# Robustness of Manifold MALA algorithms

Krzysztof Latuszynski
(University of Warwick, UK)

joint work with

Gareth O. Roberts    Katarzyna Wolny

WOGAS3, 2011

# MCMC: reversibility

- $\pi$ the target probability distribution on $\mathcal{X}$ (known up to a normalizing constant)
- MCMC algorithms aim at Monte Carlo sampling from $\pi$ by designing a Markov chain $P$ s.t.

$$\pi P = \pi$$

- they use the reversibility condition (aka detailed balance)

$$\pi(x)P(x,y) = \pi(y)P(y,x) \qquad \text{for every } x, y \in \mathcal{X}$$

  to design appropriate $P$ .
- Fact: if $\pi$ and $P$ satisfy the reversibility condition then $\pi P = \pi$
- the Metropolis-Hastings algorithm takes virtually any transition kernel $Q$ and adjusts it by accepting moves from $Q$ with probability

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

  to enforce reversibility wrt $\pi$

# MCMC: reversibility

- $\pi$ the target probability distribution on $\mathcal{X}$ (known up to a normalizing constant)
- MCMC algorithms aim at Monte Carlo sampling from $\pi$ by designing a Markov chain $P$ s.t.

$$\pi P = \pi$$

- they use the reversibility condition (aka detailed balance)

$$\pi(x)P(x,y) = \pi(y)P(y,x) \qquad \text{for every } x, y \in \mathcal{X}$$

to design appropriate $P$.
- Fact: if $\pi$ and $P$ satisfy the reversibility condition then $\pi P = \pi$
- the Metropolis-Hastings algorithm takes virtually any transition kernel $Q$ and adjusts it by accepting moves from $Q$ with probability

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

to enforce reversibility wrt $\pi$

# MCMC: reversibility

- $\pi$ the target probability distribution on $\mathcal{X}$ (known up to a normalizing constant)
- MCMC algorithms aim at Monte Carlo sampling from $\pi$ by designing a Markov chain $P$ s.t.

$$\pi P = \pi$$

- they use the reversibility condition (aka detailed balance)

$$\pi(x)P(x,y) = \pi(y)P(y,x) \qquad \text{for every } x,y \in \mathcal{X}$$

to design appropriate $P$.
- Fact: if $\pi$ and $P$ satisfy the reversibility condition then $\pi P = \pi$
- the Metropolis-Hastings algorithm takes virtually any transition kernel $Q$ and adjusts it by accepting moves from $Q$ with probability

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

to enforce reversibility wrt $\pi$

# MCMC: reversibility

- $\pi$ the target probability distribution on $\mathcal{X}$ (known up to a normalizing constant)
- MCMC algorithms aim at Monte Carlo sampling from $\pi$ by designing a Markov chain $P$ s.t.

$$\pi P = \pi$$

- they use the reversibility condition (aka detailed balance)

$$\pi(x)P(x,y) = \pi(y)P(y,x) \qquad \text{for every } x, y \in \mathcal{X}$$

  to design appropriate $P$ .
- Fact: if $\pi$ and $P$ satisfy the reversibility condition then $\pi P = \pi$
- the Metropolis-Hastings algorithm takes virtually any transition kernel $Q$ and adjusts it by accepting moves from $Q$ with probability

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

  to enforce reversibility wrt $\pi$

# MCMC: reversibility

- $\pi$ the target probability distribution on $\mathcal{X}$ (known up to a normalizing constant)
- MCMC algorithms aim at Monte Carlo sampling from $\pi$ by designing a Markov chain $P$ s.t.

$$\pi P = \pi$$

- they use the reversibility condition (aka detailed balance)

$$\pi(x)P(x,y) = \pi(y)P(y,x) \qquad \text{for every } x, y \in \mathcal{X}$$

  to design appropriate $P$ .
- Fact: if $\pi$ and $P$ satisfy the reversibility condition then $\pi P = \pi$
- the Metropolis-Hastings algorithm takes virtually any transition kernel $Q$ and adjusts it by accepting moves from $Q$ with probability

