

#1) _____

```
#setting n to 1000
n <- 1000

#setting random seed to 1
set.seed(1)

#setting x to a random uniform of n values, from 0 to 1
x <- runif(n, 0, 1)

#generating the mean of x
mean(x)
#mean is 0.4996917

#generating the variance of x
var(x)
#variance is 0.08316708

#generating the cumulative sum of x
cs <- cumsum(x)

#plotting 1:n vs cs/1:n as both type l
plot(1:n, cs/1:n, type = "l")

#adding horizontal line with height .5
lines(c(0,n),c(0.5,0.5))
```

#2) _____

```
#function for piecewise function
#f(x) { a >= 1 and a <= 0, f(a) = 1 otherwise f(a) = 0
#so the integration function becomes either 1a da or 0a da
f = function(a) {
  ifelse(a >= 0 & a <= 1, a*1, a*0)
}

#integration of f(x) for mean of uniform(0,1)
integrate(f,-Inf, Inf)
#the mean is 0.5

#function for piecewise function
#g(x) { a >= 1 and a <= 0, f(a) = 1 otherwise g(a) = 0
#so the integration function becomes either (a-mean)^2*1 da or
# (a-mean)^2*0 da
g = function(a) {
  ifelse(a >= 0 & a <= 1, (a-(integrate(f,-Inf, Inf)$value))^2*1,
        (a-(integrate(f,-Inf, Inf)$value))^2*0)
}
```

```
#integration of g(x) for mean of uniform(0,1)
integrate(g,-Inf, Inf)
#mean is 0.8333333
```

#3)

```
#set n to 1,000,000
n <- 1000000
```

```
#create x random uniform(-1,1) of n values
x <- runif(n, -1, 1)
```

```
#create y random uniform(-1,1) of n values
y <- runif(n, -1, 1)
```

```
#sum all values where  $x^2 + y^2 < 1$ 
sum(x^2 + y^2 < 1)
#number of points that satisfy is 785630
#figure is a circle
```

```
#area of circle  $x^2 + y^2 < 1$ 
radius <- 1
pi * radius^2
#given area is 3.141593
#exact area is pi
```

```
#ratio of points within  $x^2 + y^2 < 1$  vs. n
sum(x^2 + y^2 < 1) / n
#this gives 0.78563
```

```
#answer square multiplied by ratio of points vs n
4 * (sum(x^2 + y^2 < 1) / n)
#this gives 3.14252
```

#4)

```
#set n to 1,000,000
n <- 1000000
```

```
#create x random uniform(-1,1) of n values
x <- runif(n, -1, 1)
```

```
#create y random uniform(-1,1) of n values
y <- runif(n, -1, 1)
```

```
#create z random uniform(-1,1) of n values
z <- runif(n, -1, 1)
```

```
#ratio of points within
sum(x^2 + y^4 + z^6 < 1) / n
```

```
#volume of the shape
8 * (sum(x^2 + y^4 + z^6 < 1) / n)
#volume is 6.1428
```

```
#5)
```

```
#set n = 10,000
n=10000
```

```
#set count to 0
count=0
```

```
#loop to check if random 3 segments form a triangle and increments
count if true
for (t in 1:n)
{
```

```
  #creates random uniform(0,1) of 2 values
  points=runif(2,0,1)
```

```
  #sort the two points in ascending order
  x <- sort(points)
```

```
  #calculates 3 segment lengths
  unsortSegs <- c(x[1]-0,x[2]-x[1],1-x[2])
```

```
  #sorts the segments in ascending order
  segments <- sort(unsortSegs)
```

```
  #calculates lengths of the 3 segments and increments count if
  #the sum of the difference between the 1st point and 0 and
  #the difference between the 2nd point and the 1st point is
  #greater than the difference between 1 and the 1st point
  count=count+(segments[1] + segments[2] > segments[3])
```

```
}
```

```
count/n
#the ratio is approx 0.25
```