98] (a slightly different version making use of regenerations)
- Exactly this version analyzed in [Vih09]

the Adaptive Scaling Algorithm

1. draw proposal
$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, Id),$$
2. Set X_{n+1} according to the usual Metropolis acceptance rate $\alpha(X_n, Y_{n+1})$.
3. Update scale by
$$\log \sigma_{n+1} = \log \sigma_n + \gamma_n(\alpha(X_n, Y_{n+1}) - \alpha^*)$$
where $\gamma_n \to 0$.

- Recall we follow a very precise mathematical advice from diffusion limit analysis [RGG97]
- The algorithm dates back to [GRS98]
(a slightly different version making use of regenerations)
- Exactly this version analyzed in [Vih09]

parametric family of transition kernels P_θ

- typically we can design a family of ergodic transition kernels P_θ, $\theta \in \Theta$.
- Ex 1a. $\Theta = R_+$
  P_θ - Random Walk Metropolis with proposal increments
  $$q_\theta = \theta N(0, Id)$$
- Ex 1b. $\Theta = R_+ \times \{d \text{ dimensional covariance matrices}\}$
  P_θ - Random Walk Metropolis with proposal increments
  $$q_\theta = \sigma N(0, \Sigma)$$
- Ex 2. $\Theta = \Delta_{d-1} := \{(\alpha_1, \ldots, \alpha_d) \in \mathbb{R}^d : \alpha_i \geq 0, \ \sum_{i=1}^d \alpha_i = 1\}$ the $(d-1)-$dimensional probability simplex,
  P_θ - Random Scan Gibbs Sampler with coordinate selection probabilities
  $$\theta = (\alpha_1, \ldots, \alpha_n)$$
- In each case values of θ will affect efficiency of P_θ

parametric family of transition kernels P_θ

- typically we can design a family of ergodic transition kernels P_θ, $\theta \in \Theta$.
- Ex 1a. $\Theta = R_+$
  P_θ - Random Walk Metropolis with proposal increments
$$q_\theta = \theta N(0, Id)$$
- Ex 1b. $\Theta = R_+ \times \{d \text{ dimensional covariance matrices}\}$
  P_θ - Random Walk Metropolis with proposal increments
$$q_\theta = \sigma N(0, \Sigma)$$
- Ex 2. $\Theta = \Delta_{d-1} := \{(\alpha_1, \ldots, \alpha_d) \in \mathbb{R}^d : \alpha_i \geq 0, \ \sum_{i=1}^d \alpha_i = 1\}$ the $(d-1)-$dimensional probability simplex,
  P_θ - Random Scan Gibbs Sampler with coordinate selection probabilities
$$\theta = (\alpha_1, \ldots, \alpha_n)$$
- In each case values of θ will affect efficiency of P_θ

parametric family of transition kernels P_θ

- typically we can design a family of ergodic transition kernels P_θ, $\theta \in \Theta$.
- Ex 1a. $\Theta = R_+$
  P_θ - Random Walk Metropolis with proposal increments

$$q_\theta = \theta N(0, Id)$$

- Ex 1b. $\Theta = R_+ \times \{d \text{ dimensional covariance matrices}\}$
  P_θ - Random Walk Metropolis with proposal increments

$$q_\theta = \sigma N(0, \Sigma)$$

- Ex 2. $\Theta = \Delta_{d-1} := \{(\alpha_1, \ldots, \alpha_d) \in \mathbb{R}^d : \alpha_i \geq 0, \sum_{i=1}^d \alpha_i = 1\}$ the $(d-1)-$dimensional probability simplex,
  P_θ - Random Scan Gibbs Sampler with coordinate selection probabilities

$$\theta = (\alpha_1, \ldots, \alpha_n)$$

- In each case values of θ will affect efficiency of P_θ

parametric family of transition kernels P_θ

- typically we can design a family of ergodic transition kernels P_θ, $\theta \in \Theta$.
- Ex 1a. $\Theta = R_+$
  P_θ - Random Walk Metropolis with proposal increments
  $$q_\theta = \theta N(0, Id)$$
- Ex 1b. $\Theta = R_+ \times \{d \text{ dimensional covariance matrices}\}$
  P_θ - Random Walk Metropolis with proposal increments
  $$q_\theta = \sigma N(0, \Sigma)$$
- Ex 2. $\Theta = \Delta_{d-1} := \{(\alpha_1, \ldots, \alpha_d) \in \mathbb{R}^d : \alpha_i \geq 0, \ \sum_{i=1}^d \alpha_i = 1\}$ the $(d-1)-$dimensional probability simplex,
  P_θ - Random Scan Gibbs Sampler with coordinate selection probabilities
  $$\theta = (\alpha_1, \ldots, \alpha_n)$$
- In each case values of θ will affect efficiency of P_θ

parametric family of transition kernels P_θ

- typically we can design a family of ergodic transition kernels P_θ, $\theta \in \Theta$.
- Ex 1a. $\Theta = R_+$
  P_θ - Random Walk Metropolis with proposal increments
  $$q_\theta = \theta N(0, Id)$$
- Ex 1b. $\Theta = R_+ \times \{d \text{ dimensional covariance matrices}\}$
  P_θ - Random Walk Metropolis with proposal increments
  $$q_\theta = \sigma N(0, \Sigma)$$
- Ex 2. $\Theta = \Delta_{d-1} := \{(\alpha_1, \ldots, \alpha_d) \in \mathbb{R}^d : \alpha_i \geq 0, \ \sum_{i=1}^d \alpha_i = 1\}$ the $(d-1)-$dimensional probability simplex,
  P_θ - Random Scan Gibbs Sampler with coordinate selection probabilities
  $$\theta = (\alpha_1, \ldots, \alpha_n)$$
- In each case values of θ will affect efficiency of P_θ

What Adaptive MCMC is designed for?

- In a typical Adaptive MCMC setting the parameter space Θ is large
- there is an optimal $\theta_* \in \Theta$ s.t. P_{θ_*} converges quickly.
- there are arbitrary bad values in Θ, say if $\theta \in \bar{\Theta} - \Theta$ then P_θ is not ergodic.
- if $\theta \in \Theta_* :=$ a region close to θ_*, then P_θ shall inherit good convergence properties of P_{θ_*}.
- When using adaptive MCMC we hope θ_n will eventually find the region Θ_* and stay there essentially forever. And that the adaptive algorithm $\mathcal{A}$ will inherit the good convergence properties of Θ_* in the limit.
-
- We are looking for a Theorem:
  You can actually run your Adaptive MCMC algorithm $\mathcal{A}$, and it will do what it is supposed to do! (under verifiable conditions)

What Adaptive MCMC is designed for?

- In a typical Adaptive MCMC setting the parameter space Θ is large
- there is an optimal $\theta_* \in \Theta$ s.t. P_{θ_*} converges quickly.
- there are arbitrary bad values in Θ, say if $\theta \in \bar{\Theta} - \Theta$ then P_θ is not ergodic.
- if $\theta \in \Theta_* :=$ a region close to θ_*, then P_θ shall inherit good convergence properties of P_{θ_*}.
- When using adaptive MCMC we hope θ_n will eventually find the region Θ_* and stay there essentially forever. And that the adaptive algorithm $\mathcal{A}$ will inherit the good convergence properties of Θ_* in the limit.
-
- We are looking for a Theorem:
You can actually run your Adaptive MCMC algorithm $\mathcal{A}$, and it will do what it is supposed to do! (under verifiable conditions)

What Adaptive MCMC is designed for?

- In a typical Adaptive MCMC setting the parameter space Θ is large
- there is an optimal $\theta_* \in \Theta$ s.t. P_{θ_*} converges quickly.
- there are arbitrary bad values in Θ, say if $\theta \in \bar{\Theta} - \Theta$ then P_θ is not ergodic.
- if $\theta \in \Theta_* :=$ a region close to θ_*, then P_θ shall inherit good convergence properties of P_{θ_*}.
- When using adaptive MCMC we hope θ_n will eventually find the region Θ_* and stay there essentially forever. And that the adaptive algorithm $\mathcal{A}$ will inherit the good convergence properties of Θ_* in the limit.
-
- We are looking for a Theorem:
You can actually run your Adaptive MCMC algorithm $\mathcal{A}$, and it will do what it is supposed to do! (under verifiable conditions)

What Adaptive MCMC is designed for?

- In a typical Adaptive MCMC setting the parameter space Θ is large
- there is an optimal $\theta_* \in \Theta$ s.t. P_{θ_*} converges quickly.
- there are arbitrary bad values in Θ, say if $\theta \in \bar{\Theta} - \Theta$ then P_θ is not ergodic.
- if $\theta \in \Theta_* :=$ a region close to θ_*, then P_θ shall inherit good convergence properties of P_{θ_*}.
- When using adaptive MCMC we hope θ_n will eventually find the region Θ_* and stay there essentially forever. And that the adaptive algorithm $\mathcal{A}$ will inherit the good convergence properties of Θ_* in the limit.
-
- We are looking for a Theorem:
You can actually run your Adaptive MCMC algorithm $\mathcal{A}$, and it will do what it is supposed to do! (under verifiable conditions)

What Adaptive MCMC is designed for?

- In a typical Adaptive MCMC setting the parameter space Θ is large
- there is an optimal $\theta_* \in \Theta$ s.t. P_{θ_*} converges quickly.
- there are arbitrary bad values in Θ, say if $\theta \in \bar{\Theta} - \Theta$ then P_θ is not ergodic.
- if $\theta \in \Theta_* :=$ a region close to θ_*, then P_θ shall inherit good convergence properties of P_{θ_*}.
- When using adaptive MCMC we hope θ_n will eventually find the region Θ_* and stay there essentially forever. And that the adaptive algorithm $\mathcal{A}$ will inherit the good convergence properties of Θ_* in the limit.
-
- We are looking for a Theorem:
  You can actually run your Adaptive MCMC algorithm $\mathcal{A}$, and it will do what it is supposed to do! (under verifiable conditions)

What Adaptive MCMC is designed for?