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

  to enforce reversibility wrt $\pi$

# MCMC: proposals form Langevin diffusions

- Since any $Q$ is OK, we can take $Q(x,\cdot) \sim N(x,\sigma)$
- this yields the basic Random Walk Metropolis
- is it optimal?
- HAHAHA!!!
- can we improve the algorithm by using a better $Q$ ?
- Any choice of $\sigma(x)$ in

$$dx_t = \left(\frac{\sigma^2(x_t)}{2}\nabla\log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t)\right)dt + \sigma(x_t)dB(t),$$

  yields a Langevin diffusion with the correct stationary distribution $\pi$.
- For any such diffusion we can use its Euler discretization to produce proposals.
- For fixed $\sigma(x) = \sigma$ one obtains the standard MALA algorithm with

$$Q(x,\cdot) \quad \sim \quad N(x + \frac{\sigma}{2}\nabla\log(\pi(x)), \sigma^2)$$

# MCMC: proposals form Langevin diffusions

- Since any $Q$ is OK, we can take $Q(x,\cdot) \sim N(x,\sigma)$
- this yields the basic Random Walk Metropolis
- is it optimal?
- HAHAHA!!!
- can we improve the algorithm by using a better $Q$ ?
- Any choice of $\sigma(x)$ in

$$dx_t = \left(\frac{\sigma^2(x_t)}{2}\nabla \log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t)\right)dt + \sigma(x_t)dB(t),$$

  yields a Langevin diffusion with the correct stationary distribution $\pi$.
- For any such diffusion we can use its Euler discretization to produce proposals.
- For fixed $\sigma(x) = \sigma$ one obtains the standard MALA algorithm with

$$Q(x,\cdot) \quad \sim \quad N(x + \frac{\sigma}{2}\nabla \log(\pi(x)), \sigma^2)$$

# MCMC: proposals form Langevin diffusions

- Since any $Q$ is OK, we can take $Q(x, \cdot) \sim N(x, \sigma)$
- this yields the basic Random Walk Metropolis
- is it optimal?
- HAHAHA!!!
- can we improve the algorithm by using a better $Q$ ?
- Any choice of $\sigma(x)$ in

$$dx_t = \left( \frac{\sigma^2(x_t)}{2} \nabla \log(\pi(x_t)) + \sigma(x_t) \nabla \sigma(x_t) \right) dt + \sigma(x_t) dB(t),$$

  yields a Langevin diffusion with the correct stationary distribution $\pi$.
- For any such diffusion we can use its Euler discretization to produce proposals.
- For fixed $\sigma(x) = \sigma$ one obtains the standard MALA algorithm with

$$Q(x, \cdot) \quad \sim \quad N(x + \frac{\sigma}{2} \nabla \log(\pi(x)), \sigma^2)$$

# MCMC: proposals form Langevin diffusions

- Since any $Q$ is OK, we can take $Q(x,\cdot) \sim N(x,\sigma)$
- this yields the basic Random Walk Metropolis
- is it optimal?
- HAHAHA!!!
- can we improve the algorithm by using a better $Q$ ?
- Any choice of $\sigma(x)$ in

$$dx_t = \left(\frac{\sigma^2(x_t)}{2}\nabla \log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t)\right)dt + \sigma(x_t)dB(t),$$

  yields a Langevin diffusion with the correct stationary distribution $\pi$.
- For any such diffusion we can use its Euler discretization to produce proposals.
- For fixed $\sigma(x) = \sigma$ one obtains the standard MALA algorithm with

$$Q(x,\cdot) \quad \sim \quad N(x + \frac{\sigma}{2}\nabla \log(\pi(x)), \sigma^2)$$

