Homework 3

공간자료의 통계분석

2022-22742 진수정

Problem 4

(b) Starting from appropriate initial values of the parameters, generate m = 10000 MCMC samples sequentially from the posterior conditional distributions obtained in (a) using elevation data in geoR package and obtain posterior mean and posterior variance for all the parameters (β, σ2, □, v2) using the last 1000 samples by assuming the MCMC chain converges after 9000 iterations. Here, consider x(s) = (1, x1(s), x2(s)), where x1(s) is the x-coordinate at s and x2(s) is the y-coordinate at s. Also, provide histograms of MCMC samples for all the parameters. Do not use built-in functions for implementing MCMC. Here are some further specifications for implementation.

- Fix α = 1.5.
- Consider discrete priors for □ and v2. For □, π(□) ∼ 1/□ with support {0.1, 0.2, 0.3, · · · , 5}. For v2, π(v2) ∼ unif({0, 0.05, 0.1, 0.15, · · · , 5}).

```
library(geoR)
library(spatstat)
library(rockchalk)
library(MCMCpack)

data(elevation)
n = length(elevation$data)
X = cbind(rep(1,n), elevation$coords)
Z = elevation$data
```

```
# hyper parameter
sigma0 = 1000
beta0 = c(913.800731, -1.694949, -25.251291)
alpha0 = 1.5
b0 = 1

# initial value
nu = 1.4
phi = 0.1
beta = c(913.800731, -1.694949, -25.251291)
sigma = 500
```

