Sequential Monte Carlo methods

Lecture 11 – Metropolis-Hastings

Thomas Schön, Uppsala University

2017-08-28

Outline – Lecture 11

Aim: Introduce the idea underlying Markov chain Monte Carlo and start looking at how the Metropolis Hastings algorithm can be used for Bayesian inference in dynamical systems.

Outline:

1. Summary of day 2
2. Bayesian inference
3. Markov chain Monte Carlo (MCMC)
4. Metropolis Hastings (MH) algorithm
5. Using MH for Bayesian inference in dynamical systems

Summary of day 2

Summary of day 2

Auxiliary variables u are introduced with the hope that it is simpler to sample from $\pi(x, u)$ than from $\pi(x)$.

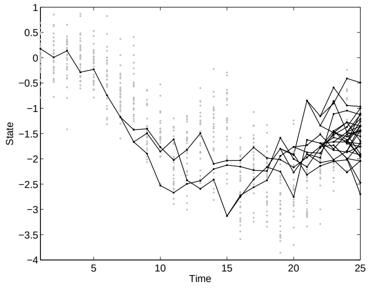
By introducing the **ancestor indices** $\{a_t^i\}_{i=1}^N$ (representing the mixture indices) as auxiliary variables within the particle filter we:

- keep the freedom in choosing our proposal $q(x_t \mid x_{t-1}, y_t)$
- at a linear computational cost!

The result is called the **auxiliary particle filter**.

The **fully adapted particle filter** makes use of locally optimal proposals both for the ancestor indices (auxiliary variables) and for the state variable.

Summary of day 2



Path degeneracy: The resampling step will by construction result in that for any time s there exists a time $t > s$ such that the PF approximation $\widehat{p}^N(x_{0:t} \mid y_{1:t})$ consists of a single particle at time s.

Maximum likelihood problem: Select the θ that according to the observed data $y_{1:T}$ is "as likely as possible" in the sense that

$$\widehat{\theta} = \arg\max_{\theta} \sum_{t=1}^{T} \log \int p(y_t \mid x_t, \theta) p(x_t \mid y_{1:t-1}, \theta) \mathrm{d}x_t$$

Summary of day 2

The particle filter likelihood estimator,

$$\widehat{Z} = \prod_{t=1}^{T} \left\{ \frac{1}{N} \sum_{i=1}^{N} \widetilde{W}_t^i \right\}$$

is a **random variable** providing an **unbiased** estimator of the likelihood $\mathbb{E}_{\psi_{N,T}}\left[\widehat{Z}\right] = p(y_{1:T})$ for any number of particles $N \geq 1$.

The distribution of **all the random variables** sampled by the bootstrap PF is,

$$\psi_{N,T}(\mathbf{x}_{0:T}, \mathbf{a}_{1:T}) = \left\{ \prod_{i=1}^{N} p(x_0^i) \right\} \prod_{t=1}^{T} \left\{ \prod_{i=1}^{N} w_{t-1}^{a_t^i} p(x_t^i \mid x_{t-1}^{a_t^i}) \right\}.$$

Executing the particle filter algorithm can be viewed as a way of generating **one sample** from this distribution!

Bayesian inference

Bayesian inference

Bayesian inference comes down to computing the target distribution $\pi(x)$.

More commonly our interest lies in some integral of the form:

$$\mathbb{E}_{\pi}[\varphi(x) \mid y_{1:T}] = \int \varphi(x) p(x \mid y_{1:T}) \mathrm{d}x.$$

Ex. (nonlinear dynamical systems)

Here our interest is often $x = \theta$ and $\pi(\theta) = p(\theta \mid y_{1:T})$

or $x = (x_{1:T}, \theta)$ and $\pi(x_{1:T}, \theta) = p(x_{1:T}, \theta \mid y_{1:T})$.

We keep the development general for now and specialize later.

How?

The two main strategies for the Bayesian inference problem:

1. **Variational methods** provides an approximation by assuming a certain functional form containing unknown parameters, which are found using optimization, where some distance measure is minimized.
2. **Markov chain Monte Carlo (MCMC)** works by simulating a Markov chain which is designed in such a way that its stationary distribution coincides with the target distribution.