# MCMC: proposals form Langevin diffusions

- Since any $Q$ is OK, we can take $Q(x,\cdot) \sim N(x,\sigma)$
- this yields the basic Random Walk Metropolis
- is it optimal?
- HAHAHA!!!
- can we improve the algorithm by using a better $Q$ ?
- Any choice of $\sigma(x)$ in

$$dx_t = \left( \frac{\sigma^2(x_t)}{2} \nabla \log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t) \right) dt + \sigma(x_t) dB(t),$$

  yields a Langevin diffusion with the correct stationary distribution $\pi$.
- For any such diffusion we can use its Euler discretization to produce proposals.
- For fixed $\sigma(x) = \sigma$ one obtains the standard MALA algorithm with

$$Q(x,\cdot) \quad \sim \quad N(x + \frac{\sigma}{2}\nabla \log(\pi(x)), \sigma^2)$$

# MCMC: proposals form Langevin diffusions

- Since any $Q$ is OK, we can take $Q(x,\cdot) \sim N(x,\sigma)$
- this yields the basic Random Walk Metropolis
- is it optimal?
- HAHAHA!!!
- can we improve the algorithm by using a better $Q$ ?
- Any choice of $\sigma(x)$ in

$$dx_t = \left(\frac{\sigma^2(x_t)}{2}\nabla \log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t)\right)dt + \sigma(x_t)dB(t),$$

  yields a Langevin diffusion with the correct stationary distribution $\pi$.
- For any such diffusion we can use its Euler discretization to produce proposals.
- For fixed $\sigma(x) = \sigma$ one obtains the standard MALA algorithm with

$$Q(x,\cdot) \quad \sim \quad N(x + \frac{\sigma}{2}\nabla \log(\pi(x)), \sigma^2)$$

# MCMC: proposals form Langevin diffusions

- Since any $Q$ is OK, we can take $Q(x, \cdot) \sim N(x, \sigma)$
- this yields the basic Random Walk Metropolis
- is it optimal?
- HAHAHA!!!
- can we improve the algorithm by using a better $Q$ ?
- Any choice of $\sigma(x)$ in

$$dx_t = \left( \frac{\sigma^2(x_t)}{2} \nabla \log(\pi(x_t)) + \sigma(x_t) \nabla \sigma(x_t) \right) dt + \sigma(x_t) dB(t),$$

  yields a Langevin diffusion with the correct stationary distribution $\pi$.
- For any such diffusion we can use its Euler discretization to produce proposals.
- For fixed $\sigma(x) = \sigma$ one obtains the standard MALA algorithm with

$$Q(x, \cdot) \quad \sim \quad N(x + \frac{\sigma}{2} \nabla \log(\pi(x)), \sigma^2)$$

# MCMC: proposals form Langevin diffusions

- Since any $Q$ is OK, we can take $Q(x, \cdot) \sim N(x, \sigma)$
- this yields the basic Random Walk Metropolis
- is it optimal?
- HAHAHA!!!
- can we improve the algorithm by using a better $Q$ ?
- Any choice of $\sigma(x)$ in

$$dx_t = \left( \frac{\sigma^2(x_t)}{2} \nabla \log(\pi(x_t)) + \sigma(x_t) \nabla \sigma(x_t) \right) dt + \sigma(x_t) dB(t),$$

  yields a Langevin diffusion with the correct stationary distribution $\pi$.
- For any such diffusion we can use its Euler discretization to produce proposals.
- For fixed $\sigma(x) = \sigma$ one obtains the standard MALA algorithm with

$$Q(x, \cdot) \quad \sim \quad N(x + \frac{\sigma}{2} \nabla \log(\pi(x)), \sigma^2)$$

# Link with Manifold MALA

- It is difficult to find a reasonable $\sigma(x) \neq \sigma$ in

$$dx_t = \left( \frac{\sigma^2(x_t)}{2} \nabla \log(\pi(x_t)) + \sigma(x_t) \nabla \sigma(x_t) \right) dt + \sigma(x_t) dB(t)$$

- Girolami and Calderhead [GC11] consider Langevin diffusions evolving on a manifold rather then on a flat surface.
- abuse of notation: in the Bayesian setting the parameter space $\mathcal{X} := \Theta$ and

$$\pi(x) \; := \; \pi(\theta) \; = \; \text{prior}(\theta) l(\text{data}|\theta)$$

- The geometry of the manifold is defined by a metric tensor $G(\theta)$ of users choice.