- In a typical Adaptive MCMC setting the parameter space Θ is large
- there is an optimal $\theta_* \in \Theta$ s.t. P_{θ_*} converges quickly.
- there are arbitrary bad values in Θ, say if $\theta \in \bar{\Theta} - \Theta$ then P_θ is not ergodic.
- if $\theta \in \Theta_* :=$ a region close to θ_*, then P_θ shall inherit good convergence properties of P_{θ_*}.
- When using adaptive MCMC we hope θ_n will eventually find the region Θ_* and stay there essentially forever. And that the adaptive algorithm $\mathcal{A}$ will inherit the good convergence properties of Θ_* in the limit.
-
- We are looking for a Theorem:
  You can actually run your Adaptive MCMC algorithm $\mathcal{A}$, and it will do what it is supposed to do! (under verifiable conditions)

a fundamental problem

- adaptive MCMC algorithms learn about π on the fly and use this information during the simulation
- the transition kernel P_n used for obtaining $X_n|X_{n-1}$ is allowed to depend on $\{X_0, \dots, X_{n-1}\}$
- consequently the algorithms are **not Markovian!**
- standard MCMC theory of validating the simulation does not apply

a fundamental problem

- adaptive MCMC algorithms learn about π on the fly and use this information during the simulation
- the transition kernel P_n used for obtaining $X_n|X_{n-1}$ is allowed to depend on $\{X_0, \ldots, X_{n-1}\}$
- consequently the algorithms are **not Markovian!**
- standard MCMC theory of validating the simulation does not apply

a fundamental problem

- adaptive MCMC algorithms learn about π on the fly and use this information during the simulation
- the transition kernel P_n used for obtaining $X_n|X_{n-1}$ is allowed to depend on $\{X_0, \ldots, X_{n-1}\}$
- consequently the algorithms are **not Markovian!**
- standard MCMC theory of validating the simulation does not apply

a fundamental problem

- adaptive MCMC algorithms learn about π on the fly and use this information during the simulation
- the transition kernel P_n used for obtaining $X_n|X_{n-1}$ is allowed to depend on $\{X_0, \ldots, X_{n-1}\}$
- consequently the algorithms are **not Markovian!**
- standard MCMC theory of validating the simulation does not apply

ergodicity: a toy counterexample

- Let $\mathcal{X} = \{0, 1\}$ and π be uniform.
- $$P_1 = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix} \quad \text{and} \quad P_2 = (1-\varepsilon)\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \varepsilon P_1 \quad \text{for some} \quad \varepsilon > 0.$$
- π is the stationary distribution for both, P_1 and P_2.
- Consider X_n , evolving for $n \geq 1$ according to the following adaptive kernel:

$$Q_n = \begin{cases} P_1 & \text{if} \quad X_{n-1} = 0 \\ P_2 & \text{if} \quad X_{n-1} = 1 \end{cases}$$

- Note that after two consecutive 1 the adaptive process X_n is trapped in 1 and can escape only with probability ε.
- Let $\bar{q}_1 := \lim_{n\to\infty} P(X_n = 1)$ and $\bar{q}_0 := \lim_{n\to\infty} P(X_n = 0)$.
- Now it is clear, that for small ε we will have $\bar{q}_1 \gg \bar{q}_0$ and the procedure fails to give the expected asymptotic distribution.

Adaptive Gibbs sampler - a generic algorithm

AdapRSG

1. Set $\alpha_n := R_n(\alpha_{n-1}, X_{n-1}, \dots, X_0) \in \mathcal{Y} \subset [0,1]^d$
2. Choose coordinate $i \in \{1, \dots, d\}$ according to selection probabilities α_n
3. Draw $Y \sim \pi(\cdot | X_{n-1,-i})$
4. Set $X_n := (X_{n-1,1}, \dots, X_{n-1,i-1}, Y, X_{n-1,i+1}, \dots, X_{n-1,d})$

- It is easy to get tricked into thinking that if step 1 is not doing anything "crazy" then the algorithm must be ergodic.
- Theorem 2.1 of [LC06] states that ergodicity of adaptive Gibbs samplers follows from the following two conditions:
  - (i) $\alpha_n \to \alpha$ a.s. for some fixed $\alpha \in (0,1)^d$; and
  - (ii) The random scan Gibbs sampler with fixed selection probabilities α induces an ergodic Markov chain with stationary distribution π.
- The above theorem is simple, neat and wrong.

Adaptive Gibbs sampler - a generic algorithm

AdapRSG

1. Set $\alpha_n := R_n(\alpha_{n-1}, X_{n-1}, \dots, X_0) \in \mathcal{Y} \subset [0,1]^d$
2. Choose coordinate $i \in \{1, \dots, d\}$ according to selection probabilities α_n
3. Draw $Y \sim \pi(\cdot | X_{n-1,-i})$
4. Set $X_n := (X_{n-1,1}, \dots, X_{n-1,i-1}, Y, X_{n-1,i+1}, \dots, X_{n-1,d})$

- It is easy to get tricked into thinking that if step 1 is not doing anything "crazy" then the algorithm must be ergodic.
- Theorem 2.1 of [LC06] states that ergodicity of adaptive Gibbs samplers follows from the following two conditions:
  - (i) $\alpha_n \to \alpha$ a.s. for some fixed $\alpha \in (0,1)^d$; and
  - (ii) The random scan Gibbs sampler with fixed selection probabilities α induces an ergodic Markov chain with stationary distribution π.
- The above theorem is simple, neat and wrong.

Adaptive Gibbs sampler - a generic algorithm

AdapRSG

1. Set $\alpha_n := R_n(\alpha_{n-1}, X_{n-1}, \dots, X_0) \in \mathcal{Y} \subset [0,1]^d$
2. Choose coordinate $i \in \{1, \dots, d\}$ according to selection probabilities α_n
3. Draw $Y \sim \pi(\cdot | X_{n-1,-i})$
4. Set $X_n := (X_{n-1,1}, \dots, X_{n-1,i-1}, Y, X_{n-1,i+1}, \dots, X_{n-1,d})$

- It is easy to get tricked into thinking that if step 1 is not doing anything "crazy" then the algorithm must be ergodic.
- Theorem 2.1 of [LC06] states that ergodicity of adaptive Gibbs samplers follows from the following two conditions:
  - (i) $\alpha_n \to \alpha$ a.s. for some fixed $\alpha \in (0,1)^d$; and
  - (ii) The random scan Gibbs sampler with fixed selection probabilities α induces an ergodic Markov chain with stationary distribution π.
- The above theorem is simple, neat and wrong.

Adaptive Gibbs sampler - a generic algorithm

AdapRSG

1. Set $\alpha_n := R_n(\alpha_{n-1}, X_{n-1}, \dots, X_0) \in \mathcal{Y} \subset [0,1]^d$
2. Choose coordinate $i \in \{1, \dots, d\}$ according to selection probabilities α_n
3. Draw $Y \sim \pi(\cdot | X_{n-1,-i})$
4. Set $X_n := (X_{n-1,1}, \dots, X_{n-1,i-1}, Y, X_{n-1,i+1}, \dots, X_{n-1,d})$

- It is easy to get tricked into thinking that if step 1 is not doing anything "crazy" then the algorithm must be ergodic.
- Theorem 2.1 of [LC06] states that ergodicity of adaptive Gibbs samplers follows from the following two conditions:
  - (i) $\alpha_n \to \alpha$ a.s. for some fixed $\alpha \in (0,1)^d$; and
  - (ii) The random scan Gibbs sampler with fixed selection probabilities α induces an ergodic Markov chain with stationary distribution π.
- The above theorem is simple, neat and wrong.

a cautionary example that disproves [LC06]

- Let $\mathcal{X} = \{(i,j) \in \mathbb{N} \times \mathbb{N} : i = j \text{ or } i = j+1\}$,
- with target distribution given by $\pi(i,j) \propto j^{-2}$
- consider a class of adaptive random scan Gibbs samplers with update rule given by:

$$R_n\Big(\alpha_{n-1}, X_{n-1} = (i,j)\Big) = \begin{cases} \left\{\frac{1}{2} + \frac{4}{a_n}, \frac{1}{2} - \frac{4}{a_n}\right\} & \text{if} \quad i = j, \\ \left\{\frac{1}{2} - \frac{4}{a_n}, \frac{1}{2} + \frac{4}{a_n}\right\} & \text{if} \quad i = j+1, \end{cases}$$

  for some choice of the sequence $(a_n)_{n=0}^{\infty}$ satisfying $8 < a_n \nearrow \infty$
- if $a_n \to \infty$ slowly enough, then X_n is **transient** with positive probability, i.e. $\mathbb{P}(X_{1,n} \to \infty) > 0$.

a cautionary example that disproves [LC06]

- Let $\mathcal{X} = \{(i,j) \in \mathbb{N} \times \mathbb{N} : i = j \text{ or } i = j+1\}$,
- with target distribution given by $\pi(i,j) \propto j^{-2}$
- consider a class of adaptive random scan Gibbs samplers with update rule given by:

$$R_n\Big(\alpha_{n-1}, X_{n-1} = (i,j)\Big) = \begin{cases} \left\{\frac{1}{2} + \frac{4}{a_n}, \frac{1}{2} - \frac{4}{a_n}\right\} & \text{if} \quad i = j, \\ \left\{\frac{1}{2} - \frac{4}{a_n}, \frac{1}{2} + \frac{4}{a_n}\right\} & \text{if} \quad i = j+1, \end{cases}$$

  for some choice of the sequence $(a_n)_{n=0}^{\infty}$ satisfying $8 < a_n \nearrow \infty$
- if $a_n \to \infty$ slowly enough, then X_n is **transient** with positive probability, i.e. $\mathbb{P}(X_{1,n} \to \infty) > 0$.

a cautionary example that disproves [LC06]

- Let $\mathcal{X} = \{(i,j) \in \mathbb{N} \times \mathbb{N} : i = j \text{ or } i = j+1\}$,
- with target distribution given by $\pi(i,j) \propto j^{-2}$
- consider a class of adaptive random scan Gibbs samplers with update rule given by:

$$R_n\Big(\alpha_{n-1}, X_{n-1} = (i,j)\Big) = \begin{cases} \left\{\frac{1}{2} + \frac{4}{a_n}, \frac{1}{2} - \frac{4}{a_n}\right\} & \text{if} \quad i = j, \\ \left\{\frac{1}{2} - \frac{4}{a_n}, \frac{1}{2} + \frac{4}{a_n}\right\} & \text{if} \quad i = j+1, \end{cases}$$

for some choice of the sequence $(a_n)_{n=0}^{\infty}$ satisfying $8 < a_n \nearrow \infty$

- if $a_n \to \infty$ slowly enough, then X_n is **transient** with positive probability, i.e. $\mathbb{P}(X_{1,n} \to \infty) > 0$.

