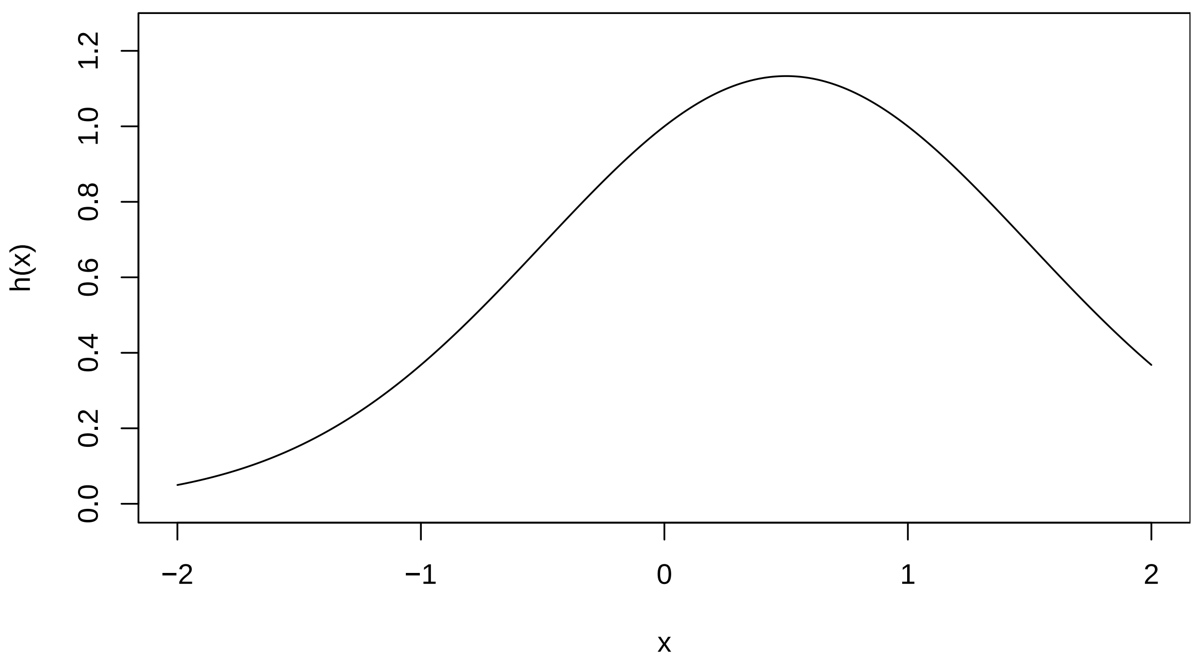
MATH 559 - EXERCISES 3 : SOLUTIONS

1. *Design sampling schemes to estimate the following integrals by Monte Carlo:*

   (a)
$$I = \int_{-2}^{2} \exp\{x(1-x)/2\}\, dx$$

Solution: Write
$$I = \int_{-2}^{2} \exp\{x(1-x)/2\} dx = \int_{-2}^{2} h(x)\, dx.$$

```
xv<-seq(-2,2,by=0.001)
yv<-exp(xv*(1-xv)/2)
par(mar=c(4,4,1,0))
plot(xv,yv,type='l',xlab='x',ylab='h(x)',ylim=range(0,1.25))
```



Method 1: Basic Monte Carlo using the $Uniform(-2,2)$

$$I = \int_{-2}^{2} \exp\{x(1-x)/2\} dx = 4\int_{-2}^{2} \exp\{x(1-x)/2\}\, \frac{1}{4}\, dx$$

```
N<-100000
nreps<-1000
I1.hat<-function(Nv){x<-runif(N,-2,2);return(mean(4*exp(x*(1-x)/2)))}
Ihat1<-replicate(nreps,I1.hat(N));Ihat1[1:10]

+  [1] 2.635866 2.635671 2.637754 2.632317 2.635548 2.627694 2.634097 2.629356
+  [9] 2.632560 2.636926
```

Method 2: Importance sampling using $Normal(0,1)$

$$I = \int_{-2}^{2} \exp\{x(1-x)/2\} dx = \sqrt{2\pi} \int_{-\infty}^{\infty} e^{x/2} \mathbb{1}_{(-2,2)}(x) \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \, dx$$

```
I2.hat<-function(Nv){x<-rnorm(N);return(mean(exp(x/2)*(x>-2)*(x<2))*sqrt(2*pi))}
Ihat2<-replicate(nreps,I2.hat(N));Ihat2[1:10]

+  [1] 2.622527 2.629624 2.632163 2.638103 2.636054 2.633263 2.627923 2.643251
+  [9] 2.632580 2.629449
```

