1.

(i) 利用Trapezoidal rule,求 積分從0到0.7 ,

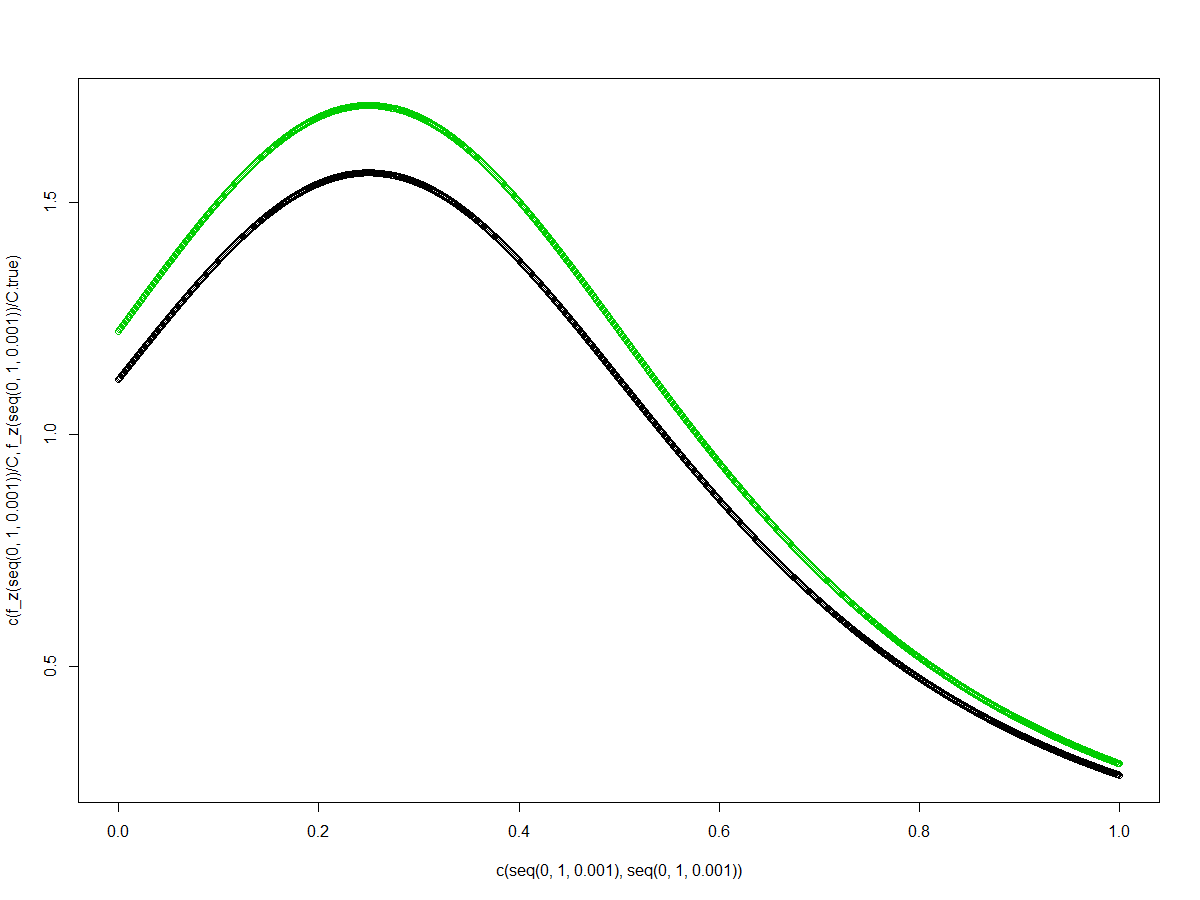
由,得 ,

取 , (用R算C=0.6396319)

