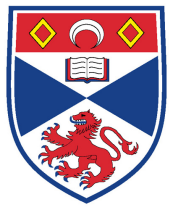
MT4531/5731: (Advanced) Bayesian Inference

The Metropolis-Hastings sampler

Nicolò Margaritella

School of Mathematics and Statistics, University of St Andrews

–



Outline

Outline

Introduction

- The main idea behind the Metropolis-Hastings (MH) sampler, is that values are drawn from some proposal distribution, (one that is relatively easy to sample from), and then "corrected" so that, after some time, they are samples from the target distribution (the posterior).
- The Metropolis-Hastings algorithm sequentially draws candidate observations from a distribution, conditional only upon the last observation, thus inducing a Markov chain.

Markov Chain properties for MH - an overview

- Assume a Markov chain with transition kernel $\mathcal{K}(\boldsymbol{\theta}, \boldsymbol{\phi})$ (intuitively, the function that gives a measure of how likely it is that the chain moves from $\boldsymbol{\theta}$ to $\boldsymbol{\phi}$)
- Assume that detailed balance is exhibited for π i.e.,

$$\pi(\boldsymbol{\theta})\mathcal{K}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \pi(\boldsymbol{\phi})\mathcal{K}(\boldsymbol{\phi}, \boldsymbol{\theta}), \tag{1}$$

- Then the chain has stationary density, $\pi(\cdot)$. (For a proof, see Question 4 in Tutorial 6.)

Outline

The Metropolis-Hastings (MH) algorithm (1)

- The proposal distribution (denoted by $q(\phi|\boldsymbol{\theta}^t)$) typically depends upon the current state of the chain. The choice of $q()$ is in principle arbitrary!
- We introduce a function $\alpha(\boldsymbol{\theta}^t, \phi)$, so that we accept the proposed ϕ, and set $\boldsymbol{\theta}^{t+1} = \phi$, with probability $\alpha(\boldsymbol{\theta}^t, \phi)$;
- ... else the chain remains at $\boldsymbol{\theta}^t$, so that $\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t$.
- It can be shown (not in this module) that the optimal acceptance function is,

$$\alpha(\boldsymbol{\theta}^t, \phi) = \min\left(1, \frac{\pi(\phi)q(\boldsymbol{\theta}^t|\phi)}{\pi(\boldsymbol{\theta}^t)q(\phi|\boldsymbol{\theta}^t)}\right).$$

Then,

$$\mathcal{K}(\boldsymbol{\theta}^t, \phi) = q(\phi|\boldsymbol{\theta}^t)\alpha(\boldsymbol{\theta}^t, \phi).$$

- For a proof that the resulting sampler satisfies detailed balance see Question 4 in Tutorial 4.

The Metropolis-Hastings (MH) algorithm (2)

- Step 1. Set an initial value for $\boldsymbol{\theta}$ denoted by $\boldsymbol{\theta}^0$.
  Step 2. Given the current $\boldsymbol{\theta} = \boldsymbol{\theta}^t$, generate a $\boldsymbol{\phi}$, from the distribution $q(\boldsymbol{\phi}|\boldsymbol{\theta})$.
  Step 3. Calculate

$$\alpha(\boldsymbol{\theta}, \boldsymbol{\phi}) = \min\left(1, \frac{\pi(\boldsymbol{\phi})q(\boldsymbol{\theta}|\boldsymbol{\phi})}{\pi(\boldsymbol{\theta})q(\boldsymbol{\phi}|\boldsymbol{\theta})}\right).$$

  Step 4. With probability $\alpha(\boldsymbol{\theta}, \boldsymbol{\phi})$, set $\boldsymbol{\theta}^{t+1} = \boldsymbol{\phi}$, else set $\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}(= \boldsymbol{\theta}^t)$.
  Step 5. Increase t by one and return to step 1 until T iterations have been performed.

Discard $\boldsymbol{\theta}^0, \ldots, \boldsymbol{\theta}^B$ as burn-in, for some suitable B, and consider $\boldsymbol{\theta}^{B+1}, \ldots, \boldsymbol{\theta}^T$ as a (dependent) sample from π.

Important notes

- We only need to know π up to proportionality, since any constants of proportionality cancel in the numerator and denominator of the calculation of α.
- The performance of the MH algorithm, is dependent on the choice of the proposal distribution q.
  - If q is chosen poorly, so that q rarely proposes values that are likely under the posterior, then the number of rejections may be high, and the efficiency of the procedure low. (High ACs)
  - If q is chosen such that only very small moves are proposed, then a high number of proposals may be accepted, but it may take a very long time for the Markov chain to properly explore the parameter space. (High ACs again!)
- Notice that the MH algorithm can be used to sample from univariate and multivariate distributions alike.