Method 3: Importance sampling using $Normal(1/2,1)$

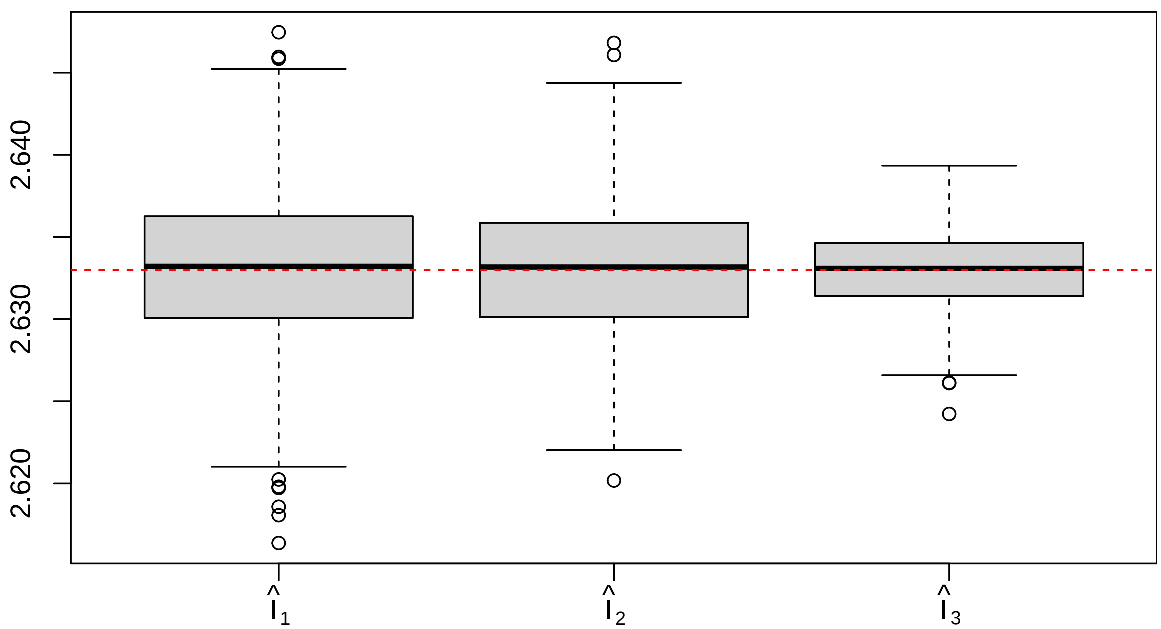
$$I = \int_{-2}^{2} \exp\{x(1-x)/2\} dx = \sqrt{2\pi} e^{1/8} \int_{-\infty}^{\infty} \mathbb{1}_{(-2,2)}(x) \frac{1}{\sqrt{2\pi}} e^{-(x-1/2)^2/2} \, dx$$

```
I3.hat<-function(Nv){x<-rnorm(N,1/2,1);return(mean((x>-2)*(x<2))*sqrt(2*pi)*exp(1/8))}
Ihat3<-replicate(nreps,I3.hat(N));Ihat3[1:10]

+  [1] 2.627581 2.630165 2.630336 2.635676 2.633574 2.629768 2.633432 2.631529
+  [9] 2.633517 2.633659
```

The exact result can be computed using the Method 3 approach and `pnorm` in `R`.

```
I1<-sqrt(2*pi)*exp(1/8)*(pnorm(2,1/2,1)-pnorm(-2,1/2,1))
```

The true value is therefore 2.632986. The three different methods have different variances



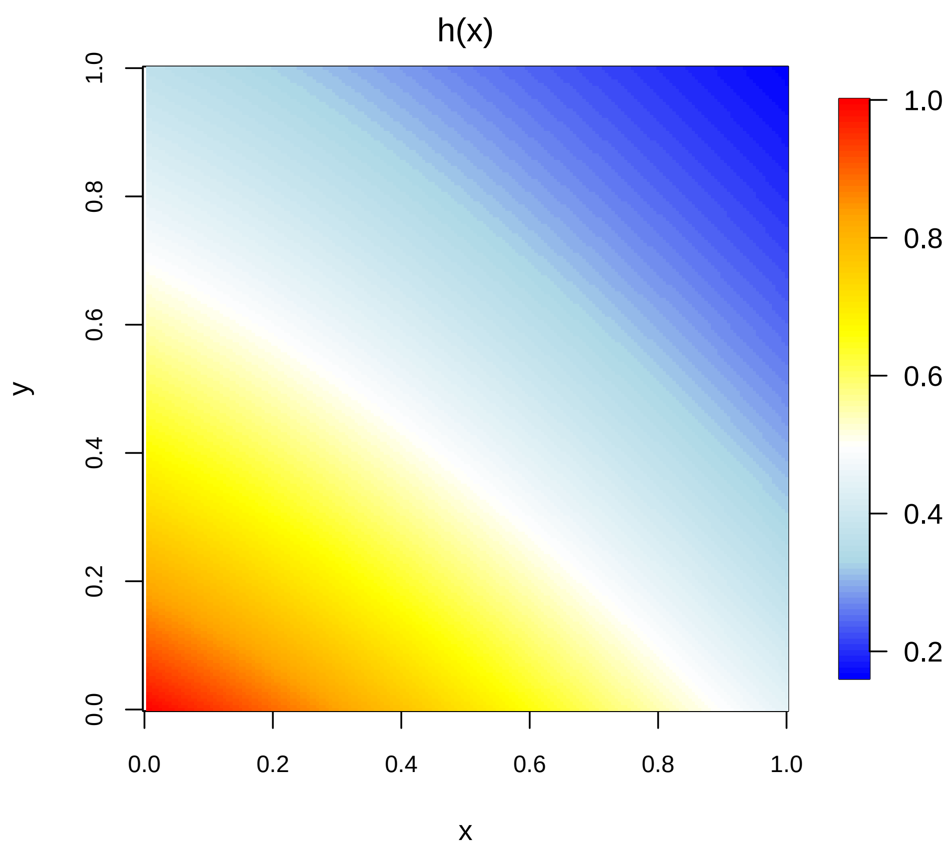

```
+ [1] "Variances (multiplied by 10^6)"
+ [1] 21.023087 17.105776  5.239979
```

