Mini-Project duo group 7

Venkata Bapanapalli

Karthik Meyyappan

Contributions: Both members equally contributed to analytically solve and implement the code of the given two questions.

1 a. P(T>15) = 1 – P(T<=15)

$$= 1 - \int_0^{15} f_T(t)\ dt$$

$$= 1 - \int_0^{15} 0.2e^{(-0.1t)} - 0.2e^{(-0.2t)}\ dt$$

$$= 1 - \genfrac{}{}{0pt}{}{15}{0}\left[0.2\left(\frac{e^{-0.1t}}{-0.1} - \frac{e^{-0.2t}}{-0.2}\right)\right]$$

$$= 1 - \genfrac{}{}{0pt}{}{15}{0}\left[-2e^{-0.1t} + e^{-0.2t}\right]$$

$$= 1 - \left[\left(-2e^{-0.1(15)} + e^{-0.2(15)}\right) - \left(-2e^{-0.1(0)} + e^{-0.2(0)}\right)\right]$$

$$= 1 - \left[-2e^{-1.5} + e^{-3} + 2e^{0} - e^{0}\right]$$

$$= 1 - \left[e^{-3} + 2e^{-1.5} + 1\right]$$

$$= 1 - [0.6035] = \mathbf{0.3965}$$

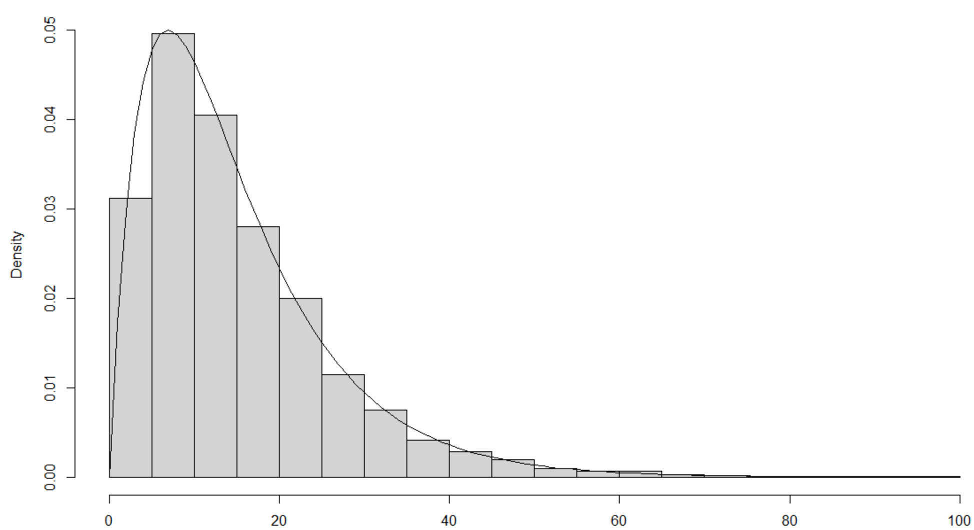
Analytically solved the above problem using integration and algebraic simplification.

TEST 1:

```
Xa = rexp(n=1, rate=1/10)
Xb = rexp(n=1, rate=1/10)
x = max(Xa, Xb)
ten_thousand= replicate(10000, max(rexp(n=1, rate=1/10), rexp(n=1, rate=1/10)))
hist(ten_thousand, prob = TRUE)
pdf = function(x){
        return (0.2*exp(-0.1*x)-0.2*exp(-0.2*x))
}
curve(pdf(x), add = TRUE)

mean(ten_thousand)
```