Outline

The Random Walk MH

- If the proposal is a symmetric function,

$$q(\phi|\theta) = q(\theta - \phi) = q(\phi - \theta) = q(\theta|\phi).$$

  Then, the acceptance function reduces to

$$\alpha(\theta, \phi) = \min\left(1, \frac{\pi(\phi)}{\pi(\theta)}\right). \tag{2}$$

- If $q(\phi|\theta) = q(\theta|\phi)$, then the candidate ϕ is of the form $\phi = \theta^t + \boldsymbol{z}$, where $\boldsymbol{z} \sim f$, where f is symmetric about zero. (i.e. a random walk!)
- Common choices for f, include the uniform distribution on the unit disk, or a multivariate normal or t-distribution.
- In the univariate case, a standard choice is the Normal distribution $N(0, \sigma^2)$.
- Let us see an example of a Normal Random Walk MH.

Example (1)

- Suppose that we are interested in sampling from the standard normal $N(0, 1)$ distribution.
- Certainly not a realistic example! (One can use the 'rnorm()' command in R to do this very easily.) But this example will help demonstrate some fundamentals on the behaviour of the MH algorithm.
- Assume that we choose to use a proposal distribution of the form

$$q(\phi|\theta) \sim N(\theta, \sigma^2),$$

  for some σ^2.
- Then the acceptance probability is given by...

Example (2)

- Then the acceptance probability is given by, $\alpha(\theta, \phi) = \min\{1, A\}$, where,

$$A = \frac{\pi(\phi)q(\theta|\phi)}{\pi(\theta)q(\phi|\theta)}$$

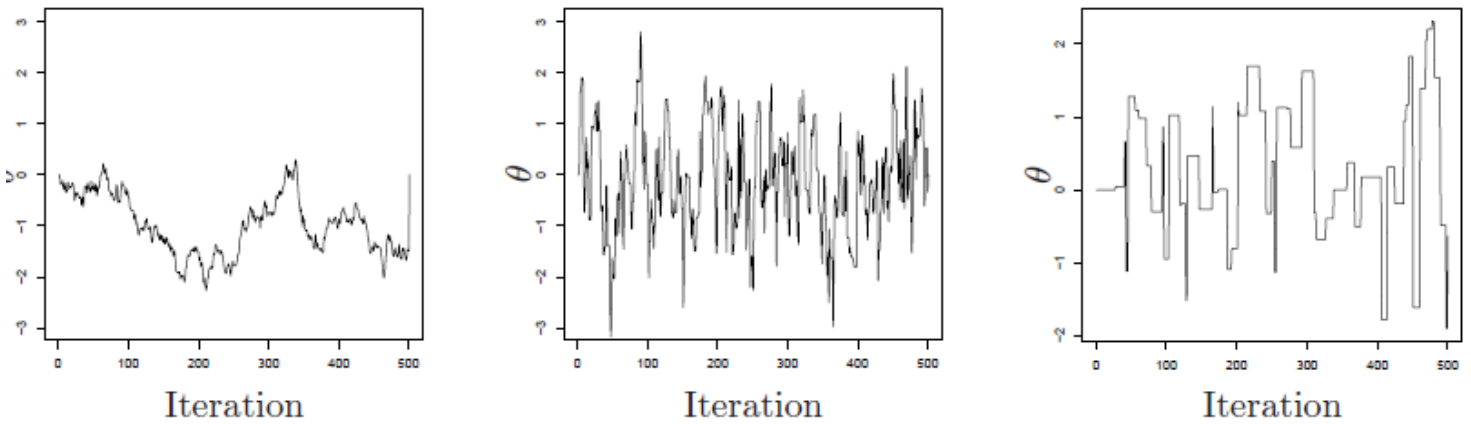
$$= \frac{\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\phi^2\right)\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2\sigma^2}(\theta-\phi)^2\right)}{\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\theta^2\right)\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2\sigma^2}(\phi-\theta)^2\right)}$$

$$= \frac{\exp\left(-\frac{1}{2}\phi^2\right)\exp\left(-\frac{1}{2\sigma^2}(\theta-\phi)^2\right)}{\exp\left(-\frac{1}{2}\theta^2\right)\exp\left(-\frac{1}{2\sigma^2}(\phi-\theta)^2\right)}$$

$$= \exp\left(-\frac{1}{2}(\phi^2-\theta^2)\right). \qquad (3)$$

Example (3)

