Peraza_Assignment4

Joaquin Peraza

2023-02-11

Question 1

$[a] = Beta(\alpha, \beta)$

$\alpha = 2, \beta = 1$

Proposal distribution

$a \sim N(a.old, 0.0001)$

$[y|a] = Bern(a)$

$[a|y] = [y|a][a]$

$MH = \frac{[y|a][a][a.old|a]}{[y|a.old][a.old][a|a.old]}$

Given the symmetric proposal distribution:

$MH = \frac{[y|a][a]}{[y|a.old][a.old]}$

```
set.seed(1)
K <- 5000
a.init <- 0.01
sigma.tune <- 0.0001

samples <- matrix(NA,K,1)
samples[1,] <- a.init

for (k in 2:K){
  a.old <- samples[k-1,]
  a.try <- rnorm(1,a.old,sigma.tune)
  mh <- ifelse(a.try>0 & a.try<1, a.try/a.old, 0)
  keep <- ifelse(mh>1,1,rbinom(1,1,mh))
  samples[k,] <- ifelse(keep==1, a.try, a.old)
}
```

Question 2

```
mean(samples[1001:5000])
```

```
## [1] 0.01313859
```

It is the approximation of the integral $\int_0^1 [a]a\, da$.

Question 3

$[a] = Beta(\alpha, \beta)$

$\alpha = 2, \beta = 1$

Proposal distribution

$a \sim N(a.old, 0.25)$

$[y|a] = Bern(a)$

$[a|y] = [y|a][a]$

$MH = \frac{[y|a][a][a.old|a]}{[y|a.old][a.old][a|a.old]}$

Given the symmetric proposal distribution:

$MH = \frac{[y|a][a]}{[y|a.old][a.old]}$

```
K <- 5000
a.init <- 0.01
sigma.tune <- 0.25

samples <- matrix(NA,K,1)
samples[1,] <- a.init

for (k in 2:K){
  a.old <- samples[k-1,]
  a.try <- rnorm(1,a.old,sigma.tune)
  mh <- ifelse(a.try>0 & a.try<1, a.try/a.old, 0)
  keep <- ifelse(mh>1,1,rbinom(1,1,mh))
  samples[k,] <- ifelse(keep==1, a.try, a.old)
}
```

Question 4

```
mean(samples[1001:5000])
```

```
## [1] 0.6672168
```

```
samples[1000:1020]
```

```
##  [1] 0.8791529 0.8858221 0.8858221 0.7527602 0.8288427 0.8203385 0.8203385
##  [8] 0.7030665 0.8334648 0.7936760 0.9810927 0.9810927 0.9810927 0.9810927
## [15] 0.7823918 0.6819744 0.9367745 0.9811770 0.9811770 0.9525869 0.9525869
```

It is the approximation of the integral $\int_0^1 [a]a\ da$.

Question 5

$[a] = Beta(\alpha, \beta)$

$\alpha = 2, \beta = 1$

Proposal distribution

$a \sim N(a.old, 10)$

$[y|a] = Bern(a)$

$[a|y] = [y|a][a]$

$MH = \frac{[y|a][a][a.old|a]}{[y|a.old][a.old][a|a.old]}$

Given the symmetric proposal distribution:

$MH = \frac{[y|a][a]}{[y|a.old][a.old]}$

```
K <- 5000
a.init <- 0.01
sigma.tune <- 10

samples <- matrix(NA,K,1)
samples[1,] <- a.init

for (k in 2:K){
  a.old <- samples[k-1,]
  a.try <- rnorm(1,a.old,sigma.tune)
  mh <- ifelse(a.try>0 & a.try<1, a.try/a.old, 0)
  keep <- ifelse(mh>1,1,rbinom(1,1,mh))
  samples[k,] <- ifelse(keep==1, a.try, a.old)
}
```

Question 6

```
mean(samples[1001:5000])
```

```
## [1] 0.6308012
```

It is the approximation of the integral $\int_0^1 [a]a\,da$, which is equal to the $E(a) = \frac{2}{3}$.

Question 7

The tuning parameter σ^2_{tune} is used to determine the standard deviation of the proposal distribution. The proposal distribution is a normal distribution that is used to generate candidate values for the next iteration of the algorithm. Because of that σ^2_{tune} determines how wide or narrow the proposal distribution will be. If σ^2_{tune} is too small (for eg. $\sigma^2_{tune} = 0.0001$) it will led to less variance in the proposed distribution and a slower exploration of the values, however if it is too high the proposal distribution is far away from the target distribution, and the acceptance rate will be very low, leading to inefficient exploration of the posterior distribution (for eg. $\sigma^2_{tune} = 10$). A value of 0.25 for sigma.tune might result to a reasonable value for a better and more efficient approximation of the target posterior distribution.