接著 , 且

得 ,

故 ,理論上所需最小樣本數約為38



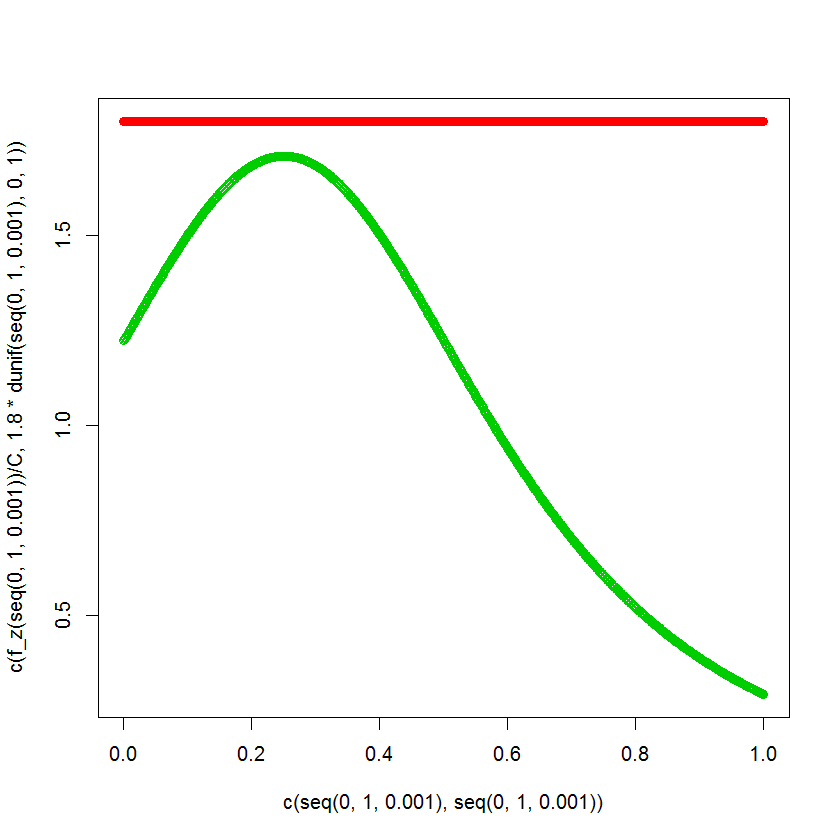
(圖為 density: 黑色為R算的C ,綠色為估計的C)

