In the name of God

Producer:  
Mehrab Atighi

Subject:

Different of Importance sampling M.c

and classical M.c

Date:

12/29/2020

Supervisor:  
Dr.Seyed Noorullah Mousavi

**Issue:** For the computation of the expectation E*f*[*h*(*X*)] when *f* is the  
normal pdf and *h*(*x*) = :  
a. Show that E*f*[*h*(*X*)] can be computed in closed form and derive its value.  
b. Construct a regular Monte Carlo approximation based on a normal *N* (0*,* 1)  
sample of size Nsim=10^3 and produce an error evaluation.  
c. Compare the above with an importance sampling approximation based on  
an importance function *g* corresponding to the *U*(*-*8*, -*1) distribution and  
a sample of size Nsim=10^3. (Warning: This choice of *g* does not provide a  
converging approximation of E*f*[*h*(*X*)]!)

d.do the part c with expontional distrubtion function and show the standard devation,error.

**solve:**

**a)**

We know that if

Now we want to solve this antegral with close form method:

If we set

so we have:

We now that the normal distrubtion( antegral is 1.

So we should find the area under the g(x) curve.

=

Now if we set

If we set .

So we have:

**b)**

if we want to use classical monte carlo method for we have:

Now we can simulate it in R:

> g<-function(x){exp(-(x^2)/2)}

> N=10^3

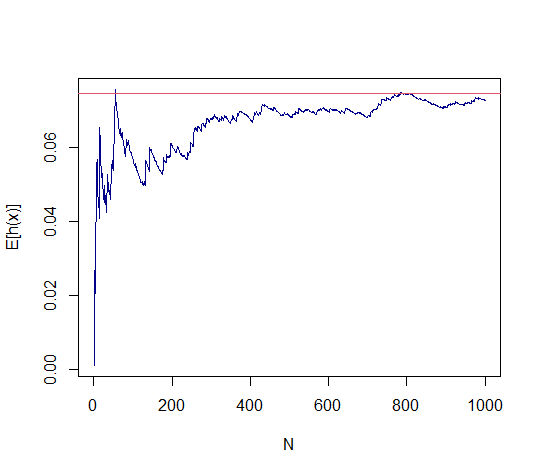
> H1<-rnorm(N,mean = 3,sd=1)

> H2<-rnorm(N,mean = 6,sd=1)

> E<-(cumsum(g(H1))/1:N)+(cumsum(g(H2))/1:N)

> plot(E , type = "l",col="Blue4")

> abline(h=a , col="850")



Our estimate for

> E[N]

[1] 0.07275448

The real value is:

> a

[1] 0.07453205

Now we can find the our monte carlo estimate error:

> abs(a-E[N])

[1] 0.001777571

now we can find the standard deviation:

> sd(E)

[1] 0.007590212

**c)**

we know that if

Now we want to use the Importance sampling Monte carlo .

According to the question part c we should use in our formula, so we have:

So now we can simulate it in R:

> N=10^3

> G1<-function(x){exp(-(x^2)/2)\*exp(-1/2\*(x-3)^2)/(sqrt(2\*pi))}

> G2<-function(x){exp(-(x^2)/2)\*exp(-1/2\*(x-6)^2)/(sqrt(2\*pi))}

> G<-integrate(G1,-8,-1)$val+integrate(G2,-8,-1)$val

> u<-runif(N,min=-8,max=-1)

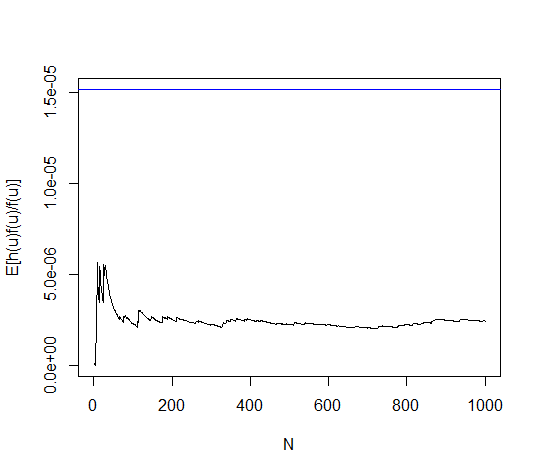
> g1<-function(x){exp(-(x^2)/2)\*exp(-1/2\*(x-3)^2)/(7\*sqrt(2\*pi))}

> g2<-function(x){exp(-(x^2)/2)\*exp(-1/2\*(x-6)^2)/(7\*sqrt(2\*pi))}

> R<-7\*((cumsum(g1(u))/1:N)+(cumsum(g2(u))/1:N))

> plot(R, type = "l" , ylim =c(min(R),G), xlab="N" , ylab = "E[h(u)f(u)/f(u)]")

> abline(h=G ,col="Blue")



Our estimate for

> R[N]

[1] 2.449683e-06

The real value is:

> G

[1] 1.516476e-05

Now we can find the our monte carlo estimate error:

> abs(G-R[N])

[1] 1.271508e-05

now we can find the standard deviation:

> sd(R)

[1] 4.939516e-07

**d)**

we know that if

the importance sampling minte carlo method formula is:

According to our qurstion pard d, we should set the .so we have:

Now we simulate in R:(here we set

> N=10^3

> F1<-function(x){exp(-(x^2)/2)\*exp(-1/2\*(x-3)^2)/(sqrt(2\*pi))}

> F2<-function(x){exp(-(x^2)/2)\*exp(-1/2\*(x-6)^2)/(sqrt(2\*pi))}

> F<-integrate(F1,0,Inf)$val+integrate(F2,0,Inf)$val

> q<-rexp(N,rate =1)

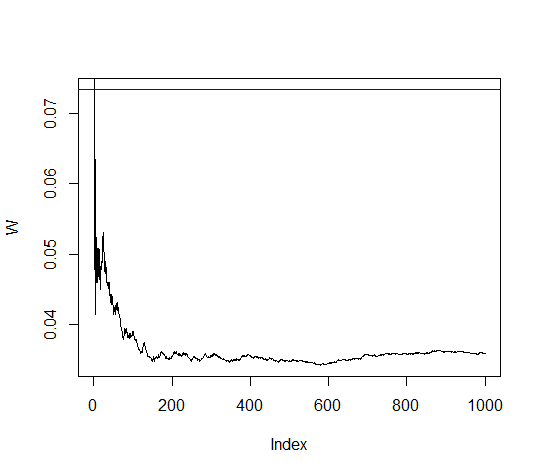
> f1<-function(x){(1/exp(-x/2))\*exp(-(x^2)/2)\*exp(-1/2\*(x-3)^2)/(sqrt(2\*pi))}

> f2<-function(x){(1/exp(-x/2))\*exp(-(x^2)/2)\*exp(-1/2\*(x-6)^2)/(sqrt(2\*pi))}

> W<-((cumsum(f1(q))/1:N)+(cumsum(f2(q))/1:N))

> plot(W, type = "l" , ylim =c(min(W),F))

> abline(h=F,col="Blue")



Our estimate for

> W[N]

[1] 0.03597266

The real value is:

> F

[1] 0.07335276

Now we can find the our monte carlo estimate error:

> abs(F-W[N])

[1] 0.0373801

now we can find the standard deviation:

> sd(W)

[1] 0.003373489

End.