

MCMC

Mahtab Nahayati

2024-12-30

Contents

Introduction	3
Problem Setup	3
Gibbs Sampler Implementation	3
Metropolis-Hasting Algorithm with Block-Wise Update	3
Chain Diagnostics	4
Autocorrelation Analysis	7

Introduction

This document implements a Gibbs sampler and a Metropolis-Hastings algorithm with block-wise updates for sampling from a bivariate normal distribution. Chain diagnostics will also be performed to evaluate the efficiency of the samplers.

Problem Setup

The standard bivariate normal distribution is defined by:

$$p(\theta_1, \theta_2) = \frac{1}{2\pi\sqrt{1-\rho^2}} e^{-\frac{1}{2(1-\rho^2)}(\theta_1^2 - 2\rho\theta_1\theta_2 + \theta_2^2)}$$

We use $\rho = 0.5$, chain size $M = 30000$, and initialize with non-informative priors.

Gibbs Sampler Implementation

```
# Set parameters
rho <- 0.5
M <- 30000
burn_in <- 5000 # Burn-in period
theta1 <- numeric(M)
theta2 <- numeric(M)

# Initialize values
theta1[1] <- 0
theta2[1] <- 0

# Gibbs sampler
for (i in 2:M) {
  # Sample from conditional of theta1 given theta2
  theta1[i] <- rnorm(1, mean = rho * theta2[i - 1], sd = sqrt(1 - rho^2))

  # Sample from conditional of theta2 given theta1
  theta2[i] <- rnorm(1, mean = rho * theta1[i], sd = sqrt(1 - rho^2))
}

# Store results in a dataframe and discard burn-in
gibbs_samples <- data.frame(
  theta1 = theta1[-(1:burn_in)],
  theta2 = theta2[-(1:burn_in)]
)
```

Metropolis-Hasting Algorithm with Block-Wise Update

```
# Proposal density standard deviation
proposal_sd <- 0.5

# Initialize values
theta <- matrix(0, nrow = M, ncol = 2)
acceptance <- 0
```

```

# Metropolis-Hastings sampler
for (i in 2:M) {
  # Propose new values
  proposal <- theta[i - 1, ] + rnorm(2, mean = 0, sd = proposal_sd)

  # Compute the log acceptance ratio
  log_accept_ratio <- (-0.5 / (1 - rho^2)) *
    ((proposal[1]^2 - 2 * rho * proposal[1] * proposal[2] + proposal[2]^2) -
     (theta[i - 1, 1]^2 - 2 * rho * theta[i - 1, 1] * theta[i - 1, 2] + theta[i - 1, 2]^2))

  # Accept or reject
  if (log(runif(1)) < log_accept_ratio) {
    theta[i, ] <- proposal
    acceptance <- acceptance + 1
  } else {
    theta[i, ] <- theta[i - 1, ]
  }
}

# Store results in a dataframe and discard burn-in
mh_samples <- data.frame(
  theta1 = theta[-(1:burn_in), 1],
  theta2 = theta[-(1:burn_in), 2]
)
cat("Acceptance rate:", acceptance / M)

