

Statistical Models & Computing Methods

Lecture 8: Advanced MCMC



Cheng Zhang

School of Mathematical Sciences, Peking University

October 21, 2024

- ▶ Simple MCMC methods, such as Metropolis algorithm and Gibbs sampler explore the posterior distribution using simple mechanism (e.g., a random walk)
- ▶ While this strategy might work well for low-dimensional distributions, it could become very inefficient (e.g., high autocorrelation, missing isolated modes) for high-dimensional distributions
- ▶ In this lecture, we discuss several advanced techniques to improve the efficiency of Markov chain Monte Carlo methods

Random walk Metropolis (RWM) is **struggling** with a banana-shaped distribution

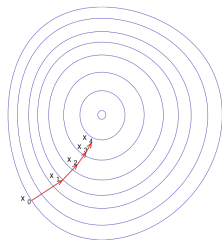
Random walk Metropolis (RWM) is **struggling** with a banana-shaped distribution

- ▶ Random proposals are likely to be inefficient, since they completely ignore the target distribution
- ▶ A better way would be to use information from the target distribution to guide our proposals
- ▶ Note that in optimization, the gradient points to an ascent direction, which would also be useful when designing the proposal distributions

$$x' = x + \epsilon \nabla \log p(x)$$

when ϵ is small,

$$\log p(x') > \log p(x)$$



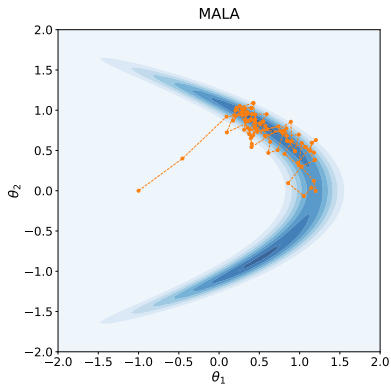
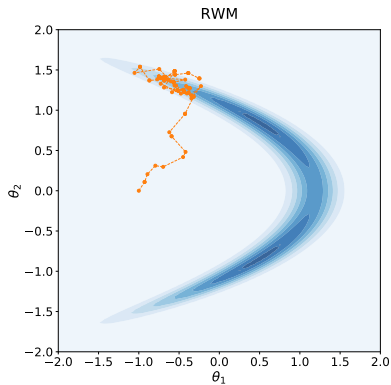
- ▶ We can incorporate the gradient information into our proposal distribution
- ▶ Let x be the current state, instead of using a random perturbation centered at x (e.g., $\mathcal{N}(x, \sigma^2)$), we can shift toward the gradient direction which leads to the following proposal distribution

$$Q(x'|x) = \mathcal{N}\left(x + \frac{\sigma^2}{2} \nabla \log p(x), \sigma^2 I\right)$$

This looks like GD with noise!

- ▶ No longer symmetric, use Metropolis-Hasting instead
- ▶ This is called **Metropolis Adjusted Langevin Algorithm (MALA)**





- ▶ It turns out that we can combine multiple MALA together, resulting in an algorithm that can generate distant proposals with high acceptance rate
- ▶ The new algorithm is based on Hamiltonian dynamics, a system introduced by Alder and Wainwright (1959) to simulate motion of molecules deterministically based on Newton's law of motion
- ▶ In 1987, Duane et al. combine the standard MCMC and the Hamiltonian dynamics, and derived a method they called *Hybrid Monte Carlo* (HMC)
- ▶ Nowadays, this abbreviation has also been used for Hamiltonian Monte Carlo

- ▶ Construct a landscape with *potential energy* $U(x)$

$$p(x) \propto e^{-U(x)}, \quad U(x) = -\log P(x)$$

- ▶ Introduce **momentum** r carrying *kinetic energy* $K(r) = \frac{1}{2}r^T M^{-1}r$, and define **total energy or Hamiltonian** $H(x, r) = U(x) + K(r)$
- ▶ **Hamiltonian equations**

$$\frac{dx}{dt} = \frac{\partial H}{\partial r}, \quad \frac{dr}{dt} = -\frac{\partial H}{\partial x}$$

- ▶ Some physics:
 - ▶ The two equations are about **velocity** and **force**, respectively.
 - ▶ Frictionless ball rolling $(x, r) \rightarrow (x', r')$ satisfies $H(x', r') = H(x, r)$



