STATS 3042

Homework 3

Yevhen Myroniuk

A01005177 4r

2018

**Homework 3 – Numerical Descriptive Statistics  
Due: Friday, October 12, at beginning of lecture, as a hard copy  
Demonstrate your programs for me during lab, week of October 15**

In this assignment, you are going to write some functions to simulate replacing batteries in an electronic device. Put all of your functions in a single .R file and submit this file into the Homework 3 assignment folder in the lab section of Learning Hub. Also submit a hard copy of all of your answers. If a question required you to write code or produce a graph, include those as part of your answer to the question.

You will be demonstrating one or more of your functions in lab during the week of October 15.

Imagine that you have to replace four AA batteries in an electronic device. As you know, if even one of the four batteries is dead, your device will not work. You have 100 AA batteries in a box, but you were careless in disposing of dead batteries and so of those 100 batteries, 20 are dead.

1. Suppose you select four batteries at random and insert them into your electronic device. What is the **exact** probability your device will work? Show your work.

To make our device work, we need to select 4 working batteries out of the pool of 100 batteries = 80 working and 20 dead. So, the chance of getting 4 batteries that are working is:

80/100 \* 79/99 \* 78/98 \* 77/97

> q1()

[1] 0.4

1. Write a script called **Batteries2(n)** that simulates selecting making **n** randomselections of four batteries from a box of 100 batteries of which 20 are dead, and returns the proportion of the time that all four batteries work.

Run **Batteries2(10000)** and submit your output. How does your answer compare to your answer in part 1?

> Batteries2(10000)

[1] 0.4

Our result compares to the q1. Both result in 0.4 (40%) success chance.

1. Suppose your method of replacing your batteries is as follows: select four batteries from the box and test your electronic device. If it works, you’re done! If not, put all four of those batteries back in the box and select four batteries again. Repeat this process until your electronic device works.   
   1. Write a function called **Batteries3a()** that simulates this procedure, and returns the number of times k that you have to replace your batteries before your device works.

> Batteries3a()

[1] 2

* 1. Now write a function called **Batteries3b(n)** that runs **Batteries3a()** **n** times and returns the following:
* A table giving the distribution of the number of times you have to replace your batteries before your device works.
* A bar plot giving the distribution of the number of times you have to replace your batteries before your device works.

Run **Batteries3b(10000)** and submit your output.

> Batteries3b(10000)

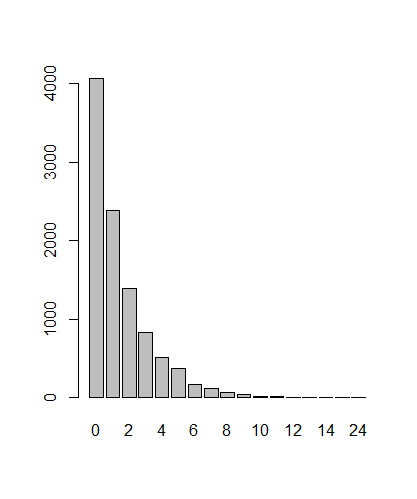
tryTable

0 1 2 3 4 5 6 7 8 9 10

4063 2388 1390 828 508 368 176 115 71 39 22

11 12 13 14 18 24

13 8 5 4 1 1



1. Suppose instead you replace your batteries by selecting four batteries from your box, as you did in Question 3; but this time, if your device doesn’t work, you throw away all four of those batteries and choose another four from the remaining batteries, and you repeat this process until your device works.
   1. What is the maximum number of times you would have to repeat this process before your device works?

The worse case scenario for us is when out of 4 picked batteries picked 1 is the bad one. Since we have 20 bad batteries it will take at most 21 attempts before we get our device working (21st attempt will result in getting the 4 good batteries)

* 1. Write a function called **Batteries4b()** that simulates this procedure, and returns the number of times k that you have to replace your batteries before your device works.
  2. Now write a function called **Batteries4c(n)** that runs **Batteries4b()** **n** times and returns the following:
* A table giving the distribution of the number of times you have to replace your batteries before your device works.
* A bar plot giving the distribution of the number of times you have to replace your batteries before your device works.

Run **Batteries4c(10000)** and submit your output.

> Batteries4c(10000)

resList

0 1 2 3 4 5 6 7 8 9 10

4102 2363 1474 820 517 312 162 92 59 44 23

11 12 13 14 15 17 18

11 6 6 4 2 2 1

