534 Homework 8

Gustavo Esparza

4/27/2019

Actual Value

For problems 1 and 2, we are considering the following integral:

$$\int_0^1 \frac{1}{1+x^2}dx$$

For reference in our Monte Carlo and Antithetic integration, we will provide the following solution for the integral:

```
fun=function(x){return(1/(1+x^2))}
integrate(fun,0,1)
```

```
## 0.7853982 with absolute error < 8.7e-15
```

Now that we know what our method estimates should approximate, we can begin our integration techniques.

1

A

Monte Carlo Estimation

let $g(x) = \frac{1}{1+x^2}$ and $X_i \sim unif(0,1)$. Then, we have the following **Monte Carlo** estimate for the evaluated integral:

$$\int_0^1 g(x)dx \approx \frac{1-0}{n}\sum_{i=1}^{n} g(X_i) = \bar{g}(X_i)$$

Then, we have the following approximation result, with a sample of 20,000 random uniform values :

```
set.seed(534) #set seed to comment on results

g = function(x){
  return(1/(1+x^2))}

n = 20000
X = runif(n,0,1)

MCest = mean(g(X))
print(MCest)
```

```
## [1] 0.7852158
```

Monte Carlo Standard Error

The standard error for our Monte Carlo estimation is defined as:

$$\sqrt{VAR(\bar{g}(X_i))} = \frac{\sigma_{g(X_i)}}{\sqrt{n}}$$

```
MCse =sd(g(X))/sqrt(n)
print(MCse)
```

```
## [1] 0.001130487
```

Result

We can see that our Monte Carlo estimate is about 0.785, which comes quite close to the actual value. We can also note that our Standard Error is $1.1 \cdot 10^{-3}$

B

Antithetic Estimate

We have $X_i \sim unif(0,1)$, then $F^{-1}(X_i) = X_i$. So, our Antithetic Estimate is defined as follows:

$$\int_0^1 g(x)dx \approx \frac{1}{2n}\sum_{i=1}^{n}\left[g(X_i) + g(1-X_i)\right]$$

Then, we have the following approximation result, with a sample of 10,000 random uniform values (to adjust for two samples):

```
set.seed(534)
n = 10000
X = runif(n,0,1)

ASest = (1/2)*mean(g(X) + g(1-X))
print(ASest)
```

```
## [1] 0.7856424
```

Antithetic Standard Error

The standard error for our Antithetic estimation is defined as:

$$\sqrt{VAR(\theta_{AS})} = \frac{\sigma_{g(X_i)+g(1-X_i)}}{\sqrt{4n}}$$

```
se_as= sd( g(X) + g(1-X))/(2*sqrt(n))
print(se_as)
```

```
## [1] 0.0001438875
```

We can see that our Antithetic estimate is about 0.7856, which comes quite close to the actual value. We can also note that our Standard Error is $1.4 \cdot 10^{-4}$

The estimates using Monte Carlo and Antithetic are nearly the same value, but the Standard Error for the Antithetic Method is smaller than that of the Monte Carlo method.

2

A

Generating Random Values From X

Since $X \sim f(x) = \frac{e^{-x}}{1-e^{-1}}$ then the CDF is $F_X = \frac{1-e^{-x}}{1-e^{-1}}$. Then, we have $F^{-1}(x) = -ln(1-(1-e^{-1})*x)$

Then, given $U \sim unif(0,1)$ then we can generate ramdom values from X such that $F^{-1}(U) \sim X$. The following function generates n random values from X:

```
#inverse function
f_inv=function(n){

u=runif(n,0,1)
inv=-log(1-(1-exp(-1))*u)
return(inv)}
```

B

Monte Carlo Estimate

Given $f(x)$ and $g(x)$ as defined previously, we can let $h(x)$ be defined as follows:

$$h(x) = \frac{g(x)}{f(x)} = \frac{\frac{1}{1+x^2}}{\frac{e^{-x}}{1-e^{-1}}} = \frac{1-e^{-1}}{(1+x^2)e^{-x}}$$

Then, our **Monte Carlo** estimate is defined as

$$\int_0^1 \frac{1}{1+x^2} \approx \frac{1}{n}\sum_{i=1}^{n} h(x_i) = \bar{h}(x_i)$$