- ▶ The joint probability of (x, r) is

$$p(x, r) \propto \exp(-H(x, r)) \propto p(x) \cdot \mathcal{N}(r|0, M)$$

- ▶ x and r are independent and r follows a Gaussian distribution
- ▶ The marginal distribution is the target distribution $p(x)$
- ▶ We then use MH to sample from the joint parameter space and x samples are collected as samples from the target distribution
- ▶ HMC is an auxiliary variable method

We follow two steps to make proposals in the joint parameter space

- ▶ Gibbs sample momentum: $r \sim \mathcal{N}(0, M)$
- ▶ Simulate Hamiltonian dynamics and flip the sign of the momentum

$$(x, r) = (x^{(0)}, r^{(0)}) \xrightarrow{\text{HD}} (x^{(t)}, r^{(t)}), \quad (x', r') = (x^{(t)}, -r^{(t)})$$

Important Properties

- ▶ **Time reversibility**: The trajectory is time reversible
- ▶ **Volume preservation**: Hamiltonian flow does not change the volume - the jacobian determinant is 1
- ▶ **Conservation of Hamiltonian**: Total energy is conserved, meaning the proposal will always be accepted

- ▶ In practice, Hamiltonian dynamics can not be simulated exactly. We need to use numerical integrators
- ▶ **Leap-frog scheme**

$$r(t + \frac{\epsilon}{2}) = r(t) - \frac{\epsilon}{2} \frac{\partial U}{\partial x}(x(t))$$

$$x(t + \epsilon) = x(t) + \epsilon \frac{\partial K}{\partial r}(r(t + \frac{\epsilon}{2}))$$

$$r(t + \epsilon) = r(t + \epsilon/2) - \frac{\epsilon}{2} \frac{\partial U}{\partial x}(x(t + \epsilon))$$

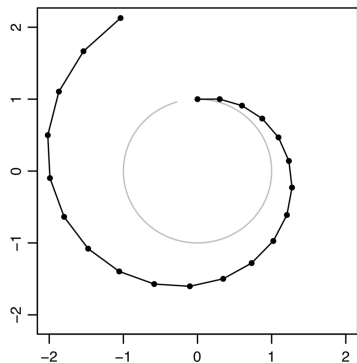
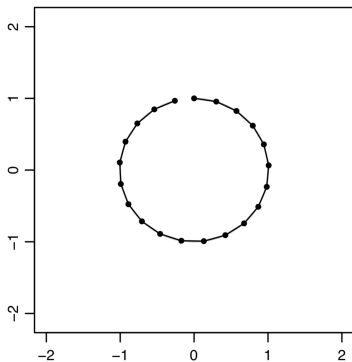
Important Properties

- ▶ **Reversibility and volume preservation:** still hold
- ▶ **Conservation of Hamiltonian:** broken. Acceptance probability becomes

$$a(x', r' | x, r) = \min(1, \exp(-H(x', r') + H(x, r)))$$



$$H(x, r) = \frac{x^2}{2} + \frac{r^2}{2}$$

Euler, $\epsilon = 0.3$ Leap-frog, $\epsilon = 0.3$ 

Adapted from Neal (2011)

HMC in one iteration

- ▶ Sample momentum $r \sim \mathcal{N}(0, M)$
- ▶ Run numerical integrators (e.g., leapfrog) for L steps
- ▶ Accept new position with probability

$$\min(1, \exp(-H(x', r') + H(x, r)))$$

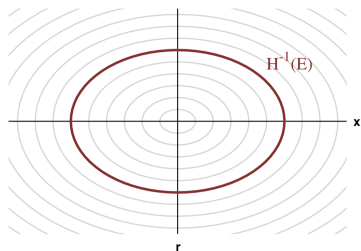
HMC in one iteration

- ▶ Sample momentum $r \sim \mathcal{N}(0, M)$
- ▶ Run numerical integrators (e.g., leapfrog) for L steps
- ▶ Accept new position with probability

$$\min(1, \exp(-H(x', r') + H(x, r)))$$

- ▶ Since Hamiltonian is conserved, every Hamiltonian trajectory is confined to an energy level set

$$H^{-1}(E) = \{x, r \mid H(x, r) = E\}$$

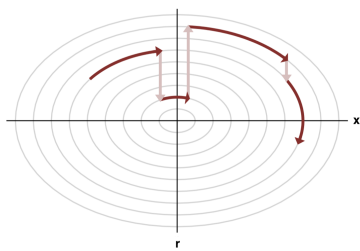
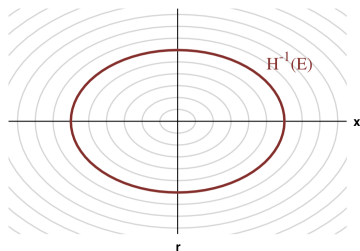