```

```
## Acceptance rate: 0.7286667
```

Chain Diagnostics

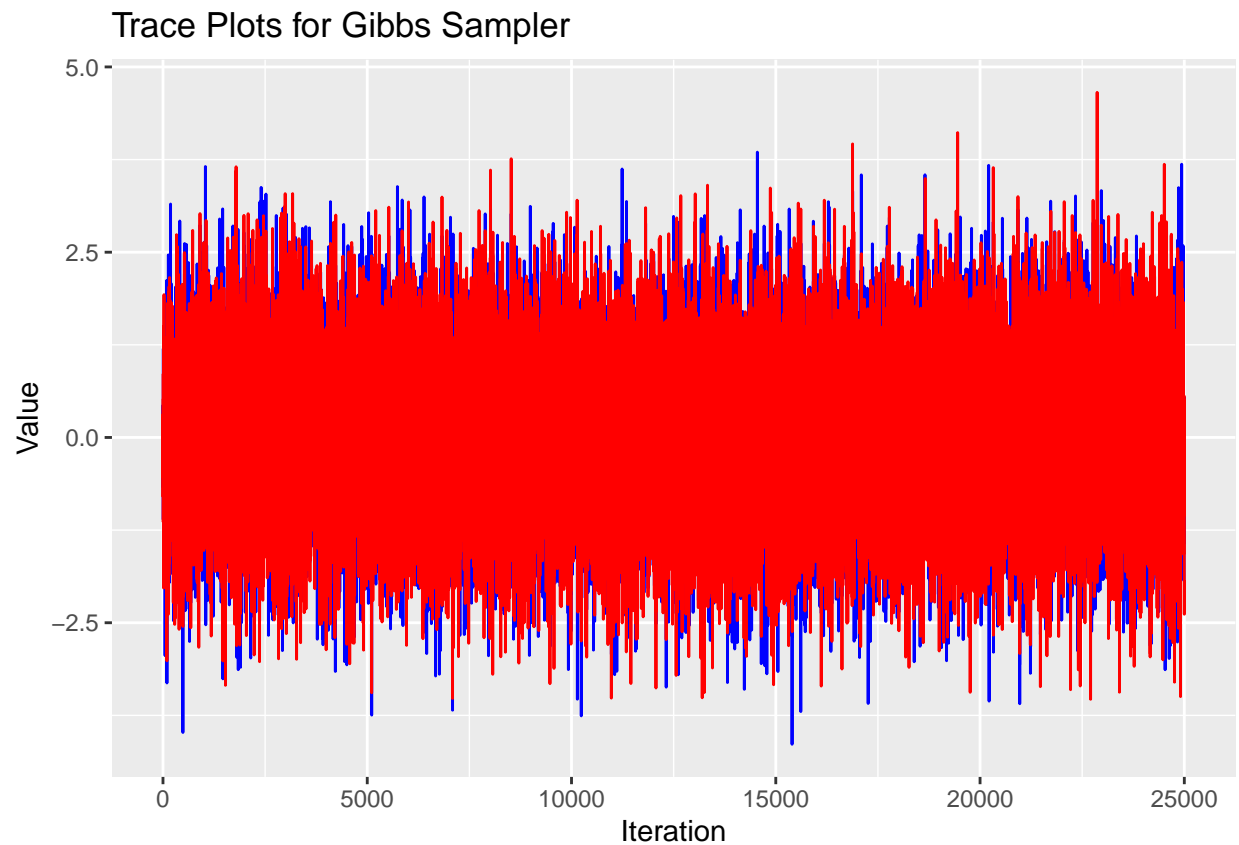
```
library(ggplot2)
```

```
## Warning: Paket 'ggplot2' wurde unter R Version 4.3.3 erstellt
```

```

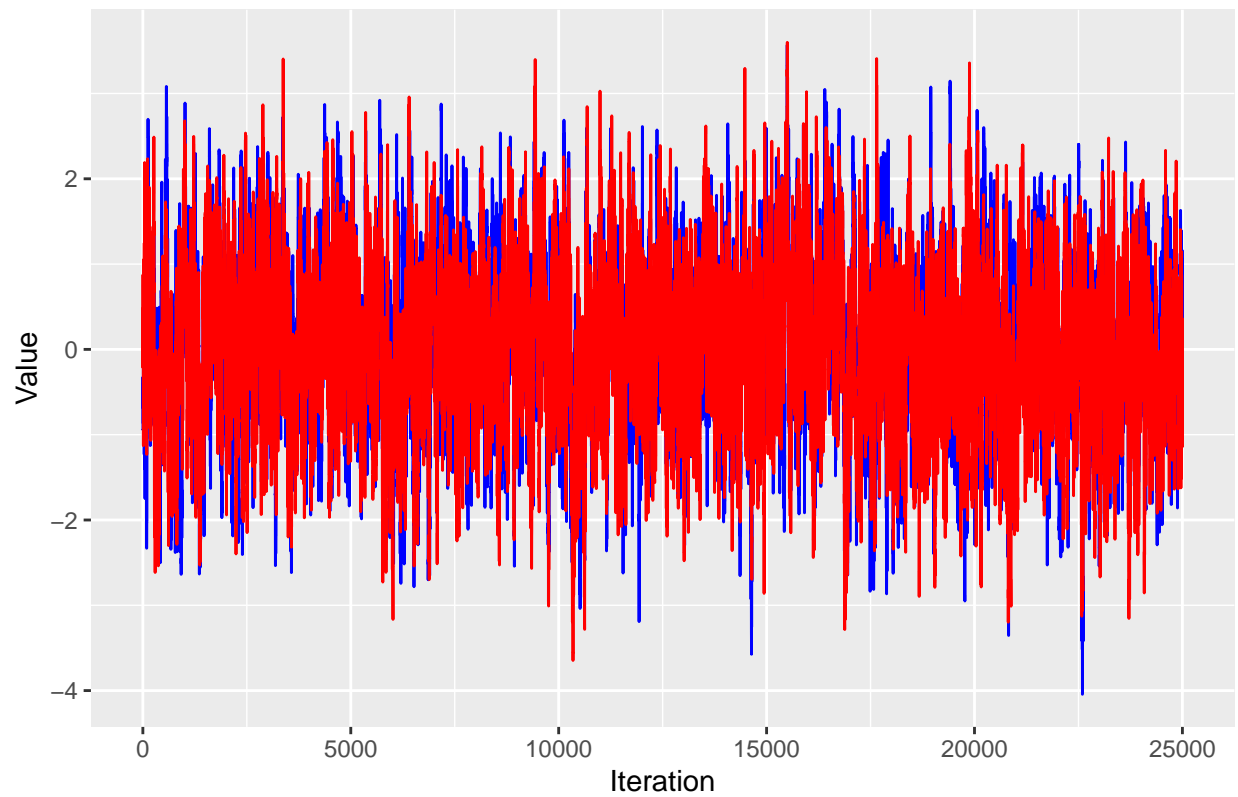
# Gibbs Sampler Trace Plot
ggplot(gibbs_samples, aes(x = seq_len(nrow(gibbs_samples)))) +
  geom_line(aes(y = theta1), color = "blue") +
  geom_line(aes(y = theta2), color = "red") +
  labs(title = "Trace Plots for Gibbs Sampler", x = "Iteration", y = "Value")