# Link with Manifold MALA

- It is difficult to find a reasonable $\sigma(x) \neq \sigma$ in

$$dx_t = \left( \frac{\sigma^2(x_t)}{2} \nabla \log(\pi(x_t)) + \sigma(x_t) \nabla \sigma(x_t) \right) dt + \sigma(x_t) dB(t)$$

- Girolami and Calderhead [GC11] consider Langevin diffusions evolving on a manifold rather then on a flat surface.
- abuse of notation: in the Bayesian setting the parameter space $\mathcal{X} := \Theta$ and

$$\pi(x) := \pi(\theta) = \text{prior}(\theta) l(\text{data}|\theta)$$

- The geometry of the manifold is defined by a metric tensor $G(\theta)$ of users choice.

# Link with Manifold MALA

- It is difficult to find a reasonable $\sigma(x) \neq \sigma$ in

$$dx_t = \left( \frac{\sigma^2(x_t)}{2} \nabla \log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t) \right) dt + \sigma(x_t) dB(t)$$

- Girolami and Calderhead [GC11] consider Langevin diffusions evolving on a manifold rather then on a flat surface.
- abuse of notation: in the Bayesian setting the parameter space $\mathcal{X} := \Theta$ and

$$\pi(x) := \pi(\theta) = \text{prior}(\theta) l(\text{data}|\theta)$$

- The geometry of the manifold is defined by a metric tensor $G(\theta)$ of users choice.

# Link with Manifold MALA

- It is difficult to find a reasonable $\sigma(x) \neq \sigma$ in

$$dx_t = \left( \frac{\sigma^2(x_t)}{2} \nabla \log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t) \right) dt + \sigma(x_t) dB(t)$$

- Girolami and Calderhead [GC11] consider Langevin diffusions evolving on a manifold rather then on a flat surface.
- abuse of notation: in the Bayesian setting the parameter space $\mathcal{X} := \Theta$ and

$$\pi(x) \;:=\; \pi(\theta) \;=\; \text{prior}(\theta) l(\text{data}|\theta)$$

- The geometry of the manifold is defined by a metric tensor $G(\theta)$ of users choice.

# Link with Manifold MALA continued

- One possibility is to take the observed Fisher information matrix plus the negative Hessian of the log-prior

$$G(\theta) \ := \ -\frac{\partial^2}{\partial\theta^2}\log\{l(\text{data}|\theta)\} - \frac{\partial^2}{\partial\theta^2}\log\{\text{prior}(\theta)\}$$

- This is equivalent to letting

$$\sigma^2(x) \ := \ \left|\left[\frac{\partial^2}{\partial x^2}\log(\pi(x))\right]^{-1}\right|$$

in $$dx_t = \left(\frac{\sigma^2(x_t)}{2}\nabla\log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t)\right)dt + \sigma(x_t)dB(t)$$

- And use its Euler discretization for proposals.
- Our goal is to systematically investigate the performance of this algorithm on a variety of target distributions.

# Link with Manifold MALA continued

- One possibility is to take the observed Fisher information matrix plus the negative Hessian of the log-prior

$$G(\theta) := -\frac{\partial^2}{\partial\theta^2}\log\{l(\text{data}|\theta)\} - \frac{\partial^2}{\partial\theta^2}\log\{\text{prior}(\theta)\}$$

- This is equivalent to letting

$$\sigma^2(x) := \left|\left[\frac{\partial^2}{\partial x^2}\log(\pi(x))\right]^{-1}\right|$$

in
$$dx_t = \left(\frac{\sigma^2(x_t)}{2}\nabla\log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t)\right)dt + \sigma(x_t)dB(t)$$

- And use its Euler discretization for proposals.
- Our goal is to systematically investigate the performance of this algorithm on a variety of target distributions.