(ii)利用empirical CDF 去估計F\_z(0.7)=P(Z < 0.7) 的值,

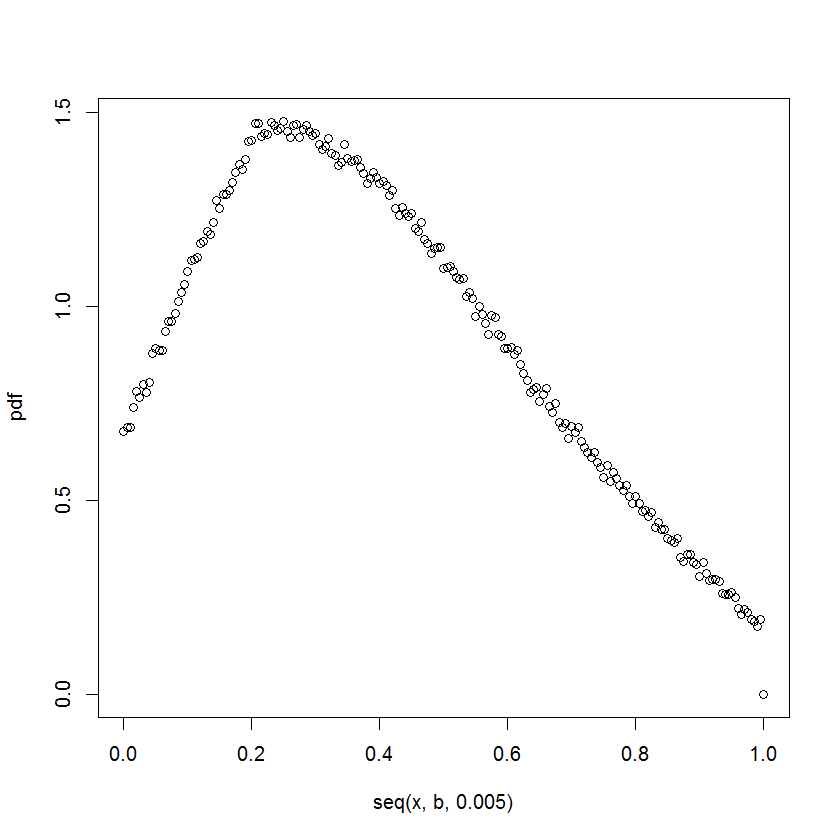
1. sample ,

首先利用Rejection sampling生成

選擇g ~ U(0,1)作為envolope , e(x)=a\*g(x) ,取a=1.8 畫圖如下



(紅色為envolope ; 黑色為 )



(此為生成後利用empirical pdf 估計 )

由於 為一個估計量，可以得到在樣本數為10^5時估計值為0.87254 ,用R算的積分值為0.872516089 ,

bias為8.608991e-05.