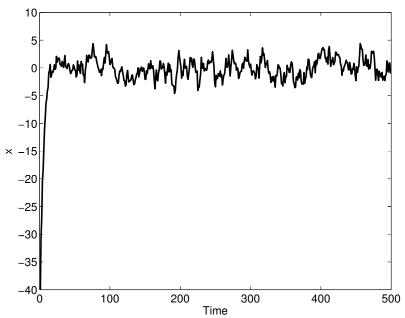
Markov chain Monte Carlo

Toy illustration – AR(1)

Let us play the game where you are asked to generate samples from

$$\pi(x) = \mathcal{N}\left(x \,\middle|\, 0, 1/(1-0.8^2)\right).$$

One realisation from $X[t+1] = 0.8X[t] + V[t]$ where $V[t] \sim \mathcal{N}(0,1)$. Initialise in $X[0] = -40$.

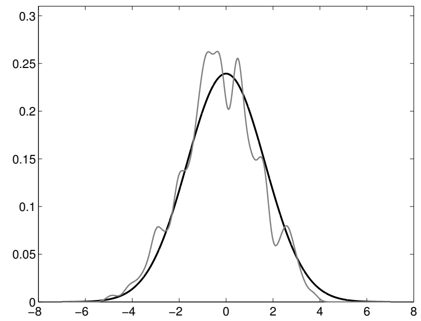


This will eventually generate samples from the following **stationary distribution**:

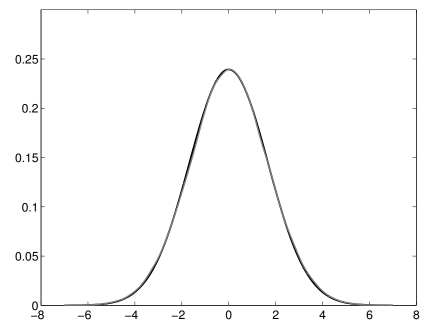
$$p^s(x) = \mathcal{N}\left(x \,\middle|\, 0, 1/(1/(1-0.8^2)\right)$$

as $t \to \infty$.

Toy illustration – AR(1)



1 000 samples



100 000 samples

The true stationary distribution is shown in black and the empirical histogram obtained by simulating the Markov chain $X[t+1] = 0.8X[t] + V[t]$ is plotted in gray.

The initial 1 000 samples are discarded (burn-in).

Metropolis Hastings algorithm

Algorithm 1 Metropolis Hastings (MH)

1. **Initialize:** Set the initial state of the Markov chain $x[1]$.
2. **For** $m = 1$ **to** M**, iterate:**
   a. Sample $x' \sim q(x \mid x[m])$.
   b. Sample $u \sim \mathcal{U}[0, 1]$.
   c. Compute the acceptance probability
   $$\alpha = \min\left(1, \frac{\pi(x')}{\pi(x[m])} \frac{q(x[m] \mid x')}{q(x' \mid x[m])}\right)$$
   d. Set the next state $x[m+1]$ of the Markov chain according to
   $$x[m+1] = \begin{cases} x' & u \leq \alpha \\ x[m] & \text{otherwise} \end{cases}$$

MH – bimodal Gaussian

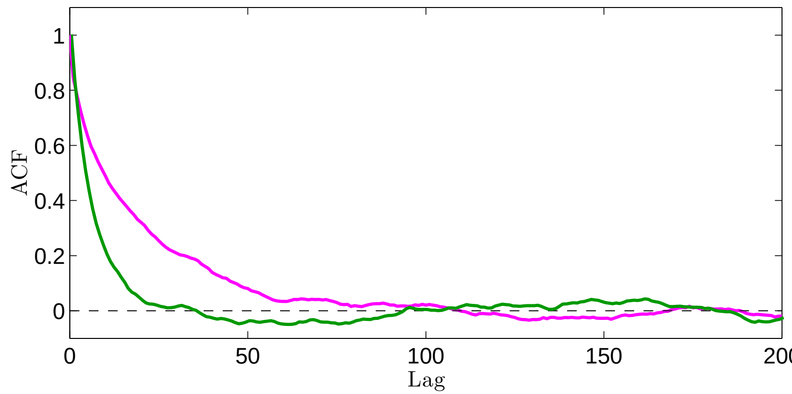
Statistical properties of MCMC

The MCMC estimator
$$\widehat{I}[\varphi] = \frac{1}{M} \sum_{m=0}^{M} \varphi(\theta[m])$$
is by the **ergodic theorem** known to be strongly consistent, i.e.
$$\underbrace{\frac{1}{M} \sum_{m=0}^{M} \varphi(\theta[m])}_{\widehat{I}[\varphi]} \xrightarrow{a.s.} \underbrace{\int \varphi(\theta) p(\theta \mid y_{1:T})}_{I[\varphi]}$$
when $M \to \infty$.