(b)

$$I = \int_0^1 \int_0^1 \frac{\sin(x)}{\log(1+x)} \exp\{-(x+y)\}\, dxdy$$

Solution:

```
pts<-seq(0,1,by=0.005)
xvec<-pts
yvec<-pts
I2.func.xy<-function(xv,yv){
    fv<-return((sin(xv)/log(1+xv))*exp(-(xv+yv)))
}
f <- Vectorize(I2.func.xy,vectorize.args=c("xv","yv"))
I2z<-outer(xvec,yvec,f)

#Plot
library(fields,quietly=TRUE)
par(pty='s',mar=c(4,3,2,2))
cols<-c("blue","lightblue","white","yellow","orange","red")
colfunc <- colorRampPalette(cols)
image.plot(xvec,yvec,I2z,col=colfunc(100),
           xlab=expression(x),ylab=expression(y),cex.axis=0.8)
contour(xvec,yvec,I2z,add=T,levels=seq(-100,-30,by=10))
title(expression(paste('h(x)')))
```



Method 1: Basic Monte Carlo using the Uniform distribution on the unit square.

$$I = \int_0^1 \int_0^1 \frac{\sin(x)}{\log(1+x)} \exp\{-(x+y)\}\, dxdy = \int_0^1 \int_0^1 h(x,y)\, dxdy$$

say.

```
N<-100000
nreps<-1000
I2.func<-function(x,y){return((sin(x)/log(1+x))*exp(-(x+y)))}
I1.hat<-function(Nv){x<-runif(N);y<-runif(N);return(mean(I2.func(x,y)))}
Ihat1<-replicate(nreps,I1.hat(N))
Ihat1[1:10]

+  [1] 0.4549497 0.4552725 0.4541699 0.4552246 0.4542833 0.4548151 0.4549934
+  [8] 0.4542877 0.4543902 0.4549678
```

Method 2: Importance sampling using the $Exponential(1)$

$$I = \int_0^1 \int_0^1 \frac{\sin(x)}{\log(1+x)} \exp\{-(x+y)\}\, dxdy = \int_0^\infty \int_0^\infty g(x,y) \exp\{-(x+y)\}\, dxdy$$

where

$$g(x,y) = \mathbb{1}_{(0,1)}(x)\mathbb{1}_{(0,1)}(y)\frac{\sin(x)}{\log(1+x)}$$

say.

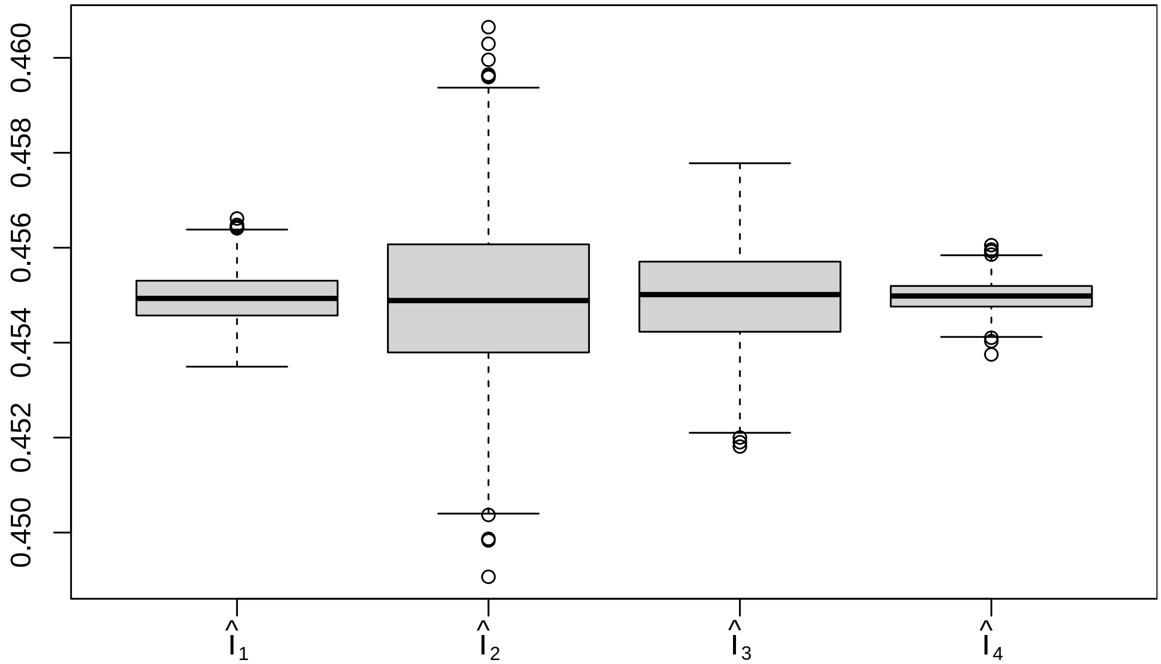
```
N<-100000
nreps<-1000
I22.func<-function(x,y){return((x<1)*(y<1)*(sin(x)/log(1+x)))}
I2.hat<-function(Nv){x<-rexp(N);y<-rexp(N);return(mean(I22.func(x,y)))}
Ihat2<-replicate(nreps,I2.hat(N))
Ihat2[1:10]

+  [1] 0.4547481 0.4558095 0.4536987 0.4569472 0.4549874 0.4541687 0.4525434
+  [8] 0.4541064 0.4536195 0.4557679
```