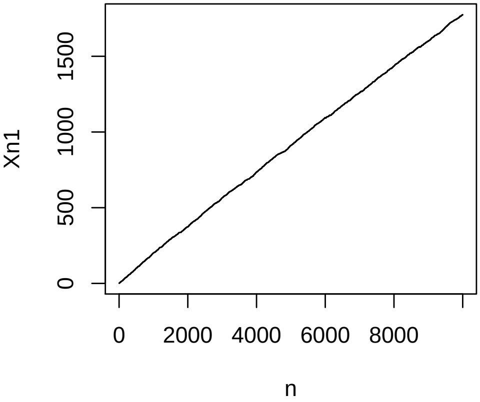
a cautionary example that disproves [LC06]

- Let $\mathcal{X} = \{(i,j) \in \mathbb{N} \times \mathbb{N} : i = j \text{ or } i = j+1\}$,
- with target distribution given by $\pi(i,j) \propto j^{-2}$
- consider a class of adaptive random scan Gibbs samplers with update rule given by:

$$R_n\Big(\alpha_{n-1}, X_{n-1} = (i,j)\Big) = \begin{cases} \left\{\frac{1}{2} + \frac{4}{a_n}, \frac{1}{2} - \frac{4}{a_n}\right\} & \text{if} \quad i = j, \\ \\ \left\{\frac{1}{2} - \frac{4}{a_n}, \frac{1}{2} + \frac{4}{a_n}\right\} & \text{if} \quad i = j+1, \end{cases}$$

  for some choice of the sequence $(a_n)_{n=0}^{\infty}$ satisfying $8 < a_n \nearrow \infty$
- if $a_n \to \infty$ slowly enough, then X_n is **transient** with positive probability, i.e. $\mathbb{P}(X_{1,n} \to \infty) > 0$.

a cautionary example...



Ergodicity of an adaptive algorithm - framework

- $\mathcal{X}$ valued process of interest X_n
- Θ valued random parameter θ_n representing the choice of kernel when updating X_n to X_{n+1}
- Define the filtration generated by $\{(X_n, \theta_n)\}$

$$\mathcal{G}_n = \sigma(X_0, \ldots, X_n, \theta_0, \ldots, \theta_n),$$

- Thus

$$P(X_{n+1} \in B \mid X_n = x, \theta_n = \theta, \mathcal{G}_{n-1}) = P_\theta(x, B)$$

- The distribution of θ_{n+1} given $\mathcal{G}_n$ depends on the algorithm.
- Define

$$A^{(n)}(x, \theta, B) = P(X_n \in B \parallel X_0 = x, \theta_0 = \theta)$$

$$T(x, \theta, n) = \|A^{(n)}(x, \theta, \cdot) - \pi(\cdot)\|_{TV}$$

- We say the adaptive algorithm is ergodic if

$$\lim_{n\to\infty} T(x, \theta, n) = 0 \qquad \text{for all } x \in \mathcal{X} \quad \text{and } \theta \in \Theta.$$

Ergodicity of an adaptive algorithm - framework

- $\mathcal{X}$ valued process of interest X_n
- Θ valued random parameter θ_n
  representing the choice of kernel when updating X_n to X_{n+1}
- Define the filtration generated by $\{(X_n, \theta_n)\}$

$$\mathcal{G}_n = \sigma(X_0, \ldots, X_n, \theta_0, \ldots, \theta_n),$$

- Thus

$$P(X_{n+1} \in B \mid X_n = x, \theta_n = \theta, \mathcal{G}_{n-1}) = P_\theta(x, B)$$

- The distribution of θ_{n+1} given $\mathcal{G}_n$ depends on the algorithm.
- Define

$$A^{(n)}(x, \theta, B) = P(X_n \in B \,\|\, X_0 = x, \theta_0 = \theta)$$
$$T(x, \theta, n) = \|A^{(n)}(x, \theta, \cdot) - \pi(\cdot)\|_{TV}$$

- We say the adaptive algorithm is ergodic if

$$\lim_{n\to\infty} T(x, \theta, n) = 0 \qquad \text{for all } x \in \mathcal{X} \quad \text{and } \theta \in \Theta.$$

Ergodicity of an adaptive algorithm - framework

- $\mathcal{X}$ valued process of interest X_n
- Θ valued random parameter θ_n
  representing the choice of kernel when updating X_n to X_{n+1}
- Define the filtration generated by $\{(X_n, \theta_n)\}$

$$\mathcal{G}_n = \sigma(X_0, \ldots, X_n, \theta_0, \ldots, \theta_n),$$

- Thus

$$P(X_{n+1} \in B \mid X_n = x, \theta_n = \theta, \mathcal{G}_{n-1}) = P_\theta(x, B)$$

- The distribution of θ_{n+1} given $\mathcal{G}_n$ depends on the algorithm.
- Define

$$A^{(n)}(x, \theta, B) = P(X_n \in B \parallel X_0 = x, \theta_0 = \theta)$$
$$T(x, \theta, n) = \|A^{(n)}(x, \theta, \cdot) - \pi(\cdot)\|_{TV}$$

- We say the adaptive algorithm is ergodic if

$$\lim_{n\to\infty} T(x, \theta, n) = 0 \qquad \text{for all } x \in \mathcal{X} \quad \text{and } \theta \in \Theta.$$

Ergodicity of an adaptive algorithm - framework

- $\mathcal{X}$ valued process of interest X_n
- Θ valued random parameter θ_n
  representing the choice of kernel when updating X_n to X_{n+1}
- Define the filtration generated by $\{(X_n, \theta_n)\}$

$$\mathcal{G}_n = \sigma(X_0, \ldots, X_n, \theta_0, \ldots, \theta_n),$$

- Thus

$$P(X_{n+1} \in B \mid X_n = x, \theta_n = \theta, \mathcal{G}_{n-1}) = P_\theta(x, B)$$

- The distribution of θ_{n+1} given $\mathcal{G}_n$ depends on the algorithm.
- Define

$$A^{(n)}(x, \theta, B) = P(X_n \in B \mid\mid X_0 = x, \theta_0 = \theta)$$
$$T(x, \theta, n) = \|A^{(n)}(x, \theta, \cdot) - \pi(\cdot)\|_{TV}$$

- We say the adaptive algorithm is ergodic if

$$\lim_{n\to\infty} T(x, \theta, n) = 0 \qquad \text{for all } x \in \mathcal{X} \quad \text{and } \theta \in \Theta.$$

Ergodicity of an adaptive algorithm - framework

- $\mathcal{X}$ valued process of interest X_n
- Θ valued random parameter θ_n
  representing the choice of kernel when updating X_n to X_{n+1}
- Define the filtration generated by $\{(X_n, \theta_n)\}$

$$\mathcal{G}_n = \sigma(X_0, \ldots, X_n, \theta_0, \ldots, \theta_n),$$

- Thus

$$P(X_{n+1} \in B \mid X_n = x, \theta_n = \theta, \mathcal{G}_{n-1}) = P_\theta(x, B)$$

- The distribution of θ_{n+1} given $\mathcal{G}_n$ depends on the algorithm.
- Define

$$A^{(n)}(x, \theta, B) = P(X_n \in B \parallel X_0 = x, \theta_0 = \theta)$$
$$T(x, \theta, n) = \|A^{(n)}(x, \theta, \cdot) - \pi(\cdot)\|_{TV}$$

- We say the adaptive algorithm is ergodic if

$$\lim_{n\to\infty} T(x, \theta, n) = 0 \qquad \text{for all } x \in \mathcal{X} \quad \text{and } \theta \in \Theta.$$

Ergodicity of an adaptive algorithm - framework

- $\mathcal{X}$ valued process of interest X_n
- Θ valued random parameter θ_n
  representing the choice of kernel when updating X_n to X_{n+1}
- Define the filtration generated by $\{(X_n, \theta_n)\}$

$$\mathcal{G}_n = \sigma(X_0, \ldots, X_n, \theta_0, \ldots, \theta_n),$$

- Thus

$$P(X_{n+1} \in B \mid X_n = x, \theta_n = \theta, \mathcal{G}_{n-1}) = P_\theta(x, B)$$

- The distribution of θ_{n+1} given $\mathcal{G}_n$ depends on the algorithm.
- Define

$$A^{(n)}(x, \theta, B) = P(X_n \in B \,\|\, X_0 = x, \theta_0 = \theta)$$
$$T(x, \theta, n) = \|A^{(n)}(x, \theta, \cdot) - \pi(\cdot)\|_{TV}$$

- We say the adaptive algorithm is ergodic if

$$\lim_{n\to\infty} T(x, \theta, n) = 0 \qquad \text{for all } x \in \mathcal{X} \quad \text{and } \theta \in \Theta.$$

