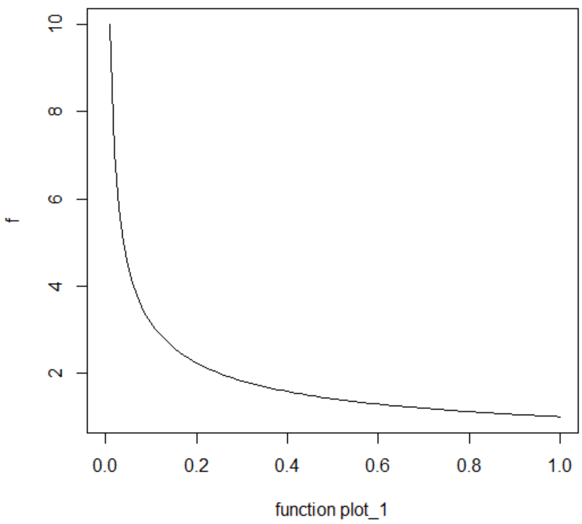
Pooja Voladoddi
Probabilistic Simulation Methods
EE 598 - Project 2b

```
(a)  "x^(-0.5)" ; x in [0.01,1]
```

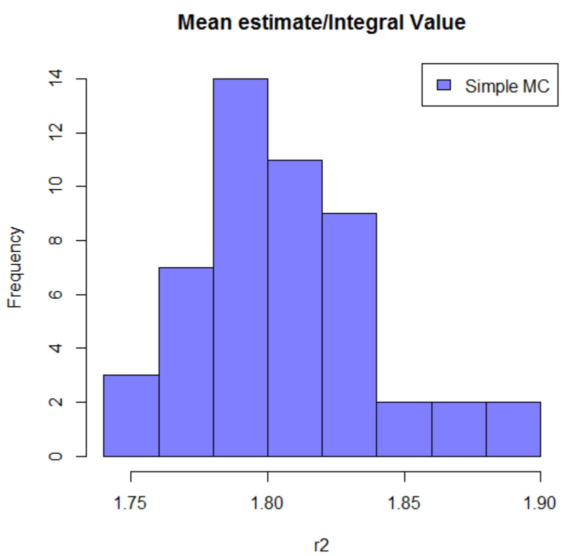
Below is the plot of the function :

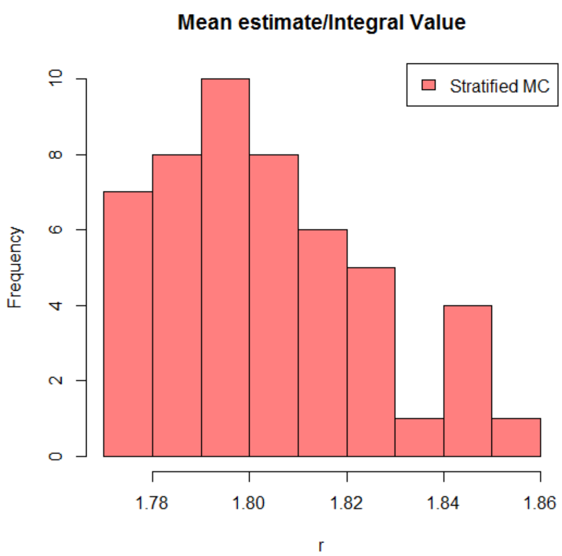


- True Integral value : 1.8

- Monte Carlo Integration yields (1000 samples) : 1.818359
  Variation : 0.001062241