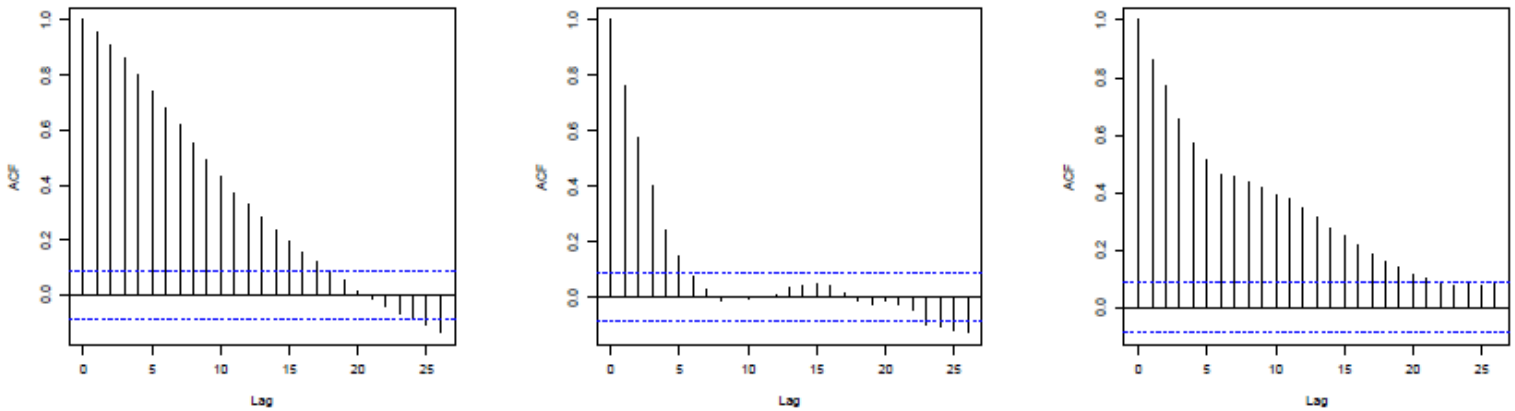
- See the lecture demonstration of the relevant R code uploaded in Moodle.
- The figure below plots the trace of the resulting output for $\sigma^2 = 0.1$, 1 and 100 respectively.



- The proposal with $\sigma^2 = 1$ provides a good sampler.
- The proposal with $\sigma^2 = 100$, generates candidate observations too far out in the tail to come from the target distribution and these are subsequently rejected.
- Conversely, the proposal with $\sigma^2 = 0.1$ generates candidates very similar to the current value (highly correlated) that are typically accepted, but it takes a long time to move over the parameter space.

Example (4)

- ACF plots for $\sigma^2 = 0.1$, $\sigma^2 = 1$ and $\sigma^2 = 100$ respectively.



- The ACF plots for both $\sigma^2 = 0.1$ and $\sigma^2 = 100$ appear similar.
- ACF plots on their own do not provide enough information on why we observe poor mixing
- This could be a result of (at least) a high rejection probability (as for $\sigma^2 = 100$) or always very small "step" sizes (as for $\sigma^2 = 0.1$).

The Independence MH sampler

- If $q(\phi|\theta) = f(\phi)$, then the candidate observation is drawn *independently* of the current state of the chain.
- This is the Independence MH sampler.
- The acceptance probability can be written as

$$\alpha(\theta, \phi) = \min\left(1, \frac{\pi(\phi)f(\theta)}{\pi(\theta)f(\phi)}\right) = \min\left(1, \frac{w(\phi)}{w(\theta)}\right),$$

where $w(\theta) = \pi(\theta)/f(\theta)$.

Outline

Single updates with the MH sampler

- The MH algorithm can update variables one-at-a-time, like Gibbs Sampler.
- Let the initial state be denoted by $\boldsymbol{\theta}^0 = (\theta_1^0, \ldots, \theta_k^0)$.
- At iteration t we cycle through each of the $\theta_1, \ldots, \theta_k$ parameters in turn.
- Consider parameter θ_p and set $\boldsymbol{\theta}_p^t = (\theta_1^{t+1}, \ldots, \theta_{p-1}^{t+1}, \theta_p^t, \theta_{p+1}^t, \ldots, \theta_k^t)$.
- Propose $\phi_p \sim q(\phi_p|\boldsymbol{\theta}_p^t)$, and set $\boldsymbol{\phi}_p = (\theta_1^{t+1}, \ldots, \theta_{p-1}^{t+1}, \phi_p, \theta_{p+1}^t, \ldots, \theta_k^t)$
- Accept the candidate value with probability $\min(1, A)$, where,