[1] 14.93673

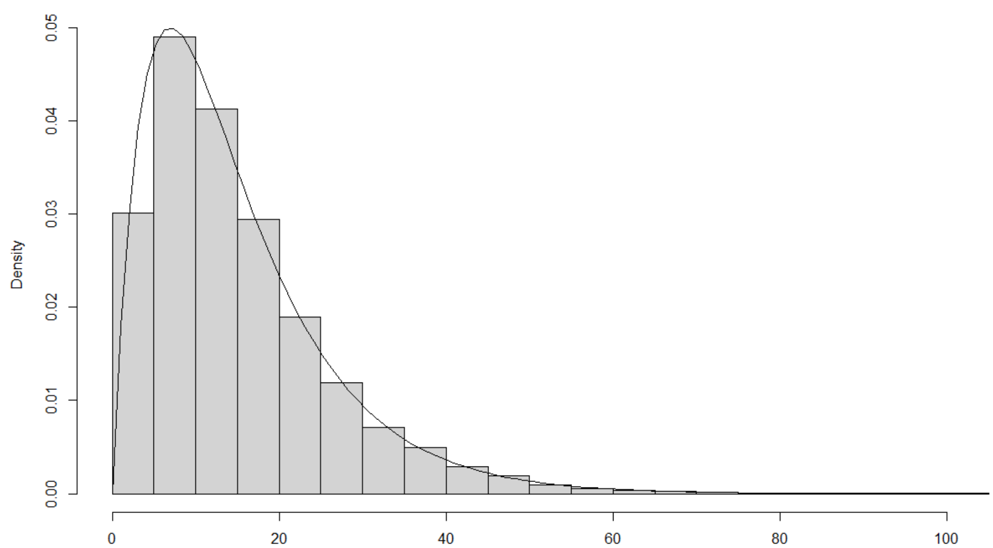


TEST 2:

```
Xa = rexp(n=1, rate=1/10)
Xb = rexp(n=1, rate=1/10)
x = max(Xa, Xb)
ten_thousand= replicate(10000, max(rexp(n=1, rate=1/10), rexp(n=1, rate=1/10)))
hist(ten_thousand, prob = TRUE)
pdf = function(x){
	return (0.2*exp(-0.1*x)-0.2*exp(-0.2*x))
}
curve(pdf(x), add = TRUE)

mean(ten_thousand)
```

[1] 15.0346

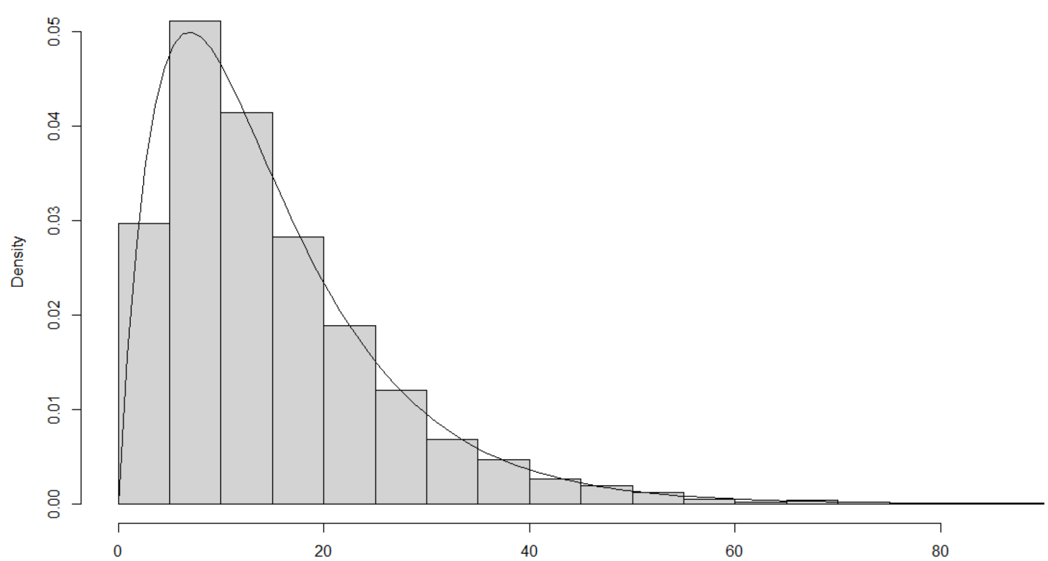


TEST 3:

```
Xa = rexp(n=1, rate=1/10)
Xb = rexp(n=1, rate=1/10)
x = max(Xa, Xb)
ten_thousand= replicate(10000, max(rexp(n=1, rate=1/10), rexp(n=1, rate=1/10)))
hist(ten_thousand, prob = TRUE)
pdf = function(x){
        return (0.2*exp(-0.1*x)-0.2*exp(-0.2*x))
}
curve(pdf(x), add = TRUE)

mean(ten_thousand)
```