Central limit theorem (CLT) stating that
$$\sqrt{M}\left(\widehat{I}[\varphi] - I[\varphi]\right) \xrightarrow{d} \mathcal{N}(0, \sigma^2_{MCMC})$$
when $M \to \infty$.

Diagnostic tool – autocorrelation function (ACF)

The autocorrelation between two states x^m and x^{m+l} (for some positive lag l) of a Markov chain is defined as the correlation between x^m and x^{m+l}.



Using MH for Bayesian inference in dynamical systems

Bayesian parameter inference in SSMs

Full probabilistic model of a nonlinear parametric SSM:

$$p(x_{1:T}, \theta, y_{1:T}) = \underbrace{p(y_{1:T} \mid x_{1:T}, \theta)}_{\text{data distribution}} \underbrace{p(x_{1:T}, \theta)}_{\text{prior}}$$

$$= \underbrace{\prod_{t=1}^{T} \underbrace{p(y_t \mid x_t, \theta)}_{\text{observation}}}_{\text{data distribution}} \underbrace{\prod_{t=1}^{T-1} \underbrace{p(x_{t+1} \mid x_t, \theta)}_{\text{dynamics}} \underbrace{p(x_1 \mid \theta)}_{\text{state}} \underbrace{p(\theta)}_{\text{param.}}}_{\text{prior}}$$

Bayesian **parameter** inference amounts to computing

$$p(\theta \mid y_{1:T}) = \frac{p(y_{1:T} \mid \theta)p(\theta)}{p(y_{1:T})}$$

or more commonly some integral of the form

$$\mathbb{E}[\varphi(\theta) \mid y_{1:T}] = \int \varphi(\theta)p(\theta \mid y_{1:T})\mathrm{d}\theta.$$

Using MH for parameter inference in a dynamical system

Algorithm 2 Metropolis Hastings (MH)

1. **Initialize:** Set the initial state of the Markov chain $\theta[1]$.
2. **For** $m = 1$ **to** M**, iterate:**
   a. Sample $\theta' \sim q(\theta \mid \theta[m])$.
   b. Sample $u \sim \mathcal{U}[0, 1]$.
   c. Compute the acceptance probability
   $$\alpha = \min\left(1, \frac{p(y_{1:T} \mid \theta')p(\theta')}{p(y_{1:T} \mid \theta[m])p(\theta[m])} \frac{q(\theta[m] \mid \theta')}{q(\theta' \mid \theta[m])}\right)$$
   d. Set the next state $\theta[m+1]$ of the Markov chain according to
   $$\theta[m+1] = \begin{cases} \theta' & u \leq \alpha \\ \theta[m] & \text{otherwise} \end{cases}$$

Important question

Problem: We cannot evaluate the acceptance probability α since the likelihood $p(y_{1:T} \mid \theta)$ is intractable.

We know that SMC provides an estimate of the likelihood.

Important question: Is it possible to use an estimate of the likelihood in computing the acceptance probability and still end up with a valid algorithm?

Valid here means that the method converges in the sense of

$$\frac{1}{M}\sum_{m=1}^{M} \varphi(\theta[m]) \xrightarrow{a.s.} \int \varphi(\theta)p(\theta \mid y_{1:T}), \quad \text{when } M \to \infty.$$

A few concepts to summarize lecture 11

Markov chain Monte Carlo (MCMC): The underlying idea is to simulate a Markov chain which is designed in such a way that its stationary distribution coincides with the target distribution.

Metropolis Hastings (MH) constructs a Markov chain with the target distribution as its stationary distribution. MH operates by first proposing a candidate sample from a proposal distribution. This candidate sample is then either accepted or rejected based on a problem-specific acceptance probability.