$$A = \frac{\pi(\boldsymbol{\phi}_p)q(\theta_p^t|\boldsymbol{\phi}_p)}{\pi(\boldsymbol{\theta}_p^t)q(\phi_p|\boldsymbol{\theta}_p^t)} = \frac{\pi(\phi_p|\boldsymbol{\theta}_{(p)}^t)\pi(\boldsymbol{\theta}_{(p)}^t)q(\theta_p^t|\boldsymbol{\phi}_p)}{\pi(\theta_p^t|\boldsymbol{\theta}_{(p)}^t)\pi(\boldsymbol{\theta}_{(p)}^t)q(\phi_p|\boldsymbol{\theta}_p^t)} = \frac{\pi(\phi_p|\boldsymbol{\theta}_{(p)}^t)q(\theta_p^t|\boldsymbol{\phi}_p)}{\pi(\theta_p^t|\boldsymbol{\theta}_{(p)}^t)q(\phi_p|\boldsymbol{\theta}_p^t)}.$$

  where $\boldsymbol{\theta}_{(p)}^t = \boldsymbol{\phi}_{(p)} = (\theta_1^{t+1}, \ldots, \theta_{p-1}^{t+1}, \theta_{p+1}^t, \ldots, \theta_k^t)$. This simplification is obtained by noting that $\pi(\boldsymbol{\phi}_p) = \pi(\phi_p|\boldsymbol{\theta}_{(p)}^t)\pi(\boldsymbol{\theta}_{(p)}^t)$ and similarly for $\pi(\boldsymbol{\theta}_p^t)$.
- If the candidate value is accepted, set $\theta_p^{t+1} = \phi_p$; else $\theta_p^{t+1} = \theta_p^t$.

Gibbs as a special case of MH

- In fact, the Gibbs Sampler is a special case of the single-update MH algorithm.
- Suppose that in the single-update MH, we break each iteration of the algorithm into k steps, and let q_j denote a proposal for candidates in the jth co-ordinate direction, so that

$$q_j(\boldsymbol{\phi}|\boldsymbol{\theta}) = \begin{cases} \pi(\phi_j|\boldsymbol{\theta}_{(j)}) & \boldsymbol{\phi}_{(j)} = \boldsymbol{\theta}_{(j)}, \quad j = 1, \ldots, k. \\ 0 & \text{else} \end{cases}$$

- The proof that for this proposal distribution (the full conditional) the proposals are always accepted is given in the lecture notes and in class.

Outline

Improving the MH efficiency

- The performance of the MH algorithm can be changed through the choice of proposal distributions.
- With the exception of the Gibbs sampler, most MCMC updates require a degree of *pilot-tuning* to obtain a chain with good mixing properties.
- In practice, this often involves adjusting the relevant proposal variances to obtain a Metropolis-Hastings acceptance rate of 20-40% (Gelman *et al*, 1996).
- This can often be achieved by implementing a pilot run, for 1000 iterations, say, calculating the mean acceptance rate for each parameter and adjusting the proposal variance accordingly to obtain a mean acceptance rate in the given interval.
- If parameters are highly correlated with each other (usually easy to assess from an initial pilot-run), multi-parameter updates can be used, proposing to update a number of parameters simultaneously. This is often referred to as *blocking*, since parameters are updated in "blocks".

Comparing the Gibbs sampler with the MH algorithm

- The Gibbs sampler can be seen as "efficient" since the parameter is always updated (i.e. with probability 1), and there are many computer programs that easily simulate from standard distributions.
- However, the MH algorithm has the advantage that it is not necessary to know (or recognise) all of the conditional distributions. We need only simulate from q, which we can choose arbitrarily.
- If q is poorly chosen, then the mixing of the chain can be slow, and the efficiency of the procedure low. (See *pilot-tuning*.)
- If parameters are highly correlated *a posteriori*, using a single-update MCMC algorithm (either Gibbs or single-update MH) will often perform poorly. This can be solved by block updates (i.e. updating more than one parameter within a single step). Block-updates can be performed within the MH, due to the arbitrary nature of the proposal distribution q (q can be any multivariate distribution, such as the multivariate Normal).
- Typically, an MCMC algorithm would update parameters with standard full conditionals via the Gibbs sampler; parameters with non-standard full conditionals are updated using MH.