Homework 2

Question 1

1,

- $\int_0^1 \frac{1}{2\sqrt{x}} dx = \frac{1}{2} \int_0^1 x^{-1/2} dx = 1$

Yes, f is a pdf

- 1, make sure f is a pdf
  2, find f' CDF

  $F_X(x) = x^{1/2}$

  3. find f' inverse/pseudo CDF

  $F^{-1}(\alpha) = x^2$ if $0 \le \alpha \le 1$

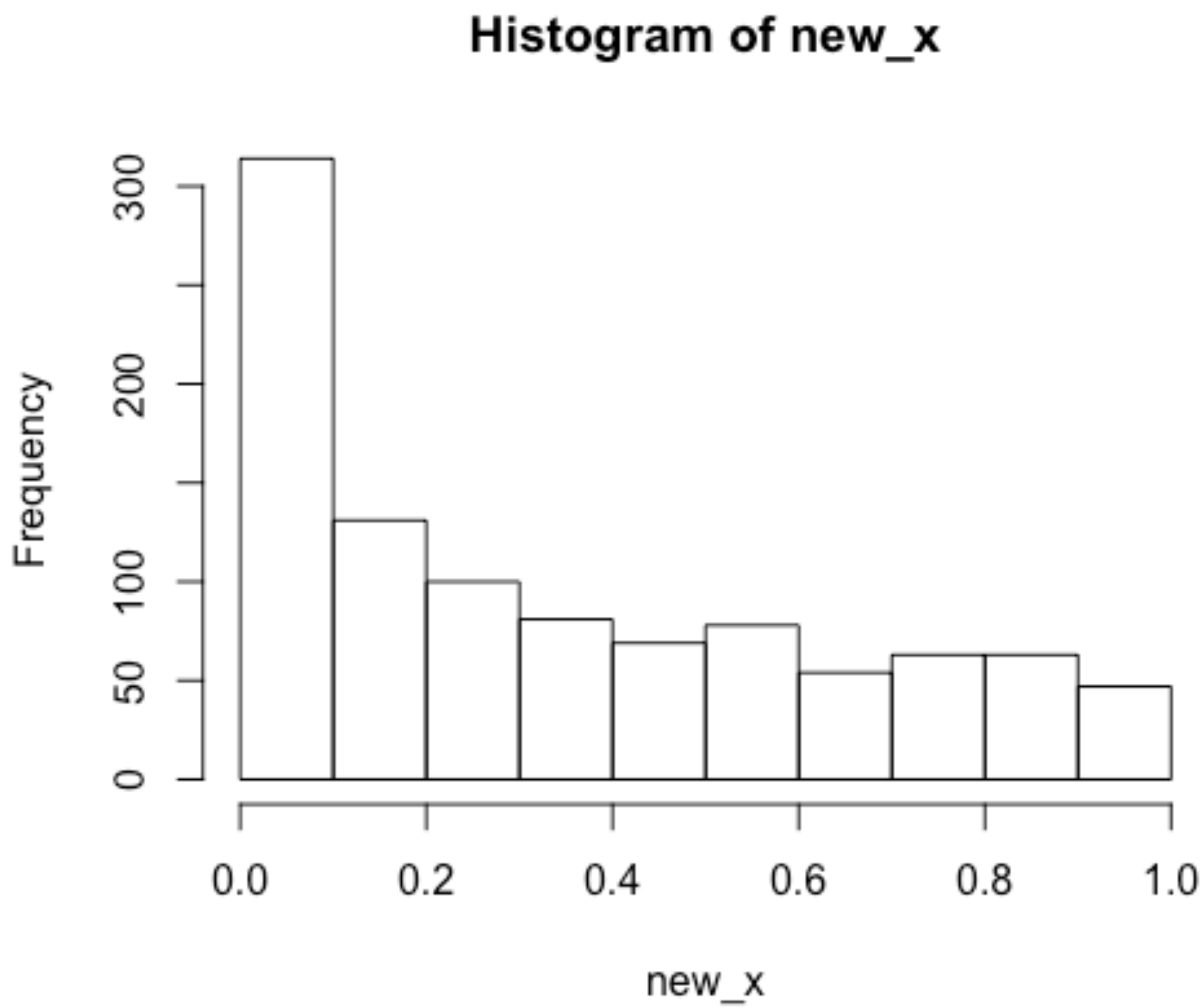
  4, Sample $U_1 \dots U_n \sim$ Uniform $([0,1])$