Method 3: Partial analytic solution integrating out y

$$I = \int_0^1 \int_0^1 \frac{\sin(x)}{\log(1+x)} \exp\{-(x+y)\}\, dxdy = (1-e^{-1}) \int_0^\infty \mathbb{1}_{(0,1)}(x)\frac{\sin(x)}{\log(1+x)} \exp\{-x\}\, dx$$

```
N<-100000
nreps<-1000
I23.func<-function(x){return((1-exp(-1))*(x<1)*(sin(x)/log(1+x)))}
I3.hat<-function(Nv){x<-rexp(N);return(mean(I23.func(x)))}
Ihat3<-replicate(nreps,I3.hat(N))
Ihat3[1:10]

+  [1] 0.4530248 0.4564074 0.4563625 0.4558437 0.4553496 0.4534164 0.4542653
+  [8] 0.4534445 0.4562923 0.4536700

I24.func<-function(x){return((1-exp(-1))*exp(-x)*(sin(x)/log(1+x)))}
I4.hat<-function(Nv){x<-runif(N);return(mean(I24.func(x)))}
Ihat4<-replicate(nreps,I4.hat(N))
Ihat4[1:10]

+  [1] 0.4549937 0.4545559 0.4548041 0.4553712 0.4543931 0.4554188 0.4549100
+  [8] 0.4550585 0.4551758 0.4550317
```

The different methods again have different variances




```
+ [1] "Variances (multiplied by 10^6)"
+ [1] 0.2699984 3.0377542 1.2004469 0.1065584
```

2. *The definite integral*

$$\int_1^2 \phi(x)\, dx = \Phi(2) - \Phi(1)$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are the standard Normal pdf and cdf respectively can be computed using the `pnorm` *function in* `R`*. However, suppose the value of the integral is to be estimated using Monte Carlo methods.*

Compare the variances of the ordinary Monte Carlo estimator ($f(x) \equiv f_0(x) \equiv \phi(x)$) and the optimal Importance Sampling estimator ($f_0(x)$ to be selected).

Solution: The value of the integral is

$$I = \Phi(2) - \Phi(1) = 0.135905$$

The ordinary Monte Carlo estimator uses the standard Normal density and estimates the expectation of the function $g(x) = \mathbb{1}_{[1,2]}(x)$, the variance of the Monte Carlo estimator is therefore

$$\frac{1}{N}\text{Var}_f[g(X)] = \frac{1}{N}\left(\mathbb{E}_f[\{g(X)\}^2] - \{\mathbb{E}_f[g(X)]\}^2\right) = \frac{I(1-I)}{N} \simeq \frac{0.117435}{N}$$

The optimal importance sampling estimator is defined by

$$f_0(x) \propto |g(x)|f(x)$$

which is the standard Normal pdf constrained to the interval $(1, 2)$; this pdf is

$$f_0(x) = \frac{g(x)f(x)}{I} \qquad \therefore \qquad \frac{g(x)f(x)}{f_0(x)} = I$$

which is constant, and hence the optimal estimator has variance zero. Of course, this estimator is infeasible, as in order to implement it one needs to know I. A feasible importance sampling estimator uses $f_0(x) \equiv Uniform(1,2)$, and sets $g(x) = \phi(x)$. This estimator has variance