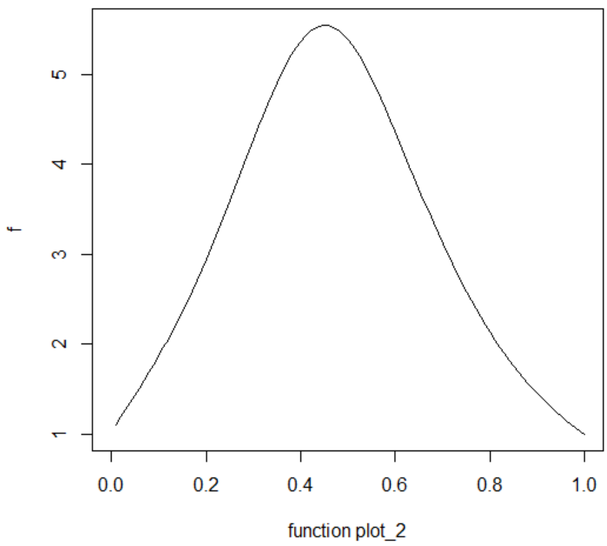
- Stratified Sampling estimate : 1.811035
  Variation (Percentage Decrease) : 0.0004853625(54.31%)





```
(b)   [1+sinh(2x)ln(x)]^-1 ; x in [0.8,3]
```

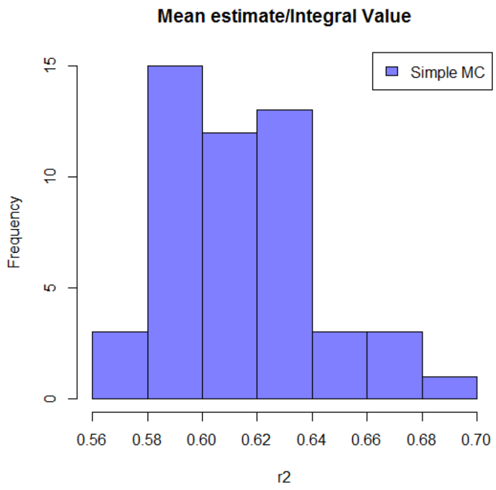
Below is the plot of the function :

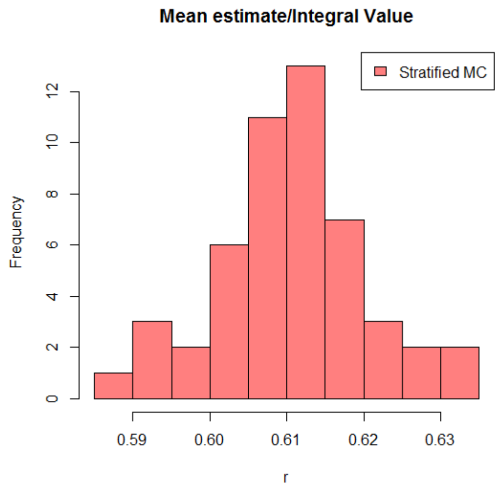


- True Integral value : 0.6095526

- Monte Carlo Integration yields (1000 samples) : 0.6223694
  Variation : 0.0007645782

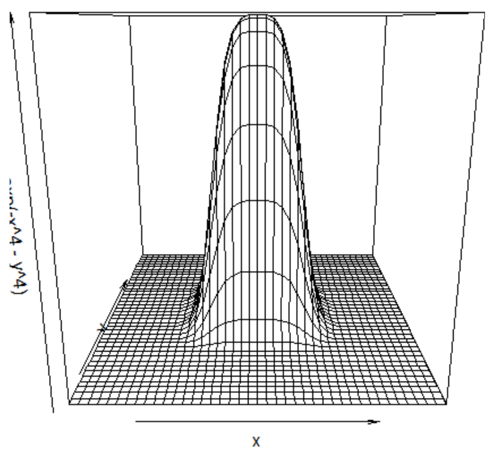
- Stratified Sampling estimate : 0.6107172
  Variation (Percentage Decrease) : 9.072309e-05 (88.13%)






```
(c)  exp(-(x^4) – (y^4))                    ; x,y in [-pi,pi]
```