So, we will generate our random values from X and use $h(x)$ to estimate our Integral:

```
n=20000
set.seed(534)
X=f_inv(n)

h = function(x){
  return( ( 1-exp(-1) ) / ( exp(-x)*(1+x^2) ) )}

MCest = mean(h(X))
print(MCest)
```

```
## [1] 0.7858983
```

Monte Carlo Standard Error

$$\sqrt{VAR(\bar{h}(X_i))} = \frac{\sigma_{h(X_i)}}{\sqrt{n}}$$

```r
se_MC =sd(h(X))/sqrt(n)
print(se_MC)
```

```
## [1] 0.0004934941
```

Thus, we can see that our estimate using the Monte Carlo Method is about .7858 with a standard error of $4.9 \cdot 10^{-4}$. Thus using values from $f(x)$ reduces the variance as opposed to using the Uniform distribution.

C

Creating new functions

In order to use the Antithetic Method, we must define the two following functions:

$Y_1 = h(F^{-1}(U))$ and $Y_2 = h(F^{-1}(1-U))$

```r
#New y functions
set.seed(534)
n=10000
U=runif(n,0,1)
U2=1-U
inv=-log(1-(1-exp(-1))*U)
inv2=-log(1-(1-exp(-1))*U2)
y1 =h(inv)
y2 =h(inv2)
```

Antithetic Estimate

Then, our Antithetic Estimate for our integral can be defined as:

$$\int_0^1 \frac{1}{1+x^2} \approx \frac{1}{2n}\sum_{i=1}^{n}(Y_1 + Y_2)$$

```r
AS_theta = (1/2)*mean(y1+y2)
print(AS_theta)
```

```
## [1] 0.7857389
```

Antithetic Standard Error

The standard error for our Antithetic estimation is defined as:

$$\sqrt{VAR(\theta_{AS})} = \frac{\sigma_{Y_1(U_i)+Y_2(1-U_i)}}{\sqrt{4n}}$$

```r
AS_se= sd( y1 + y2)/(2*sqrt(n))
print(AS_se)
```

```
## [1] 0.0001899052
```

Thus, we can see that our estimate using the Antithetic Method is about .7857 with a standard error of $1.8 \cdot 10^{-4}$. Thus using values from $f(x)$ reduces the variance as opposed to using the Uniform distribution for the Antithetic method as well.

3

```r
library(MASS) #for mvrnnorm
```

A

Hit-Miss Estimate

Since $X_i \sim f(x, \Sigma)$, then our estimate of $P(-\infty \leq X_1 \leq 1, -\infty \leq X_2 \leq 4, -\infty \leq X_3 \leq 2)$ can be defined as θ_{HM}, where:

$$\theta_{hm} = \frac{1}{n}\sum_{i=1}^{n} 1_{[-\infty \leq X_i^1 \leq 1, -\infty \leq X_i^2 \leq 4, -\infty \leq X_i^3 \leq 2]}$$

Now, we can define our distribution as given in the text and utilize our Hit-Miss Algorithm to estimate our probability.

```r
set.seed(534)
n = 20000
mu = c(0,0,0)
Sigma = matrix(c(1,3/5,1/3,3/5,1,11/15,1/3,11/15,1),3,3)

X = mvrnorm(n, mu, Sigma)

HMest = sum((X[,1]<1) & (X[,2]<4) & (X[,3]<2))/n
print(HMest)
```

```
## [1] 0.8272
```

Thus, the estimated probability is about .83.

B

Hit-Miss Standard Error

The Standard Error of our Hit-Miss estimate is defined as $\sqrt{\frac{\theta_{HM}(1-\theta_{HM})}{n}}$

```r
#standard error

HMse = sqrt( (HMest*(1-HMest) )/n )
print(HMse)
```

```
## [1] 0.002673389
```

We can see that the standard error for our estimate is about $2.6 \cdot 10^{-3}$

C

Hit-Miss Confidence Interval

The 95% Confidence interval for our Hit-Miss estimate is defined as

$$\theta_{HM} \pm 1.96 * \sigma_{HM}$$

```
L = HMest-1.96*HMse
U = HMest+1.96*HMse

cat("Confidence Interval = [",L,", ", U,"]")
```

```
## Confidence Interval = [ 0.8219602 ,  0.8324398 ]
```

We can see that approximation value provided,0.8279849897, is included in our confidence interval, so our Hit-Miss algorithm was effective in estimating the desired probability.