Mini Project 1

Names of group members: Karan Risbud(KSR190005), Shubham Vartak(SXV200115)

Contribution of each group member: Both the Project group members worked together on the project. Collaborated to solve the problem and implementation of R programming.

1) Consider Exercise 4.11 from the textbook. In this exercise, let XA be the lifetime of block A, XB be the lifetime of block B, and T be the lifetime of the satellite. The lifetimes are in years. It is given that XA and XB follow independent exponential distributions with mean 10 years. One can follow the solution of Exercise 4.6 to show that the probability density function of T is fT (t) = (0.2 exp(−0.1t) − 0.2 exp(−0.2t), 0 ≤ t < ∞, 0, otherwise, and E(T) = 15 years.

1 A) Use the above density function to analytically compute the probability that the lifetime of the satellite exceeds 15 years.

---> It’s given that,
E(XA)=10 Years
E(XB)= 10 Years
We have to find P(T>15)
It can be written as,
1-P(T<=15)
= > 1 - F(15)
= > 1 - Integral {0.2e^-0.1t – 0.2e^-0.2t dt} from 0 to 15
= > 1 – [-2e^-0.1t + e^-0.2t] from 0 to 15
= > 1 – [(-2e^-0.1*15 + e^-0.2*15) – (-2e^-0.1*0 + e^-0.2*0)]
= > 1 – [-2e^-1.5 + e^-3 + 2e^0 – e^0]
= > 1 – e^-3 + 2e^-1.5 – 1
= > 0.39647

Thus, probability of satellite’s lifetime exceeding 15 years is 0.39647

1 B) Use the following steps to take a Monte Carlo approach to compute E(T) and P(T > 15).

1 B i) Simulate one draw of the block lifetimes XA and XB. Use these draws to simulate one draw of the satellite lifetime T.

--->Here we calculate Life time of the satellite by taking the maximum between XA and XB.
We use rexp function to generate draws of XA and XB.
The rate here is specified as 0.1 because the mean lifetime of XA and XB is 10 years so that makes rate=1/10.
Code:
LifeTime.T = max(rexp(n=1,rate=0.1), rexp(n=1,rate=0.1))
LifeTime.T

```
Console  Terminal  Jobs
R 4.1.1 · ~/
> LifeTime.T = max(rexp(n=1,rate=0.1),rexp(n=1,rate=0.1))
> LifeTime.T
[1] 14.52378
>
```

1 B ii) Repeat the previous step 10,000 times. This will give you 10,000 draws from the distribution of T. Try to avoid 'for' loop. Use 'replicate' function instead. Save these draws for reuse in later steps. [Bonus: 1 bonus point for not taking more than 1 line of code for steps (i) and (ii).]

Here we repeat the above step 10,000 times using replicate function.

