

# The Metropolis–Hastings Algorithm

---

Juan F. Rubio-Ramírez

Emory University

# Metropolis–Hastings Algorithm Definition

Let

$$P_{MH}(x, dy) = p_{MH}(x, y)dy + r_{MH}(x)\Upsilon_{\{x\}}(dy)$$

where

$$p_{MH}(x, y) = p(x, y)\alpha(x, y),$$

and

$$\alpha(x, y) = \min\left\{\frac{f(y)p(y, x)}{f(x)p(x, y)}, 1\right\},$$

and  $p(x, y)$  is a density.

# Stationarity Condition

## Corollary

$$\int_A f(y) dy = \int_{\mathbb{R}^d} P_{MH}(x, A) f(x) dx.$$

## Proof.

We only need to show that

$$f(x)p_{MH}(x, y) = f(y)p_{MH}(y, x).$$

Assume without loss of generality that  $\alpha(x, y) < 1 \Rightarrow \alpha(y, x) = 1$ . Then

$$f(x)p_{MH}(x, y) = f(x)p(x, y)\alpha(x, y) = f(y)p(y, x) = f(y)p_{MH}(y, x).$$

□

## How Does it Work?

1. Initialize the algorithm with an arbitrary value  $x_0$  and  $N$ .
2. Set  $j = 1$ .
3. Generate  $x_j^*$  from  $p(x_{j-1}, x_j^*)$  and  $u \sim \text{Uniform}[0, 1]$ .
4. If  $u \leq \alpha(x_{j-1}, x_j^*)$ , set  $x_j = x_j^*$ ; otherwise  $x_j = x_{j-1}$ .
5. If  $j \leq N$ , set  $j \leftarrow j + 1$  and go to step 3.

## Remarks on Metropolis–Hastings

- We need to evaluate a function  $g(x) \propto f(x)$ .
- The algorithm is defined by  $p(x, y)$ .
- If the candidate is rejected, the current value becomes the next one.

# Uphill and Downhill Moves

Assume  $p(y, x) = p(x, y)$ :

- If the jump is “uphill”  $\left(\frac{f(y)}{f(x)} > 1\right)$ , it is always accepted.
- If it goes “downhill”  $\left(\frac{f(y)}{f(x)} < 1\right)$ , it is accepted with nonzero probability.

**Hence:**

$$\alpha(x, y) = 1 \Rightarrow p_{MH}(x, y) = p(x, y) \Rightarrow r_{MH}(x) = 0,$$

$$\alpha(x, y) < 1 \Rightarrow p_{MH}(x, y) < p(x, y) \Rightarrow r_{MH}(x) > 0.$$

## Choosing $p_{MH}(x, y)$

- Let  $p_1(z)$  be a multivariate continuous density:

$$p_{MH}(x, y) = p_1(x - y).$$

- Let  $p_2(z)$  be symmetric:

$$p_{MH}(x, y) = p_2(x - y) = p_2(y - x), \quad \alpha(x, y) = \min \left\{ \frac{f(y)}{f(x)}, 1 \right\}.$$

- Let  $p_3(z)$  be a multivariate continuous density:

$$p_{MH}(x, y) = p_3(y).$$

## Example: Approximating $\Phi(t)$

- Use MH to approximate  $F(t)$ , the standard normal CDF evaluated at  $t$ .
- Generate  $\{x_j\}_{j=1}^N$  from  $N(0, 1)$  using MH.

### Algorithm:

- 1 Initialize  $x_0 = 0$ , choose  $N$ , and set  $j = 1$ .
- 2 Generate  $x_j^* = x_{j-1} + N(0, 2.5)$ .
- 3 Compute  $\alpha(x_{j-1}, x_j^*) = \min\{\phi(x_j^*)/\phi(x_{j-1}), 1\}$ .
- 4 Draw  $u \sim \text{Uniform}[0, 1]$ .
- 5 If  $u \leq \alpha$ , set  $x_j = x_j^*$ ; else  $x_j = x_{j-1}$ .
- 6 If  $j \leq N$ , repeat from step 2.

$$F(t) \approx \frac{1}{N} \sum_{i=1}^N \mathbb{I}_{\{x_i < t\}}(x_i).$$

At what speed does the chain converge? How long should it run?

- Run multiple chains with different initial values and compare within/between variation.
- Check serial correlation of draws.
- Let  $N$  grow with the serial correlation.
- Run  $N$  chains of length  $M$  with random starts; use the last value of each chain.

If a candidate is rejected, the current value is retained. **Acceptance rate matters:**

- For normal models, the acceptance rate should be between 23% and 45% (Roberts, Gelman, and Gilks, 1994).

## M–H Example: AR(2) Model

Simulate  $T$  observations from:

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \epsilon_t = \mathbf{w}_t' \boldsymbol{\Phi} + \sigma \epsilon_t,$$

where  $\phi_1 = 1$ ,  $\phi_2 = -0.5$ ,  $\epsilon_t \sim N(0, 1)$ ,  $\mathbf{w}_t = (y_{t-1}, y_{t-2})'$  and  $\boldsymbol{\Phi} = (\phi_1, \phi_2)'$ . The stationarity conditions are:

$$\mathcal{S} = \{\boldsymbol{\Phi} : \phi_1 + \phi_2 < 1, -\phi_1 + \phi_2 < 1, \phi_2 > -1\}$$

and the data is

$$\mathbf{Y} = (y_3, \dots, y_T).$$

# Likelihood, Priors, and Posterior

Likelihood (conditional on  $y_1, y_2$ ):

$$\ell(\mathbf{Y}|\mathbf{\Phi}, \sigma, y_1, y_2) = (\sigma^2)^{-(T-2)/2} \exp\left(-\frac{1}{2\sigma^2} \sum_{t=3}^T (y_t - \mathbf{w}_t' \mathbf{\Phi})^2\right).$$

Priors:

$$\mathbf{\Phi} \in \mathcal{S} \text{ and } \sigma \in \mathbb{R}_+.$$

Posterior:

$$\pi(\mathbf{\Phi}, \sigma | \mathbf{Y}, y_1, y_2) \propto \ell(\mathbf{Y}|\mathbf{\Phi}, \sigma, y_1, y_2) 1_{\mathcal{S}}(\mathbf{\Phi}) 1_{\mathbb{R}_+}(\sigma).$$