```

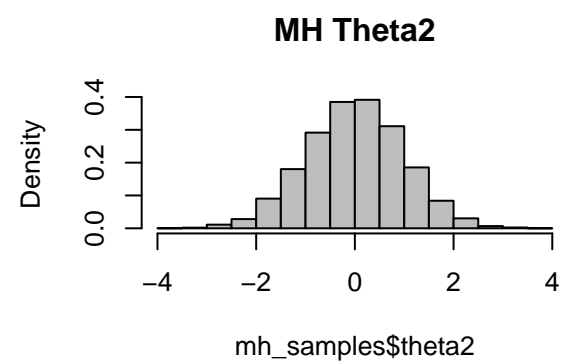
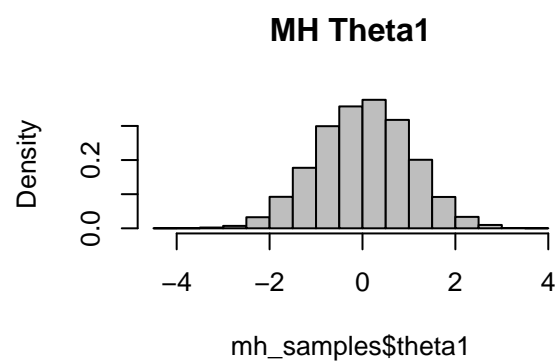
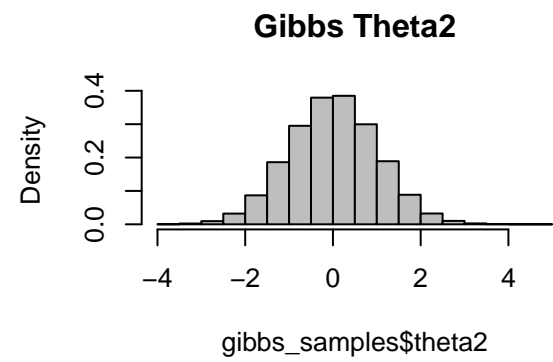
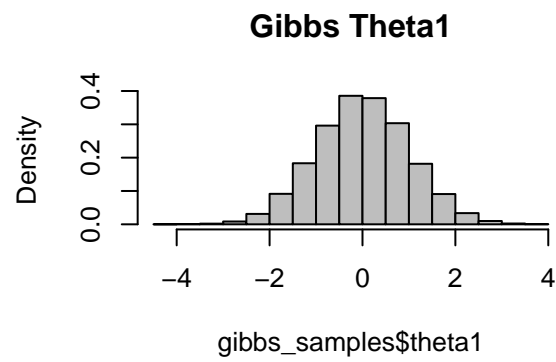


```
# Metropolis-Hastings Trace Plot  
ggplot(mh_samples, aes(x = seq_len(nrow(mh_samples)))) +  
  geom_line(aes(y = theta1, color = "blue")) +  
  geom_line(aes(y = theta2, color = "red")) +  
  labs(title = "Trace Plots for Metropolis-Hastings", x = "Iteration", y = "Value")
```

