

STAT 206 Lab 6_Lihua Xu

Due Monday, November 13, 5:00 PM

General instructions for labs: You are encouraged to work in pairs to complete the lab. Labs must be completed as an R Markdown file. Be sure to include your lab partner (if you have one) and your own name in the file. Give the commands to answer each question in its own code block, which will also produce plots that will be automatically embedded in the output file. Each answer must be supported by written statements as well as any code used.

Agenda: Accept-Reject algorithm

1. Write a function to simulate n $N(0,1)$ random variates using the Accept-Reject algorithm with a Cauchy candidate.

Here, our target density is the standard Normal distribution with PDF $f(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$. The Cauchy(0,1) random variable has the instrumental PDF

$$g(x) = \frac{1}{\pi\gamma} \left[\frac{\gamma^2}{(x - x_0)^2 + \gamma^2} \right]$$

standard PDF:

$$g(x) = \frac{1}{\pi} \frac{1}{1+x^2}, \quad -\infty < x < \infty,$$

and CDF

$$F(x) = \frac{1}{\pi} \arctan\left(\frac{x-\mu}{\sigma}\right) + \frac{1}{2}$$

Standard CDF

$$F(x) = \frac{1}{\pi} \arctan(x) + \frac{1}{2}$$

It is now easy to calculate the inverse transformation to simulate:

$$F^{-1}(y) = x = \sigma \tan(\pi(y - 1/2)) + \mu$$

We know already that Cauchy has heavier tails than the Normal distribution, suggesting that there is an M such that $Normal \leq M * Cauchy$. That is, we should find M that satisfies the following expression:

$$\frac{1}{\sqrt{2\pi}}e^{-x^2/2} \leq M \frac{1}{\pi(1+x^2)}$$

We can approximate M numerically using the following function.

$$M = \sup \frac{\frac{1}{\sqrt{2\pi}}e^{-x^2/2}}{\frac{1}{\pi} \frac{1}{1+x^2}}, \quad -\infty < x < \infty,$$

And THE maximum value is $M = 1.5203469$. This means that on the average we need to generate approximately 1.52 Cauchys before we will get a Normal.

```
Accept_Reject <- function(n){
  M <- 1.5203469
  i <- 1
  Y <- c()
  while(i < (n+1)){
    u <- runif(1)
    X <- rcauchy(1, location = 0, scale = 1)
    i <- i+1
    if (u < ((1/(sqrt(2*pi)))*exp(-X^2/2))/(M*1/pi*1/(1+X^2)))
      {Y <- c(Y,X)}
  }
  return(Y)}

```

2. Simulate 1000 $N(0, 1)$ random variates using your function to estimate $E[Y^3 \log(1 + |Y|)]$ and $Pr(Y \in [-1, 2])$. Be sure to include a Monte Carlo standard error with your estimates.

```
data <- Accept_Reject(1000)
Z <- data^3*log10(1+abs(data))
Expect <- mean(Z)
mcse <- sd(data) / sqrt(length(data))
#The expected value should be:
Expect
```

```
## [1] -0.1675298
```

```
#The Monte Carlo standard error will be:
mcse
```

```
## [1] 0.0379133
```

```
num <- length(data)
N <- 0
for (i in 1:num){
  if (-1<=data[i]&data[i]<=2){N=N+1}}
Probability <- N/num
#The probability should be:
Probability
```

```
## [1] 0.797546
```

3. What was the acceptance rate of the Accept-Reject algorithm? Is this close to the theoretical acceptance rate?

The acceptance rate resulted from my code is:

```
num/1000
```

```
## [1] 0.652
```

And the theoretical acceptance rate is:

```
M <- 1.5203469
1/M
```

```
## [1] 0.6577446
```

They are very close to each other.

4. Write a function that continues simulation until the sample size is large enough so that your Monte Carlo error is less than $\epsilon = 0.01$ for estimating a general statistic `stat` (which will be a function). Your function should also return the observed acceptance rate of the Accept-Reject algorithm and a Monte Carlo standard error.

```
stat <- function(n){
  M <- 1.5203469
  i <- -1
  Y <- c()
  value <- c()
  for(m in 1:n){
    u <- runif(1)
    X <- rcauchy(1,location = 0,scale = 1)
    i <- 1+i
    if (u < ((1/(sqrt(2*pi)))*exp(-X^2/2))/(M*1/pi*1/(1+X^2))))
    {Y <- c(Y,X)}
```

```

    if ((length(Y)>=10)&(sd(Y)/(sqrt(length(Y)))<=0.01))
    {break}}
List <- list(accpt_rate=length(Y)/i,data=Y,mcse=sd(Y)/sqrt(length(Y)),total_times=i)
return(List)}

```

5. Use your function to estimate $E[Y^3 \log(1 + |Y|)]$ and $Pr(Y \in [-1, 2])$ with $\epsilon = 0.01$. Report your estimates along with confidence intervals based on the Monte Carlo standard error. What was the acceptance rate?

*#n (# of times) is a very big value. Once the tolerance 0.01 was achieved, it will stop.
#The acceptance rate and the MCSE would be as the following values respectively:*

```

OUTPUT <- stat(1000000)
OUTPUT$accpt_rate

```

```
## [1] 0.6604726
```

```
OUTPUT$mcse
```

```
## [1] 0.009999308
```

```
OUTPUT$total_times #The total time they simulate is less than n=1000000.
```

```
## [1] 15445
```

#Estimation of the expected value and the probability

```

data_new <- OUTPUT$data
Z_new <- data_new^3*log10(1+abs(data_new))
Expect <- mean(Z_new)
#The expected value would be:
Expect

```

```
## [1] -0.003746866
```

```

C_interval <- Expect + c(-1,1)*1.96*OUTPUT$mcse
#The confidence intervals should be:
C_interval

```

```
## [1] -0.02334551 0.01585178
```

```

num_new <- length(data_new)
N_new <- 0
for (i in 1:num_new){
  if (-1<=data_new[i]&data_new[i]<=2){N_new=N_new+1}}
Probability <- N_new/num_new
#The probability would be:
Probability

```

```
## [1] 0.8154103
```