$$\frac{1}{N}\text{Var}_{f_0}[\phi(X)]$$

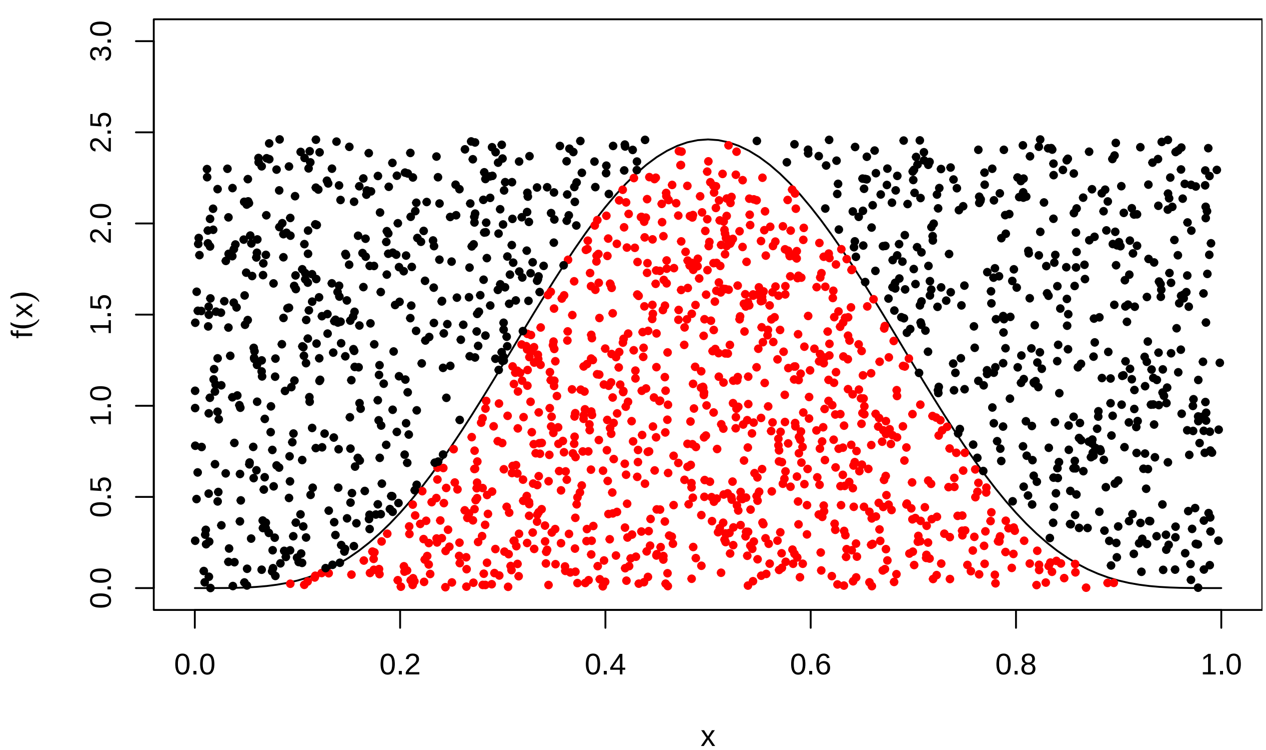
3. *Suppose $f(x) \equiv Beta(5,5)$. Compute the acceptance rate of a rejection sampling algorithm that uses $f_0(x) \equiv Uniform(0,1)$ as the proposal distribution.*

   Solution: If $x_m = 1/2$ denotes the mode of the $Beta(5,5)$,

$$\frac{f(x)}{f_0(x)} \leq f(x_m) \simeq 2.460938 = M$$

   so the acceptance rate is $1/M = 0.406349$

```
M<-dbeta(1/2,5,5)
N<-10000
X<-runif(N);U<-runif(N)
acc<-U < dbeta(X,5,5)/M
sum(acc)/N

+ [1] 0.4056

xv<-seq(0,1,by=0.01);yv<-dbeta(xv,5,5)
par(mar=c(4,4,2,0))
plot(xv,yv,type='l',ylim=range(0,3),xlab='x',ylab=expression(f(x)))
points(X[acc][1:1000],U[acc][1:1000]*M,pch=19,cex=0.5,col='red')
points(X[!acc][1:1000],U[!acc][1:1000]*M,pch=19,cex=0.5,col='black')
```



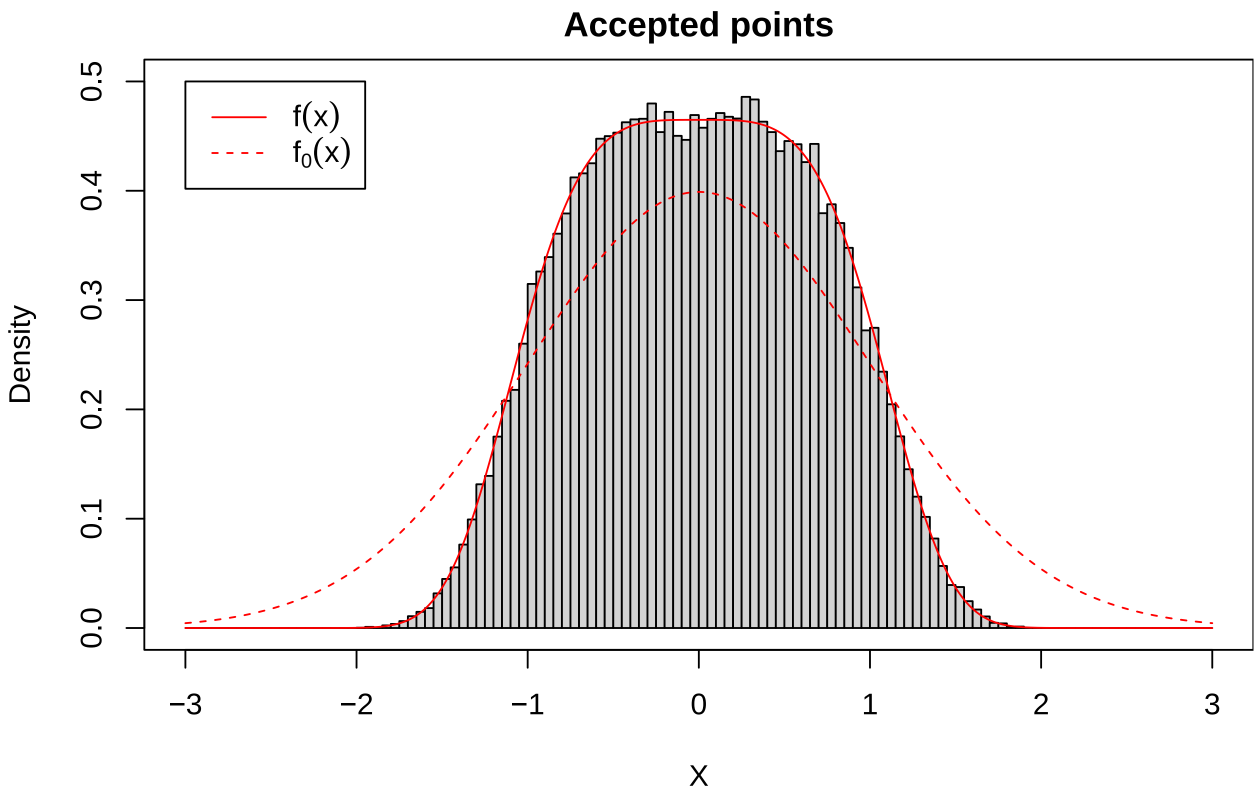
4. *Consider the density*
$$f(x) = c\exp\{-x^4/2\} \qquad x \in \mathbb{R}$$
where $c < \infty$ is a constant. Design a rejection sampling scheme to produce samples from $f(x)$.