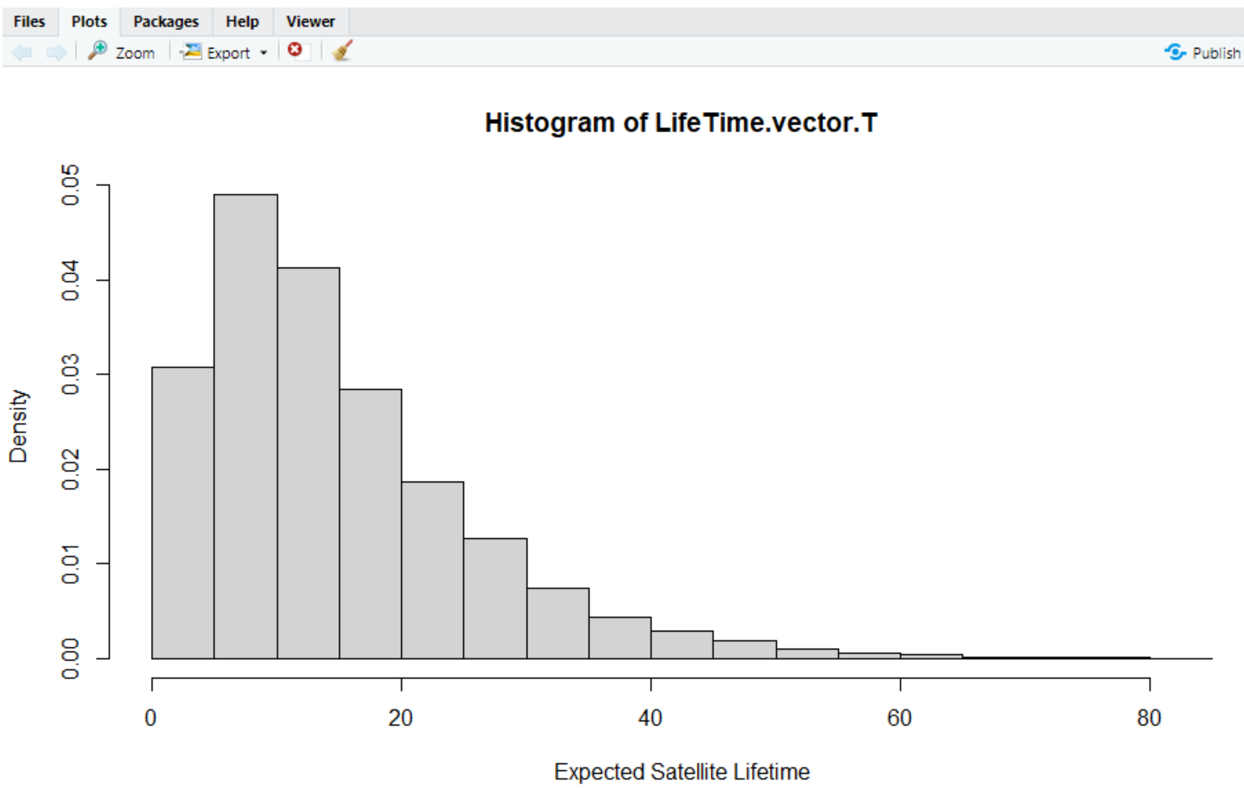
LifeTime.vector.T = replicate(n=10000,
max(rexp(n=1,rate=0.1),
rexp(n=1,rate=0.1)))
head(LifeTime.vector.T)

```
Console  Terminal  Jobs
R 4.1.1 · ~/
> LifeTime.vector.T = replicate(n=10000,
+                               max(rexp(n=1,rate=0.1),
+                                   rexp(n=1,rate=0.1)))
> head(LifeTime.vector.T)
[1]  8.974255  7.469344 13.572334  5.938948 17.231071 10.529070
>
```