  5 $F^{-1}(U_1) \dots F^{-1}(U_n) \sim F$

  The steps above show how to construct a rv X with pdf f

-

```
# find inverse CDF and put uniform dist in it to generate new samples wi
unif<-runif(1000,0,1)
new_x<-unif^2
hist(new_x)
```



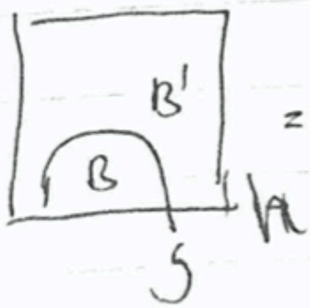
How to calculate C for Question 2 and Question 3

to find c

$\rightarrow f(x) = c\, g(x) \qquad \int f(x) = c \int g(x) \qquad 1 = c \int g(x)$

$\rightarrow c = \dfrac{1}{\int g(x)}$

$\rightarrow$ to find $\int g(x)$



$= \dfrac{\text{Area}(B)}{\text{Area}(B')} = \dfrac{\text{accept}}{\text{total}} = \dfrac{\int g(x)}{\int h(x)}$

$\int g(x) = \int h(x)$ (Accept/Total)

$\rightarrow c = \dfrac{1}{g(x)}$

Question 2

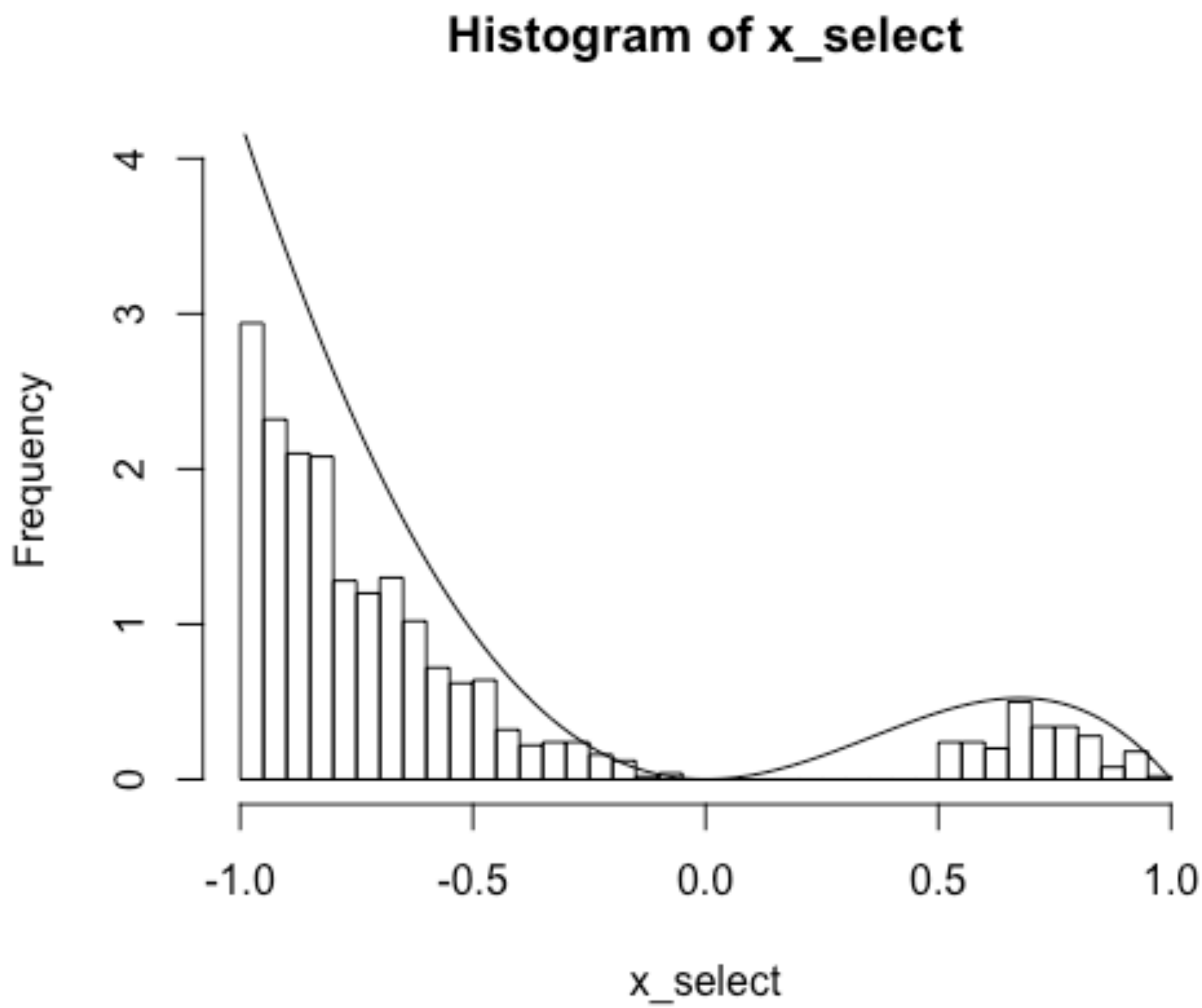
```r
g <- function(x) {
  ((sin(x))**2)*abs(x**3+2*x-3)

}

re_run<-0
f_x<-rep(0,1000)
x_select<-c()
y_select<-c()
M<-4.5

for (i in 1:length(f_x)) {
  repeat{

    x<-runif(1,-1,1)
    y<-runif(1,0,M)
    re_run<-re_run+1
    if(((x > -1 & x<0)| (x>0.5 &x<1)) & 0<=y & y<=g(x)){break}
  }
  x_select<-c(x_select,x)
  y_select<-c(y_select,y)
}
```




```
## ----c-----
g_val<- 1000/re_run*2*4.5
c2<-1/g_val
c2
# c = 0.6196667
```