Ergodicity of an adaptive algorithm - framework

- $\mathcal{X}$ valued process of interest X_n
- Θ valued random parameter θ_n
  representing the choice of kernel when updating X_n to X_{n+1}
- Define the filtration generated by $\{(X_n, \theta_n)\}$

$$\mathcal{G}_n = \sigma(X_0, \ldots, X_n, \theta_0, \ldots, \theta_n),$$

- Thus

$$P(X_{n+1} \in B \mid X_n = x, \theta_n = \theta, \mathcal{G}_{n-1}) = P_\theta(x, B)$$

- The distribution of θ_{n+1} given $\mathcal{G}_n$ depends on the algorithm.
- Define

$$A^{(n)}(x, \theta, B) = P(X_n \in B \parallel X_0 = x, \theta_0 = \theta)$$
$$T(x, \theta, n) = \|A^{(n)}(x, \theta, \cdot) - \pi(\cdot)\|_{TV}$$

- We say the adaptive algorithm is ergodic if

$$\lim_{n\to\infty} T(x, \theta, n) = 0 \qquad \text{for all } x \in \mathcal{X} \quad \text{and } \theta \in \Theta.$$

Tools for establishing ergodicity

- **(Diminishing Adaptation)** Let $D_n = \sup_{x \in \mathcal{X}} \|P_{\Gamma_{n+1}}(x, \cdot) - P_{\Gamma_n}(x, \cdot)\|$ and assume $\lim_{n \to \infty} D_n = 0$ in probability
- **(Simultaneous uniform ergodicity)** For all $\varepsilon > 0$, there exists $N = N(\varepsilon)$ s.t. $\|P_\gamma^N(x, \cdot) - \pi(\cdot)\| \leq \varepsilon$ for all $x \in \mathcal{X}$ and $\gamma \in \mathcal{Y}$
- **(Containment condition)** Let $M_\varepsilon(x, \gamma) = \inf\{n \geq 1 : \|P_\gamma^n(x, \cdot) - \pi(\cdot)\| \leq \varepsilon\}$ and assume $\{M_\varepsilon(X_n, \gamma_n)\}_{n=0}^\infty$ is bounded in probability, i.e. given $X_0 = x_*$ and $\Gamma_0 = \gamma_*$, for all $\delta > 0$, there exists N s.t. $\mathbb{P}[M_\varepsilon(X_n, \Gamma_n) \leq N | X_0 = x_*, \Gamma_0 = \gamma_*] \geq 1 - \delta$ for all $n \in \mathbb{N}$.

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) $+$ (simultaneous uniform ergodicity) $\Rightarrow$ ergodicity.

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) $+$ (containment) $\Rightarrow$ ergodicity.

Tools for establishing ergodicity

- **(Diminishing Adaptation)** Let $D_n = \sup_{x \in \mathcal{X}} \|P_{\Gamma_{n+1}}(x, \cdot) - P_{\Gamma_n}(x, \cdot)\|$ and assume $\lim_{n\to\infty} D_n = 0$ in probability
- **(Simultaneous uniform ergodicity)** For all $\varepsilon > 0$, there exists $N = N(\varepsilon)$ s.t. $\|P_\gamma^N(x, \cdot) - \pi(\cdot)\| \leq \varepsilon$ for all $x \in \mathcal{X}$ and $\gamma \in \mathcal{Y}$
- **(Containment condition)** Let $M_\varepsilon(x, \gamma) = \inf\{n \geq 1 : \|P_\gamma^n(x, \cdot) - \pi(\cdot)\| \leq \varepsilon\}$ and assume $\{M_\varepsilon(X_n, \gamma_n)\}_{n=0}^\infty$ is bounded in probability, i.e. given $X_0 = x_*$ and $\Gamma_0 = \gamma_*$, for all $\delta > 0$, there exists N s.t. $\mathbb{P}[M_\varepsilon(X_n, \Gamma_n) \leq N | X_0 = x_*, \Gamma_0 = \gamma_*] \geq 1 - \delta$ for all $n \in \mathbb{N}$.

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) $+$ *(simultaneous uniform ergodicity)* $\Rightarrow$ *ergodicity.*

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) $+$ *(containment)* $\Rightarrow$ *ergodicity.*

Tools for establishing ergodicity

- **(Diminishing Adaptation)** Let $D_n = \sup_{x\in\mathcal{X}} \|P_{\Gamma_{n+1}}(x,\cdot) - P_{\Gamma_n}(x,\cdot)\|$ and assume $\lim_{n\to\infty} D_n = 0$ in probability
- **(Simultaneous uniform ergodicity)** For all $\varepsilon > 0$, there exists $N = N(\varepsilon)$ s.t. $\|P_\gamma^N(x,\cdot) - \pi(\cdot)\| \leq \varepsilon$ for all $x \in \mathcal{X}$ and $\gamma \in \mathcal{Y}$
- **(Containment condition)** Let $M_\varepsilon(x,\gamma) = \inf\{n \geq 1 : \|P_\gamma^n(x,\cdot) - \pi(\cdot)\| \leq \varepsilon\}$ and assume $\{M_\varepsilon(X_n,\gamma_n)\}_{n=0}^\infty$ is bounded in probability, i.e. given $X_0 = x_*$ and $\Gamma_0 = \gamma_*$, for all $\delta > 0$, there exists N s.t. $\mathbb{P}[M_\varepsilon(X_n,\Gamma_n) \leq N | X_0 = x_*, \Gamma_0 = \gamma_*] \geq 1-\delta$ for all $n \in \mathbb{N}$.

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) + (simultaneous uniform ergodicity) $\Rightarrow$ ergodicity.

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) + (containment) $\Rightarrow$ ergodicity.

Tools for establishing ergodicity

- **(Diminishing Adaptation)** Let $D_n = \sup_{x\in\mathcal{X}} \|P_{\Gamma_{n+1}}(x,\cdot) - P_{\Gamma_n}(x,\cdot)\|$ and assume $\lim_{n\to\infty} D_n = 0$ in probability
- **(Simultaneous uniform ergodicity)** For all $\varepsilon > 0$, there exists $N = N(\varepsilon)$ s.t. $\|P_\gamma^N(x,\cdot) - \pi(\cdot)\| \leq \varepsilon$ for all $x \in \mathcal{X}$ and $\gamma \in \mathcal{Y}$
- **(Containment condition)** Let $M_\varepsilon(x,\gamma) = \inf\{n \geq 1 : \|P_\gamma^n(x,\cdot) - \pi(\cdot)\| \leq \varepsilon\}$ and assume $\{M_\varepsilon(X_n,\gamma_n)\}_{n=0}^\infty$ is bounded in probability, i.e. given $X_0 = x_*$ and $\Gamma_0 = \gamma_*$, for all $\delta > 0$, there exists N s.t. $\mathbb{P}[M_\varepsilon(X_n,\Gamma_n) \leq N | X_0 = x_*, \Gamma_0 = \gamma_*] \geq 1 - \delta$ for all $n \in \mathbb{N}$.

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) $+$ *(simultaneous uniform ergodicity)* $\Rightarrow$ *ergodicity.*

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) $+$ *(containment)* $\Rightarrow$ *ergodicity.*

Tools for establishing ergodicity

- **(Diminishing Adaptation)** Let $D_n = \sup_{x\in\mathcal{X}} \|P_{\Gamma_{n+1}}(x,\cdot) - P_{\Gamma_n}(x,\cdot)\|$ and assume $\lim_{n\to\infty} D_n = 0$ in probability
- **(Simultaneous uniform ergodicity)** For all $\varepsilon > 0$, there exists $N = N(\varepsilon)$ s.t. $\|P_\gamma^N(x,\cdot) - \pi(\cdot)\| \leq \varepsilon$ for all $x \in \mathcal{X}$ and $\gamma \in \mathcal{Y}$
- **(Containment condition)** Let $M_\varepsilon(x,\gamma) = \inf\{n \geq 1 : \|P_\gamma^n(x,\cdot) - \pi(\cdot)\| \leq \varepsilon\}$ and assume $\{M_\varepsilon(X_n,\gamma_n)\}_{n=0}^\infty$ is bounded in probability,
i.e. given $X_0 = x_*$ and $\Gamma_0 = \gamma_*$, for all $\delta > 0$,
there exists N s.t. $\mathbb{P}[M_\varepsilon(X_n,\Gamma_n) \leq N | X_0 = x_*, \Gamma_0 = \gamma_*] \geq 1 - \delta$ for all $n \in \mathbb{N}$.

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) $+$ *(simultaneous uniform ergodicity)* $\Rightarrow$ *ergodicity.*

Theorem (Roberts Rosenthal 2007)

(diminishing adaptation) $+$ *(containment)* $\Rightarrow$ *ergodicity.*

Containment: a closer look

- **(Containment condition)** $M_\varepsilon(x,\gamma) = \inf\{n \geq 1 : \|P_\gamma^n(x,\cdot) - \pi(\cdot)\| \leq \varepsilon\}$ given $X_0 = x_*$ and $\Gamma_0 = \gamma_*$, for all $\delta > 0$, there exists N s.t. $\mathbb{P}[M_\varepsilon(X_n,\Gamma_n) \leq N | X_0 = x_*, \Gamma_0 = \gamma_*] \geq 1-\delta$ for all $n \in \mathbb{N}$.
- Containment can be verified using simultaneous geometrical ergodicity or simultaneous polynomial ergodicity. (details in [BRR10])
- The family $\{P_\gamma : \gamma \in \mathcal{Y}\}$ is Simultaneously Geometrically Ergodic if
  - there exist a uniform ν_m-small set C i.e. for each γ $P_\gamma^m(x,\cdot) \geq \delta\nu_\gamma(\cdot)$ for all $x \in C$.
  - $P_\gamma V \leq \lambda V + b\mathbb{I}_C$ for all γ.
- S.G.E. implies containment