# Link with Manifold MALA continued

- One possibility is to take the observed Fisher information matrix plus the negative Hessian of the log-prior

$$G(\theta) := -\frac{\partial^2}{\partial\theta^2}\log\{l(\text{data}|\theta)\} - \frac{\partial^2}{\partial\theta^2}\log\{\text{prior}(\theta)\}$$

- This is equivalent to letting

$$\sigma^2(x) := \left|\left[\frac{\partial^2}{\partial x^2}\log(\pi(x))\right]^{-1}\right|$$

in $$dx_t = \left(\frac{\sigma^2(x_t)}{2}\nabla\log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t)\right)dt + \sigma(x_t)dB(t)$$

- And use its Euler discretization for proposals.
- Our goal is to systematically investigate the performance of this algorithm on a variety of target distributions.

# Link with Manifold MALA continued

- One possibility is to take the observed Fisher information matrix plus the negative Hessian of the log-prior

$$G(\theta) \;:=\; -\frac{\partial^2}{\partial\theta^2}\log\{l(\text{data}|\theta)\} - \frac{\partial^2}{\partial\theta^2}\log\{\text{prior}(\theta)\}$$

- This is equivalent to letting

$$\sigma^2(x) \;:=\; \left|\left[\frac{\partial^2}{\partial x^2}\log(\pi(x))\right]^{-1}\right|$$

in $$dx_t = \left(\frac{\sigma^2(x_t)}{2}\nabla\log(\pi(x_t)) + \sigma(x_t)\nabla\sigma(x_t)\right)dt + \sigma(x_t)dB(t)$$

- And use its Euler discretization for proposals.
- Our goal is to systematically investigate the performance of this algorithm on a variety of target distributions.

# Geometric ergodicity

- Under very mild conditions all these algorithms are ergodic, i.e.

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \;\to\; 0 \qquad \text{as} \;\; n \to \infty$$

- But how to assess performance of an MCMC algorithm?
- We say that an algorithm $P$ is geometrically ergodic if there exists $\gamma < 1$ s.t.

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \;\leq\; M(x)\gamma^n$$

- Thm: If $f : \mathcal{X} \to \mathbb{R}$ is such that $\int f^2(x)\pi(x)dx < \infty$, and the MCMC algorithm $P$ is geometrically ergodic and reversible then the CLT holds for estimating $\int f(x)\pi(x)dx.$

# Geometric ergodicity

- Under very mild conditions all these algorithms are ergodic, i.e.

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \to 0 \qquad \text{as } n \to \infty$$

- But how to assess performance of an MCMC algorithm?
- We say that an algorithm $P$ is geometrically ergodic if there exists $\gamma < 1$ s.t.

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \leq M(x)\gamma^n$$

- Thm: If $f : \mathcal{X} \to \mathbb{R}$ is such that $\int f^2(x)\pi(x)dx < \infty$, and the MCMC algorithm $P$ is geometrically ergodic and reversible then the CLT holds for estimating $\int f(x)\pi(x)dx$.

# Geometric ergodicity

- Under very mild conditions all these algorithms are ergodic, i.e.

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \;\to\; 0 \qquad \text{as} \;\; n \to \infty$$

- But how to assess performance of an MCMC algorithm?
- We say that an algorithm $P$ is geometrically ergodic if there exists $\gamma < 1$ s.t.

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \;\leq\; M(x)\gamma^n$$

- Thm: If $f : \mathcal{X} \to \mathbb{R}$ is such that $\int f^2(x)\pi(x)dx < \infty$, and the MCMC algorithm $P$ is geometrically ergodic and reversible then the CLT holds for estimating $\int f(x)\pi(x)dx.$

# Geometric ergodicity

- Under very mild conditions all these algorithms are ergodic, i.e.

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \to 0 \qquad \text{as } n \to \infty$$

- But how to assess performance of an MCMC algorithm?
- We say that an algorithm $P$ is geometrically ergodic if there exists $\gamma < 1$ s.t.

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \leq M(x)\gamma^n$$

- Thm: If $f : \mathcal{X} \to \mathbb{R}$ is such that $\int f^2(x)\pi(x)dx < \infty$, and the MCMC algorithm $P$ is geometrically ergodic and reversible then the CLT holds for estimating $\int f(x)\pi(x)dx$.