Question 3

```
#question 3
p_3 <- function(x) {
  (x^-0.5)/2

}
g_3<-function(x){
  (x^-0.5)/2*abs((cos(x**2+x)))**0.5
}
run_3<-0
p_y<-rep(0,1000)
x_select_3<-c()
y_select_3<-c()
for (i in 1:length(p_y)) {
    repeat{
      run_3<-run_3+1
      in_unif<-runif(1,0,1)
      x<-in_unif^2
      y<-runif(1,0,p_3(x))
      re_run<-re_run+1
      if(0<=y & y<=g_3(x)){break}
    }
  x_select_3<-c(x_select_3,x)
  y_select_3<-c(y_select_3,y)
  }

  ##calcuate c
  c_3<- 1/(1000/run_3)
  # c_3 = 1.157
```

Question 4

, a, no

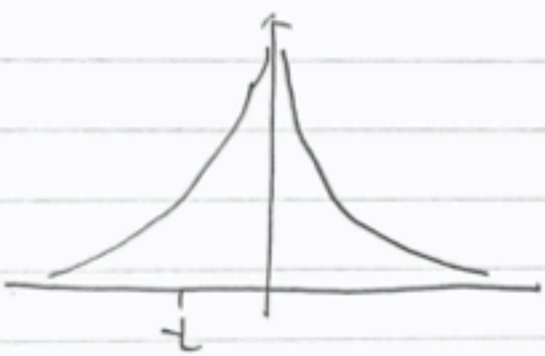
$$f(x) = \frac{1}{z} e^{-|x|}$$

$X \sim f$

$$F_X(t) = \mathbb{P}(X \leq t) = \int_{-\infty}^{t} \frac{1}{z} e^{-|x|} dx$$

$$= \frac{1}{z} \int_{-\infty}^{t} e^{-|x|} dx$$

$$= \frac{1}{z} \begin{cases} \int_{-\infty}^{t} e^{x} dx & t<0 \\ \int_{-\infty}^{0} e^{-|x|} dx + \int_{0}^{t} e^{-x} & t>0 \end{cases}$$