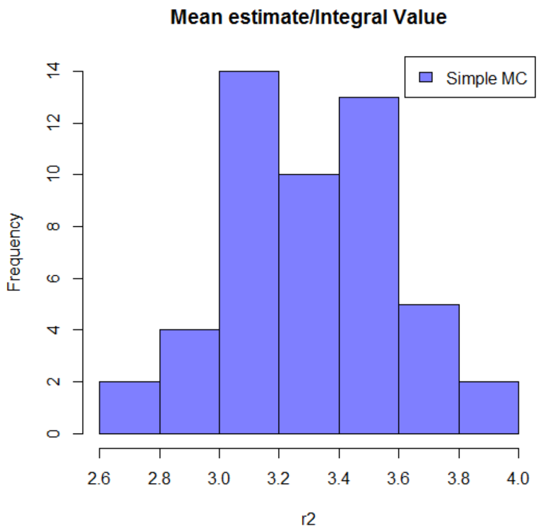
Below is the plot of the function:

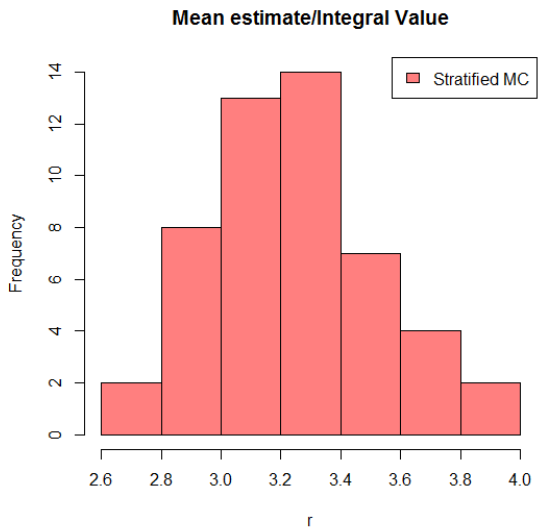


- True Integral value : 3.286266

- Monte Carlo Integration yields (1000 samples) : 2.778039
  Variation : 0.08035447

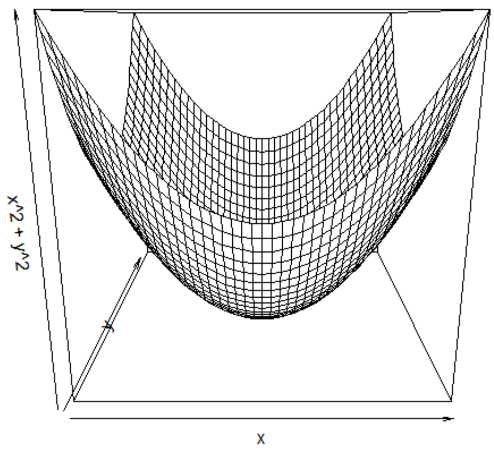
- Stratified Sampling estimate : 3.257167
  Variation (Percentage Decrease) : 0.07477476 (6.94%)





```
(d)  ((x^2) + (y^2))                      ; x,y in [-1,1]
```

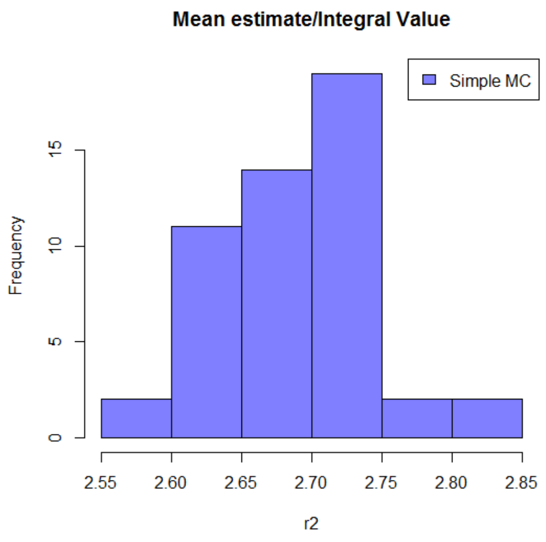
Below is the plot of the function:

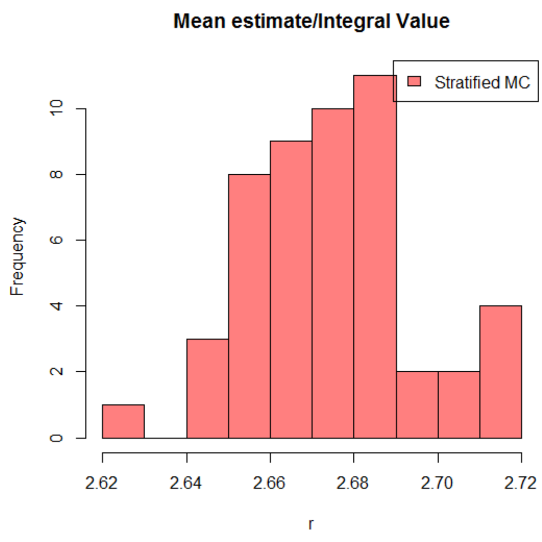


- True Integral value : 2.666667

- Monte Carlo Integration yields (1000 samples) : 2.694137
  Variation : 0.002965507

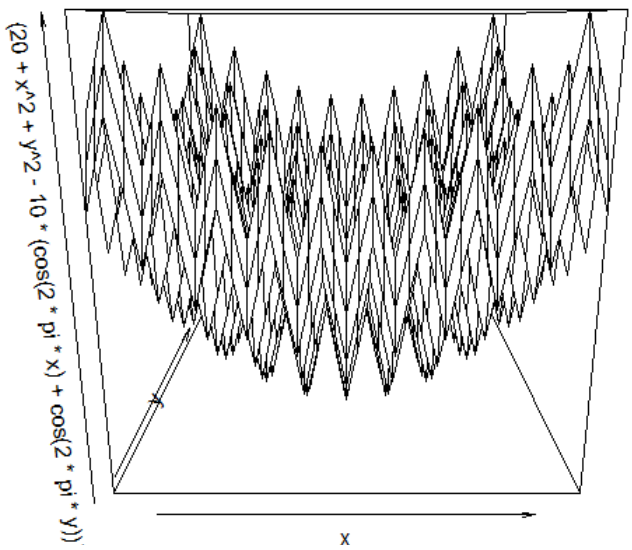
- Stratified Sampling estimate : 2.67433
  Variation (Percentage Decrease) : 0.0003954741(86.67%)





```
(e)  20+(x^2)+(y^2)-10(cos(2pi*x)+cos(2pi*y))          ; x,y in
[-5,5]
```

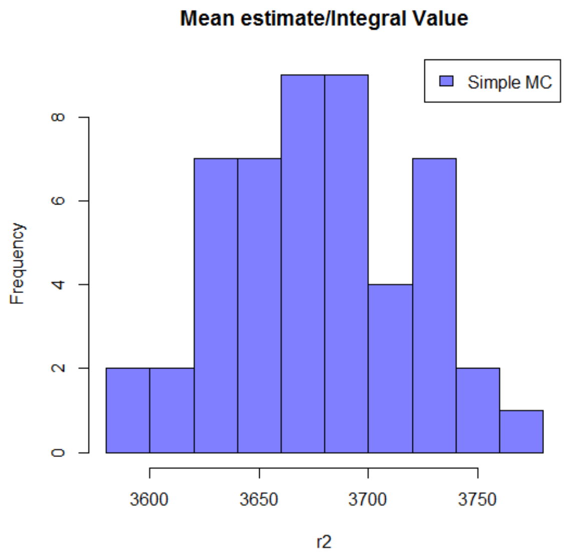
Below is the plot of the function:

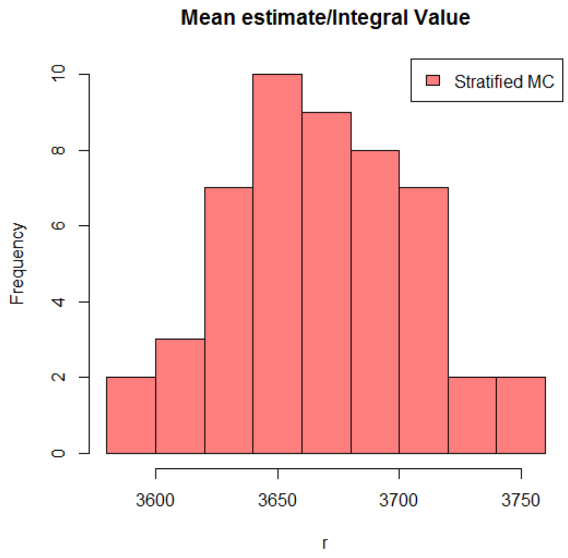


- True Integral value : 3666.667

- Monte Carlo Integration yields (1000 samples) : 3678.238
  Variation : 1833.514