# Drift conditions

- Geometric ergodicity is implied by the following Drift Conditions

$$PV(x) \le \lambda V(x) \qquad \text{for} \quad x \notin C,$$
$$PV(x) \le K \qquad \text{for} \quad x \in C,$$

where $V : \mathcal{X} \to [1, \infty)$, the constants $\lambda < 1$ and $K < \infty$ and $C$ is a small set satisfying for some $\varepsilon > 0$ and a some probability measure $\nu$

$$P(x, \cdot) \ge \varepsilon \nu(\cdot).$$

- To conclude lack of geometric ergodicity define

$$\alpha_x := \int_{\mathcal{X}} \alpha(x, y) q(x, y) dy$$

- Now if $\lim_{\|x\| \to \infty} \alpha_x = 0$, then the algorithm is not geometrically ergodic.

# Drift conditions

- Geometric ergodicity is implied by the following Drift Conditions

$$PV(x) \leq \lambda V(x) \qquad \text{for} \quad x \notin C,$$
$$PV(x) \leq K \qquad \text{for} \quad x \in C,$$

where $V : \mathcal{X} \to [1, \infty)$, the constants $\lambda < 1$ and $K < \infty$ and $C$ is a small set satisfying for some $\varepsilon > 0$ and a some probability measure $\nu$

$$P(x, \cdot) \geq \varepsilon \nu(\cdot).$$

- To conclude lack of geometric ergodicity define

$$\alpha_x \quad := \quad \int_{\mathcal{X}} \alpha(x, y) q(x, y) dy$$

- Now if $\lim_{\|x\| \to \infty} \alpha_x \;=\; 0,$
then the algorithm is not geometrically ergodic.

# Drift conditions

- Geometric ergodicity is implied by the following Drift Conditions

$$PV(x) \leq \lambda V(x) \qquad \text{for} \quad x \notin C,$$
$$PV(x) \leq K \qquad \text{for} \quad x \in C,$$

where $V : \mathcal{X} \to [1, \infty)$, the constants $\lambda < 1$ and $K < \infty$ and $C$ is a small set satisfying for some $\varepsilon > 0$ and a some probability measure $\nu$

$$P(x, \cdot) \geq \varepsilon \nu(\cdot).$$

- To conclude lack of geometric ergodicity define

$$\alpha_x := \int_{\mathcal{X}} \alpha(x, y) q(x, y) dy$$

- Now if $\lim_{\|x\| \to \infty} \alpha_x = 0,$
then the algorithm is not geometrically ergodic.

# Drift conditions

- Geometric ergodicity is implied by the following Drift Conditions

$$PV(x) \leq \lambda V(x) \qquad \text{for} \quad x \notin C,$$
$$PV(x) \leq K \qquad \text{for} \quad x \in C,$$

where $V : \mathcal{X} \to [1, \infty)$ , the constants $\lambda < 1$ and $K < \infty$ and $C$ is a small set satisfying for some $\varepsilon > 0$ and a some probability measure $\nu$

$$P(x, \cdot) \geq \varepsilon \nu(\cdot).$$

- To conclude lack of geometric ergodicity define

$$\alpha_x := \int_{\mathcal{X}} \alpha(x, y) q(x, y) dy$$

- Now if $\lim_{\|x\| \to \infty} \alpha_x = 0,$
then the algorithm is not geometrically ergodic.

# Results for benchmark targets

- We establish Drift Conditions for the Manifold MALA for a family of benchmark target distributions

$$\pi(x) \quad \propto \quad \exp\{-\gamma\|x\|^{\beta}\}$$

- The summary of our results for manifold MALA together with the respective properties of random walk Metropolis and standard MALA:

| algorithm | $0 < \beta < 1$ | $\beta = 1$ | $1 < \beta < 2$ | $\beta = 2$ | $2 < \beta$ |
|---|---|---|---|---|---|
| RWM | N | Y | Y | Y | Y |
| MALA | N | Y | Y | Y | N |
| MMALA | Y | | Y | Y | Y |

Table 1. Geometric Ergodicity of random walk Metropolis (RWM), MALA [RT96, MT96] and manifold MALA (MMALA). N = geometric ergodicity fails, Y = geometric ergodicity holds.