$$= \frac{1}{z} \begin{cases} e^{t} & t<0 \\ 1 + -e^{-x}\Big|_0^t & t>0 \end{cases}$$

$$= \frac{1}{z} \begin{cases} e^{t} & t<0 \\ 1 - e^{-t} + 1 & t>0 \end{cases}$$

$\mathbb{P}(X \leq \infty) = 1$

$$1 = \frac{1}{z}(2 - e^{-t}) = \frac{1}{z}(2) \Rightarrow z = 2$$

how would you sample from the distribution that has density f ?

1, find CDF and inverse/pseudo CDF

$$\text{CDF} = \begin{cases} \frac{1}{2}e^{t} & t<0 \\ 1-\frac{1}{2}e^{-t} & t\geq 0 \end{cases}$$

inverse $F^{-1}(x) = -\log(2-2x)$

2, Sample X’ from Inverse CDF, by plugging in uniform samples(unif[0,1]) in the inverse CDF to generate samples with density f.

```
###Question 4
#construct standard normal want to draw from normal so set g
g_n<-function(x){
  first<-1/sqrt(2*pi)
  second<-exp(-x^2/2)
  first*second
}

h<-function(x){
  0.5*exp(-x) # only need to consider half becuase it is symetric
}
range<-seq(-3,3,0.01)
M=1.5
plot(range,g_n(range),type = 'l',ylim = c(0,1))
lines(range,M*h(range))
#----find M such that g(x)<=Mh(x)


inv_cdf_mh_p<-function(x){
  -log(2-2*x) #x is from unif[0,1]
  #-log(x)
  #-log(1-1*x)
}
```

```r
run_4<-0
s_4<-rep(0,100)
x_select_4<-c()
y_select_4<-c()
for (i in 1:length(s_4)) {
  repeat{
    run_4<-run_4+1
    x<-inv_cdf_mh_p(runif(1,0,1))
    y<-runif(1,0,M*h(x))
    if( 0<=y&y<=g_n(x)){break}
  }
  x_select_4<-c(x_select_4,x)
  y_select_4<-c(y_select_4,y)
}
```