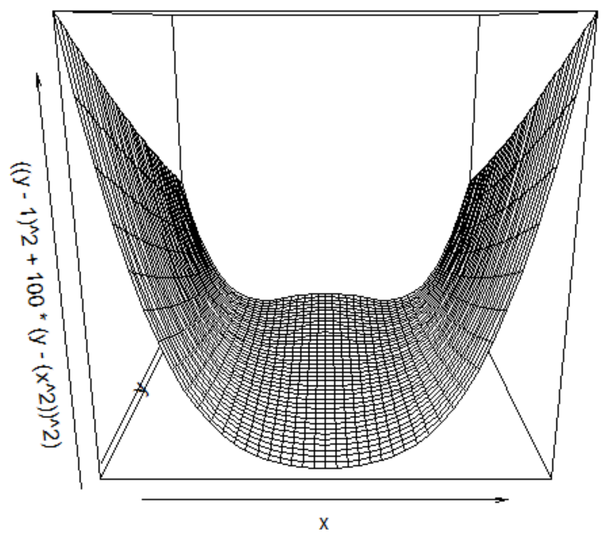
- Stratified Sampling estimate : 3667.708
  Variation (Percentage Decrease) : 1539.72 (16.02%)






```
(f)  (x-1)^2 + 100(y-(x^2))^2                    ; x,y in [-2,2]
```

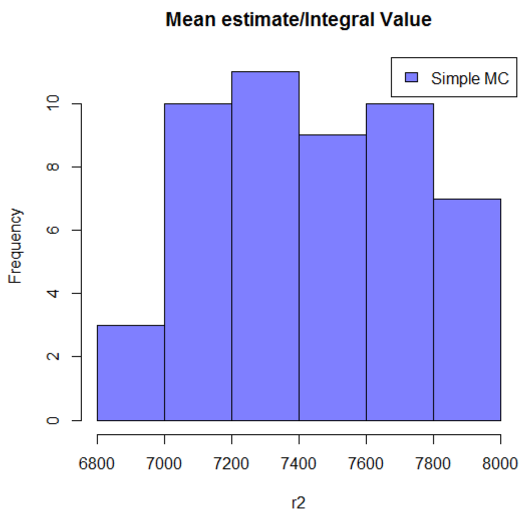
Below is the function plot :

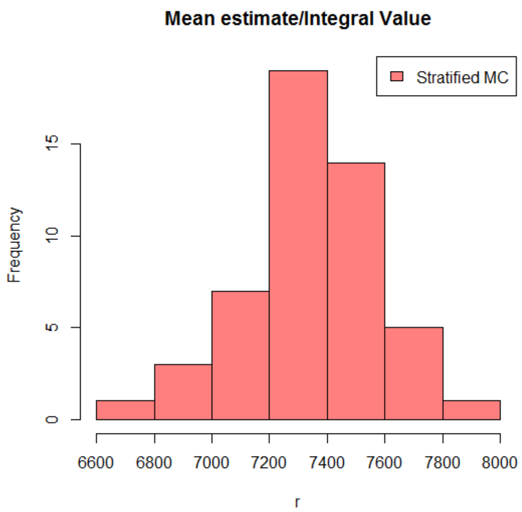


- True Integral value : 7290.667

- Monte Carlo Integration yields (1000 samples) : 7196.672
  Variation : 91851.6

- Stratified Sampling estimate : 7348.153
  Variation (Percentage Decrease) : 54940.54 (40.19%)





DISCUSSION :

The problem statement is to perform Simple Monte Carlo , Stratified Sampling Monte Carlo and Importance Sampling Monte Carlo Integration on the given six functions.