Adapted from Betancourt (2017)



- ▶ Since Hamiltonian is conserved, every Hamiltonian trajectory is confined to an energy level set

$$H^{-1}(E) = \{x, r \mid H(x, r) = E\}$$



Adapted from Betancourt (2017)

- ▶ The choice of the conditional probability distribution over the momentum, or equivalently, the kinetic energy, affects HMC's behavior over different energy level sets
- ▶ Ideally, the kinetic energy will interact with the target distribution to ensure that the energy level sets are uniformly distributed
- ▶ In HMC, we often use Euclidean-Gaussian kinetic energy $K(r) = \frac{r^T r}{2}$. This sets $M = I$ and completely ignore local geometric information of the target distribution
- ▶ Preconditioning mass matrix may help, but it is also quite limited
- ▶ Instead of using a fixed M , how about using an **adaptive** one?



- ▶ Consider the symmetric KL divergence between two densities p and q

$$D_{\text{KL}}^{\text{S}}(p||q) = D_{\text{KL}}(p||q) + D_{\text{KL}}(q||p)$$

- ▶ Let $p(y|x)$ be the likelihood. Then $D_{\text{KL}}^{\text{S}}(p(y|x + \delta x)||p(y|x))$ is approximately

$$\delta x^T \mathbb{E}_{y|x} (\nabla_x \log p(y|x) \nabla_x \log p(y|x)^T) \delta x = \delta x^T G(x) \delta x$$

where $G(x)$ is the **Fisher Information** matrix

- ▶ This induces a **Riemannian manifold** (Amari 2000) over the parameter space of a statistical model, which defines the **natural geometric structure** of density $p(x)$

- ▶ Based on the Riemannian manifold formulation, Girolami and Calderhead (2011) introduce a new method, called **Riemannian manifold HMC** (RMHMC)
- ▶ Hamiltonian on a Riemannian manifold

$$H(x, r) = U(x) + \frac{1}{2} \log((2\pi)^d |G(x)|) + \frac{1}{2} r^T G(x)^{-1} r$$

- ▶ The joint probability is

$$p(x, r) \propto \exp(-H(x, r)) \propto p(x) \cdot \mathcal{N}(r|0, G(x))$$

- ▶ x and r now are correlated, and the conditional distribution of r given x follows a Gaussian distribution
- ▶ The marginal distribution is the target distribution



- ▶ The resulting dynamics is non-separable, so instead of the standard leapfrog we need to use the *generalized* leapfrog method (Leimkuhler and Reich, 2004)
- ▶ **The generalized leapfrog scheme**

$$r(t + \frac{\epsilon}{2}) = r(t) - \frac{\epsilon}{2} \nabla_x H(x(t), r(t + \frac{\epsilon}{2}))$$

$$x(t + \epsilon) = x(t) + \frac{\epsilon}{2} (G(x(t)))^{-1} + G(x(t + \epsilon))^{-1} r(t + \frac{\epsilon}{2})$$

$$r(t + \epsilon) = r(t + \frac{\epsilon}{2}) - \frac{\epsilon}{2} \nabla_x H(x(t + \epsilon), r(t + \frac{\epsilon}{2}))$$

- ▶ The above scheme is time reversible and volume preserving. However, the first two equations are defined implicitly (can be solved via several fixed point iterations)