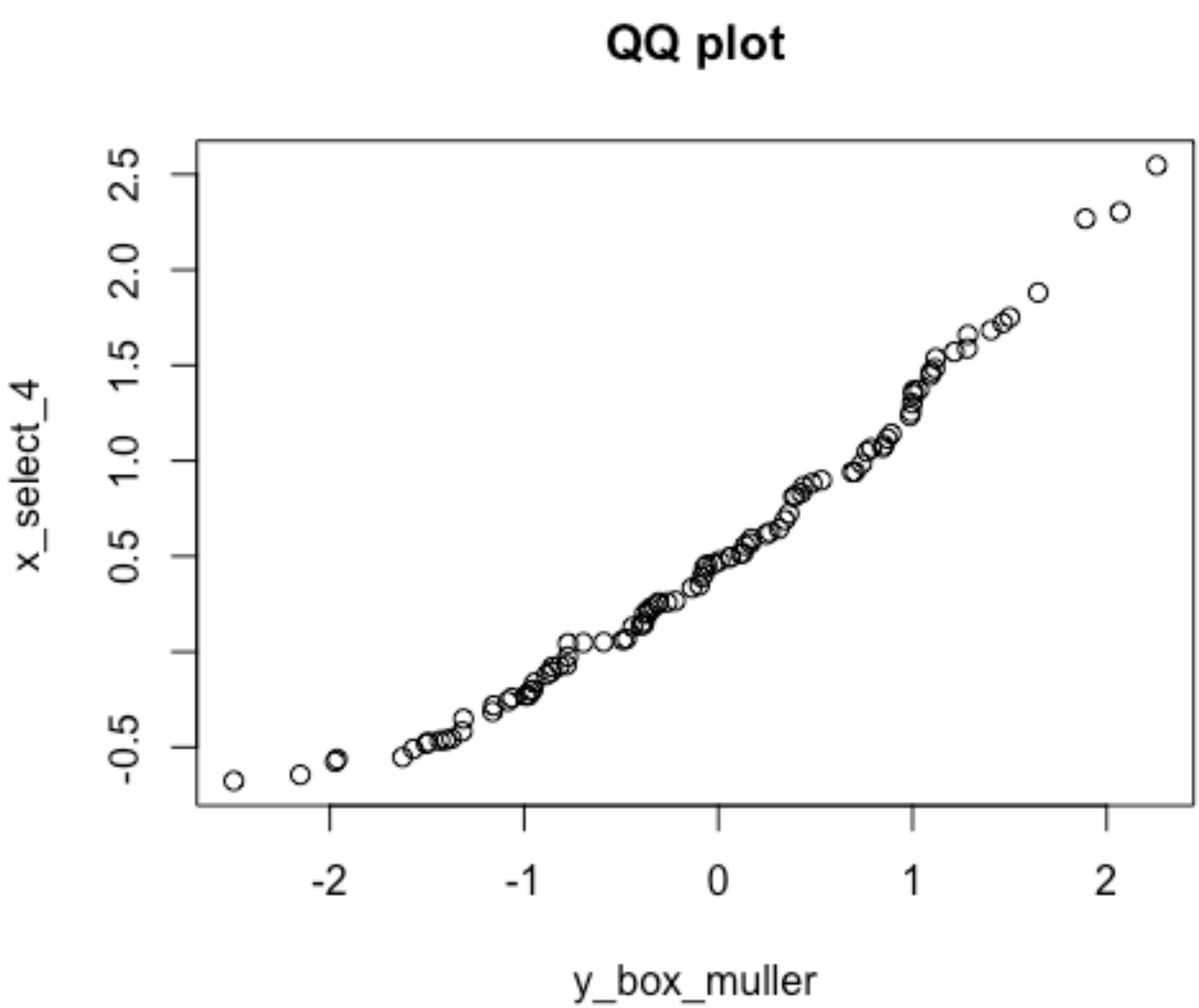
In your algorithm, on average how many samples from f were you rejecting for every accepted sample? Is this what you were expecting?

On average 0.72 samples are rejected for every accepted sample. This is what I am expecting by taking the ratio of the probability of rejecting and accepting.

This make sense because my M = 1.5 so it is relatively small.

```r
> # reject/accept
> rej_act<-(run_4-100)/100
> rej_act
[1] 0.72
> #rej_act = 0.72
> p_reject<-1-100/run_4
> p_accpet <- 100/run_4
> expected<-p_reject/p_accpet
> expected
[1] 0.72
```

```r
# box muler
u_1<-runif(100)
u_2<-runif(100)
x_box_muller<-sqrt(-2*log(u_1))*cos(2*pi*u_2)
y_box_muller<-sqrt(-2*log(u_1))*sin(2*pi*u_2)
plot(x_box_muller,y_box_muller)
qqplot(y_box_muller,x_select_4,main = "QQ plot")
```



This QQ plot shows an identity relationship between sample generated through rejection algorithm and the box muller method.
This means those two samples comes from the same distribution and they are normally distributed.

Question 5

5, a, no

$$1 = c\int g(x,y) \Rightarrow \frac{1}{c} = \int g(x,y) \Rightarrow \frac{1}{c} = \int_1^2 \int_0^1 x e^{xy}\, dy\, dx$$