The major disadvantage of the Simple MC is the uniform sampling: many samples from the uniform distribution are wasted and unnecessary for a particular function depending on the shape , this results in highly variant results with inaccuracy.

A more efficient implementation of sampling is allowed for, using the Stratified Sampling and Importance Sampling. The results (plots, histograms, integral estimates and variances) above show the comparison of Simple MC and Stratified MC integration.

An obvious thing to observe would be the variance reduction. However a disadvantage of this method is that the user would require to know how exactly the strata should be spaced, and where the intervals should vary in length for appropriate sampling. Lesser variance reduction implies a not-so-efficient selection of strata or the intervals for sampling, and thus for integration.

Importance Sampling is known to reduce variance even further, but which is absolutely dependant on the selection of the Importance PDF. If the selected PDF varies slightly in mean than it is supposed to, it gives disastrous results when compared to Stratified Sampling. The disadvantage of Importance Sampling is the need to select an absolutely precise PDF to sample from.

Computation wise, Stratified Sampling takes lesser time than Importance Sampling.

There are three appendices attached for this project :
[A] – R code to plot the functions and calculate the true integral value.
[B] – R code to perform simple MC.
[C] – R code to perform Stratified MC .

APPENDIX – A

```
################################### function plots #########################
library(cubature)
library(emdbook)

#*********function 1************
f <- function(x) x^(-0.5)
plot_values <- integrate(f,0.01,1)
plot_values
plot(f,xlab='function plot_1')

#*********function 2************
f <- function(x) 1/(1+sinh(2*x)*log(x))
plot_values <- integrate(f,0.8,3)
plot_values
x11()
plot(f,xlab='function plot_2')

#********function 3*************

f <- function(x) exp(-(x[1]^4)-(x[2]^4))
plot_values <- adaptIntegrate(f,c(-pi,-pi),c(pi,pi),tol=1e-5)
print (plot_values)
x11()
curve3d(exp(-x^4-y^4),from=c(-pi,-pi),to=c(pi,pi))

#******** function 4 ************
f <- function(x) (x[1]^2)+(x[2]^2)
plot_values <- adaptIntegrate(f,c(-1,-1),c(1,1),tol=1e-5)
print (plot_values)
x11()
curve3d(x^2+y^2,from=c(-1,-1),to=c(1,1))

#********* function 5**************
f <- function(x) 20 + x[1]^2 + x[2]^2 - 10*(cos(2*pi*x[1]) + cos(2*pi*x[2]))
plot_values <- adaptIntegrate(f,c(-5,-5),c(5,5),tol=1e-5)
print (plot_values)
x11()
curve3d( (20 + x^2 + y^2 - 10*(cos(2*pi*x) + cos(2*pi*y))),from=c(-5,-5),to=c(5,5))
```

```
#********* function 6**************
f <- function(x)  (x[1]-1)^2 + 100*(x[2]-(x[1]^2))^2
plot_values <- adaptIntegrate(f,c(-2,-2),c(2,2),tol=1e-5)
print (plot_values)
x11()
curve3d( ((y-1)^2+100*(y-(x^2))^2),from=c(-5,-5),to=c(5,5))
```

