# 10 - Continuous MCMC - 01

Master in Foundations of Data Science

Bayesian Statistics and Probabilistic Programming

Fall 2018-2019

Josep Fortiana

Wednesday, November 28, 2018

Universitat de Barcelona

Basic Metropolis algorithm

Metropolis-Hastings algorithm

# 10 - Continuous MCMC - 01

Basic Metropolis algorithm

Metropolis–Hastings algorithm

# Setting

Statistical model with a parameter $\theta \in \Theta \subset \mathbb{R}^p$,
$n$ i.i.d. $d$-dimensional observations: $x = (x_1, \ldots, x_n)$.

*Likelihood:* $f(x_i|\theta)$, *Prior pdf:* $h(\theta)$.

*Posterior pdf:*

$$h_x(\theta) = \frac{\prod_{i=1}^n f(x_i|\theta) \cdot h(\theta)}{Z_x} = \frac{f(x,\theta)}{Z_x},$$

$Z_x$ is the *normalization constant*, marginal density of $x$, evaluated at the observed point.

# Metropolis algorithm

We construct a sequence $\{\theta^{(t)}\}_{t\in\mathbb{N}}$ of points in $\Theta$:

A trajectory of a Markov chain whose limit pdf is $h_x(\theta)$.

A step in the chain $\theta^{(t)} \to \theta^{(t+1)}$ is as follows:

# Metropolis algorithm

1. Change proposal (Sample from a candidate generation pdf)
2. Acceptance-rejection

# [1] Change proposal (Uniform)

Set a length scale $\Delta > 0$. Write $\theta = \theta^{(t)}$.

Generate $\theta' = u$, a random uniform vector in:

$$\prod_{j=1}^{p} (\theta_j - \Delta, \theta_j + \Delta), \qquad p = \dim(\theta),$$

that is $\theta'_j - \theta_j \sim \mathrm{Unif}(-\Delta, \Delta), \quad 1 \leq j \leq p$.

## [1] Change proposal (General pdf)

A *candidate generation* pdf $g$, such that: $g(-\theta) = g(\theta)$.

Then $\theta' = \theta + u$, where $u \sim g$.

By analogy with the finite case, write:

$$k(\theta'|\theta) = g(\theta' - \theta). \qquad \text{(Sort of ``matrix'')}$$

Since $g$ is a pdf,

$$\int_{\theta'} k(\theta'|\theta)\, d\theta' = 1. \qquad \text{(A ``stochastic matrix'')}$$

## Notation: Transition kernel

The function of the two variables $\theta$, $\theta'$:

$$k(\theta'|\theta)$$

is the *transition kernel.*

Here “kernel” means “a function of two variables”, sort of “matrix”.
Just as in “Kernel Learning”.

## Notation: Remark

Some books prefer:

$$k(\theta, \theta') \quad \text{(equivalent to our)} \quad k(\theta'|\theta).$$

Our motivation is to keep analogy with the matrix and conditional density-oriented notation.

$$k_{ij} = \mathrm{P}(\text{transition to state } j \,|\, \text{current state is } i).$$

($j$-th entry in row $i$)

## Metropolis algorithm - [2] Acceptance-rejection step

Generate a random indicator $I \sim \mathrm{Ber}(p)$ with:

$$p = \min\left\{ 1,\ \frac{h_x(\theta')}{h_x(\theta)} \right\},$$

- If $I = 1$, we accept the update: $\theta^{(t+1)} = \theta'$,
- If $I = 0$, we keep: $\theta^{(t+1)} = \theta$.

# Intuitive explanation

A jump to $\theta'$ is proposed. Then:

- If the target distribution density at $\theta'$ is higher we go there.
- If it is lower we go there only conditionally, with a probability proportional to the decrease in density.

## Intuitive explanation

We wander around the state space $\Theta$, and we want to go more often, and spend more time, at regions where the probability density $h_x(\theta)$ is higher.

Hence, when on a high density area we try to stay, whereas when on a low density area, we move on, perchance we will improve.

