Assignment 4

Wenhao Li 250959325

16/03/2020

# Question 1

# (a)

directpoly<-function(x,coef){  
 # Error checking  
 if(length(coef)<2){  
 stop("Length of the coefficient must be greater than 2")  
 }  
 sum\_1<-0  
 n<-length(coef)  
 for (i in 1:n) {  
 sum\_1<-sum\_1+coef[i]\*x^(i-1)  
 }  
 return(sum\_1)  
}  
# Test function  
x<-1:3  
coef<-c(2,17,-3)  
directpoly(x,coef)

## [1] 16 24 26

# (b)

hornerpoly<-function(x,coef){  
 # Error checking  
 if(length(coef)<2){  
 stop("Length of the coefficient must be greater than 2")  
 }  
 n<-length(coef)  
 output<-coef[n]  
 for(i in (n-1):1){  
 output<-output\*x+coef[i]  
 }  
 return(output)  
}  
# Test function  
x<-1:3  
coef<-c(2,17,-3)  
hornerpoly(x,coef)

## [1] 16 24 26

# (c)

system.time(directpoly(x=seq(-10,10,length=5000000),c(1,-2,2,3,4,6,7,8)))

## user system elapsed   
## 0.927 0.065 0.999

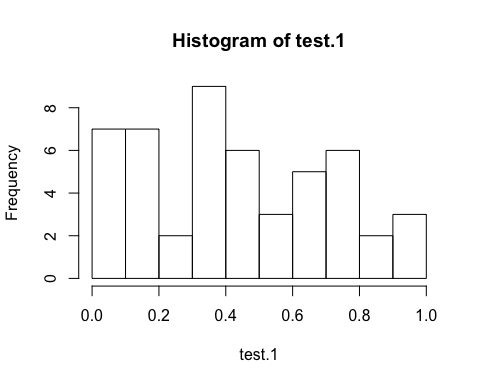
system.time(hornerpoly(x=seq(-10,10,length=5000000),c(1,-2,2,3,4,6,7,8)))

## user system elapsed   
## 0.177 0.017 0.195

Comment: Horner’s algorithm is much faster than the direct sum algorithm. Horner’s algorithm is about 10 times faster than the direct sum algorithm for elasped and user time and about 6 times faster for system time.

# Question 2

my.unif<-function(n,a,c=0,m,x0){  
 x<-integer(n)  
 k=x0  
 for (i in 1:n) {  
 x[i]<-(a\*k+c)%%m  
 k=x[i]  
 }  
 return(x/m)  
}  
  
test.1=my.unif(n=50,a=172,c=13,m=30307,x0=17218)  
hist(test.1,breaks=seq(0,1,0.1))



test.2=my.unif(n=50,a=171,c=51,m=32767,x0=2020)  
hist(test.2)

 Comment: Both histogrms show that both distributions do not follow Uniform distributin ([0,1]).

# Question 3

# (a)

set.seed(2020)  
U<-runif(1000)  
average<-mean(U)  
variance<-var(U)  
standard.deviation<-sd(U)  
c(average,variance,standard.deviation)

## [1] 0.49216701 0.08345052 0.28887803

# (b)

true.average<-0.5  
true.variance<-((1-0)^2)/12  
true.standard.deviation<-sqrt(true.variance)  
c(true.average,true.variance,true.standard.deviation)

## [1] 0.50000000 0.08333333 0.28867513

c(average,variance,standard.deviation)

## [1] 0.49216701 0.08345052 0.28887803

Comment: The differnece between calculated and true mean, variance and standard.deviation is very small. The true average is a litte bigger than calculated mean by 0.00783299. The true variance is a little smaller than calculated variance by 0.00011719. The true standard deviation is a little smaller than calculated standard deviation by 0.0002029.

# (c)

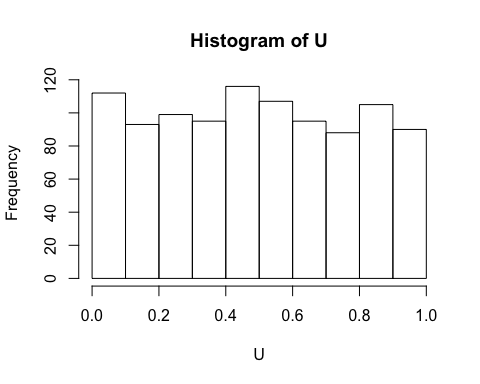
proportion<-length(U[U<0.6])/1000  
true.proportion<-0.6  
c(proportion,true.proportion)

## [1] 0.622 0.600

Comment: The difference between two values is small. The true value is a little smaller than calculated one by 0.022.

#(d)

hist(U,breaks=seq(0,1,0.1))

 Comment: The histogram shows that the distribution of U is likely to follow Uniform distributin ([0,1]).