Trace Plots for Metropolis–Hastings



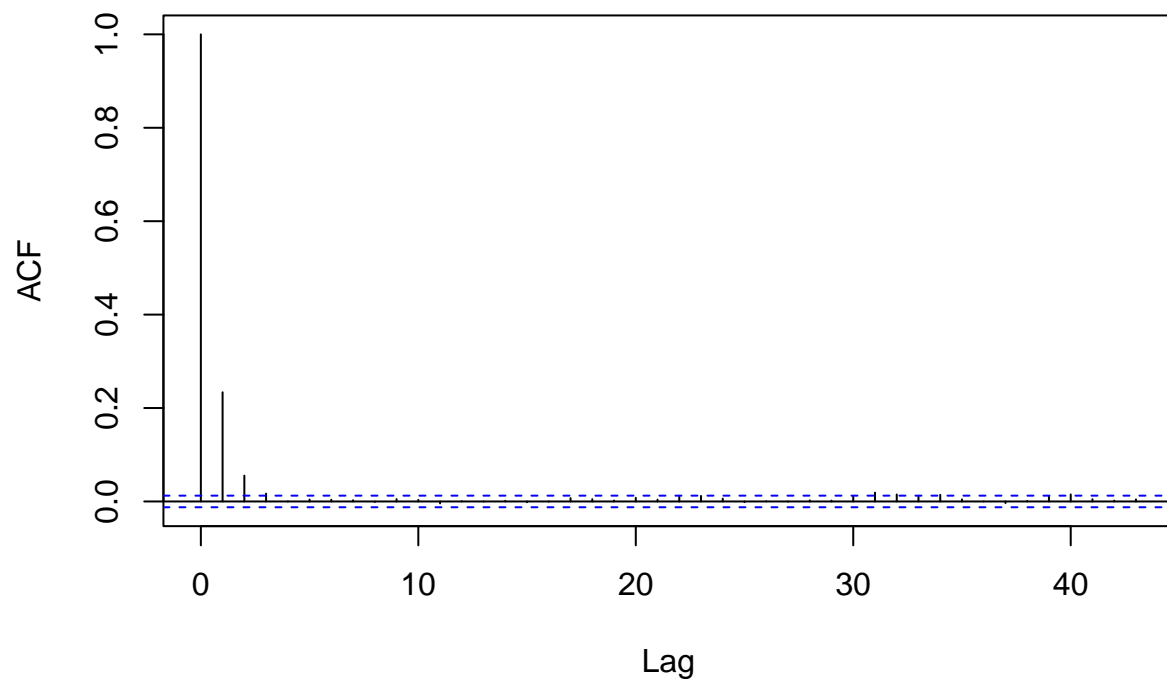
```
par(mfrow = c(2, 2))
hist(gibbs_samples$theta1, freq = FALSE, main = "Gibbs Theta1", col = "grey")
hist(gibbs_samples$theta2, freq = FALSE, main = "Gibbs Theta2", col = "grey")
hist(mh_samples$theta1, freq = FALSE, main = "MH Theta1", col = "grey")
hist(mh_samples$theta2, freq = FALSE, main = "MH Theta2", col = "grey")
```