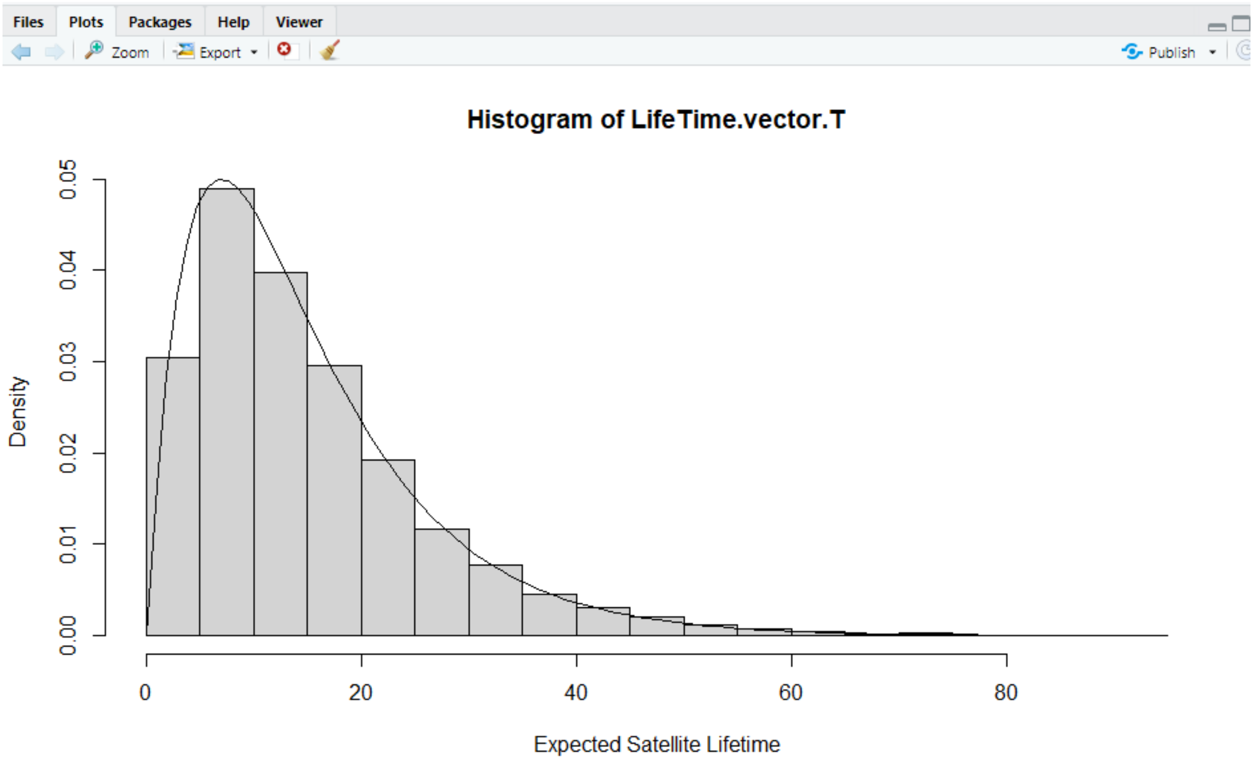
1 B iii) Make a histogram of the draws of T using 'hist' function. Superimpose the density function given above. Try using 'curve' function for drawing the density. Note what you see.

Here, we used hist function to generate a histogram. We used the curve function to superimpose the density curve. In the curve function, we used add=T parameter to superimpose the pdf curve on the existing histogram curve.

```
hist(LifeTime.vector.T,probability = T,
   xlab = "Expected Satellite Lifetime")
```




```
hist(LifeTime.vector.T,probability = T,
   xlab = "Expected Satellite Lifetime")
PDF.F.T = function(a) 0.2*exp(-0.1*a) - 0.2*exp(-0.2*a)
curve(PDF.F.T,add = T)
```



1 B iv) Use the saved draws to estimate E(T). Compare your answer with the exact answer given above.

Using the simulation technique, we generated the mean 15.033 which is very close to the calculated mean of 15 years.

mean(LifeTime.vector.T)

```
> #Q1.b.4
> mean(LifeTime.vector.T)
[1] 15.03358
>
```

1 B v) Use the saved draws to estimate the probability that the satellite lasts more than 15 years. Compare with the exact answer computed in part (a).

Prob = sum(LifeTime.vector.T>15)/length(LifeTime.vector.T)
Prob

```
Console   Terminal ×   Jobs ×
R 4.1.1 · ~/
> Prob = sum(LifeTime.vector.T>15)/length(LifeTime.vector.T)
> Prob
[1] 0.3943
>
```

Using the simulation technique, we generated the P(T>15) = 0.3943 which is very close to the calculated P(T>15) = 0.39647