- It appears that Manifold MALA outperforms the RWM and the standard MALA in both convergence rates (empirical experience) and robustness (the above table)
- Extensions towards more general target distributions are work in progress

# Results for benchmark targets

- We establish Drift Conditions for the Manifold MALA for a family of benchmark target distributions

$$\pi(x) \quad \propto \quad \exp\{-\gamma\|x\|^{\beta}\}$$

- The summary of our results for manifold MALA together with the respective properties of random walk Metropolis and standard MALA:

| algorithm | $0<\beta<1$ | $\beta=1$ | $1<\beta<2$ | $\beta=2$ | $2<\beta$ |
|---|---|---|---|---|---|
| RWM | N | Y | Y | Y | Y |
| MALA | N | Y | Y | Y | N |
| MMALA | Y | | Y | Y | Y |

Table 1. Geometric Ergodicity of random walk Metropolis (RWM), MALA [RT96, MT96] and manifold MALA (MMALA). N = geometric ergodicity fails, Y = geometric ergodicity holds.

- It appears that Manifold MALA outperforms the RWM and the standard MALA in both convergence rates (empirical experience) and robustness (the above table)
- Extensions towards more general target distributions are work in progress

# Results for benchmark targets

- We establish Drift Conditions for the Manifold MALA for a family of benchmark target distributions

$$\pi(x) \quad \propto \quad \exp\{-\gamma \|x\|^{\beta}\}$$

- The summary of our results for manifold MALA together with the respective properties of random walk Metropolis and standard MALA:

| algorithm | $0 < \beta < 1$ | $\beta = 1$ | $1 < \beta < 2$ | $\beta = 2$ | $2 < \beta$ |
|---|---|---|---|---|---|
| RWM | N | Y | Y | Y | Y |
| MALA | N | Y | Y | Y | N |
| MMALA | Y | | Y | Y | Y |

Table 1. Geometric Ergodicity of random walk Metropolis (RWM), MALA [RT96, MT96] and manifold MALA (MMALA). N = geometric ergodicity fails, Y = geometric ergodicity holds.

- It appears that Manifold MALA outperforms the RWM and the standard MALA in both convergence rates (empirical experience) and robustness (the above table)
- Extensions towards more general target distributions are work in progress

# Results for benchmark targets

- We establish Drift Conditions for the Manifold MALA for a family of benchmark target distributions

$$\pi(x) \quad \propto \quad \exp\{-\gamma\|x\|^{\beta}\}$$

- The summary of our results for manifold MALA together with the respective properties of random walk Metropolis and standard MALA:

| algorithm | $0 < \beta < 1$ | $\beta = 1$ | $1 < \beta < 2$ | $\beta = 2$ | $2 < \beta$ |
|---|---|---|---|---|---|
| RWM | N | Y | Y | Y | Y |
| MALA | N | Y | Y | Y | N |
| MMALA | Y | | Y | Y | Y |

Table 1. Geometric Ergodicity of random walk Metropolis (RWM), MALA [RT96, MT96] and manifold MALA (MMALA). N = geometric ergodicity fails, Y = geometric ergodicity holds.

- It appears that Manifold MALA outperforms the RWM and the standard MALA in both convergence rates (empirical experience) and robustness (the above table)
- Extensions towards more general target distributions are work in progress

# Results for benchmark targets

- We establish Drift Conditions for the Manifold MALA for a family of benchmark target distributions

$$\pi(x) \quad \propto \quad \exp\{-\gamma\|x\|^{\beta}\}$$

- The summary of our results for manifold MALA together with the respective properties of random walk Metropolis and standard MALA:

| algorithm | $0 < \beta < 1$ | $\beta = 1$ | $1 < \beta < 2$ | $\beta = 2$ | $2 < \beta$ |
|---|---|---|---|---|---|
| RWM | N | Y | Y | Y | Y |
| MALA | N | Y | Y | Y | N |
| MMALA | Y | | Y | Y | Y |

Table 1. Geometric Ergodicity of random walk Metropolis (RWM), MALA [RT96, MT96] and manifold MALA (MMALA). N = geometric ergodicity fails, Y = geometric ergodicity holds.