APPENDIX – B

```
#*********************** SIMPLE MC INTEGRATION******************

#*****************      function 1      *************************
#**************************************************************
f <- function(x) x^ (-0.5)

n <- 1000
a <- 0.01
b <- 1
set.seed(5)

x <- runif(n,a,b)
summing <- sum(f(x))*(b-a)/n
print(summing)

r <- replicate(50, mean(f(runif(n,a,b)))*(b-a))
hist(r)
print (var(r))

#**************************************************************
#***************         function 2      **********************
#**************************************************************
f <- function(x) 1/(1+sinh(2*x)*log(x))

n <- 1000
a <- 0.8
b <- 3
set.seed(5)
x <-  runif(n,a,b)
summing <- sum(f(x))*(b-a)/n
print(summing)

r <- replicate(50, mean(f(runif(n,a,b)))*(b-a))
```

```
x11()
hist(r)
print (var(r))


#*******************************************************************
#**************          function 3          ***********************
#*******************************************************************
f <- function(x,y) exp(-(x^4)-(y^4))
n <- 1000
a <- -pi
b <- pi
set.seed(5)
x <-     runif(n,a,b)
y <-     runif(n,a,b)

summing <- sum(f(x,y))*(b-a)*(b-a)/n
print(summing)

r <- replicate(50, mean(f (runif(n,a,b),runif(n,a,b)) )*(b-a)*(b-a))
x11(); hist(r)
#variance of the evaluated integral
print (var(r))

#**********************************************************************
#*********************      function 4      **************************
f <- function(x,y) ((x^2)+(y^2))
n <- 1000
a <- -1
b <- 1
set.seed(5)
x <-   runif(n,a,b)
y <-   runif(n,a,b)
integral_estimate <- (b-a)*(b-a)*sum(f(x,y))/n
print (integral_estimate)

r <- replicate(50, mean(f (runif(n,a,b),runif(n,a,b)) )*(b-a)*(b-a))
x11()
hist(r)
print (var(r))

#*********************************************************************
#*************************    function 5  **************************
f <- function(x,y) 20 + x^2 + y^2 - 10*(cos(2*pi*x) + cos(2*pi*y))
n <- 1000
a <- -5
b <- 5
set.seed(5)
x <-     runif(n,a,b)
```

```
y <-     runif(n,a,b)
integral_estimate <- (b-a) * (b-a) * sum(f(x,y)) / n
print(integral_estimate)
x11(); r <- replicate(50, mean(f (runif(n,a,b),runif(n,a,b)) )*(b-a)*(b-a))

hist(r)                                                  #histogram plot of 'integral
estimates'
print (var(r))        #variance of the evaluated integral

#*************************************************************************
#*************************    function 6    ***************************
f <- function(x,y)  (x-1)^2 + 100*(y-(x^2))^2
n <- 1000
a <- -2
b <- 2
set.seed(5)
x <-     runif(n,a,b)
y <-     runif(n,a,b)
integral_estimate <- (b-a)*(b-a)*sum(f(x,y))/n
print(integral_estimate)

r <- replicate(50, mean(f (runif(n,a,b),runif(n,a,b)) )*(b-a)*(b-a))
x11(); hist(r)                #histogram plot of 'integral estimates'
print(var(r))            #variance of estiamtes over 50 runs

**************************************************************************
*
```