# Question 4

# (a)

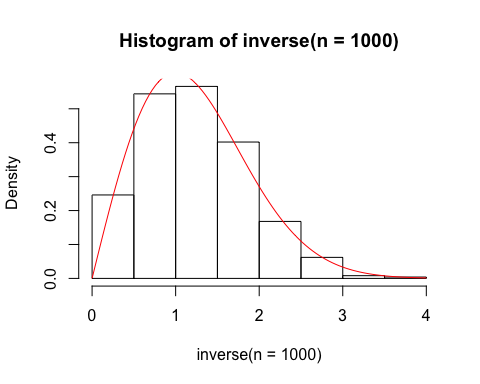
白板上写着字

描述已自动生成

inverse<-function(n,a=1){  
 jk<-runif(n)  
 x<-sqrt(-2\*(a^2)\*log(1-jk))  
 return(x)  
}  
fx<-function(x,a=1){  
 (x\*exp(-x^2/(2\*a^2)))/(a^2)  
}  
  
# Test  
set.seed(721)  
inverse(n=20)

## [1] 0.4273861 0.4511862 0.4247111 1.6698325 1.9348687 1.5088176 0.2634750  
## [8] 1.4622280 0.8161548 1.0361257 2.0160578 0.3806599 0.5066131 1.1698287  
## [15] 1.1606557 0.7492447 0.8920261 1.2527818 2.1908355 1.4843690

hist(inverse(n=1000),probability=TRUE)  
curve(fx,col=2,add=TRUE)

 Comment: The histogram and the density curve fits well.

# (b)

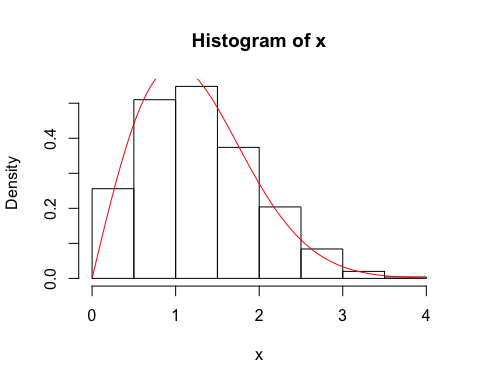
白板上写着字

描述已自动生成

reject<-function(n,M,a=1){  
 x<-runif(n,min=0,max=5)  
 jk<-runif(n)  
 x<-x[jk<=fx(x,a)/M]  
while (length(x)<n) {  
 other.length<-n-length(x)  
 other.x<-runif(other.length,min=0,max=5)  
 jk<-runif(other.length)  
 other.x<-other.x[jk<=fx(other.x)/M]  
 x<-c(x,other.x)  
}  
 return(x)   
}  
  
  
set.seed(721)  
reject(20,M=exp(-1/2),a=1)

## [1] 0.4312111 1.4163438 0.6021777 1.2236646 1.6412039 1.4441653 0.1397879  
## [8] 0.7516135 1.1608881 3.3075433 0.4797271 1.6855978 1.5344197 1.4602601  
## [15] 2.0398102 0.6840323 1.4242271 1.0307856 1.1165852 1.7017674

x<-reject(1000,M=exp(-1/2),a=1)  
hist(x,probability=TRUE)  
curve(fx,col=2,add=TRUE)

 Comment: The histogram and the density curve fits well.

# (c)

system.time(inverse(n=10000,a=1))

## user system elapsed   
## 0.001 0.000 0.001

system.time(reject(n=10000,M=exp(-1/2),a=1))

## user system elapsed   
## 0.004 0.000 0.004

Comment: Inverse method is faster than reject method.

# Question 5

# (a)

set.seed(721)  
# Estimeate E[U^2]  
u<-runif(n=10000)  
a<-mean(u^2)  
  
# Confidence Interval  
b<-a-1.96\*sd(u^2)/sqrt(10000)  
c<-a+1.96\*sd(u^2)/sqrt(10000)  
c(b,c)

## [1] 0.3246256 0.3363184

# Compare with the true value  
a

## [1] 0.330472

true.a<-((1+0)/2)^2+((1-0)^2)/12  
true.a

## [1] 0.3333333

accuracy<-sd(u^2)/sqrt(10000)  
accuracy

## [1] 0.002982847

Comment: The 95% confidence interval is (0.3246256,0.3363184), it covers the estimeate value of 0.330472 and the true value of 0.3333333. The estimate E[U^2] is close to the true value and is a little samller than the true vaule. The accuracy value is small.

# (b)

v<-(u^2+(1-u)^2)/2  
d<-mean(v)  
e<-d-1.96\*sd(v)/sqrt(10000)  
f<-d+1.96\*sd(v)/sqrt(10000)  
c(e,f)

## [1] 0.3320817 0.3350044

d

## [1] 0.333543

accuracy<-sd(v)/sqrt(10000)  
accuracy

## [1] 0.0007456055

Comment: The 95% confidence interval is (0.3320817,0.3350044), it covers the estimate value of 0.333543. The estimate value in (b) is also close to the true value. The estimate value in (b) is more closr to the true value than the estimate value in (a). The extimate value in (b) is more accuracy than in (a). The estimate value in (b) is a better estimator than the estimator value in (a).

# (c)

p<-((u/2)^2+(1-u/2)^2)/2  
g<-mean(p)  
h<-g-1.96\*sd(p)/sqrt(10000)  
i<-g+1.96\*sd(p)/sqrt(10000)  
c(h,i)

## [1] 0.3326898 0.3356172

g

## [1] 0.3341535

accuracy<-sd(p)/sqrt(10000)  
accuracy

## [1] 0.0007467794

Comment: The 95% confidence interval is (0.3326898 0.3356172), it covers the estimate value of 0.3341535. The estimate value in (c) is also close to the true value. The estimate value in (c) is a litter far away from the true value than (b), but is still closr to the true value than the estimate value in (a) The estimate value in (c) is less accuracy than (b), but is still more accuracy than (a). The estimate value in (c) is a better estimator than the estimate value in (a).