- It appears that Manifold MALA outperforms the RWM and the standard MALA in both convergence rates (empirical experience) and robustness (the above table)
- Extensions towards more general target distributions are work in progress

# Establishing Drift Conditions for Manifold MALA

- Verifying Drift Conditions is technical, we outline main steps.
- For the drift function we choose

$$V(x) = \|x\|$$

(and use the fact that if $PV(x) \leq \lambda V(x)$ then also $P(V(x)+1) \leq \lambda(V(x)+1)$.)

- First prove that the drift condition holds conditionally on accepting the proposal.
- We then prove that since $\alpha_x \nrightarrow 0$ the drift condition holds unconditionally.
- The argument uses the fact that the expected value of the Manifold MALA is a contraction in $\|x\|$.

# Establishing Drift Conditions for Manifold MALA

- Verifying Drift Conditions is technical, we outline main steps.
- For the drift function we choose

$$V(x) = \|x\|$$

(and use the fact that if $PV(x) \leq \lambda V(x)$ then also $P(V(x)+1) \leq \lambda(V(x)+1)$.)

- First prove that the drift condition holds conditionally on accepting the proposal.
- We then prove that since $\alpha_x \nrightarrow 0$ the drift condition holds unconditionally.
- The argument uses the fact that the expected value of the Manifold MALA is a contraction in $\|x\|$.

# Establishing Drift Conditions for Manifold MALA

- Verifying Drift Conditions is technical, we outline main steps.
- For the drift function we choose

$$V(x) = \|x\|$$

(and use the fact that if $PV(x) \leq \lambda V(x)$ then also $P(V(x)+1) \leq \lambda(V(x)+1)$.)

- First prove that the drift condition holds conditionally on accepting the proposal.
- We then prove that since $\alpha_x \nrightarrow 0$ the drift condition holds unconditionally.
- The argument uses the fact that the expected value of the Manifold MALA is a contraction in $\|x\|$.

# Establishing Drift Conditions for Manifold MALA

- Verifying Drift Conditions is technical, we outline main steps.
- For the drift function we choose

$$V(x) = \|x\|$$

(and use the fact that if $PV(x) \leq \lambda V(x)$ then also $P(V(x)+1) \leq \lambda(V(x)+1)$.)

- First prove that the drift condition holds conditionally on accepting the proposal.
- We then prove that since $\alpha_x \nrightarrow 0$ the drift condition holds unconditionally.
- The argument uses the fact that the expected value of the Manifold MALA is a contraction in $\|x\|$.

# Establishing Drift Conditions for Manifold MALA

- Verifying Drift Conditions is technical, we outline main steps.
- For the drift function we choose

$$V(x) = \|x\|$$

(and use the fact that if $PV(x) \leq \lambda V(x)$ then also $P(V(x)+1) \leq \lambda(V(x)+1)$.)

- First prove that the drift condition holds conditionally on accepting the proposal.
- We then prove that since $\alpha_x \nrightarrow 0$ the drift condition holds unconditionally.
- The argument uses the fact that the expected value of the Manifold MALA is a contraction in $\|x\|$.

M. Girolami and B. Calderhead.
Riemann manifold Langevin and Hamiltonian Monte Carlo methods.
*Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(2):123–214, 2011.

K.L. Mengersen and R.L. Tweedie.
Rates of convergence of the Hastings and Metropolis algorithms.
*The Annals of Statistics*, 24(1):101–121, 1996.

G.O. Roberts and R.L. Tweedie.
Exponential convergence of Langevin distributions and their discrete approximations.
*Bernoulli*, 2(4):341–363, 1996.