由 ,

根據中央極限定理 ,

我們構造level 信賴區間 ,

取

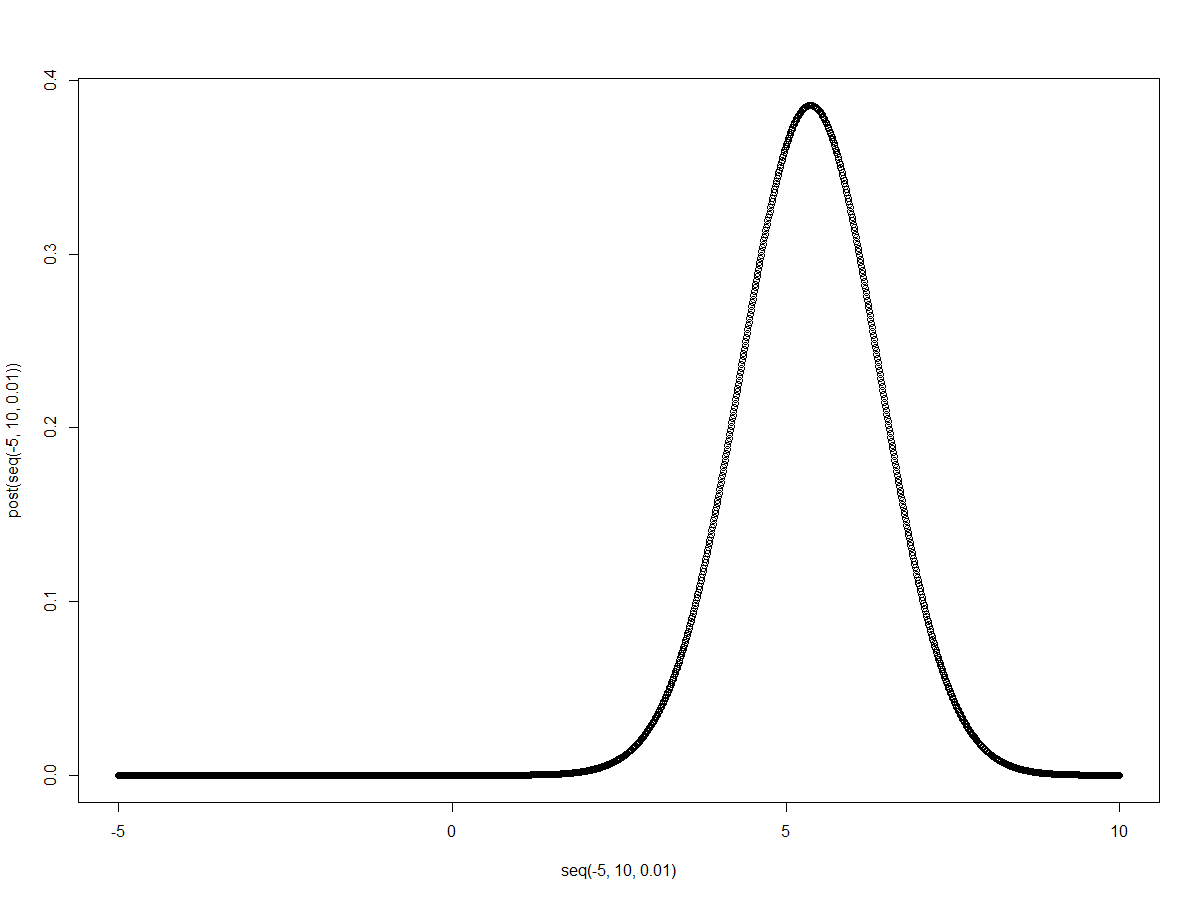
亦即我們有99%信心水準 ,取出的樣本數滿足

模擬結果n約為10^6.

2.估計mean and variance of posterior density ,

由於此處無法理論推得posterior density屬於某個特定分配 ,

Step1.模擬 theta=5 ,X ~N(5,1) , 先畫出posterior density ,



(posterior density給定 theta=5 ,X ~N(5,1)的畫圖 )

因

我們生成x1,x2,…,xn ~N(5,1) ,取n=10^4 ,可以利用Simpson method算出對

於theta=5的估計值.

Step 2.由此我們每給定一個theta ,可以算出相對應的

利用

,取n=100 ,

重複3次得到估計值為120.9677, -67.09211, 197.5573  
 發現對於mean of posterior 的估計在正負值間跳動 ,

我們對於theta的先驗分布為cauchy ,而理論上cauchy分布的mean是不

存在的 ,而這裡即使給定了資料X ,對於theta的mean估計仍沒有很好.

(附上程式碼部分)

1.

|  |
| --- |
| f\_z<-function(z){ (1+( (z-0.25)/0.5 )^2 )^-1.5}  C.true<-integrate(f\_z,0,1)$value  I\_true<-integrate(f\_z,0,0.7)$value /C.true  par(mfrow=c(1,1))  #est. C  C<-( max(f\_z(seq(0,1,0.001)) )+min(f\_z(seq(0,1,0.001)) ) )/2  plot(c(seq(0,1,0.001),seq(0,1,0.001) ) ,c( f\_z(seq(0,1,0.001))/C , f\_z(seq(0,1,0.001))/C.true ),  col= c(rep(3,1001),rep(1,1001)) )  plot(seq(0,1,0.001), f\_z(seq(0,1,0.001))/C )  #conv. rate  secdiff.f\_z<-function(z){ 60/C\*( 1+(2\*z-0.5)^2 )^(-3.5) \*(2\*z-0.5)^2 -12/C\*( 1+(2\*z-0.5)^2 )^(-2.5) }  plot(seq(0,1,0.001),secdiff.f\_z(seq(0,1,0.001)));lines(c(0.6,0.6),c(-22,5))  n<-sqrt( (0.7)^3\*secdiff.f\_z(0.6)/12\* (10^4) )  windows();plot(x=c(seq(0,1,0.001),seq(0,1,0.001)),y=c(f\_z(seq(0,1,0.001))/C,1.8\*dunif(seq(0,1,0.001),0,1)),  col =c(rep(3,1001),rep(2,1001)) )  c<-1.8 ;C\*c  envo<-function(x,c){ f\_z(x)/C / c }  acc.rej.exp<-function(n,c=1.8){  u1<-runif(n,0,1)  Y<-u1  u2<-runif(n,0,1)  X<-rep(0,n)  N<-length(which(u2 <= envo(Y,c) ))  for(i in 1:N){ X[i]<-Y[which(u2 <=envo(Y,c))][i] } #exp(-(Y-1)^2/2)  while(N<n){  uu1<-runif(n-N,0,1)  Y<-uu1  uu2<-runif(n-N,0,1)  if(length(which(uu2<=envo(Y,c)))>0){  for(i in 1:length(which(uu2<=envo(Y,c)))){ X[N+i]<-Y[which(uu2<=envo(Y,c))][i] }}  N<-N+length(which(uu2 <= envo(Y,c)))  }  return(X)  }  PDF<-function(a,b,n,band){  #a=0;b=1;n=1000;band=0.2  x<-a  pdf<-rep(0,length(seq(x,b,0.005)))  j<-1  while(a<=b){  I<-0  for(i in 1:n){ if( abs(a-acc.rej.exp(1))<=band){ I<-I+1 } }  pdf[j]<-I/(2\*n\*band)  a<-a+0.005  j<-j+1  }  windows();plot(seq(x,b,0.005),pdf) #;lines(c(0,0),c(0,1)) ;lines(c(-x,b),c(0.5,0.5))  return(pdf)  }  PDF(0,1,5000,0.2)  #sample size n  cdf<-0 ;n<-10^6  I<-0  for(i in 1:n){ if(acc.rej.exp(1)<=0.7){ I<-I+1 } }  cdf<-I/n  while(n<=cdf\*(1-cdf)\*qnorm(1-0.4,0,1)^2\*10^8 ){  if(n<10^6){n<-n\*10}else{n<-n+10^5}  cdf<-0  I<-0  for(i in 1:n){ if(acc.rej.exp(1)<=0.7){ I<-I+1 } }  cdf<-I/n  cat("\n","The sample size is",n," ,bias is",abs(cdf-I\_true),"\n")  } |