Solution: Using the proposal $f_0(x) \equiv Normal(0,1)$, we have that
$$\frac{f(x)}{f_0(x)} = \sqrt{2\pi}c\exp\left\{-\frac{x^2}{2}(x^2-1)\right\}$$
which is minimized when
$$4x^3 - 2x = 0 \qquad \therefore \qquad x = 1/\sqrt{2}$$
and the supremum of the ratio is therefore
$$M = \sqrt{2\pi}c\exp\left\{\frac{1}{8}\right\}.$$
If we sample $X \sim Normal(0,1)$, and $U \sim Uniform(0,1)$, we accept if
$$u \leq \frac{c\exp\{-x^4/2\}}{M\dfrac{1}{\sqrt{2\pi}}\exp\{-x^2/2\}} = \exp\{-1/8\}\frac{\exp\{-x^4/2\}}{\exp\{-x^2/2\}}.$$
Thus we do not need the value of c to implement the algorithm, but we can estimate it by using the relation that
$$\text{Acceptance rate} = \frac{1}{M} = \frac{1}{\sqrt{2\pi}c}\exp\left\{-\frac{1}{8}\right\}$$

```
M<-sqrt(2*pi)*exp(1/8)
N<-100000
X<-rnorm(N);U<-runif(N)
acc<-U < exp(-1/8-0.5*(X^4-X^2))
acc.rate<-sum(acc)/N
acc.rate

+ [1] 0.75735

M.est<-1/acc.rate
cval<-M.est/(sqrt(2*pi)*exp(1/8))
cval

+ [1] 0.4648648
```

The estimated value of c is therefore 0.464865.

```
xv<-seq(-3,3,by=0.01)
yv<-cval*exp(-0.5*xv^4)
par(mar=c(4,4,2,0))
hist(X[acc],br=seq(-3,3,by=0.05),xlab='X',ylim=range(0,0.5),
    freq=FALSE,main='Accepted points');box()
lines(xv,yv,col='red')
lines(xv,dnorm(xv),lty=2,col='red')
legend(-3,0.5,c(expression(f(x)),expression(f[0](x))),lty=c(1,2),col='red')
```



5. *It is often necessary to sample uniformly on a region to compute various integrals numerically.*

   (a) *Design a method to sample uniformly from the region $\mathcal{D}$ with boundary the unit circle:*

$$\mathcal{D} = \{(x, y) \in \mathbb{R}^2 : x^2 + y^2 < 1\}.$$

Solution: Here, we could use rejection based on the proposal density $f_0(x, y)$ that is jointly uniform on the set $(-1, 1) \times (-1, 1)$, accepting points only if $x^2 + y^2 < 1$. This method has acceptance rate $\pi/4 \doteq 0.785398$. However, we can actually sample directly using a polar coordinates formulation. Suppose $R \sim Beta(2, 1)$ and $T \sim Uniform(0, 2\pi)$ are independent, and define