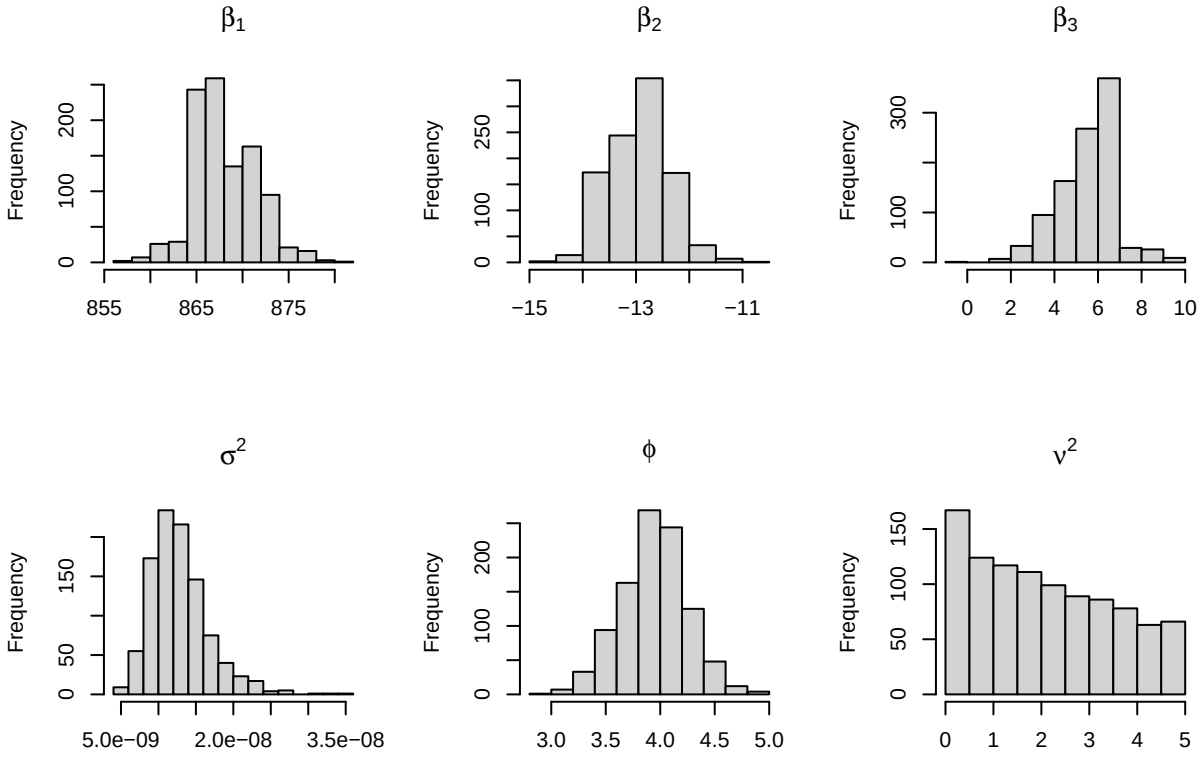
```r
# R(phi)
R = function(phi) {
  R.phi = matern(pairdist(X[,2:3]), phi, 1.5)
  return(R.phi)
}

# inverse of (R(phi) + nu^2*I)
temp = array(NA,c(n,n,50,100))
for (i in 1:50) {
  for (j in 1:100) {
    temp[,,i,j] = solve(R(i/10) + matrix(j/20,n,n))
  }
}

# MCMC
set.seed(42)
params = NULL
for (i in 1:10000) {
  # beta
  beta.sigma = solve(t(X)%*%temp[,,phi*10,nu*20]%*%X/sigma + matrix(1/sigma0,3,3))
  beta.mu = beta.sigma %*% (t(X)%*%temp[,,phi*10,nu*20]%*%Z/sigma + beta0/sigma0)
  beta = mvrnorm(1,beta.mu,beta.sigma)

  # sigma
  sigma = rinvgamma(1, n/2+alpha0, (t(Z-X%*%beta)%*%temp[,,phi*10,nu*20]%*%(Z-X%*%beta)/2 + 1/b0)^(-1))

  # phi
  prob.phi = rep(NA,50)
  scaler = t(Z-X%*%beta) %*% temp[,,1,nu*20] %*% (Z-X%*%beta) / (2*sigma)
  for (j in 1:50) {
    prob.phi[j] = (det(temp[,,j,nu*20]))^(0.5) * exp(-t(Z-X%*%beta)%*%temp[,,j,nu*20]%*%(Z-X%*%beta)/(2
  }
  phi = sample(seq(0.1,5,0.1), 1, replace = T, prob.phi)

  # nu
  prob.nu = rep(NA,50)
  scaler = t(Z-X%*%beta)%*%temp[,,phi*10,1]%*%(Z-X%*%beta)/(2*sigma)
  for (k in 1:100) {
    prob.nu[k] = (det(temp[,,phi*10,k]))^(0.5) * exp(-t(Z-X%*%beta)%*%temp[,,phi*10,k]%*%(Z-X%*%beta)/(1
  }
  nu = sample(seq(0.05,5,0.05), 1, replace = T, prob.nu)

  params = rbind(params, c(beta,sigma,phi,nu))
}
```

```r
# posterior mean and variance
cat("# ------------------- beta ------------------- #\n",
    "posterior mean of beta: ", c(mean(params[9001:10000,1]),
                                  mean(params[9001:10000,2]),
                                  mean(params[9001:10000,3])),"\n",
    "posterior variance of beta: ", c(var(params[9001:10000,1]),
                                      var(params[9001:10000,2]),
                                      var(params[9001:10000,3])),"\n\n",
```

```
    "# -------------------- sigma2 -------------------- #\n",
    "posterior mean of sigma2: ", mean(params[9001:10000,4]),"\n",
    "posterior variance of sigma2: ", var(params[9001:10000,4]),"\n\n",
    "# -------------------- phi -------------------- #\n",
    "posterior mean of phi: ", mean(params[9001:10000,5]),"\n",
    "posterior variance of phi: ", var(params[9001:10000,5]),"\n\n",
    "# -------------------- nu2 -------------------- #\n",
    "posterior mean of nu2: ", mean(params[9001:10000,6]),"\n",
    "posterior variance of nu2: ", var(params[9001:10000,6])
    )
```

```
## # -------------------- beta -------------------- #
##  posterior mean of beta:  868.0627 -12.96036 5.549155
##  posterior variance of beta:  11.72857 0.2903264 1.979464
##
##  # -------------------- sigma2 -------------------- #
##  posterior mean of sigma2:  1.278316e-08
##  posterior variance of sigma2:  1.511863e-17
##
##  # -------------------- phi -------------------- #
##  posterior mean of phi:  3.9948
##  posterior variance of phi:  0.09432729
##
##  # -------------------- nu2 -------------------- #
##  posterior mean of nu2:  2.10895
##  posterior variance of nu2:  2.050123
```

```
# histogram
par(mfrow = c(2,3))
hist(params[9001:10000,1], main = expression(beta[1]), xlab = "")
hist(params[9001:10000,2], main = expression(beta[2]), xlab = "")
hist(params[9001:10000,3], main = expression(beta[3]), xlab = "")
hist(params[9001:10000,4], main = expression(sigma^2), xlab = "")
hist(params[9001:10000,5], main = expression(phi), xlab = "")
hist(params[9001:10000,6], main = expression(nu^2), xlab = "")
```


β1
Frequency
0
100
200
855
865
875
β2
Frequency
0
100
250
−15
−13
−11
β3
Frequency
0
100
300
0
2
4
6
8
10
σ2
Frequency
0
50
150
5.0e−09
2.0e−08
3.5e−08
ϕ
Frequency
0
100
200
3.0
3.5
4.0
4.5
5.0
ν2
Frequency
0
50
100
150
0
1
2
3
4
5