**Mini project #1**

**Group Member:** Chaoran Li, Wenting Wang

**Contribution of each member:**

Firstly, we discussed the mathematical models and code details together. Then, we divided the project into two part and finished our respective work. Wenting Wang mainly worked on Q1-a and Q2 while Chaoran Li worked on Q1-b and Q1-c. Then, we merged our code and solution into one report.

Each member makes contribution to each sub task of this project and combines all to finish this project, as the details shown in table 1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Question1-a | Question1-b | Question1-c | Question2 |
| Chaoran li | 20% | 80% | 80% | 20% |
| Wenting wang | 80% | 20% | 20% | 80% |

Table 1: Member contribution table

**Question 1:**

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

**Solution**:

According to the description of the question, the probability that the lifetime of the satellite exceeds 15 years denoted as P(T>15) can be get by:

We can get with the probability density function :

Thus，the probability that the lifetime of the satellite exceeds 15 years is：

0.3964733

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

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

**Solution**:

A screenshot of a cell phone

Description automatically generated

A picture containing bird, flower

Description automatically generated

In this question, we used two methods to get Exponential Distribution. The first one used the function rexp() directly which we would use in our following code. The second one was a self-designed function myExp() which used runif() to build an Exponential Distribution based on the mathematical model taught in our class. Both of them worked and got the right result.

We wrote the second version for practicing and we recommended the first one because the R language directly provides it.

ii.) Repeat the previous step 10000 times. This will give you 10000 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).]

**Solution**:



A screenshot of a cell phone

Description automatically generated

We solved (i) and (ii) here in one line for **Bonus** with 'replicate' function.

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.

**Solution**:

A picture containing knife

Description automatically generated

A close up of a logo

Description automatically generated

We drew the histogram with an interval of 3 and it fit the density function given above well. This proved that previous Exponential Distribution function worked well.

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

**Solution**:



A close up of a logo

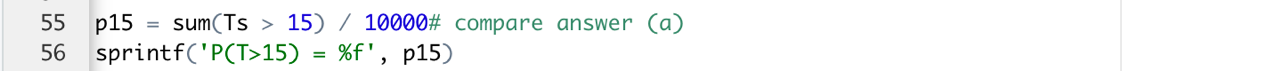
Description automatically generated

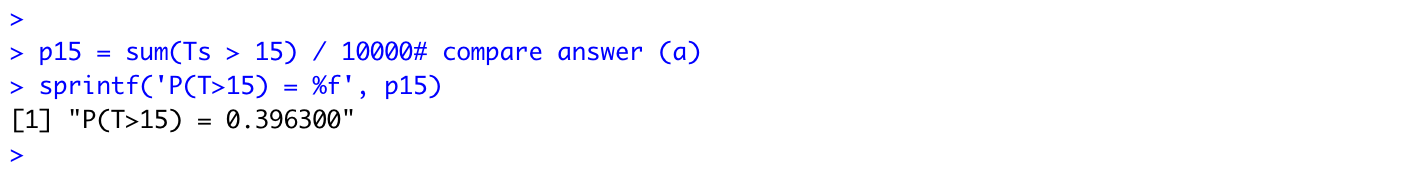
The given estimated E(T) is 15.

15.007074 > 15 (but close enough)

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).

**Solution**:





The probability that the satellite lasts more than 15 years this time is 0.396300 which is calculated in (a).

0.396300 < 0.3964733 (but close enough)

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.

**Solution**:

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

We wrote a 'simulate' function here and it could help us solve (b-iv) and (c) concisely and quickly.

A close up of a logo

Description automatically generatedA close up of a logo

Description automatically generated

A close up of a logo

Description automatically generatedA close up of a logo

Description automatically generated

A close up of a logo

Description automatically generated

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Theoretical Value | Simulation #1 | Simulation #2 | Simulation #3 | Simulation #4 | Simulation #5 |
| E(T) | 15 | 15.007074 | 15.0564 | 15.02805 | 14.84717 | 15.02667 |
| P(T>15) | 0.3964733 | 0.3963 | 0.3962 | 0.4026 | 0.3909 | 0.4052 |

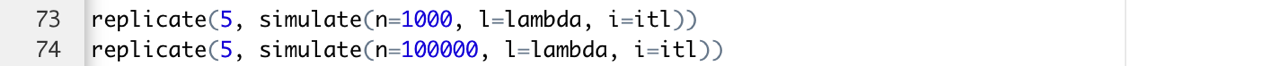
Table 2: Results of 10000 times of draw

The theoretical value and simulation value fit well for both E(T) and P(T>15). Simulation values change because of the existence of error. But all simulation values are close to the theoretical value which support the Law of Large Numbers (LLN).