Containment: a closer look

- **(Containment condition)** $M_\varepsilon(x,\gamma) = \inf\{n \geq 1 : \|P_\gamma^n(x,\cdot) - \pi(\cdot)\| \leq \varepsilon\}$ given $X_0 = x_*$ and $\Gamma_0 = \gamma_*$, for all $\delta > 0$, there exists N s.t. $\mathbb{P}[M_\varepsilon(X_n,\Gamma_n) \leq N | X_0 = x_*, \Gamma_0 = \gamma_*] \geq 1-\delta$ for all $n \in \mathbb{N}$.
- Containment can be verified using simultaneous geometrical ergodicity or simultaneous polynomial ergodicity. (details in [BRR10])
- The family $\{P_\gamma : \gamma \in \mathcal{Y}\}$ is Simultaneously Geometrically Ergodic if
  - there exist a uniform ν_m-small set C i.e. for each γ $P_\gamma^m(x,\cdot) \geq \delta\nu_\gamma(\cdot)$ for all $x \in C$.
  - $P_\gamma V \leq \lambda V + b\mathbb{I}_C$ for all γ.
- S.G.E. implies containment

Containment: a closer look

- **(Containment condition)** $M_\varepsilon(x,\gamma) = \inf\{n \geq 1 : \|P^n_\gamma(x,\cdot) - \pi(\cdot)\| \leq \varepsilon\}$ given $X_0 = x_*$ and $\Gamma_0 = \gamma_*$, for all $\delta > 0$, there exists N s.t. $\mathbb{P}[M_\varepsilon(X_n,\Gamma_n) \leq N | X_0 = x_*, \Gamma_0 = \gamma_*] \geq 1 - \delta$ for all $n \in \mathbb{N}$.
- Containment can be verified using simultaneous geometrical ergodicity or simultaneous polynomial ergodicity. (details in [BRR10])
- The family $\{P_\gamma : \gamma \in \mathcal{Y}\}$ is Simultaneously Geometrically Ergodic if
  - there exist a uniform ν_m-small set C i.e. for each γ $\quad P^m_\gamma(x,\cdot) \geq \delta\nu_\gamma(\cdot)$ $\quad$ for all $x \in C$.
  - $P_\gamma V \leq \lambda V + b\mathbb{I}_C$ $\quad$ for all γ.
- S.G.E. implies containment

Containment: a closer look

- **(Containment condition)** $M_\varepsilon(x,\gamma) = \inf\{n \geq 1 : \|P_\gamma^n(x,\cdot) - \pi(\cdot)\| \leq \varepsilon\}$
  given $X_0 = x_*$ and $\Gamma_0 = \gamma_*$, for all $\delta > 0$,
  there exists N s.t. $\mathbb{P}[M_\varepsilon(X_n,\Gamma_n) \leq N | X_0 = x_*, \Gamma_0 = \gamma_*] \geq 1 - \delta$ for all $n \in \mathbb{N}$.
- Containment can be verified using simultaneous geometrical ergodicity or simultaneous polynomial ergodicity. (details in [BRR10])
- The family $\{P_\gamma : \gamma \in \mathcal{Y}\}$ is Simultaneously Geometrically Ergodic if
  - there exist a uniform ν_m-small set C i.e.
    for each γ $\quad P_\gamma^m(x,\cdot) \geq \delta\nu_\gamma(\cdot)$ $\quad$ for all $x \in C$.
  - $P_\gamma V \leq \lambda V + b\mathbb{I}_C$ $\quad$ for all γ.
- S.G.E. implies containment

Adaptive random scan Metropolis within Gibbs

AdapRSMwG

1. Set $\alpha_n := R_n(\alpha_{n-1}, X_{n-1}, \dots, X_0) \in \mathcal{Y}$
2. Choose coordinate $i \in \{1, \dots, d\}$ according to selection probabilities α_n
3. Draw $Y \sim Q_{X_{n-1,-i}}(X_{n-1,i}, \cdot)$
4. With probability

$$\min\left(1, \ \frac{\pi(Y|X_{n-1,-i})\, q_{X_{n-1,-i}}(Y, X_{n-1,i})}{\pi(X_{n-1}|X_{n-1,-i})\, q_{X_{n-1,-i}}(X_{n-1,i}, Y)}\right), \tag{1}$$

accept the proposal and set

$$X_n = (X_{n-1,1}, \dots, X_{n-1,i-1}, Y, X_{n-1,i+1}, \dots, X_{n-1,d})\,;$$

otherwise, reject the proposal and set $X_n = X_{n-1}$.

Adaptive random scan adaptive Metropolis within Gibbs

`AdapRSadapMwG`

1. Set $\alpha_n := R_n(\alpha_{n-1}, X_{n-1}, \dots, X_0, \gamma_{n-1}, \dots, \gamma_0) \in \mathcal{Y}$
2. Set $\gamma_n := R'_n(\alpha_{n-1}, X_{n-1}, \dots, X_0, \gamma_{n-1}, \dots, \gamma_0) \in \Gamma_1 \times \dots \times \Gamma_n$
3. Choose coordinate $i \in \{1, \dots, d\}$ according to selection probabilities α, i.e. with $\Pr(i = j) = \alpha_j$
4. Draw $Y \sim Q_{X_{n-1,-i}, \gamma_{n-1}}(X_{n-1,i}, \cdot)$
5. With probability (1),

$$\min\left(1, \frac{\pi(Y|X_{n-1,-i})\, q_{X_{n-1,-i},\gamma_{n-1}}(Y, X_{n-1,i})}{\pi(X_{n-1}|X_{n-1,-i})\, q_{X_{n-1,-i},\gamma_{n-1}}(X_{n-1,i}, Y)}\right),$$

   accept the proposal and set

$$X_n = (X_{n-1,1}, \dots, X_{n-1,i-1}, Y, X_{n-1,i+1}, \dots, X_{n-1,d});$$

   otherwise, reject the proposal and set $X_n = X_{n-1}$.

Ergodicity Adaptive Random Scan Gibbs [ŁRR13]

- Assuming that `RSG(`β`)` is uniformly ergodic and $|\alpha_n - \alpha_{n-1}| \to 0$, we can prove ergodicity of
  - `AdapRSG`
  - `AdapRSMwG`
  - `AdapRSadapMwG`

  by establishing diminishing adaptation and simultaneous uniform ergodicity
- Assuming that $|\alpha_n - \alpha_{n-1}| \to 0$ and regularity conditions for the target and proposal distributions (in the spirit of Roberts Rosenthal 98, Fort et al 03) ergodicity of
  - `AdapRSMwG`
  - `AdapRSadapMwG`

  can be verified by establishing diminishing adaptation and containment (by simultaneous geometrical ergodicity, using results of Bai et al 2008)

Ergodicity Adaptive Random Scan Gibbs [ŁRR13]

- Assuming that `RSG(`β`)` is uniformly ergodic and $|\alpha_n - \alpha_{n-1}| \to 0$, we can prove ergodicity of
  - `AdapRSG`
  - `AdapRSMwG`
  - `AdapRSadapMwG`

  by establishing diminishing adaptation and simultaneous uniform ergodicity
- Assuming that $|\alpha_n - \alpha_{n-1}| \to 0$ and regularity conditions for the target and proposal distributions (in the spirit of Roberts Rosenthal 98, Fort et al 03) ergodicity of
  - `AdapRSMwG`
  - `AdapRSadapMwG`

  can be verified by establishing diminishing adaptation and containment (by simultaneous geometrical ergodicity, using results of Bai et al 2008)

Adaptive Metropolis - shape of the distribution

- Recall the Adaptive Scaling Metropolis Algorithm with proposals

$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, I_d),$$

- the proposal uses I_d for covariance and does not depend on the shape of the target...
- in a certain setting, if the covariance of the target is Σ and one uses $\tilde{\Sigma}$ for proposal increments, the suboptimality factor is computable [RR01]

$$b = d \frac{\sum_{i=1}^{d} \lambda_i^{-2}}{(\sum_{i=1}^{d} \lambda_i^{-1})^2},$$

where $\{\lambda_i\}$ are eigenvalues of $\tilde{\Sigma}^{1/2}\Sigma^{-1/2}$.
- the optimal proposal increment is

$$N(0, (2.38)^2 \Sigma / d).$$

- Again we have a very precise guidance. One should estimate Σ and use it for proposals.

Adaptive Metropolis - shape of the distribution

- Recall the Adaptive Scaling Metropolis Algorithm with proposals

$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, I_d),$$

- the proposal uses I_d for covariance and does not depend on the shape of the target...
- in a certain setting, if the covariance of the target is Σ and one uses $\tilde{\Sigma}$ for proposal increments, the suboptimality factor is computable [RR01]

$$b = d \frac{\sum_{i=1}^{d} \lambda_i^{-2}}{(\sum_{i=1}^{d} \lambda_i^{-1})^2},$$

where $\{\lambda_i\}$ are eigenvalues of $\tilde{\Sigma}^{1/2}\Sigma^{-1/2}$.
- the optimal proposal increment is

$$N(0, (2.38)^2\Sigma/d).$$

- Again we have a very precise guidance. One should estimate Σ and use it for proposals.

Adaptive Metropolis - shape of the distribution

- Recall the Adaptive Scaling Metropolis Algorithm with proposals

$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, I_d),$$

- the proposal uses I_d for covariance and does not depend on the shape of the target...
- in a certain setting, if the covariance of the target is Σ and one uses $\tilde{\Sigma}$ for proposal increments, the suboptimality factor is computable [RR01]

$$b = d\frac{\sum_{i=1}^{d} \lambda_i^{-2}}{(\sum_{i=1}^{d} \lambda_i^{-1})^2},$$

where $\{\lambda_i\}$ are eigenvalues of $\tilde{\Sigma}^{1/2}\Sigma^{-1/2}$.
- the optimal proposal increment is

$$N(0, (2.38)^2\Sigma/d).$$

- Again we have a very precise guidance. One should estimate Σ and use it for proposals.

