Project #1: Part #2

Antoinette Lynne, Harini Shanmugam and Aanya Shrivastava

2021-09-13

Problem #1. (10 points)

Read and understand the description of the original Monty Hall game. Now, write down your own description in your own words. Resist the temptation to copy-paste from an external resource since that will earn you zero points. It might help to imagine that you have a five-year old niece or nephew you are describing the game to.

Solution:
The Monty Hall Problem is a problem from a game show about probability. In short, there are three doors. One has a car behind it and the other two doors have goats behind them. You ultimately want to choose the door with the car behind it. First, you are given the choice to randomly pick one of the three doors. Then, the host will choose a door that is not the door you choose nor the one with the car behind it; it will be revealed to have a goat behind it. So the door that has the car behind it is either your original choice or the remaining door which has not been picked by you or the host. From there, you have the option to stick with your original choice, or you can switch your choice to the remaining door. After you decide, the door you chose will be revealed to either have the car or a goat behind it. The question posed is whether it is better to stick with your original choice or to switch.

Problem #2. (10 points)

Write and execute code in `R` which will represent **one round** of the original Monty Hall game. More precisely, it should do the following:

- "choose" at random behind which door the prize will be,
- "choose" at random the door that the contestant picks,
- choose which door the host will open,
- determine whether it was optimal to **switch** or **stick** for this particular round and return this determination.

Solution:

```r
montyhall <- function() {
  choices <- 1:3
  car <- sample(1:3,1)
  my_choice <- sample(choices,1)
  host <- sample(choices[choices != car & choices != my_choice], 1)
  if (car == my_choice) {
    return("stick")
  } else {
    return("switch")
  }
}
```

```r
montyhall()
## [1] "switch"
```

Problem #3. (10 points)

Write and execute code in `R` which repeats 100 rounds of the Monty Hall game. For which proportion of these rounds was it optimal to **switch**? *Hint:* If you answer to the previous problem was not in the form of a function, make it so now.

Solution:

```r
proportion <- function(n){
  switches <- 0
  for (i in 1:n){
    if (montyhall() == "switch"){
      switches <- switches + 1
    }
  }
  return(switches/n)
}

proportion(100)
## [1] 0.62
```

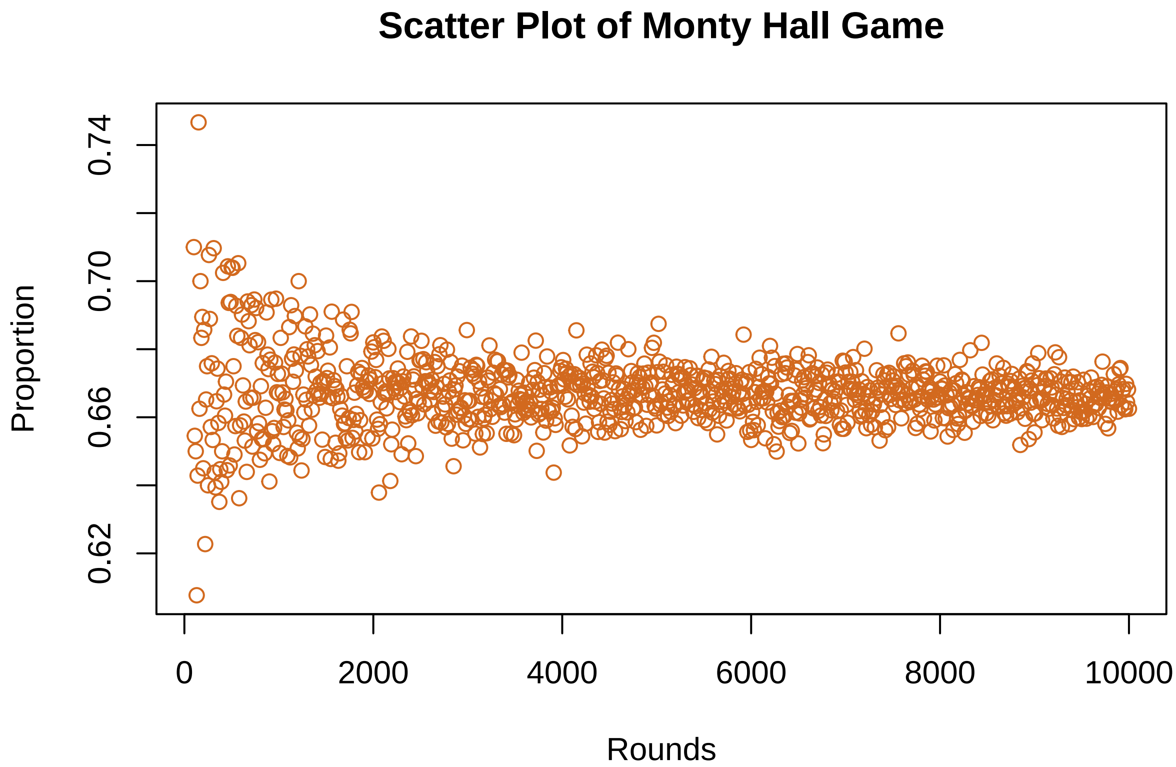
Problem #4 (10 points)

Write and execute code in `R` which repeats n rounds of the Monty Hall game for $n = 100, 110, 120, ..., 10000$. For each n, your code should record the proportion of individual rounds for which it was optimal to **switch**. In `R`, plot this recorded proportion as a function of n. Does your computational result agree with the analysis available at Wikipedia: the Monty Hall problem? Why? It might also be useful to consult: Wikipedia: The Law of Large Numbers.

Solution:

```r
b <- c()
n <- seq(100,10000, 10)
for(i in n){
  prop <- proportion(i)
  b <- c(b, prop)
}

plot(n, b,
     col = "chocolate",
     main = "Scatter Plot of Monty Hall Game",
     xlab = "Rounds",
     ylab = "Proportion")
```



Our computational result agrees with the analysis on the Wikipedia page. According to the page, by switching your choice, the probability of choosing the car becomes 2/3. By looking at our plot, we can see that the points create a trend that converges to the proportion of 0.66. This is supported by the law of large numbers which states that a large number of trials will tend to average a result that is closer to the expected value. In this case, our expected value was 2/3, or 0.66, and our plot shows a large set of data points that average to the expected value.