p-dimensional Gaussian

Metropolis-Hastings

Hedibert F. Lopes

2026-05-19

Contents

---

1 Target Distribution

Consider the p-dimensional Gaussian target

$$\boldsymbol{\theta} = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_p \end{pmatrix} \sim \mathcal{N}_p(\mathbf{0}_p, \, \boldsymbol{\Sigma}), \quad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & \rho & \rho & \cdots & \rho \\ \rho & 1 & \rho & \cdots & \rho \\ \vdots & & \ddots & & \vdots \\ \rho & \rho & \cdots & \rho & 1 \end{pmatrix}, \quad \rho = 0.5.$$

The log-density (up to a constant) is $\log p(\boldsymbol{\theta}) = -\frac{1}{2}\boldsymbol{\theta}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\theta}$.

2 Single-move random walk Metropolis-Hastings

Initialise $\boldsymbol{\theta}^{(0)} = \mathbf{0}_p$;

At iteration $i-1$ and for each component $j = 1, \ldots, p$:

1. Propose $\theta_j^* \sim \mathcal{N}\left(\theta_j^{(i-1)}, \, v\right)$, keeping all other components fixed.
2. Compute the acceptance probability

$$\alpha = \min\left\{1, \; \frac{\exp\left(-\dfrac{1}{2}\boldsymbol{\theta}^{*\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{\theta}^*\right)}{\exp\left(-\dfrac{1}{2}\boldsymbol{\theta}^{(i-1)\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{\theta}^{(i-1)}\right)}\right\}.$$

3. Set $\theta_j^{(i)} = \theta_j^*$ with probability α, else $\theta_j^{(i)} = \theta_j^{(i-1)}$.

3 Block-move random walk Metropolis-Hastings

Initialise $\boldsymbol{\theta}^{(0)} = \mathbf{0}_p$;

At iteration $i-1$:

1. Propose the **full vector** $\boldsymbol{\theta}^* \sim \mathcal{N}_p\Big(\boldsymbol{\theta}^{(i-1)},\, v\,\mathbf{I}_p\Big)$.
2. Compute the acceptance probability

$$\alpha = \min\left\{1,\ \frac{\exp\left(-\frac{1}{2}\boldsymbol{\theta}^{*\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{\theta}^*\right)}{\exp\left(-\frac{1}{2}\boldsymbol{\theta}^{(i-1)\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{\theta}^{(i-1)}\right)}\right\}.$$

3. Set $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^*$ with probability α, else $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^{(i-1)}$.

4 R Implementation

```
# --- Target: N_p(0, Sigma) with equicorrelation Sigma ---
p   <- 10
rho <- 0.5

Sigma    <- matrix(rho, nrow = p, ncol = p)
diag(Sigma) <- 1
SigmaInv <- solve(Sigma)

# Log-density of N_p(0, Sigma) up to constant
log_target <- function(theta) {
  -0.5 * as.numeric(t(theta) %*% SigmaInv %*% theta)
}

# MCMC settings
burnin <- 10000
lag    <- 100
M_mcmc <- 1000
niter  <- burnin + lag * M_mcmc
v      <- 0.5         # proposal variance
```

4.1 Single-move

```
theta       <- rep(0, p)     # starting value theta^(0) = 0_p
thetas.draw <- NULL
accept_s    <- 0

for (iter in 1:niter) {
  for (j in 1:p) {                  # update each component in turn
    theta.star    <- theta
    theta.star[j] <- rnorm(1, theta[j], sqrt(v))
    log_alpha     <- log_target(theta.star) - log_target(theta)
    if (log(runif(1)) < log_alpha) {
      theta    <- theta.star
      accept_s <- accept_s + 1
    }
```