$$X = R\cos(T) \qquad Y = R\sin(T)$$

so that

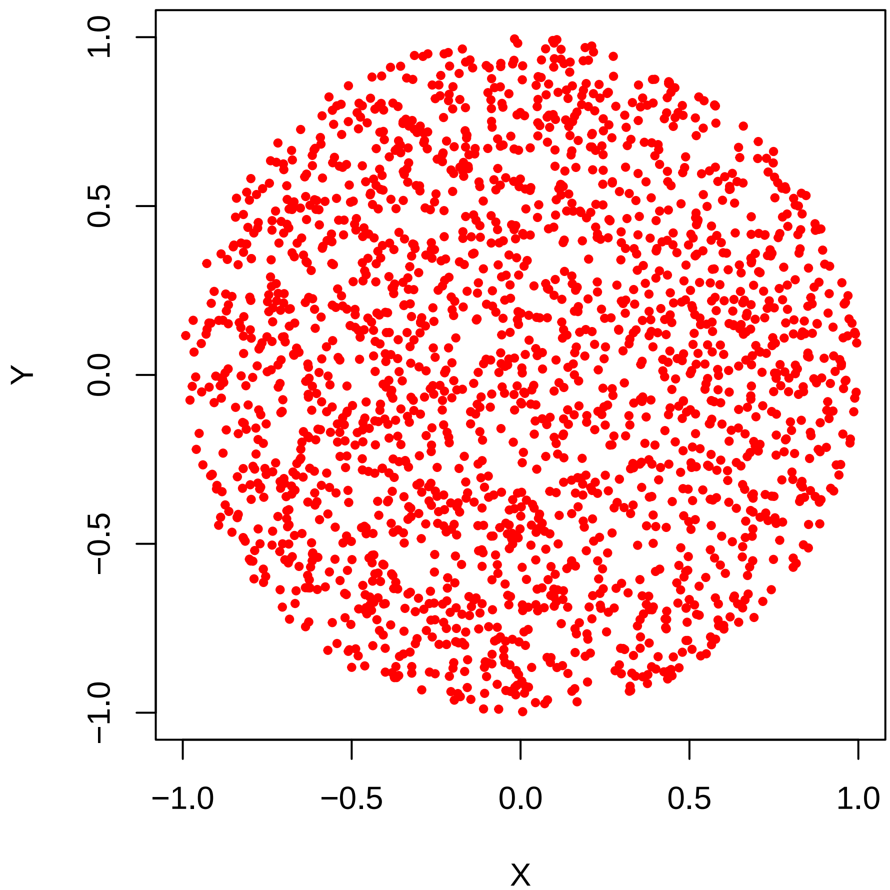
$$R = \sqrt{X^2 + Y^2} \qquad T = \arctan(Y/X).$$

Then, using 1-1 transformation methods,

$$\begin{aligned} f_{X,Y}(x, y) &= f_{R,T}(\sqrt{x^2 + y^2}, \arctan(y/x)) \frac{1}{\sqrt{x^2 + y^2}} \\ &= \frac{\Gamma(3)}{\Gamma(2)\Gamma(1)} \sqrt{x^2 + y^2} \times \frac{1}{2\pi} \times \frac{1}{\sqrt{x^2 + y^2}} \qquad\qquad R, T \text{ independent} \\ &= \frac{1}{\pi} \end{aligned}$$

for (x, y) inside the unit disk.

```
N<-2000
R<-rbeta(N,2,1)
Th<-runif(N,0,2*pi)
X<-R*cos(Th)
Y<-R*sin(Th)
par(pty='s',mar=c(4,4,2,0))
plot(X,Y,xlim=range(-1,1),ylim=range(-1,1),pch=19,cex=0.5,col='red')
```



(b) *Design a method to sample uniformly from the region $\mathcal{R}_d$ where*

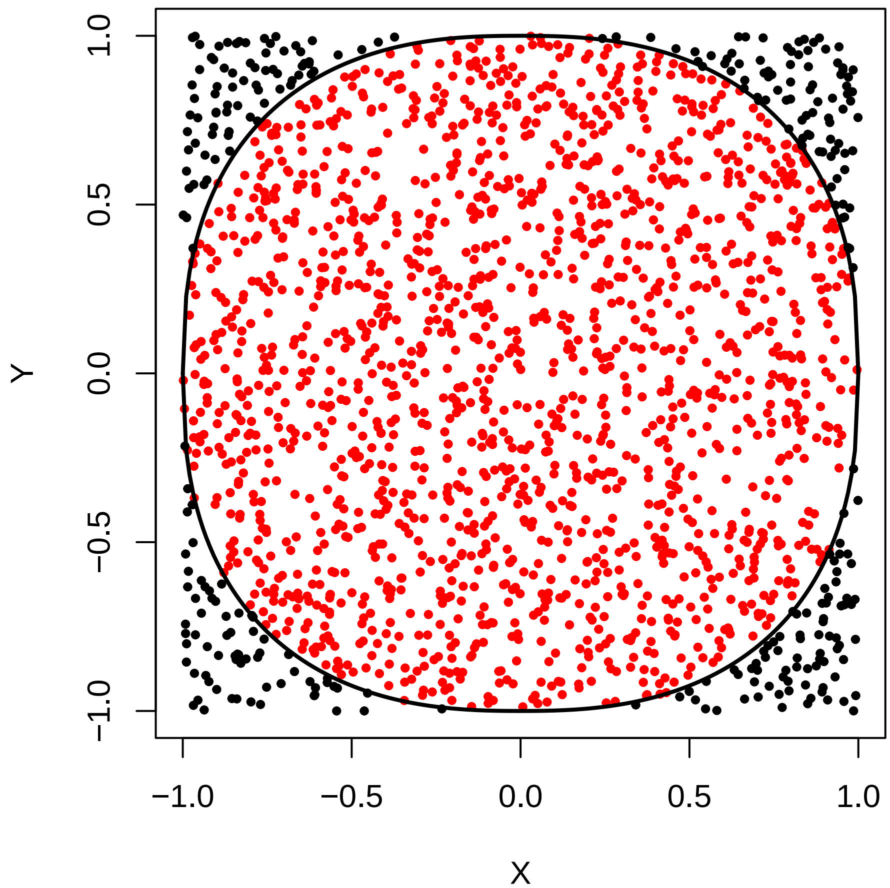
$$\mathcal{R}_d = \{(x,y) \in \mathbb{R}^2 : (|x|^d + |y|^d)^{1/d} < 1\}.$$

where $d > 1$.