1 B vi) Repeat the above process of obtaining an estimate of E(T) and an estimate of the probability four more times. Note what you see.

```
for(i in 1:4)
{
 LifeTime.vector.T = replicate(n=10000,  max(rexp(n=1,rate=0.1),rexp(n=1,rate=0.1)))
 mean = mean(LifeTime.vector.T)
 Prob = sum(LifeTime.vector.T>15)/length(LifeTime.vector.T)
 print(paste("E(T) = ",mean, "Probability of Satellite lasting more than 15 years = ",Prob))

}
```

```
Console  Terminal ×  Jobs ×
R 4.1.1 · ~/

> hist(LifeTime.vector.T,probability = T,
+      xlab = "Expected Satellite Lifetime")
> for(i in 1:4)
+ {
+   LifeTime.vector.T = replicate(n=10000,
+                               max(rexp(n=1,rate=0.1),
+                                   rexp(n=1,rate=0.1)))
+   mean = mean(LifeTime.vector.T)
+   Prob = sum(LifeTime.vector.T>15)/length(LifeTime.vector.T)
+   hist(LifeTime.vector.T,probability = T,
+        xlab = "Expected Satellite Lifetime")
+   PDF.F.T = function(a) 0.2*exp(-0.1*a) - 0.2*exp(-0.2*a)
+   curve(PDF.F.T,add = T)
+   print(paste("E(T) = ",mean, "Probability of Satellite lasting more than 15 years = ",Prob))
+
+ }
[1] "E(T) =  14.883925118184 Probability of Satellite lasting more than 15 years =  0.3918"
[1] "E(T) =  15.1294586032809 Probability of Satellite lasting more than 15 years =  0.4021"
[1] "E(T) =  15.0360614259894 Probability of Satellite lasting more than 15 years =  0.4001"
[1] "E(T) =  14.9839690506514 Probability of Satellite lasting more than 15 years =  0.3947"
> |
```

Observation:

After simulating 4 more times, we get 4 different values of mean and probability which are close to the calculated probability and the mean.

1 C) Repeat part (vi) five times using 1,000 and 100,000 Monte Carlo replications instead of 10,000. Make a table of results. Comment on what you see and provide an explanation.

```
#For n = 1000
for(i in 1:5)
{
  LifeTime.vector.T = replicate(n=1000,
                          max(rexp(n=1,rate=0.1),
                            rexp(n=1,rate=0.1)))
  mean = mean(LifeTime.vector.T)
  Prob = sum(LifeTime.vector.T>15)/length(LifeTime.vector.T)
  print(paste("E(T) = ",mean, "Probability of Satellite lasting more than 15 years = ",Prob,"where N =
",1000))

}
```

```
> #For n = 1000
> for(i in 1:5)
+ {
+   LifeTime.vector.T = replicate(n=1000,
+                               max(rexp(n=1,rate=0.1),
+                                   rexp(n=1,rate=0.1)))
+   mean = mean(LifeTime.vector.T)
+   Prob = sum(LifeTime.vector.T>15)/length(LifeTime.vector.T)
+   print(paste("E(T) = ",mean, "Probability of Satellite lasting more than 15 years = ",Prob,"where N = ",1000))
+
+ }
[1] "E(T) =  14.5589413703001 Probability of Satellite lasting more than 15 years =  0.392 where N =  1000"
[1] "E(T) =  14.8588764138867 Probability of Satellite lasting more than 15 years =  0.395 where N =  1000"
[1] "E(T) =  14.6765354418693 Probability of Satellite lasting more than 15 years =  0.378 where N =  1000"
[1] "E(T) =  14.9186470593604 Probability of Satellite lasting more than 15 years =  0.386 where N =  1000"
[1] "E(T) =  14.8356899144774 Probability of Satellite lasting more than 15 years =  0.388 where N =  1000"
> 
```

```
#For n = 100000
for(i in 1:5)
{
  LifeTime.vector.T = replicate(n=100000,
                          max(rexp(n=1,rate=0.1),
                            rexp(n=1,rate=0.1)))
  mean = mean(LifeTime.vector.T)
  Prob = sum(LifeTime.vector.T>15)/length(LifeTime.vector.T)
  print(paste("E(T) = ",mean, "Probability of Satellite lasting more than 15 years = ",Prob,"where N =
",100000))

}
```

```
> #For n = 100000
> for(i in 1:5)
+ {
+   LifeTime.vector.T = replicate(n=100000,
+                                 max(rexp(n=1,rate=0.1),
+                                     rexp(n=1,rate=0.1)))
+   mean = mean(LifeTime.vector.T)
+   Prob = sum(LifeTime.vector.T>15)/length(LifeTime.vector.T)
+   print(paste("E(T) = ",mean, "Probability of Satellite lasting more than 15 years = ",Prob,"where N = ",10000
0))
+
+ }
[1] "E(T) =  15.0109123377279 Probability of Satellite lasting more than 15 years =  0.39662 where N =  1e+05"
[1] "E(T) =  14.968761418814 Probability of Satellite lasting more than 15 years =  0.39667 where N =  1e+05"
[1] "E(T) =  15.0430726423364 Probability of Satellite lasting more than 15 years =  0.39915 where N =  1e+05"
[1] "E(T) =  15.0551293245203 Probability of Satellite lasting more than 15 years =  0.39802 where N =  1e+05"
[1] "E(T) =  14.9929296898005 Probability of Satellite lasting more than 15 years =  0.39659 where N =  1e+05"
> |
```