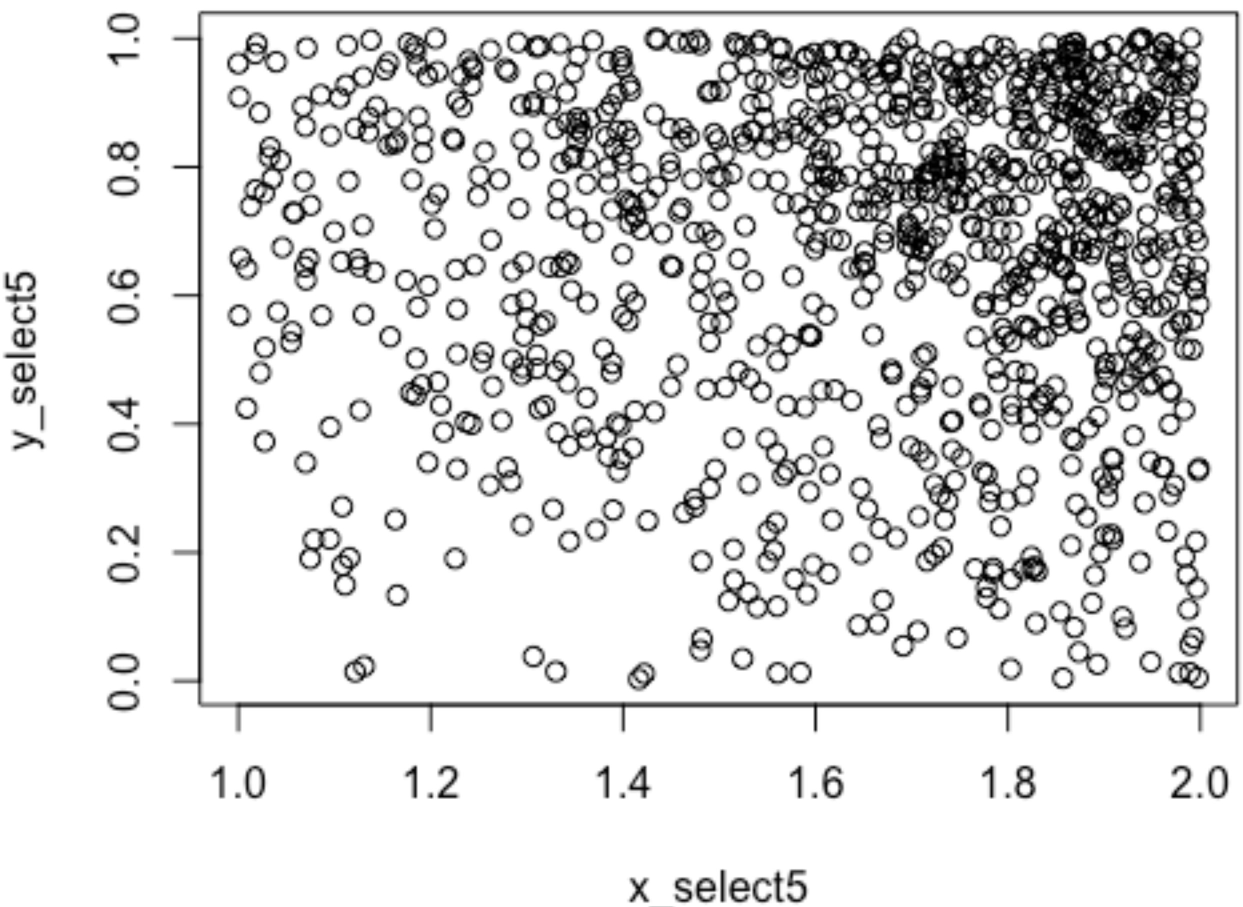
$$= \int_1^2 e^x - 1\, dx$$

$$= e^2 - e - 1$$

$$c = \frac{1}{e^2 - e - 1} \approx 0.272$$

```r
# Question 5
g_5<-function(x,y){
  x*exp(x*y)
}
run_5<-0
f5_x<-rep(0,1000)
x_select5<-c()
y_select5<-c()

for (i in 1:length(f5_x)) {
  repeat{
    x<-runif(1,1,2)
    y<-runif(1,0,1)
    g_x_y<-runif(1,1,g_5(2,1))
    run_5<-run_5+1
    if( 0<=g_x_y & g_x_y<=g_5(x,y)){break}
  }
  x_select5<-c(x_select5,x)
  y_select5<-c(y_select5,y)
}
plot(x_select5,y_select5)
```


y_select5
x_select5
1.0
1.2
1.4
1.6
1.8
2.0
0.0
0.2
0.4
0.6
0.8
1.0