[1] 14.88006

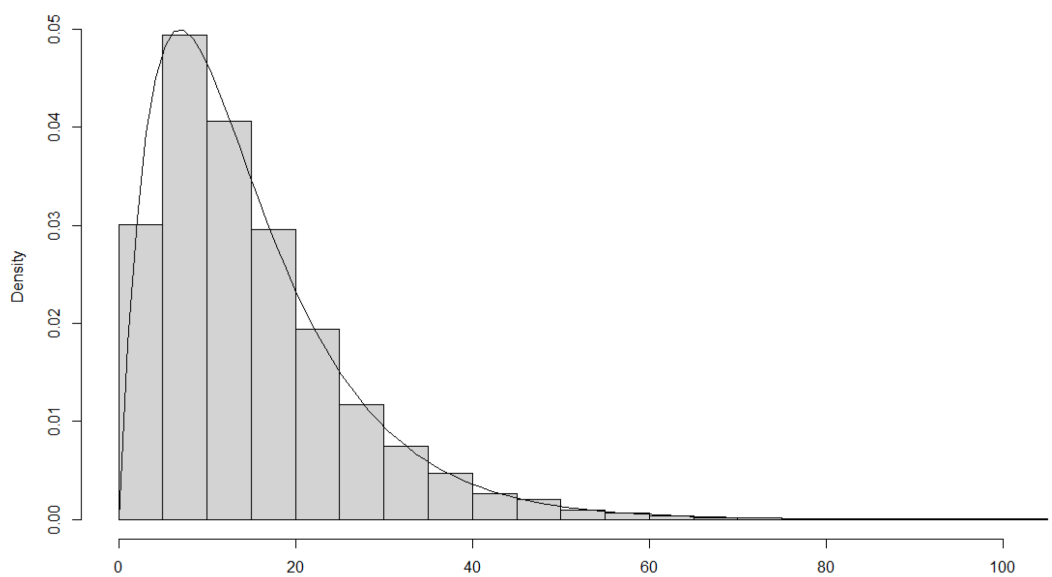


TEST 4:

```
Xa = rexp(n=1, rate=1/10)
Xb = rexp(n=1, rate=1/10)
x = max(Xa, Xb)
ten_thousand= replicate(10000, max(rexp(n=1, rate=1/10), rexp(n=1, rate=1/10)))
hist(ten_thousand, prob = TRUE)
pdf = function(x){
	return (0.2*exp(-0.1*x)-0.2*exp(-0.2*x))
}
curve(pdf(x), add = TRUE)

mean(ten_thousand)
```

[1] 15.00983

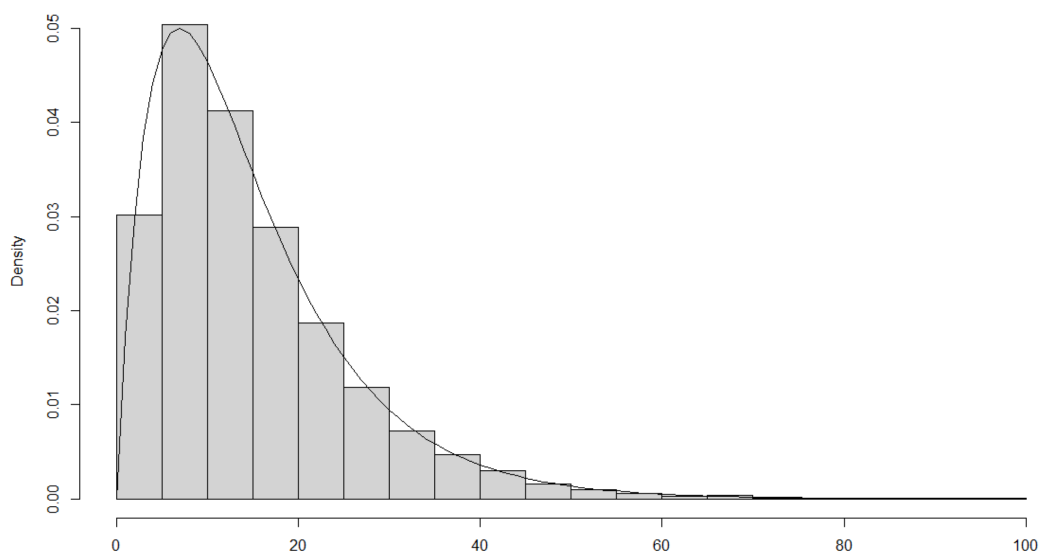


TEST 5:

```
Xa = rexp(n=1, rate=1/10)
Xb = rexp(n=1, rate=1/10)
x = max(Xa, Xb)
ten_thousand= replicate(10000, max(rexp(n=1, rate=1/10), rexp(n=1, rate=1/10)))
hist(ten_thousand, prob = TRUE)
pdf = function(x){
	return (0.2*exp(-0.1*x)-0.2*exp(-0.2*x))
}
curve(pdf(x), add = TRUE)

mean(ten_thousand)
```

[1] 14.86456



1,000 replications:	Mean	Probability
Test 1	15.27268	0.3745067
Test 2	14.81808	0.3633907
Test 3	14.99583	0.3677772
Test 4	14.68535	0.3600811
Test 5	14.6079	0.3581364

100,000 replications:	Mean	Probability
Test 1	14.99465	0.3677482
Test 2	14.96038	0.3669065
Test 3	14.96291	0.3669686
Test 4	15.0092	0.3681049
Test 5	14.98172	0.3674308

We can conclude from the collected data that as we increased the sample size from 1,000 to 100,000 the E(T) and P(T >15) reduces in variation. The larger the sample size the closer the mean is to 15.

2.

First, the probability of the number of points that fall within the circle that is inscribed in the square needs to be found. The probability of that would be the number of points that satisfy the function,

$$x^2 + y^2 \leq 1\ ,\ 0 \leq x \leq 1 \text{ and } 0 \leq y \leq 1$$

divided by the number of total points generated, in this case 10000.

$$\frac{\pi}{4} = \frac{x^2 + y^2 \leq 1}{10000}$$

Code:

```
x = runif(10000, min=0, max=1)
y = runif(10000, min=0, max=1)
inscribed = (x-0.5)^2 + (y-0.5)^2 <= 0.5^2
4*(sum(inscribed)/10000)
```