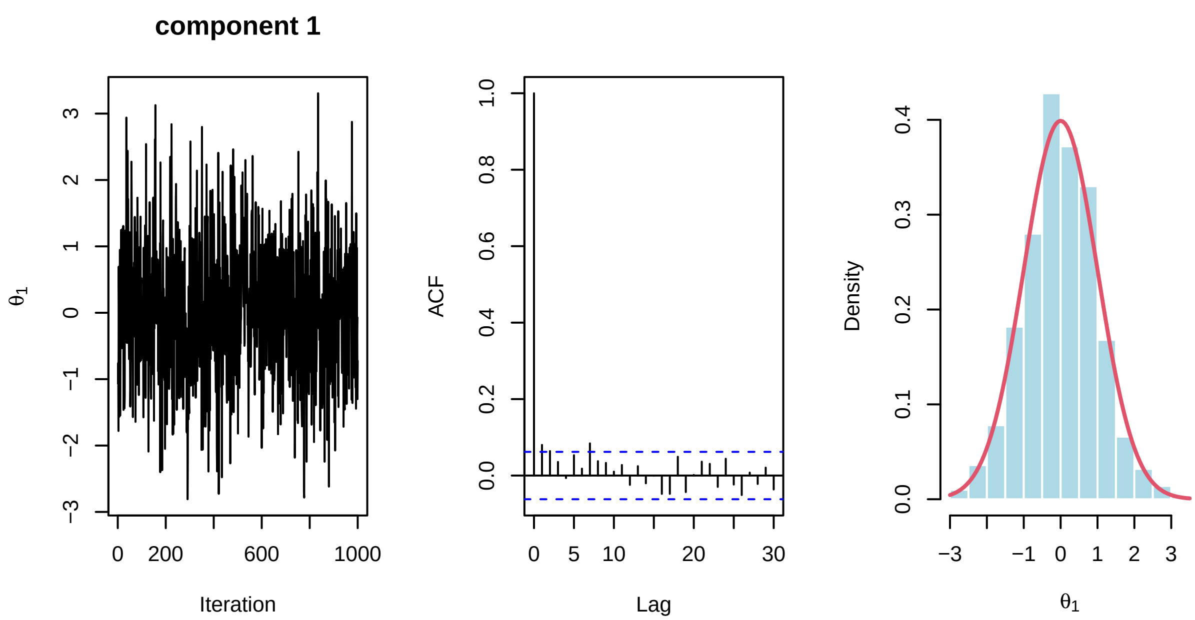
```
  }
  thetas.draw <- rbind(thetas.draw, theta)
}

# Burn-in and thinning
idx         <- seq(burnin + 1, niter, by = lag)
thetas.draw <- thetas.draw[idx, ]

cat("Single-move acceptance rate:", round(accept_s / (niter * p), 3), "\n")
```

```
## Single-move acceptance rate: 0.716
```

```
par(mfrow = c(1, 3))
ts.plot(thetas.draw[, 1],
        xlab = "Iteration", ylab = expression(theta[1]),
        main = "component 1")
acf(thetas.draw[, 1], main = "")
hist(thetas.draw[, 1], freq = FALSE, col = "lightblue", border = "white",
     xlab = expression(theta[1]),
     main = "")
curve(dnorm(x, 0, sqrt(Sigma[1, 1])), add = TRUE, col = 2, lwd = 2)
```



4.2 Block-move

```
theta        <- rep(0, p)   # starting value theta^(0) = 0_p
thetas.draw1 <- NULL
accept_b     <- 0

for (iter in 1:niter) {
  theta.star <- rnorm(p, theta, sqrt(v))          # propose full vector
  log_alpha  <- log_target(theta.star) - log_target(theta)
  if (log(runif(1)) < log_alpha) {
    theta    <- theta.star
    accept_b <- accept_b + 1
```

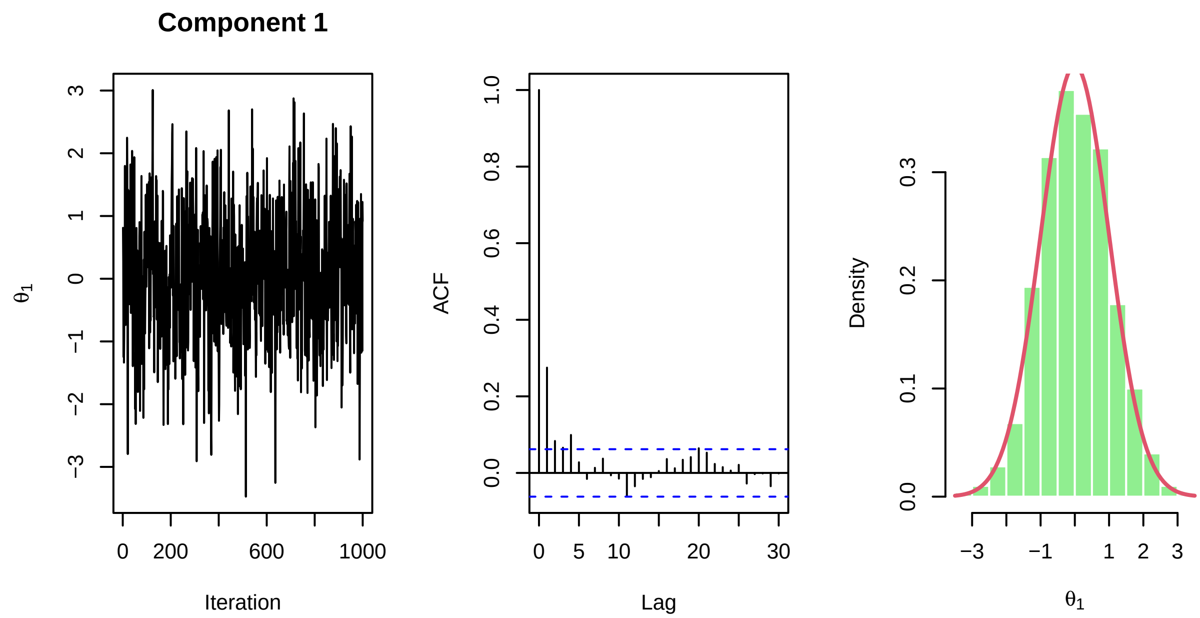
```r
  }
  thetas.draw1 <- rbind(thetas.draw1, theta)
}

# Burn-in and thinning
thetas.draw1 <- thetas.draw1[idx, ]

cat("Block-move acceptance rate:", round(accept_b / niter, 3), "\n")
```