Solution: Here, as $d > 1$, the region $\mathcal{R}_d$ is convex, is interior to the region $[-1,1] \times [-1,1]$, but not interior to the unit disk. Therefore acceptance is likely to be an effective method. Below we implement the method for $d = 2.5$.

```
N<-2000
d<-2.5
X<-runif(N,-1,1)
Y<-runif(N,-1,1)
acc<-abs(X)^d+abs(Y)^d < 1
sum(acc)/N

+ [1] 0.846
```

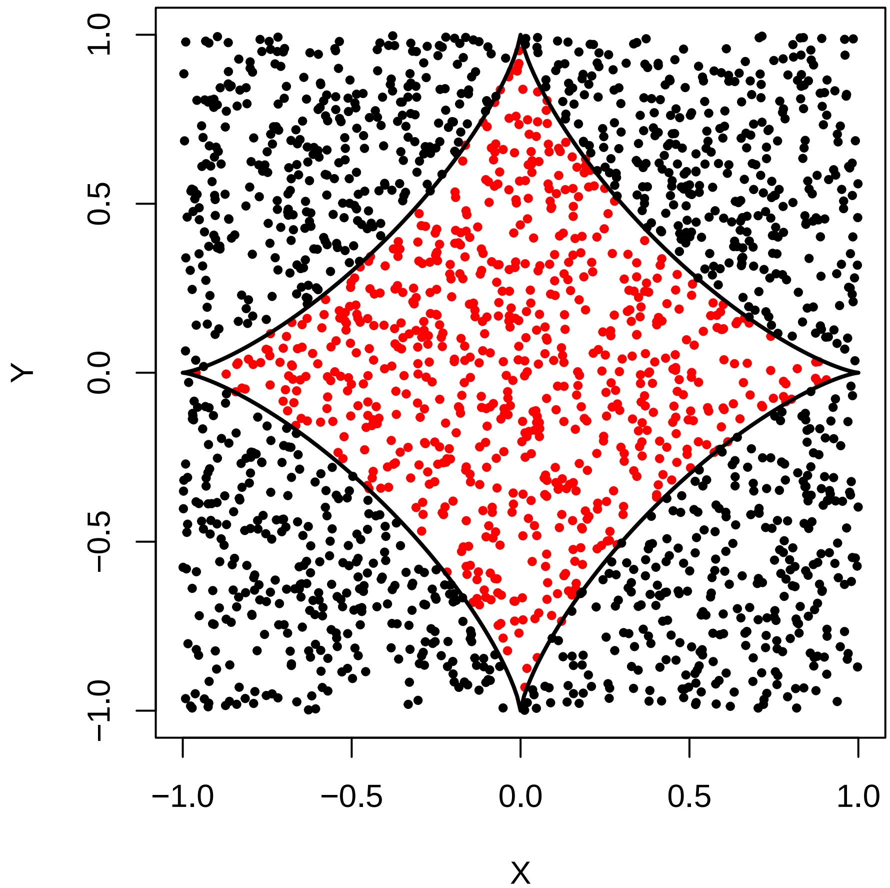
```
par(pty='s',mar=c(4,4,2,0))
plot(X[acc],Y[acc],xlim=range(-1,1),xlab='X',ylab='Y',
    ylim=range(-1,1),pch=19,cex=0.5,col='red')
points(X[!acc],Y[!acc],pch=19,cex=0.5)
xv<-seq(-1,1,by=0.01)
yv<-(1-abs(xv)^d)^(1/d)
lines(xv,yv,lwd=2)
lines(xv,-yv,lwd=2)
```



The method also works when $d < 1$, but then the method is less effective as the region is smaller.

```
N<-2000
d<-0.75
X<-runif(N,-1,1)
Y<-runif(N,-1,1)
acc<-abs(X)^d+abs(Y)^d < 1
sum(acc)/N

+ [1] 0.3535

par(pty='s',mar=c(4,4,2,0))
plot(X[acc],Y[acc],xlim=range(-1,1),xlab='X',ylab='Y',
    ylim=range(-1,1),pch=19,cex=0.5,col='red')
points(X[!acc],Y[!acc],pch=19,cex=0.5)
```

```
xv<-seq(-1,1,by=0.01)
yv<-(1-abs(xv)^d)^(1/d)
lines(xv,yv,lwd=2)
lines(xv,-yv,lwd=2)
```



In this case, we can simulate instead from the unit circle as a proposal, using the method from (a).

```
N<-2000
d<-0.75
R<-rbeta(N,2,1)
Th<-runif(N,0,2*pi)
X<-R*cos(Th)
Y<-R*sin(Th)
acc<-abs(X)^d+abs(Y)^d < 1
sum(acc)/N

+ [1] 0.4625

par(pty='s',mar=c(4,4,2,0))
plot(X[acc],Y[acc],xlim=range(-1,1),xlab='X',ylab='Y',
     ylim=range(-1,1),pch=19,cex=0.5,col='red')
points(X[!acc],Y[!acc],pch=19,cex=0.5)
xv<-seq(-1,1,by=0.01)
yv<-(1-abs(xv)^d)^(1/d)
```

```
lines(xv,yv,lwd=2)
lines(xv,-yv,lwd=2)
```