

```

y_dot_bar <- c(28,8,-3,7,-1,1,18,12)

sigma_j <- c(15,10,16,11,9,11,10,18)

J <- 8

mu <- 10

tau <- 2

iteration <- 5000

M <- diag(1,8)

epsilon <- 0.01

L <- 100

accept <- 0


alpha <- array(NA,dim=c(J,iteration))

phi <- array(NA,dim=c(J,iteration))  # alpha와 phi는 dimension이 동일


library(mvtnorm)  # for mvtnorm

gradient <- function(alpha){
  -(alpha - y_dot_bar)/sigma_j - (alpha - mu)/(tau^2)
}

alpha_hat <- ((y_dot_bar/sigma_j)+(mu/tau^2))/((1/sigma_j)+(1+tau^2))

variance_j <- 1/((1/sigma_j)+(1+tau^2))

posterior <- function(alpha){
  dmvnorm(alpha,alpha_hat,diag(variance_j))
}


alpha[,1] <- 1

phi[,1] <- 1

```

```

# leapfrog_step

# 처음은 1번째에서 무조건시작

# 중간은 2번째와 3번째 에서 랜덤하게 L-2번 반복

# 마지막은 무조건 3번째

# 대각행렬이므로 역행렬과 기존행렬이 동일

for(i in 2:iteration){

  alpha[,i] <- alpha[,i-1]

  phi[,i] <- rmvnorm(1,rep(0,J),M)

  phi[,i] <- phi[,i] + 0.5*epsilon*gradient(alpha[,i-1])

  for(j in 1:L-2){

    if(runif(1)>0.5){

      alpha[,i] <- alpha[,i] + epsilon*(M%%phi[,i])

    }

    else{

      phi[,i] <- phi[,i] + 0.5*epsilon*gradient(alpha[,i])

    }

  }

  phi[,i] <- phi[,i] + 0.5*epsilon*gradient(alpha[,i])

  num <- posterior(alpha[,i])/posterior(phi[,i])

  den <- posterior(alpha[,i-1])/posterior(phi[,i-1])

  a <- min(num/den,1)

  if(runif(1,0,1) > a){

    alpha[,i] <- alpha[,i]

  }

  else{

    alpha[,i] <- alpha[,i-1]

    accept <- accept+1

  }

}

```

마지막 acceptance ratio 부분에서 에러가 자꾸 나서

확인해보니 50번째 이후부터 발산을 하게됩니다.

해결방법을 찾으려고 했으나, 찾지못하였습니다.

죄송합니다ㅜㅜ