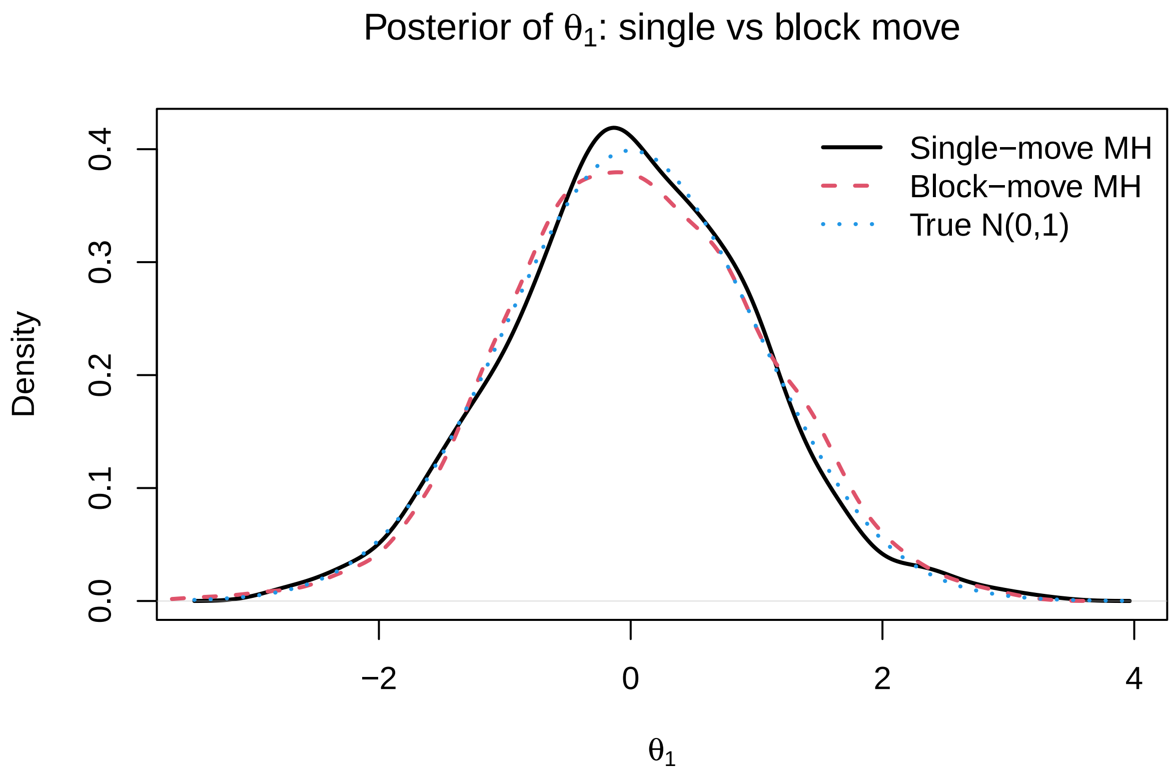
```
## Block-move acceptance rate: 0.165
```

```r
par(mfrow = c(1, 3))
ts.plot(thetas.draw1[, 1],
        xlab = "Iteration", ylab = expression(theta[1]),
        main = "Component 1")
acf(thetas.draw1[, 1], main = "")
hist(thetas.draw1[, 1], freq = FALSE, col = "lightgreen", border = "white",
     xlab = expression(theta[1]),
     main = "")
curve(dnorm(x, 0, sqrt(Sigma[1, 1])), add = TRUE, col = 2, lwd = 2)
```



4.2.1 Comparison: Single-Move vs Block-Move

```r
par(mfrow = c(1, 1))
plot(density(thetas.draw[, 1]),
     type = "l", lwd = 2, col = 1,
     xlab = expression(theta[1]), ylab = "Density",
     main = expression(paste("Posterior of ", theta[1], ": single vs block move")))
lines(density(thetas.draw1[, 1]), col = 2, lwd = 2, lty = 2)
curve(dnorm(x, 0, sqrt(Sigma[1, 1])), add = TRUE,
      col = 4, lwd = 2, lty = 3)
legend("topright",
       legend = c("Single-move MH", "Block-move MH", "True N(0,1)"),
       col = c(1, 2, 4), lty = c(1, 2, 3), lwd = 2, bty = "n")
```



Posterior of θ_1: single vs block move
Single−move MH
Block−move MH
True N(0,1)
Density
0.0
0.1
0.2
0.3
0.4
−2
0
2
4
θ_1