

Problem Set 3

(Due Feb. 23, 1:00 PM)

Instructions

1. The following questions should each be answered within an R script. Be sure to provide many comments in the script to facilitate grading. Undocumented code will not be graded.
2. Work on git. Fork the repository found at `github.com/jeonghkim/PS3` and add your code, committing and pushing frequently. Use meaningful commit messages – these may affect your grade.
3. You may work in teams, but each student should develop their own R script. To be clear, there should be no copy and paste. Each keystroke in the assignment should be your own.
4. If you have any questions regarding the Problem Set, contact the TAs or use their office hours.
5. For students new to programming, this may take a while. Get started.

Let's Make a Deal¹

In the game show “Let's Make a Deal”, the candidate gets to choose one of three closed doors, and receives the prize behind the door they choose. Behind one door is a new car; behind the other two doors are goats. After the contestant selects one of the 3 doors, the host opens one of the other two doors, and reveals a goat. Now, the candidate has the option of either sticking with the door they originally selected, or switching to the only other door that is still closed. What should the candidate do, and why? What are the probabilities of winning the car if they stay versus if they switch? This question has become known as the Monty Hall Problem.

¹https://en.wikipedia.org/wiki/Let's_Make_a_Deal

Your tasks

For this problem set, you will not solve the Monty Hall Problem, but you will have to code a slightly simplified version of the “Let’s Make a Deal” game. More specifically, you will set up a new class, which contains information regarding the door a player chooses, and a method that simulates a modified version of the game. You will have to do this using first the S3 and then the S4 class system. Here are the specific instructions:

1. Define a new S3 class: `door`. Objects of this class simply take on one numeric value: 1, 2, or 3 – indicating which door a candidate chooses.
2. Create a method for `door` objects that is called `PlayGame`. This method is supposed to do the following:
 - take the numeric value that is stored in the `door` object,
 - draw a random number between 1 and 3 that presents the door behind which the car is hidden,
 - compare the two numbers, and print a message congratulating a winning candidate that chose the correct door, or expressing sympathies for a losing candidate that chose the wrong door.
3. Then do it again, this time using the S4 system. Under S4, you will need to write:
 - a construction function that allows the user to create a `door` object,
 - a validation function that checks whether the value stored in `door` is actually an integer,
 - and the new generic method `PlayGame` explained above.