1. Repeat part (vi) five times using 1000 and 100000 Monte Carlo replications instead of 10000. Make a table of results. Comments on what you see and provide an explanation.

**Solution**:

Use the 'simulate' function we designed in (b-vi), we can solve (c) concisely and quickly.

****

**A screenshot of a cell phone

Description automatically generated**

**A close up of a logo

Description automatically generatedA close up of a logo

Description automatically generated**

**A close up of a logo

Description automatically generatedA close up of a logo

Description automatically generated**

**A close up of a logo

Description automatically generatedA close up of a logo

Description automatically generated**

**A close up of a logo

Description automatically generatedA close up of a logo

Description automatically generated**

**A close up of a logo

Description automatically generatedA close up of a logo

Description automatically generated**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1000 times | Theoretical Value | Simulation #6 | Simulation #7 | Simulation #8 | Simulation #9 | Simulation #10 |
| E(T) | 15 | 14.89791 | 14.53424 | 15.00756 | 14.74868 | 14.76678 |
| P(T>15) | 0.3964733 | 0.41 | 0.373 | 0.402 | 0.373 | 0.383 |
| 100000 times | Theoretical Value | Simulation #11 | Simulation #12 | Simulation #13 | Simulation #14 | Simulation #15 |
| E(T) | 15 | 14.96408 | 14.98647 | 15.07583 | 14.9321 | 15.06806 |
| P(T>15) | 0.3964733 | 0.3947 | 0.3966 | 0.39963 | 0.39343 | 0.39838 |

Table 3: Results of 1000 and 100000 times of draw

If the time of draw is big enough, we will get some really 'large' results which cause the whole histogram become wider. These results will not affect the overall distribution, but they would not be observed in small time of draw simulation.

If we use to evaluate the influence of times of draw on simulation results, we can have the table below:

|  |  |  |  |
| --- | --- | --- | --- |
| Times of Draw | 1000 | 10000 | 100000 |
| Error(E(T)) | 3.066E-4 | 2.496E-5 | 1.464E-5 |
| Error(P(T>15)) | 1.905E-3 | 1.843E-4 | 3.311E-5 |

Table 4: The Influence of times of draw on Simulation results

This result proved the Law of Large Numbers (LLN) again. While the times of draw increasing, the simulation value will be close to the theoretical value.

In addition, we found that to obtain a sufficiently accurate E(T) is always easier than P(T>15). This means that we need a relatively small time of draw if we are only interested in E(T).

**Question ２:**

**Solution**:

To estimate the value of pi, first we draw a circle with center (0.5, 0.5) and radius=0.5. Then we can get the relation between and the probability

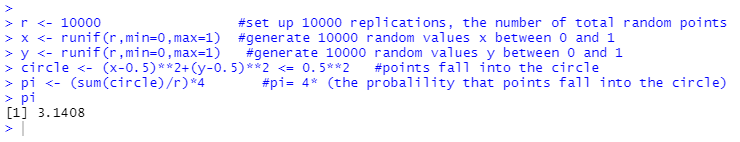
that a randomly selected point in a unit square with coordinates (0, 0), (0, 1),

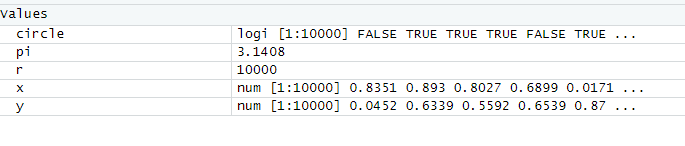
(1, 0) and (1, 1) falls the circle is :

P = (Area of circle) / (Area of square) =

Thus,

Next, we use RStudio to generate 10000 pairs of number x and y between 0 and 1. The code and output show below:





The result shows that the simulated value of is 3.1408 which is very close to 3.1415926 in mathematical value.