- ▶ Consider a 2D banana-shaped posterior distribution as follows

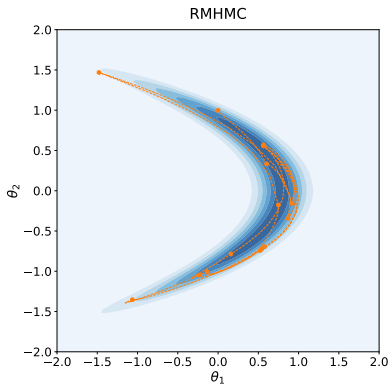
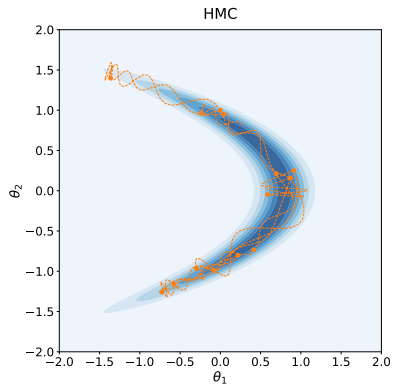
$$y_i \sim \mathcal{N}(\theta_1 + \theta_2^2, \sigma_y^2), \quad \theta = (\theta_1, \theta_2) \sim \mathcal{N}(0, \sigma_\theta^2 I)$$

- ▶ the log-posterior is (up to an ignorable constant)

$$\log p(\theta|Y, \sigma_y^2, \sigma_\theta^2) = -\frac{\sum_i (y_i - \theta_1 - \theta_2^2)^2}{2\sigma_y^2} - \frac{\theta_1^2 + \theta_2^2}{2\sigma_\theta^2}$$

- ▶ Fisher information for the joint likelihood

$$G(\theta) = \mathbb{E}_{Y|\theta} (-\nabla_\theta^2 \log p(Y, \theta)) = \frac{n}{\sigma_y^2} \begin{bmatrix} 1 & 2\theta_2 \\ 2\theta_2 & 4\theta_2^2 \end{bmatrix} + \frac{1}{\sigma_\theta^2} I$$



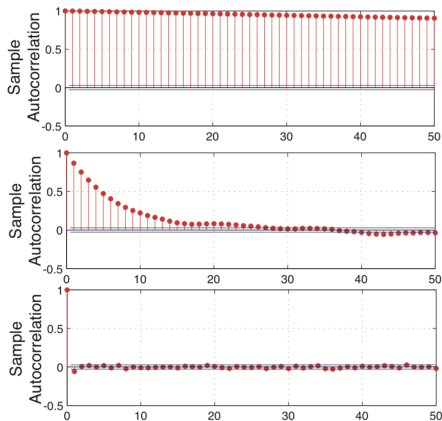
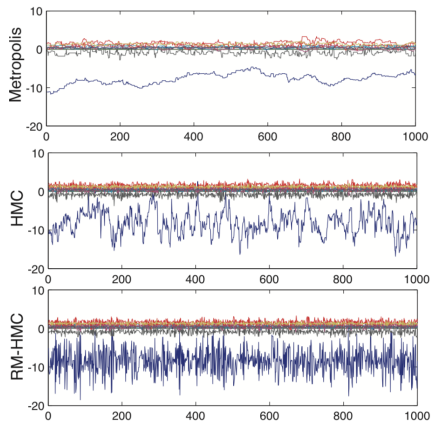
- ▶ Consider a Bayesian logistic regression model with design matrix X and regression coefficients $\beta \in \mathbb{R}^d$, with a simple prior $\beta \sim \mathcal{N}(0, \alpha I_d)$
- ▶ Neglecting constants, the log-posterior is

$$\begin{aligned}\log p(\beta|X, Y, \alpha) &= L(\beta) - \frac{1}{2\alpha}\beta^T\beta \\ &= \beta^T X^T Y - \sum_i \log(1 + \exp(x_i^T \beta)) - \frac{1}{2\alpha}\beta^T\beta\end{aligned}$$

- ▶ Use the joint likelihood to compute the fisher information

$$G(\beta) = \mathbb{E}_{Y|X, \beta, \alpha} \left(-\nabla_{\beta}^2 L(\beta) + \frac{1}{\alpha} I_d \right) = X^T W X + \frac{1}{\alpha} I_d$$





Adapted from Girolami and Calderhead (2011)