Autocorrelation Analysis

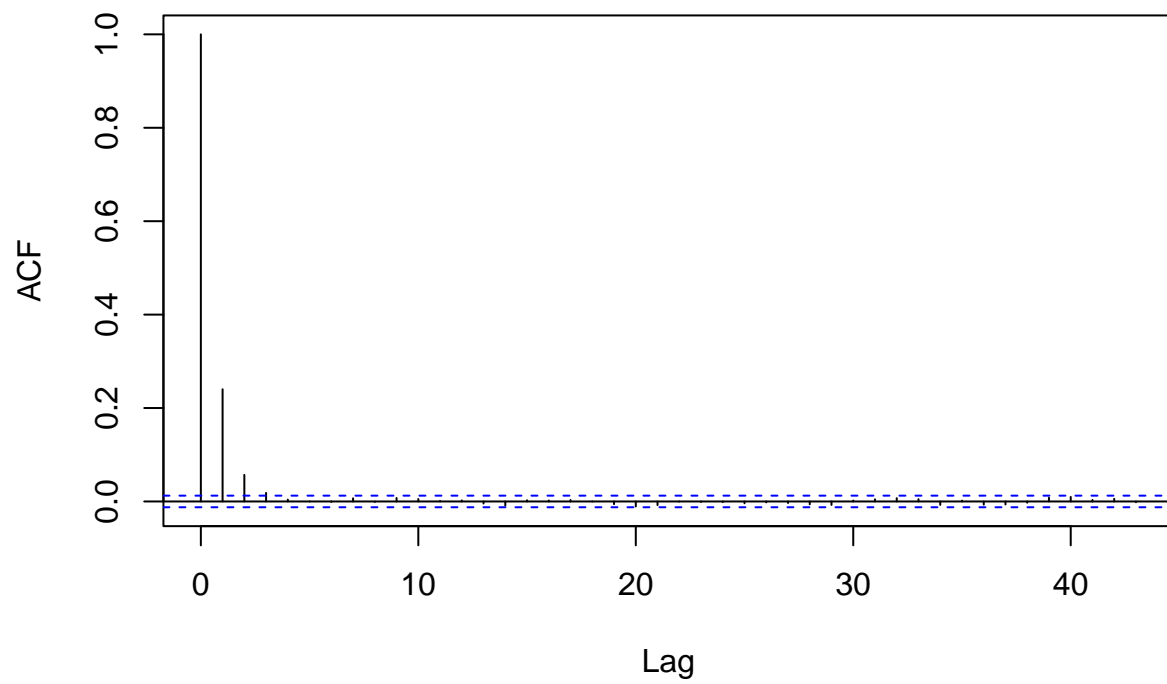
```
acf(gibbs_samples$theta1, main = "ACF Gibbs Theta1")
```

ACF Gibbs Theta1



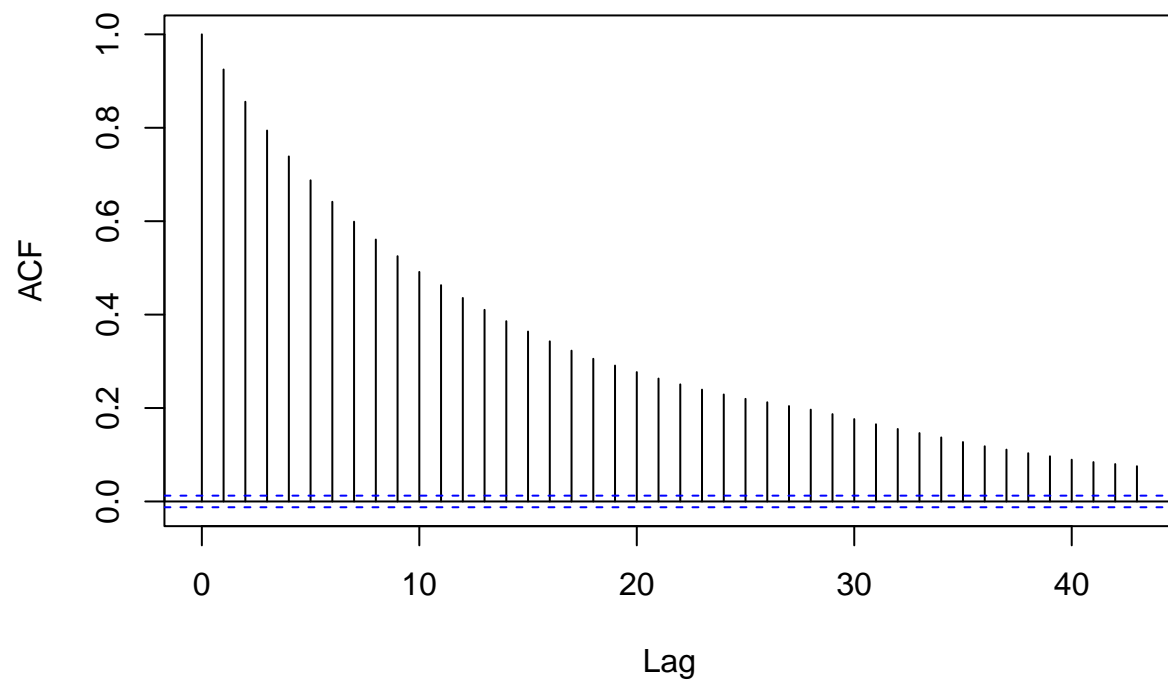
```
acf(gibbs_samples$theta2, main = "ACF Gibbs Theta2")
```