APPENDIX – C

```
#####******** stratified sampling************

#*************************************************************
########################## Function 1 #########################
f <- function(x) x^(-0.5)

set.seed(5)
                                                n <-900
                                                a <- 0.01
                                                b <- 0.5
r1 <- replicate(50, mean(f(runif(n,a,b)))*(b-a))
                                                n <-100
                                                a <- 0.5
                                                b <- 1
r2 <- replicate(50, mean(f(runif(n,a,b)))*(b-a))

#mean estimate on one run
mean_estimate <- (r1+r2)
print (mean_estimate[1])

r <- (r1+r2)
```

```
hist(r)

#variance of the evaluated integral on fifty runs
print (var(r1+r2))




##########################################################################
###################### function 2 ####################################

f <- function(x) 1/(1+sinh(2*x)*log(x))
# 100 700 200
set.seed(5)

                                             n <-100
                                             a <- 0.8
                                             b <- 1
r1 <- replicate(50, mean(f(runif(n,a,b)))*(b-a))
                                             n <-700
                                             a <- 1
                                             b <- 2
r2 <- replicate(50, mean(f(runif(n,a,b)))*(b-a))
                                             n <-200
                                             a <- 2
                                             b <- 3
r3 <- replicate(50, mean(f(runif(n,a,b)))*(b-a))

#mean estimate
print (mean(r1)+mean(r2)+mean(r3))

#mean estimate _ 2
print (r1[1]+r2[1]+r3[1])

#variance of the evaluated integral
print (var(r1+r2+r3))

rr <- r1 + r2 + r3
hist(rr)

##########################################################################
###################### function 3 ####################################
f <- function(x,y) exp(-(x^4)-(y^4))
####
n <- 500
a1 <- -pi
b1 <- 0

a2 <- -pi
```

```
b2 <- pi
set.seed(5)
r1 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))

####
n <-500
a1 <- 0
b1 <- pi

a2 <- -pi
b2 <- pi
set.seed(5)
r2 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))

#mean estimate
print (mean(r1)+mean(r2))

#mean estimate _ 3
print (r1[1]+r2[1])

#variance of the evaluated integral
print (var(r1+r2))

#########**************************************************##########
#########*********function 4****************************###########
set.seed(5)
f <- function(x,y) ((x^2)+(y^2))

                                                  n <- 250
                                                  a1 <- -1
                                                  b1 <- -0.97

                                                  a2 <- -1
                                                  b2 <- 1

r1 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))

                                                  n <- 5000
                                                  a1 <- -0.97
                                                  b1 <- 0.97

                                                  a2 <- -1
                                                  b2 <- 1

r2 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))

                                                  n <- 250
                                                  a1 <- 0.97
                                                  b1 <- 1
```

```
                                  a2 <- -1
                                  b2 <- 1
r3 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))

#mean estimate
print (mean(r1)+mean(r2)+mean(r3))

#mean estimate _ 4
print (r1[1]+r2[1]+r3[1])

#variance of the evaluated integral
print (var(r1+r2+r3))

#######*************************************************########
#######******************function 5**************************
f <- function(x,y) 20 + x^2 + y^2 - 10*(cos(2*pi*x) + cos(2*pi*y))

set.seed(5)

                                  n <-200
                                  a1 <- -5
                                  b1 <- -3

                                  a2 <- -5
                                  b2 <- 5
r1 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b2-a2)*(b1-a1))

                                  n <-300
                                  a1 <- -3
                                  b1 <- 0

                                  a2 <- -5
                                  b2 <- 5
r2 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b2-a2)*(b1-a1))


                                  n <-300
                                  a1 <- 0
                                  b1 <- 3

                                  a2 <- -5
                                  b2 <-  5
r3 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b2-a2)*(b1-a1))

                                  n <-200
                                  a1 <- 3
                                  b1 <- 5
```

```
                    a2 <- -5
                    b2 <-  5
r4 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b2-a2)*(b1-a1))


#mean estimate
print (mean(r1)+mean(r2)+mean(r3)+mean(r4))

#variance of the evaluated integral
print (var(r1+r2+r3+r4))


#######*************************************************######
####################****function 6 ********************##########
f <- function(x,y) (x-1)^2 + 100*(y-(x^2))^2

set.seed(5)
                    n <- 100
                    a1 <- -2
                    b1 <- -1.5

                    a2 <- -2
                    b2 <- 2
r1 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))

                    n <- 350
                    a1 <- -1.5
                    b1 <- -0.5

                    a2 <- -2
                    b2 <- 2
r2 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))

                    n <- 100
                    a1 <- -0.5
                    b1 <- 0.5

                    a2 <- -2
                    b2 <- 2
r3 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))


                    n <- 350
                    a1 <- 0.5
                    b1 <- 1.5

                    a2 <- -2
                    b2 <- 2
r4 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))
```

```
n <- 100
a1 <- 1.5
b1 <- 2

a2 <- -2
b2 <- 2
r5 <- replicate(50, (mean(f(runif(n,a1,b1),runif(n,a2,b2))))*(b1-a1)*(b2-a2))

#mean estimate
print (mean(r1)+mean(r2)+mean(r3)+mean(r4)+mean(r5))

#variance of the evaluated integral
print (var(r1+r2+r3+r4+r5))

rr <- r1 + r2 + r3 + r4 + r5
```