Adaptive Metropolis - shape of the distribution

- Recall the Adaptive Scaling Metropolis Algorithm with proposals
$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, I_d),$$
- the proposal uses I_d for covariance and does not depend on the shape of the target...
- in a certain setting, if the covariance of the target is Σ and one uses $\tilde{\Sigma}$ for proposal increments, the suboptimality factor is computable [RR01]
$$b = d\frac{\sum_{i=1}^{d} \lambda_i^{-2}}{(\sum_{i=1}^{d} \lambda_i^{-1})^2},$$
where $\{\lambda_i\}$ are eigenvalues of $\tilde{\Sigma}^{1/2}\Sigma^{-1/2}$.
- the optimal proposal increment is
$$N(0, (2.38)^2\Sigma/d).$$
- Again we have a very precise guidance. One should estimate Σ and use it for proposals.

Adaptive Metropolis - shape of the distribution

- Recall the Adaptive Scaling Metropolis Algorithm with proposals

$$Y_{n+1} \sim q_{\sigma_n}(X_n, \cdot) = X_n + \sigma_n N(0, I_d),$$

- the proposal uses I_d for covariance and does not depend on the shape of the target...
- in a certain setting, if the covariance of the target is Σ and one uses $\tilde{\Sigma}$ for proposal increments, the suboptimality factor is computable [RR01]

$$b = d\frac{\sum_{i=1}^{d} \lambda_i^{-2}}{(\sum_{i=1}^{d} \lambda_i^{-1})^2},$$

where $\{\lambda_i\}$ are eigenvalues of $\tilde{\Sigma}^{1/2}\Sigma^{-1/2}$.
- the optimal proposal increment is

$$N(0, (2.38)^2\Sigma/d).$$

- Again we have a very precise guidance. One should estimate Σ and use it for proposals.

Adaptive Metropolis - versions and stability

- The theory suggests increment

$$N(0, (2.38)^2\Sigma_n/d)$$

- The AM version of [HST01] (the original one) uses

$$N(0, \Sigma_n + \varepsilon Id)$$

- Modification due to [RR09] is to use

$$Q_n = (1-\beta)N(0, (2.38)^2\Sigma_n/d) + \beta N(0, \varepsilon Id/d).$$

- the above modification appears more tractable: containment has been verified for both, exponentially and super-exponentially decaying tails (Bai et al 2009).
- the original version has been analyzed in [SV10] and [FMP10] using different techniques.

Adaptive Metropolis - versions and stability

- The theory suggests increment

$$N(0, (2.38)^2 \Sigma_n / d)$$

- The AM version of [HST01] (the original one) uses

$$N(0, \Sigma_n + \varepsilon Id)$$

- Modification due to [RR09] is to use

$$Q_n = (1-\beta)N(0, (2.38)^2 \Sigma_n / d) + \beta N(0, \varepsilon Id/d).$$

- the above modification appears more tractable: containment has been verified for both, exponentially and super-exponentially decaying tails (Bai et al 2009).
- the original version has been analyzed in [SV10] and [FMP10] using different techniques.

Adaptive Metropolis - versions and stability

- The theory suggests increment

$$N(0, (2.38)^2 \Sigma_n / d)$$

- The AM version of [HST01] (the original one) uses

$$N(0, \Sigma_n + \varepsilon Id)$$

- Modification due to [RR09] is to use

$$Q_n = (1 - \beta) N(0, (2.38)^2 \Sigma_n / d) + \beta N(0, \varepsilon Id / d).$$

- the above modification appears more tractable: containment has been verified for both, exponentially and super-exponentially decaying tails (Bai et al 2009).
- the original version has been analyzed in [SV10] and [FMP10] using different techniques.

Adaptive Metropolis - versions and stability

- The theory suggests increment

$$N(0, (2.38)^2\Sigma_n/d)$$

- The AM version of [HST01] (the original one) uses

$$N(0, \Sigma_n + \varepsilon Id)$$

- Modification due to [RR09] is to use

$$Q_n = (1-\beta)N(0, (2.38)^2\Sigma_n/d) + \beta N(0, \varepsilon Id/d).$$

- the above modification appears more tractable: containment has been verified for both, exponentially and super-exponentially decaying tails (Bai et al 2009).
- the original version has been analyzed in [SV10] and [FMP10] using different techniques.

Adaptive Metropolis - versions and stability

- The theory suggests increment

$$N(0, (2.38)^2\Sigma_n/d)$$

- The AM version of [HST01] (the original one) uses

$$N(0, \Sigma_n + \varepsilon Id)$$

- Modification due to [RR09] is to use

$$Q_n = (1-\beta)N(0, (2.38)^2\Sigma_n/d) + \beta N(0, \varepsilon Id/d).$$

- the above modification appears more tractable: containment has been verified for both, exponentially and super-exponentially decaying tails (Bai et al 2009).
- the original version has been analyzed in [SV10] and [FMP10] using different techniques.

Technicques of Fort et al.

- The Theory is very delicate and is building on the following crucial conditions.
- A1: For any $\theta \in \Theta$, there exists π_θ, s.t. $\pi_\theta = P_\theta \pi_\theta$.
- A2(a): For any $\epsilon > 0$, there exists a non-decreasing sequence $r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\left\|P_{\theta_{n-r_\epsilon(n)}}^{r_\epsilon(n)}(X_{n-r_\epsilon(n)},\cdot) - \pi_{\theta_{n-r_\epsilon(n)}}\right\|_{TV}\right] \leq \epsilon.$$

- A2(b): For any $\epsilon > 0$,

$$\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)})\right] = 0.$$

- the dependence on θ in π_θ above, is crucial for other algorithms like Interacting Tempering, however I will drop it for clarity in subsequent slides.

Technicques of Fort et al.

- The Theory is very delicate and is building on the following crucial conditions.
- A1: For any $\theta \in \Theta$, there exists π_θ, s.t. $\pi_\theta = P_\theta \pi_\theta$.
- A2(a): For any $\epsilon > 0$, there exists a non-decreasing sequence $r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\|P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)}, \cdot) - \pi_{\theta_{n-r_\epsilon(n)}}\|_{TV}\right] \leq \epsilon.$$

- A2(b): For any $\epsilon > 0$,

$$\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)})\right] = 0.$$

- the dependence on θ in π_θ above, is crucial for other algorithms like Interacting Tempering, however I will drop it for clarity in subsequent slides.

Technicques of Fort et al.

- The Theory is very delicate and is building on the following crucial conditions.
- A1: For any $\theta \in \Theta$, there exists π_θ, s.t. $\pi_\theta = P_\theta \pi_\theta$.
- A2(a): For any $\epsilon > 0$, there exists a non-decreasing sequence $r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\|P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)}, \cdot) - \pi_{\theta_{n-r_\epsilon(n)}}\|_{TV}\right] \leq \epsilon.$$

- A2(b): For any $\epsilon > 0$,

$$\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)})\right] = 0.$$

- the dependence on θ in π_θ above, is crucial for other algorithms like Interacting Tempering, however I will drop it for clarity in subsequent slides.

Technicques of Fort et al.

- The Theory is very delicate and is building on the following crucial conditions.
- A1: For any $\theta \in \Theta$, there exists π_θ, s.t. $\pi_\theta = P_\theta \pi_\theta$.
- A2(a): For any $\epsilon > 0$, there exists a non-decreasing sequence $r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\| P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)}, \cdot) - \pi_{\theta_{n-r_\epsilon(n)}} \|_{TV} \right] \le \epsilon.$$

- A2(b): For any $\epsilon > 0$,

$$\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)}) \right] = 0.$$

- the dependence on θ in π_θ above, is crucial for other algorithms like Interacting Tempering, however I will drop it for clarity in subsequent slides.

Technicques of Fort et al.

- The Theory is very delicate and is building on the following crucial conditions.
- A1: For any $\theta \in \Theta$, there exists π_θ, s.t. $\pi_\theta = P_\theta \pi_\theta$.
- A2(a): For any $\epsilon > 0$, there exists a non-decreasing sequence $r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\|P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)},\cdot) - \pi_{\theta_{n-r_\epsilon(n)}}\|_{TV}\right] \leq \epsilon.$$

- A2(b): For any $\epsilon > 0$,

$$\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)})\right] = 0.$$

- the dependence on θ in π_θ above, is crucial for other algorithms like Interacting Tempering, however I will drop it for clarity in subsequent slides.

Comparison to containment

- A2(a): For any $\epsilon > 0$, $\exists\ r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\|P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)}, \cdot) - \pi\|_{TV} \right] \leq \epsilon.$$

- A2(b): For any $\epsilon > 0$, $\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)}) \right] = 0.$
- Containment C(a): recall $M_\epsilon(x,\theta) := \inf_n\{\|P^n_\theta(x,\cdot) - \pi\|_{TV} \leq \epsilon\}$, and assume

$$\forall \delta > 0, \epsilon > 0, \ \exists\ M_{\epsilon,\delta} \quad \text{s.t.} \quad \forall n\ \ P(M_\epsilon(X_n,\theta_n) \leq M_{\epsilon,\delta}) \geq 1-\delta.$$

- Diminishing Adaptation C(b): $\lim_{n\to\infty} \mathbb{E}\left[D(\theta_{n-1},\theta_n)\right] = 0.$
- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \text{const}(\epsilon) = r_\epsilon$, then
  A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.

Comparison to containment

- A2(a): For any $\epsilon > 0$, $\exists\ r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\|P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)},\cdot) - \pi\|_{TV}\right] \leq \epsilon.$$

- A2(b): For any $\epsilon > 0$, $\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)})\right] = 0.$
- Containment C(a): recall $M_\epsilon(x,\theta) := \inf_n\{\|P^n_\theta(x,\cdot) - \pi\|_{TV} \leq \epsilon\}$, and assume

$$\forall\delta > 0, \epsilon > 0,\ \ \exists\ M_{\epsilon,\delta} \quad \text{s.t.} \quad \forall n\ \ P(M_\epsilon(X_n,\theta_n) \leq M_{\epsilon,\delta}) \geq 1-\delta.$$

- Diminishing Adaptation C(b): $\lim_{n\to\infty} \mathbb{E}\left[D(\theta_{n-1},\theta_n)\right] = 0.$
- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \text{const}(\epsilon) = r_\epsilon$, then
  A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.