2.

|  |
| --- |
| px.theta<-function(x,theta){ 1/sqrt(2\*pi)\*exp(-(x-theta)^2/2) }  p.theta<-function(theta){ 1/(pi\*(1+theta^2)) }  px<-function(theta){ px.theta(x,theta)\*p.theta(theta) } #plot(seq(-10,10,0.01),px(seq(-10,10,0.01))) #plot p(x)  post<-function(theta){ px.theta(x,theta)\*p.theta(theta)/I\_Spx } #p(theta|x) (posterior) #x<-rnorm(1,5,1);plot(seq(-10,10,0.01),post(seq(-10,10,0.01)))#plot p(theta|x)  #f<-function(theta){ theta\*post(theta) } #theta\*p(theta|x)  #plot p(theta|x)  theta<-5 ;x<-rnorm(1,theta,1)  a<-theta-10 ;b<-theta+10 ;n<-100  h<-seq(a,b,(b-a)/n)  if(n%%2==0){  z1<-0;z2<-0  for(i in 2:(n/2)){ z1<-z1+2\*px(h[2\*i-1]) }  for(i in 1:(n/2)){ z2<-z2+4\*px(h[2\*i]) }  I\_Spx<-( px(h[1])+px(h[n+1])+z1+z2 )\*(b-a)/n/3  rm(z1,z2)  } else { print("n need to be even") }  plot(seq(-5,10,0.01),post(seq(-5,10,0.01))) ;lines(c(theta,theta),c(-0.1,0.6));lines(c(seq(-5,10,0.01)[which(post(seq(-5,10,0.01))==max(post(seq(-5,10,0.01))))],seq(-5,10,0.01)[which(post(seq(-5,10,0.01))==max(post(seq(-5,10,0.01))))]),c(-0.1,max(post(seq(-5,10,0.01)))) )  #estmator  E<-rep(0,100)  for(k in 1:100){  y<-c()  for(j in -1000:1000){  theta<-j  x<-rnorm(1,theta,1)  a<-theta-10 ;b<-theta+10 ;n<-100  h<-seq(a,b,(b-a)/n)  z1<-0;z2<-0  for(i in 2:(n/2)){ z1<-z1+2\*px(h[2\*i-1]) }  for(i in 1:(n/2)){ z2<-z2+4\*px(h[2\*i]) }  I\_Spx<-( px(h[1])+px(h[n+1])+z1+z2 )\*(b-a)/n/3  y<-c(y,theta\*post(theta))  }  E[k]<-sum(y)  }  mean(E) |