# No denominators

Since the target pdf appears only in the quotient:

$$\frac{h_x(\theta')}{h_x(\theta)},$$

$Z_x$ is NOT required. Only the joint pdf $f(x, \theta)$:

$$p = \min\left\{1, \frac{f(x, \theta')}{f(x, \theta)}\right\}.$$

## Scale in candidate generation

$\Delta$ in the uniform case, in general dispersion parameter(s) in the candidate generation pdf $g$.

Tradeoff between:

- Small $\Delta$, high acceptance probability, slow displacement in $\Theta$,
- Large $\Delta$, a swift displacement, small acceptance probability.

## Resulting Markov chain

In the continuous state space $\Theta$.

Transition kernel ("matrix"):

$$P(\theta' \mid \theta) = k(\theta' \mid \theta) \cdot \min \left\{ 1, \; \frac{h_x(\theta')}{h_x(\theta)} \right\}$$

## Detailed balance condition

Multiplying by $h_x(\theta)$,

$$h_x(\theta) \cdot P(\theta' \,|\, \theta) = k(\theta' \,|\, \theta) \cdot \min\left\{h_x(\theta), h_x(\theta')\right\},$$

By the symmetry of $k(\theta'|\theta)$ this is equal to:

$$k(\theta \,|\, \theta') \cdot \min\left\{h_x(\theta'), h_x(\theta)\right\} = h_x(\theta') \cdot P(\theta \,|\, \theta').$$

Hence it is a *time-reversible* Markov chain.

## The target pdf $h_x(\theta)$ is the limit probability

Indeed:

$$\int_{\theta\in\Theta} h_x(\theta)\cdot P(\theta'\,|\,\theta)\,d\theta = \int_{\theta\in\Theta} h_x(\theta')\cdot P(\theta\,|\,\theta')\,d\theta$$

$$= \; h_x(\theta')\cdot\int_{\theta\in\Theta} P(\theta\,|\,\theta')\,d\theta = h_x(\theta').$$

# 10 - Continuous MCMC - 01

Basic Metropolis algorithm

Metropolis-Hastings algorithm

# References

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller (1953), *Equation of State Calculations by Fast Computing Machines*, J. Chemical Physics, Vol. 21, pp. 1087–1092.

Wilfred Keith Hastings (1970), *Monte Carlo sampling methods using Markov chains and their applications*, Biometrika 57, 97-109.

## Description

A generalization, with a non-symmetric candidate proposal kernel $k(\cdot \mid \cdot)$.

The acceptation rule is modified to compensate.

Now we accept $\theta^{(m+1)} = \theta'$ with probability:

$$\min\left\{ 1,\ \frac{h_x(\theta') \cdot k(\theta \mid \theta')}{h_x(\theta) \cdot k(\theta' \mid \theta)} \right\},$$

# Description

The transition kernel is:

$$P(\theta' \mid \theta) = k(\theta' \mid \theta) \cdot \min \left\{ 1, \frac{h_x(\theta') \cdot k(\theta \mid \theta')}{h_x(\theta) \cdot k(\theta' \mid \theta)} \right\}.$$

## Detailed balance condition

Multiplying by $h_x(\theta)$,

$$h_x(\theta) \cdot P(\theta' \,|\, \theta) = k(\theta' \,|\, \theta) \cdot \min \left\{ h_x(\theta), \frac{h_x(\theta') \cdot k(\theta \,|\, \theta')}{k(\theta' \,|\, \theta)} \right\}$$

$$= \min \{ h_x(\theta) \cdot k(\theta' \,|\, \theta), h_x(\theta') \cdot k(\theta \,|\, \theta') \}$$

$$= h_x(\theta') \cdot P(\theta \,|\, \theta'),$$

hence the chain is reversible with respect to $h_x(\theta)$, the stationary distribution.

## Construction of $k(\cdot\,,\cdot)$

Based on a random walk.

From $x$, the *proposed* $y$ is equal to $x$ plus a random

$$z = y - x,$$

generated following a pdf $g$.

$$k(x, y) = g(z) = g(y - x).$$

When $g$ is a symmetric pdf, we recover the Metropolis algorithm, where $k(\cdot\,,\cdot)$ is a symmetric kernel.