Andrew Seamon

CPS 3320-01

Project 1 Writeup

3/3/2021

**Theory**

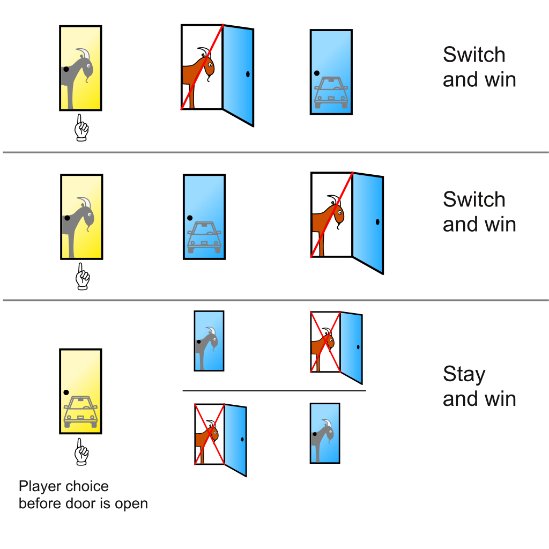
 The Monty Hall problem is a probability puzzle based on the gameshow “Let’s Make a Deal”. The name of this puzzle refers to the gameshows’ original host, Monty Hall. In this gameshow, the contestant will choose one of three doors (which either conceal a prize or a goat), the host will reveal what is behind a different door, and then the contestant will have the option to switch doors. I would like to test whether or not it is beneficial to always switch doors after seeing a goat behind one of the other doors. The following diagram explains the concept of the problem:

Figure 1: Shown in the diagram is the outcome of each possible option in the Monty Hall Problem. The contestant choses the yellow door, followed by the host revealing a different door, finally allowing the contestant to either switch doors or stay with their initial pick.

In studying this show and the strategies therein, I realized that the puzzle should have an answer, and I was going to find it for myself. My dataset is created at random by the code and then subsequently tested. To remove as many variables as possible, I forced the prize to always be behind door #1 and allowed the “contestant” to choose on their own (at random). The only problem I encountered is that a computer can never produce a 100% random number, which may have skewed the results. To conduct my analysis, I let the code complete five runs of 1,000,000 trials. To produce pseudorandom values, I used the random library included with Python.

**Results**

Any given execution of this program will produce slightly different results (within 0.008 across runs/executions). This is due to the use of random in my code. Shown below is an example of the output produced by this program.

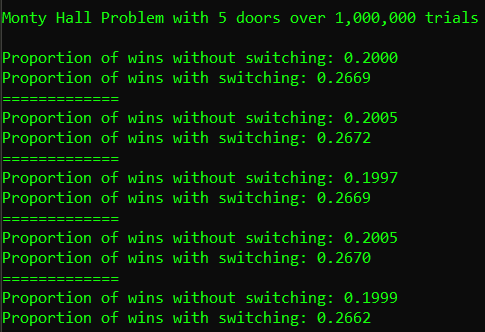


Figure 2: shown is the output of running python montyHall.py 5 1000000 in the command line. This run considers using 5 doors and running 1,000,000 trials per execution. Larger numbers are better.

\* The following page contains a table of my findings. Your results may vary slightly. \*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Monty Hall Problem** | | | | | | | |
| Number of Trials  Number of Doors | | 10 Trials | 100 Trials | 500 Trials | 1,000 Trials | 500,000 Trials | 1,000,000 Trials |
| 3 Doors | w/o switch | **0.3000** | 0.3300 | 0.3120 | 0.3380 | 0.3329 | 0.3335 |
| w/ switch | 0.4000 | 0.6300 | 0.6680 | **0.6770** | 0.6668 | 0.6673 |
| 5 Doors | w/o switch | 0.3000 | 0.1700 | 0.1920 | 0.2160 | 0.1998 | 0.1996 |
| w/ switch | 0.2000 | 0.3400 | 0.2600 | 0.2530 | 0.2682 | 0.2675 |
| 8 Doors | w/o switch | 0.1000 | 0.1600 | 0.1280 | 0.1330 | 0.1245 | 0.1251 |
| w/ switch | 0.4000 | 0.2000 | 0.1500 | 0.1590 | 0.1458 | 0.1462 |
| 14 Doors | w/o switch | 0.1000 | 0.0600 | 0.0580 | 0.0620 | 0.0716 | 0.0715 |
| w/ switch | 0.1000 | 0.0700 | 0.0700 | 0.0880 | 0.0778 | 0.0775 |
| 20 Doors | w/o switch | 0.0000 | 0.0600 | 0.0720 | 0.0520 | 0.0506 | 0.0501 |
| w/ switch | 0.0000 | 0.0600 | 0.0500 | 0.0550 | 0.0530 | 0.0524 |

Shown in the table above is the proportion of wins with and without switching doors. This was done for ndoors = [3, 5, 8, 14, 20] and ntrials = [10, 100, 500, 1000, 500000, 1000000]. In a normal game of *Let’s Make a Deal*, there are three doors, and you only get one chance to either switch doors or keep your initial choice. I thought it may be interesting to take these numbers to an extreme. Colored green is the most likely opportunity for a contestant to win using 3 doors. According to the table, 1 out of 1000 contestants will win by switching their door choice upon seeing a goat. Colored red is the least likely opportunity for a contestant to win. According to the table, a contestant is least likely to win by not switching their door choice.

**Reflection**

Overall, this was an interesting idea to experiment with. At the beginning, I hypothesized that always switching doors would produce more wins. According to the data gathered, I was correct in my hypothesis.