ACF Gibbs Theta2



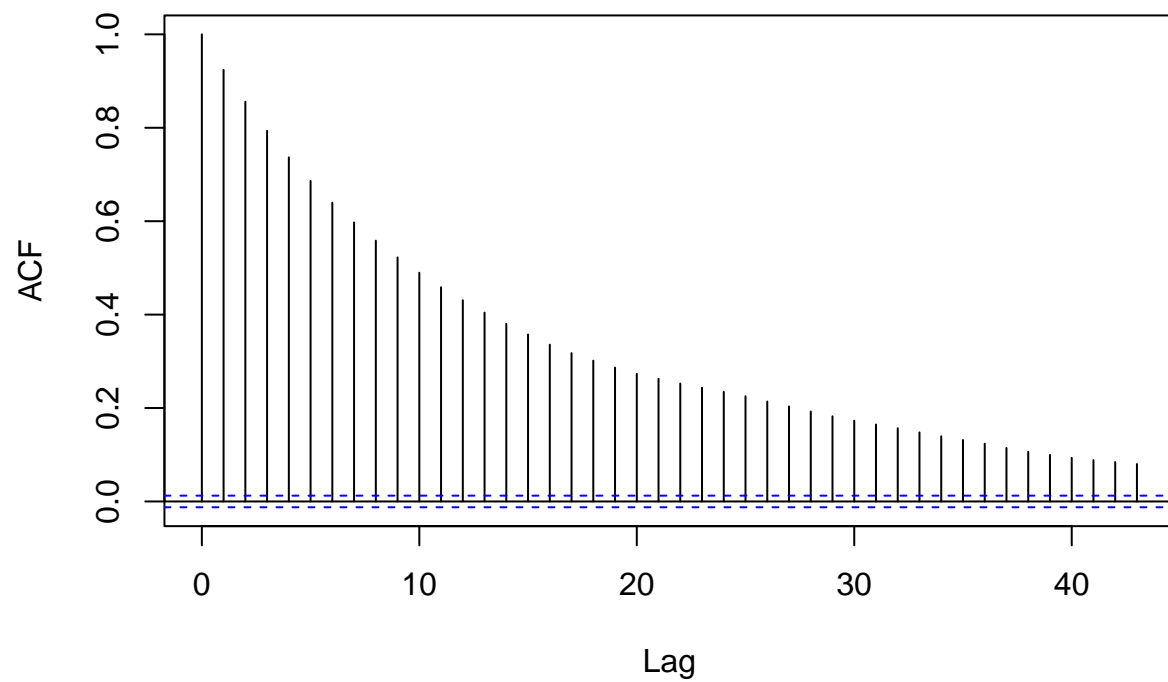
```
acf(mh_samples$theta1, main = "ACF MH Theta1")
```

ACF MH Theta1



```
acf(mh_samples$theta2, main = "ACF MH Theta2")
```

ACF MH Theta2



Discussion

Convergence: Both samplers converge after discarding the burn-in period. *Efficiency:* The Gibb sampler demonstrates better mixing (lower autocorrelation) than Metropolis-Hastings. *Choice of M :* The chain size $M=30,000$ was sufficient to achieve convergence and independence of samples after burn-in.