Comparison to containment

- A2(a): For any $\epsilon > 0$, $\exists\, r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\|P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)},\cdot) - \pi\|_{TV}\right] \leq \epsilon.$$

- A2(b): For any $\epsilon > 0$, $\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)})\right] = 0.$
- Containment C(a): recall $M_\epsilon(x,\theta) := \inf_n\{\|P^n_\theta(x,\cdot) - \pi\|_{TV} \leq \epsilon\}$, and assume

$$\forall \delta > 0, \epsilon > 0, \ \exists\, M_{\epsilon,\delta} \quad \text{s.t.} \quad \forall n \ \ P(M_\epsilon(X_n,\theta_n) \leq M_{\epsilon,\delta}) \geq 1-\delta.$$

- Diminishing Adaptation C(b): $\lim_{n\to\infty} \mathbb{E}\left[D(\theta_{n-1},\theta_n)\right] = 0.$
- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \text{const}(\epsilon) = r_\epsilon$, then
  A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.

Comparison to containment

- A2(a): For any $\epsilon > 0$, $\exists\, r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\|P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)},\cdot) - \pi\|_{TV}\right] \leq \epsilon.$$

- A2(b): For any $\epsilon > 0$, $\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)})\right] = 0.$
- Containment C(a): recall $M_\epsilon(x,\theta) := \inf_n\{\|P^n_\theta(x,\cdot) - \pi\|_{TV} \leq \epsilon\}$, and assume

$$\forall \delta > 0, \epsilon > 0, \ \exists\, M_{\epsilon,\delta} \quad \text{s.t.} \quad \forall n \ \ P(M_\epsilon(X_n,\theta_n) \leq M_{\epsilon,\delta}) \geq 1-\delta.$$

- Diminishing Adaptation C(b): $\lim_{n\to\infty} \mathbb{E}\left[D(\theta_{n-1},\theta_n)\right] = 0.$
- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \text{const}(\epsilon) = r_\epsilon$, then A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.

Comparison to containment

- A2(a): For any $\epsilon > 0, \ \exists\ r_\epsilon(n),$ s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and
$$\limsup_{n\to\infty} \mathbb{E}\left[\|P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)},\cdot) - \pi\|_{TV}\right] \leq \epsilon.$$
- A2(b): For any $\epsilon > 0,$ $\qquad \lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)})\right] = 0.$
- Containment C(a): recall $M_\epsilon(x,\theta) := \inf_n\{\|P^n_\theta(x,\cdot) - \pi\|_{TV} \leq \epsilon\}$, and assume
$$\forall\delta > 0, \epsilon > 0, \ \ \exists\ M_{\epsilon,\delta} \qquad \text{s.t.} \qquad \forall n \ \ P(M_\epsilon(X_n,\theta_n) \leq M_{\epsilon,\delta}) \geq 1-\delta.$$
- Diminishing Adaptation C(b): $\lim_{n\to\infty} \mathbb{E}\left[D(\theta_{n-1},\theta_n)\right] = 0.$
- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \text{const}(\epsilon) = r_\epsilon,$ then
A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.

Comparison to containment

- A2(a): For any $\epsilon > 0$, $\exists\ r_\epsilon(n)$, s.t. $\limsup_{n\to\infty} r_\epsilon(n)/n = 0$ and

$$\limsup_{n\to\infty} \mathbb{E}\left[\|P^{r_\epsilon(n)}_{\theta_{n-r_\epsilon(n)}}(X_{n-r_\epsilon(n)},\cdot) - \pi\|_{TV}\right] \leq \epsilon.$$

- A2(b): For any $\epsilon > 0$, $\lim_{n\to\infty} \sum_{j=0}^{r_\epsilon(n)-1} \mathbb{E}\left[D(\theta_{n-r_\epsilon(n)+j}, \theta_{n-r_\epsilon(n)})\right] = 0.$
- Containment C(a): recall $M_\epsilon(x,\theta) := \inf_n\{\|P^n_\theta(x,\cdot) - \pi\|_{TV} \leq \epsilon\}$, and assume

$$\forall \delta > 0, \epsilon > 0, \ \ \exists\ M_{\epsilon,\delta} \qquad \text{s.t.} \qquad \forall n\ \ P(M_\epsilon(X_n,\theta_n) \leq M_{\epsilon,\delta}) \geq 1-\delta.$$

- Diminishing Adaptation C(b): $\lim_{n\to\infty} \mathbb{E}\left[D(\theta_{n-1},\theta_n)\right] = 0.$
- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \mathrm{const}(\epsilon) = r_\epsilon$, then
  A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.

Comparison to containment

- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \text{const}(\epsilon) = r_\epsilon,$ then A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.
- Therefore A2(a), A2(b) generalize C(a), C(b) (rather then weaken) and the generalization is in settings where $r_\epsilon(n)$ needs to grow to ∞ as $n \to \infty$.
- We shall **try to investigate**, what happens if $r_\epsilon(n)$ needs to grow to ∞ as $n \to \infty$.

Comparison to containment

- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \text{const}(\epsilon) = r_\epsilon,$ then A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.
- Therefore A2(a), A2(b) generalize C(a), C(b) (rather then weaken) and the generalization is in settings where $r_\epsilon(n)$ needs to grow to ∞ as $n \to \infty$.
- We shall **try to investigate**, what happens if $r_\epsilon(n)$ needs to grow to ∞ as $n \to \infty$.

Comparison to containment

- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \mathrm{const}(\epsilon) = r_\epsilon$, then A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.
- Therefore A2(a), A2(b) generalize C(a), C(b) (rather then weaken) and the generalization is in settings where $r_\epsilon(n)$ needs to grow to ∞ as $n \to \infty$.
- We shall **try to investigate**, what happens if $r_\epsilon(n)$ needs to grow to ∞ as $n \to \infty$.

Comparison to containment

- C(a), C(b) $\Rightarrow$ A2(a), A2(b) by taking e.g. $r_\epsilon(n) = M_{\epsilon/2,\epsilon/2}$.
- if $r_\epsilon(n) = \mathrm{const}(\epsilon) = r_\epsilon$, then A2(a), A2(b) $\Rightarrow$ C(a), C(b) by taking e.g. $M_{\epsilon,\delta} := r_{\epsilon\delta}$.
- Therefore A2(a), A2(b) generalize C(a), C(b) (rather then weaken) and the generalization is in settings where $r_\epsilon(n)$ needs to grow to ∞ as $n \to \infty$.
- We shall **try to investigate**, what happens if $r_\epsilon(n)$ needs to grow to ∞ as $n \to \infty$.

a new class: AdapFail Algorithms

- an adaptive algorithm $\mathcal{A} \in$ AdapFail, if with positive probability, it is asymptotically less efficient then ANY MCMC algorithm with fixed θ.
- more formally, AdapFail can be defined e.g. as follows: $\mathcal{A} \in$ AdapFail, if

$$\forall_{\epsilon_* > 0}, \quad \exists_{0 < \epsilon < \epsilon_*}, \quad \text{s.t.} \quad \lim_{K \to \infty} \inf_{\theta \in \Theta} \lim_{n \to \infty} P\Big(M_\epsilon(X_n, \theta_n) > K M_\epsilon(\tilde{X}_n, \theta)\Big) > 0,$$

  where $\{\tilde{X}_n\}$ is a Markov chain independent of $\{X_n\}$, which follows the fixed kernel P_θ.
- QuasiLemma: If containment doesn't hold for $\mathcal{A}$ then $\mathcal{A} \in$ AdapFail.
- If A2(a), A2(b) hold but C(a), C(b) do not hold, then $\mathcal{A} \in$ AdapFail, but it deteriorates slowly enough (due to more restrictive A2(b)), so that marginal distributions (still) converge, and SLLN (still) holds.
- However, if $\mathcal{A} \in$ AdapFail, then **we do not want to use it anyway!!**

a new class: `AdapFail` Algorithms

- an adaptive algorithm $\mathcal{A} \in$ `AdapFail`, if with positive probability, it is asymptotically less efficient then ANY MCMC algorithm with fixed θ.
- more formally, `AdapFail` can be defined e.g. as follows: $\mathcal{A} \in$ `AdapFail`, if

$$\forall_{\epsilon_*>0}, \quad \exists_{0<\epsilon<\epsilon_*}, \quad \text{s.t.} \quad \lim_{K\to\infty} \inf_{\theta\in\Theta} \lim_{n\to\infty} P\Big(M_\epsilon(X_n,\theta_n) > KM_\epsilon(\tilde{X}_n,\theta)\Big) > 0\,,$$

  where $\{\tilde{X}_n\}$ is a Markov chain independent of $\{X_n\}$, which follows the fixed kernel P_θ.
- QuasiLemma: If containment doesn't hold for $\mathcal{A}$ then $\mathcal{A} \in$ `AdapFail`.
- If A2(a), A2(b) hold but C(a), C(b) do not hold, then $\mathcal{A} \in$ `AdapFail`, but it deteriorates slowly enough (due to more restrictive A2(b)), so that marginal distributions (still) converge, and SLLN (still) holds.
- However, if $\mathcal{A} \in$ `AdapFail`, then **we do not want to use it anyway!!**

a new class: AdapFail Algorithms

- an adaptive algorithm $\mathcal{A} \in$ AdapFail, if with positive probability, it is asymptotically less efficient then ANY MCMC algorithm with fixed θ.
- more formally, AdapFail can be defined e.g. as follows: $\mathcal{A} \in$ AdapFail, if

$$\forall_{\epsilon_*>0}, \quad \exists_{0<\epsilon<\epsilon_*}, \quad \text{s.t.} \quad \lim_{K\to\infty} \inf_{\theta\in\Theta} \lim_{n\to\infty} P\Big(M_\epsilon(X_n,\theta_n) > K M_\epsilon(\tilde{X}_n,\theta)\Big) > 0\,,$$

