CS 6313 Statistical Methods for Data Science
Mini Project 1

2021794249 Muil Yang

1

1.1 (a)

1.1.1 Explanation

We need to calculate $P(T > 15)$.

$$P(T > 15) = \int_{15}^{\infty} \left[0.2e^{-0.1t} - 0.2e^{-0.2t}\right] dt = \left[-2e^{-0.1t} + e^{-0.2t}\right]_{15}^{\infty} = 2e^{-1.5} - e^{-3}$$

1.1.2 Code

```
f_T <- function(t) {
  0.2 * exp(-0.1 * t) - 0.2 * exp(-0.2 * t)
}

P_T_greater_15 <- integrate(f_T, lower = 15, upper = Inf)$value
P_T_greater_15
```

Result:

$$P(T > 15) = 0.3964733$$

1.2 (b)

1.2.1 Explanation

Satellite's work is based on a block A, and it also has an independent backup B. The satellite performs well its task until both A and B fail. So, the lifetime of the satellite T will be

$$T = max(X_A, X_B)$$

I will run the simulation 1000 times to obtain 1000 independent results. They will be

$$T_1, T_2, ..., T_{1000}$$

With the 1000 simulated T values, we can estimate $E(T)$ and $P(T > 15)$

1.2.2 Code

```
simulate_T <- function(n) {
    XA <- rexp(n, rate = 0.1)
    XB <- rexp(n, rate = 0.1)
    T_vals <- pmax(XA, XB)

    return(T_vals)
}

n <- 1000
Ts <- simulate_T(n)
```

```
E_T <- mean(Ts)
E_T
P_T_greater_15 <- sum(Ts > 15) / length(Ts)
P_T_greater_15
```

Result:

$$E(T) = 14.87148$$

$$P(T > 15) = 0.395$$

	Real Value	1000 Samples
$E(T)$	15	14.871
$P(T > 15)$	0.396	0.395

1.2.3 Histogram and Density Function

```
hist(Ts, probability = TRUE, xlim = c(0, 60), ylim=c(0, 0.06),
     main = "Histogram␣of␣T␣with␣Density",
     xlab = "T",
     col = "lightblue")

curve(0.2 * exp(-0.1 * x) - 0.2 * exp(-0.2 * x),
      from = 0, to = 60, add = TRUE, col = "red", lwd = 2)
```

Result:

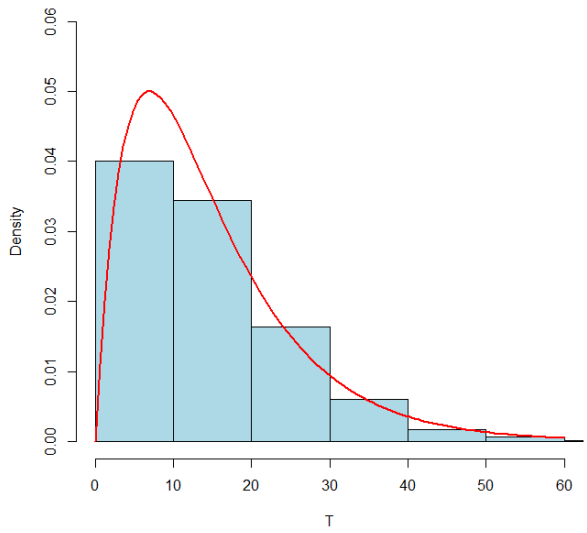

Figure 1: Histogram and Density Function

1.2.4 Sampling Four More Times

	True Value	1st	2nd	3rd	4th	5th
$E(T)$	15	14.871	15.016	14.932	15.483	14.923
$P(T > 15)$	0.396	0.395	0.412	0.404	0.418	0.383

Estimates fluctuate due to random sampling. But 1000 samples are fairly large numbers of samples, so the number fluctuates near the real value.

1.3 (c)

1.3.1 100 simulation

```
simulate_T <- function(n) {
    XA <- rexp(n, rate = 0.1)
    XB <- rexp(n, rate = 0.1)
    T_vals <- pmax(XA, XB)

    return(T_vals)
}

n <- 100
Ts <- simulate_T(n)

E_T <- mean(Ts)
E_T
P_T_greater_15 <- sum(Ts > 15) / length(Ts)
P_T_greater_15
```

Result:

$$E(T) = 15.9662$$
$$P(T > 15) = 0.42$$

1.3.2 10,000 simulation

```
simulate_T <- function(n) {
    XA <- rexp(n, rate = 0.1)
    XB <- rexp(n, rate = 0.1)
    T_vals <- pmax(XA, XB)

    return(T_vals)
}

n <- 10000
Ts <- simulate_T(n)

E_T <- mean(Ts)
E_T
P_T_greater_15 <- sum(Ts > 15) / length(Ts)
P_T_greater_15
```

Result:

$$E(T) = 14.86336$$
$$P(T > 15) = 0.3933$$

1.3.3 Result

	True Value	**100 Samples**	**1,000 Samples**	**10,000 Samples**
$E(T)$	15	15.966	14.871	14.863
$P(T > 15)$	0.396	0.42	0.395	0.393

Due to the Law of Large Numbers, increasing the number of samples leads to more accurate results. With 1,000 samples, the estimates are closer to the true values compared to using only 100 samples. However, beyond a certain point, the improvement becomes less significant. As a result, the difference between the estimates from 10,000 samples and 1,000 samples is relatively small.

2

2.1 Explanation

If a unit square has corners at (0, 0), (0, 1), (1, 0), and (1, 1), the area of the square is 1. And if a circle has center at (0.5, 0.5) and a radius of 0.5, the area of the circle is $\frac{\pi}{4}$. Therefore, the probability that a randomly selected point from the square falls inside the circle is

$$P = \frac{\frac{\pi}{4}}{1} = \frac{\pi}{4}$$

Using this probability, we can estimate π as $4P$.

2.2 Code

```
n <- 10000

x <- runif(n)
y <- runif(n)

is_inside_circle <- (x - 0.5)^2 + (y - 0.5)^2 <= 0.25
p <- sum(is_inside_circle) / length(is_inside_circle)

estimate_pi <- 4 * p
estimate_pi
```

Result:

$$\pi = 4P = 3.1312$$

The estimated value of π is 3.1312, while the true value of π is 3.1416. Given the large number of samples, the estimate is quite close to the actual value.