Comparison Table:

	N=1,000		N=10,000		N=100,000	
Tests	**Mean**	**Probability**	**Mean**	**Probability**	**Mean**	**Probability**
1	14.5589	0.392	15.0335	0.3943	15.0109	0.39662
2	14.8588	0.395	14.8839	0.3918	14.9687	0.39667
3	14.6765	0.378	15.1294	0.4021	15.0431	0.39915
4	14.9186	0.386	15.0361	0.4001	15.0551	0.39802
5	14.8356	0.388	14.9839	0.3947	14.9929	0.39659

Observation:

From the table we observe that as the number of simulations increase, the values get closer to the actual calculated results. Thus, the accuracy increases with the increase in the number of simulations.

2) (10 points) Use a Monte Carlo approach estimate the value of π based on 10, 000 replications. [Ignorable hint: First, get a relation between π and the probability that a randomly selected point in a unit square with coordinates — (0, 0), (0, 1), (1, 0), and (1, 1) — falls in a circle with center (0.5, 0.5) inscribed in the square. Then, estimate this probability, and go from there.]

- The probability that a point lies inside a circle which is inscribed in a square is given by SA (circle)/SA(Square) where SA -> Surface Area. The probability turns out to be pie/4
- Thus pie = 4 * Probability.
- Equation of a Circle is given by (x-a) ^2 + (y-b) ^2 = r^2 where (a, b) is center of circle and r is radius.
- For the point to lie inside the circle we get (x-a) ^2 + (y-b) ^2 <= r^2
- We generate 10000 uniform sample points between (0,1) for x & y coordinates.
- We take center = (0.5,0.5) and r = 1

R-Code

```
#number of iterations=10000
#Uniformly generated n random variable from 0,1
x.coordinate = runif(n,min=0,max=1)
y.coordinate = runif(n,min=0,max=1)
points.inside.circle = length(which((x.coordinate-0.5)^2 + (y.coordinate-0.5)^2 <= 0.5^2))
#Probability is points inside circle/ total no. of points
probability = points.inside.circle/n
result = 4 * probability
result
```

R-Code-Output:

```
Console  Terminal  Jobs
R 4.1.1 · ~/
> #Uniformly generated n random variable from 0,1
> x.coordinate = runif(n,min=0,max=1)
> y.coordinate = runif(n,min=0,max=1)
> points.inside.circle = length(which((x.coordinate-0.5)^2 + (y.coordinate-0.5)^2 <=
 0.5^2))
> #Probability is points inside circle/ total no. of points
> probability = points.inside.circle/n
> result = 4 * probability
> result
[1] 3.1388
```