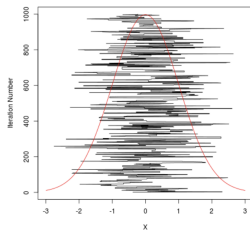
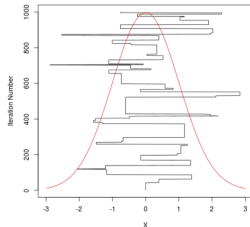
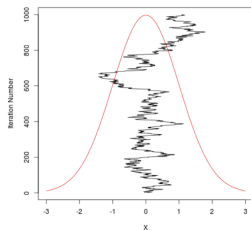
- ▶ Integration time determines the exploration efficiency of Hamiltonian trajectory in each energy level set
 - ▶ Too short integration time lose the advantage of the coherent exploration of the Hamiltonian trajectory (e.g., one step HMC is equivalent to MALA)
 - ▶ Too long integration time wastes computation since trajectories are likely to return to explored regions
- ▶ The No-U-Turn Sampler (Hoffman and Gelman, 2011).
 - ▶ Idea: use the distance to the initial position as a criteria for selecting integration time - avoid U-Turn
 - ▶ Naive implementation is not time reversible. Use a strategy similar to the doubling procedure in slice sampling (Neal 2003).

- ▶ Generally speaking, the efficiency of MCMC depends on its proposal distribution, which usually involves several hyper-parameters
- ▶ Most MCMC algorithms, therefore, need tuning to be efficient and reliable in large scale applications
- ▶ However, tuning could be painful and sometimes not practical (requires computing time, human time, and typically expert knowledge, too many variables, when to stop tuning, tuning criterion not clear, etc)
- ▶ Adaptive MCMC is about tuning MCMC without human intervention
- ▶ It uses the trajectory so far to tune the sampling kernel on the fly (so it is not a Markov chain anymore)

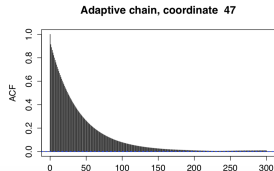
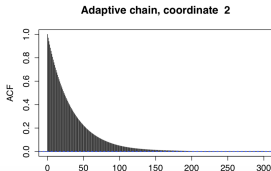
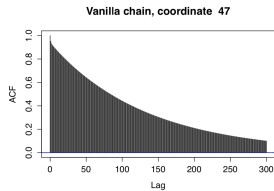
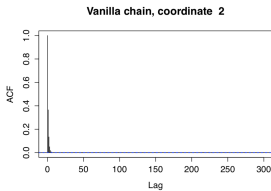
- ▶ Proposal distribution:

$$x' \sim Q_\sigma(\cdot|x) = x + \sigma\mathcal{N}(0, I_d)$$

- ▶ Plots for different σ - Goldilock's principle



- ▶ Random Scan Gibbs Sampler for 50-d Truncated Multivariate Normals. Are uniform $1/d$ selection probabilities optimal?



- ▶ First, we need a parameterized family of proposal distributions for a given MCMC class
- ▶ We also need an optimization rule that is mathematically sound and computationally cheap
- ▶ We need it to work in practice

Ergodicity of Adaptive MCMC

- ▶ How do we know that the chain will converge to the target distribution if it is not even Markovian?
- ▶ **Two conditions** (see Roberts and Rosenthal 2007):
 - ▶ *Diminishing adaption*: the dependency on earlier states of the chain goes to zero
 - ▶ *Bounded convergence*: convergence times for all adapted transition kernels are bounded in probability

- ▶ Consider random walk Metropolis for a d -dimensional target distribution with proposal $Q(x'|x_n) = \mathcal{N}(x_n, \sigma^2 \Sigma^{(n)})$
- ▶ If the target distribution is Gaussian with covariance Σ , the optimal proposal is $\mathcal{N}(x_n, \frac{2.38^2}{d} \Sigma)$, which leads to an acceptance rate $\alpha^* \approx 0.23$ (see Gelman et al 1996)
- ▶ This gives a simple criterion for random walk Metropolis in practice
- ▶ We can use it to design an **adaptive Metropolis algorithm**



- ▶ Draw proposal

$$x' \sim Q(\cdot|x_n) = x_n + \sigma_n \mathcal{N}(0, I_d)$$

- ▶ select the value x_{n+1} according to the Metropolis acceptance rate $\alpha_n = \alpha(x'|x_n)$
- ▶ Update scale by

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

where the adaptation parameter $\gamma_n \rightarrow 0$

- ▶ Optimal scaling is not the whole story. In fact, the optimal proposal suggests to learn the covariance matrix of the target distribution (e.g., use the empirical estimates)
- ▶ The algorithm runs as follows:
 - ▶ Sample a candidate value from $\mathcal{N}(x_n, \frac{2.38^2}{d}\Sigma_n)$
 - ▶ Select the value x_{n+1} as in the usual Metropolis (or MH)
 - ▶ Update the proposal distribution in two steps:

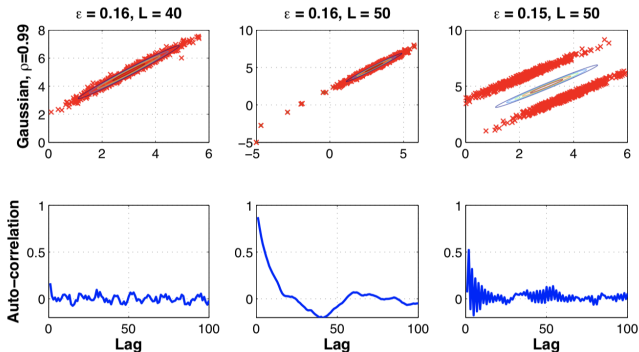
$$\mu_{n+1} = \mu_n + \gamma_{n+1}(x_{n+1} - \mu_n)$$

$$\Sigma_{n+1} = \Sigma_n + \gamma_{n+1} ((x_{n+1} - \mu_n)(x_{n+1} - \mu_n)^T - \Sigma_n)$$

where $\gamma_n \rightarrow 0$

- ▶ Many variants exist (e.g., adapting the scale, block updates, and batch adaption, etc)





- The performance of HMC would be sensitive to its hyperparameters, mainly the stepsize ϵ and trajectory length L



- ▶ Optimal acceptance rate strategy might not work well. The example shown on the previous slides all have similar acceptance rate
- ▶ Effective sample size is impractical since high order auto-correlation are hard to estimate
- ▶ Wang et al (2013) uses normalized expected squared jumping distance (ESJD)

$$\text{ESJD}_\gamma = \mathbb{E}_\gamma \|x^{(t+1)} - x^{(t)}\|^2 / \sqrt{L}$$

where $\gamma = (\epsilon, L)$

- ▶ Update γ via **Bayesian optimization**, with an annealing adapting rate

- ▶ Instead of using a fixed trajectory length L , we can sample it from some distribution (e.g., $\mathcal{U}(1, L_{\max})$)
- ▶ Split the Hamiltonian

$$H(x, r) = H_1(x, r) + H_2(x, r) + \cdots + H_k(x, r)$$

simulate Hamiltonian dynamics on each H_i (sequentially or randomly) give the Hamiltonian dynamics on H . Can save computation if some of the H_i are analytically solvable

- ▶ Partial momentum refreshment
- ▶ Acceptance using windows of states
- ▶ See Neal (2010) for more complete and detailed discussion

- ▶ Duane, S, Kennedy, A D, Pendleton, B J, and Roweth, D. Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222, 1987.
- ▶ Neal, Radford M. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 54:113–162, 2010.
- ▶ Michael Betancourt. A conceptual introduction to Hamiltonian Monte Carlo. arXiv preprint arXiv:1701.02434, 2017.
- ▶ Amari. S. and Nagaoka. H. (2000) *Methods of Information Geometry*, Oxford University Press.

- ▶ Girolami, Mark and Calderhead, Ben. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B*, 73(2):123– 214, 2011.
- ▶ Hoffman, Matthew D and Gelman, Andrew. The No-U-Turn Sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. Preprint arXiv:1111.4246, 2011.
- ▶ Leimkuhler. B. and Reich. S. (2004) *Simulating Hamiltonian Dynamics*, Cambridge University Press.
- ▶ Roberts, Gareth O. and Rosenthal, Jeffrey S. Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms. *Journal of applied probability*, 44(2):458– 475, 2007.

- ▶ Gelman, A., Roberts, G., Gilks, W.: Efficient Metropolis jumping rules. *Bayesian Statistics*, 5:599–608, 1996.
- ▶ Roberts, G.O., Gelman, A., Gilks, W.: Weak convergence and optimal scaling of random walk Metropolis algorithms. *Ann. Appl. Probab.* 7, 110–120 (1997)
- ▶ Z. Wang, S. Mohamed, and N. Freitas. Adaptive Hamiltonian and Riemann manifold Monte Carlo. In *International Conference on Machine Learning*, pages 1462–1470, 2013.