  where $\{\tilde{X}_n\}$ is a Markov chain independent of $\{X_n\}$, which follows the fixed kernel P_θ.
- QuasiLemma: If containment doesn't hold for $\mathcal{A}$ then $\mathcal{A} \in$ AdapFail.
- If A2(a), A2(b) hold but C(a), C(b) do not hold, then $\mathcal{A} \in$ AdapFail, but it deteriorates slowly enough (due to more restrictive A2(b)), so that marginal distributions (still) converge, and SLLN (still) holds.
- However, if $\mathcal{A} \in$ AdapFail, then **we do not want to use it anyway!!**

a new class: AdapFail Algorithms

- an adaptive algorithm $\mathcal{A} \in$ AdapFail, if with positive probability, it is asymptotically less efficient then ANY MCMC algorithm with fixed θ.
- more formally, AdapFail can be defined e.g. as follows: $\mathcal{A} \in$ AdapFail, if

$$\forall_{\epsilon_*>0}, \quad \exists_{0<\epsilon<\epsilon_*}, \quad \text{s.t.} \quad \lim_{K\to\infty} \inf_{\theta\in\Theta} \lim_{n\to\infty} P\Big(M_\epsilon(X_n,\theta_n) > KM_\epsilon(\tilde{X}_n,\theta)\Big) > 0\,,$$

  where $\{\tilde{X}_n\}$ is a Markov chain independent of $\{X_n\}$, which follows the fixed kernel P_θ.
- QuasiLemma: If containment doesn't hold for $\mathcal{A}$ then $\mathcal{A} \in$ AdapFail.
- If A2(a), A2(b) hold but C(a), C(b) do not hold, then $\mathcal{A} \in$ AdapFail, but it deteriorates slowly enough (due to more restrictive A2(b)), so that marginal distributions (still) converge, and SLLN (still) holds.
- However, if $\mathcal{A} \in$ AdapFail, then **we do not want to use it anyway!!**

a new class: AdapFail Algorithms

- an adaptive algorithm $\mathcal{A} \in$ AdapFail, if with positive probability, it is asymptotically less efficient then ANY MCMC algorithm with fixed θ.
- more formally, AdapFail can be defined e.g. as follows: $\mathcal{A} \in$ AdapFail, if

$$\forall_{\epsilon_*>0}, \quad \exists_{0<\epsilon<\epsilon_*}, \quad \text{s.t.} \quad \lim_{K\to\infty} \inf_{\theta\in\Theta} \lim_{n\to\infty} P\Big(M_\epsilon(X_n, \theta_n) > K M_\epsilon(\tilde{X}_n, \theta)\Big) > 0\,,$$

  where $\{\tilde{X}_n\}$ is a Markov chain independent of $\{X_n\}$, which follows the fixed kernel P_θ.
- QuasiLemma: If containment doesn't hold for $\mathcal{A}$ then $\mathcal{A} \in$ AdapFail.
- If A2(a), A2(b) hold but C(a), C(b) do not hold, then $\mathcal{A} \in$ AdapFail, but it deteriorates slowly enough (due to more restrictive A2(b)), so that marginal distributions (still) converge, and SLLN (still) holds.
- However, if $\mathcal{A} \in$ AdapFail, then **we do not want to use it anyway!!**

a new class: AdapFail Algorithms

- an adaptive algorithm $\mathcal{A} \in$ AdapFail, if with positive probability, it is asymptotically less efficient then ANY MCMC algorithm with fixed θ.
- more formally, AdapFail can be defined e.g. as follows: $\mathcal{A} \in$ AdapFail, if

$$\forall_{\epsilon_*>0}, \quad \exists_{0<\epsilon<\epsilon_*}, \quad \text{s.t.} \quad \lim_{K\to\infty} \inf_{\theta\in\Theta} \lim_{n\to\infty} P\Big(M_\epsilon(X_n,\theta_n) > K M_\epsilon(\tilde{X}_n,\theta)\Big) > 0\,,$$

  where $\{\tilde{X}_n\}$ is a Markov chain independent of $\{X_n\}$, which follows the fixed kernel P_θ.
- QuasiLemma: If containment doesn't hold for $\mathcal{A}$ then $\mathcal{A} \in$ AdapFail.
- If A2(a), A2(b) hold but C(a), C(b) do not hold, then $\mathcal{A} \in$ AdapFail, but it deteriorates slowly enough (due to more restrictive A2(b)), so that marginal distributions (still) converge, and SLLN (still) holds.
- However, if $\mathcal{A} \in$ AdapFail, then **we do not want to use it anyway!!**

Current Challenges - theory and methodology

- Simplify the theoretical analysis of Adaptive MCMC
- Prove THE THEOREM that you can actually do it under verifiable conditions
- Design algorithms that are easier to analyse (recall the Adaptive Metropolis sampler)
- Devise other sound criteria that would guide adaptation (similarly as the 0.234 acceptance rule does)
- Adaptive MCMC is increasingly popular among practitioners - a research opportunity with large impact
- Good review articles: [AT08], [RR09], [Ros08], [Ros13] (from which I took the Goldilock principle plots)

Current Challenges - theory and methodology

- Simplify the theoretical analysis of Adaptive MCMC
- Prove THE THEOREM that you can actually do it under verifiable conditions
- Design algorithms that are easier to analyse (recall the Adaptive Metropolis sampler)
- Devise other sound criteria that would guide adaptation (similarly as the 0.234 acceptance rule does)
- Adaptive MCMC is increasingly popular among practitioners - a research opportunity with large impact
- Good review articles: [AT08], [RR09], [Ros08], [Ros13] (from which I took the Goldilock principle plots)

Current Challenges - theory and methodology

- Simplify the theoretical analysis of Adaptive MCMC
- Prove THE THEOREM that you can actually do it under verifiable conditions
- Design algorithms that are easier to analyse (recall the Adaptive Metropolis sampler)
- Devise other sound criteria that would guide adaptation (similarly as the 0.234 acceptance rule does)
- Adaptive MCMC is increasingly popular among practitioners - a research opportunity with large impact
- Good review articles: [AT08], [RR09], [Ros08], [Ros13] (from which I took the Goldilock principle plots)

Current Challenges - theory and methodology

- Simplify the theoretical analysis of Adaptive MCMC
- Prove THE THEOREM that you can actually do it under verifiable conditions
- Design algorithms that are easier to analyse (recall the Adaptive Metropolis sampler)
- Devise other sound criteria that would guide adaptation (similarly as the 0.234 acceptance rule does)
- Adaptive MCMC is increasingly popular among practitioners - a research opportunity with large impact
- Good review articles: [AT08], [RR09], [Ros08], [Ros13] (from which I took the Goldilock principle plots)

Current Challenges - theory and methodology

- Simplify the theoretical analysis of Adaptive MCMC
- Prove THE THEOREM that you can actually do it under verifiable conditions
- Design algorithms that are easier to analyse (recall the Adaptive Metropolis sampler)
- Devise other sound criteria that would guide adaptation (similarly as the 0.234 acceptance rule does)
- Adaptive MCMC is increasingly popular among practitioners - a research opportunity with large impact
- Good review articles: [AT08], [RR09], [Ros08], [Ros13] (from which I took the Goldilock principle plots)

Current Challenges - theory and methodology

- Simplify the theoretical analysis of Adaptive MCMC
- Prove THE THEOREM that you can actually do it under verifiable conditions
- Design algorithms that are easier to analyse (recall the Adaptive Metropolis sampler)
- Devise other sound criteria that would guide adaptation (similarly as the 0.234 acceptance rule does)
- Adaptive MCMC is increasingly popular among practitioners - a research opportunity with large impact
- Good review articles: [AT08], [RR09], [Ros08], [Ros13] (from which I took the Goldilock principle plots)

C. Andrieu and J. Thoms.
A tutorial on adaptive MCMC.
Statistics and Computing, 18(4):343–373, 2008.

Y. Bai, G.O. Roberts, and J.S. Rosenthal.
On the containment condition for adaptive Markov chain Monte Carlo algorithms.
Preprint, 2010.

G. Fort, E. Moulines, and P. Priouret.
Convergence of adaptive mcmc algorithms: Ergodicity and law of large numbers.
2010.

W.R. Gilks, G.O. Roberts, and S.K. Sahu.
Adaptive markov chain monte carlo through regeneration.
Journal of the American Statistical Association, 93(443):1045–1054, 1998.

H. Haario, E. Saksman, and J. Tamminen.
An adaptive Metropolis algorithm.
Bernoulli, 7(2):223–242, 2001.

R.A. Levine and G. Casella.
Optimizing random scan Gibbs samplers.
Journal of Multivariate Analysis, 97(10):2071–2100, 2006.

K. Łatuszyński, G.O. Roberts, and J.S. Rosenthal.
Adaptive Gibbs samplers and related MCMC methods.
Ann. Appl. Probab., 23(1):66–98, 2013.

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(1):110–120, 1997.

J.S Rosenthal.
Optimal proposal distributions and adaptive MCMC.
Preprint, 2008.

- J.S. Rosenthal.
  Optimising and adapting the metropolis algorithm.
  preprint, 2013.

- G.O. Roberts and J.S. Rosenthal.
  Optimal scaling for various Metropolis-Hastings algorithms.
  Statistical Science, 16(4):351–367, 2001.

- G.O. Roberts and J.S. Rosenthal.
  Examples of adaptive MCMC.
  Journal of Computational and Graphical Statistics, 18(2):349–367, 2009.

- E. Saksman and M. Vihola.
  On the ergodicity of the adaptive metropolis algorithm on unbounded domains.
  The Annals of Applied Probability, 20(6):2178–2203, 2010.

M. Vihola.
On the stability and ergodicity of an adaptive scaling Metropolis algorithm.
Arxiv preprint arXiv:0903.4061, 2009.