

# CSc 217 – Probability and Statistics for Computer Science

## Programming Lab Practicum №1

by

Alex Washburn

Due at 23:59 on

2023 – 10 – 17

# A CURIOUS CASE OF CHOICE AND CHANCE

In this practicum, we will explore the Monty Hall Problem. This is a famous thought experiment in probability theory which was loosely based on a similar American television game show named *Let's Make a Deal*. Monty Hall was the original host of *Let's Make a Deal* during its debut broadcast, and the Monty Hall Problem is named after him. The scenario under scrutiny is the following.

You are on a game show with Monty Hall. There are *three* doors  $D_1$ ,  $D_2$ , and  $D_3$ . Behind one of the doors, there is a car  $\mathcal{C}$ , the most desirable prize. Behind the other two doors are goats  $\mathcal{G}$ , which despite being cute are still the least desirable prize. The one car  $\mathcal{C}$  and the two goats  $\mathcal{G}$  are distributed uniformly at random behind the three doors  $D_1$ ,  $D_2$ , and  $D_3$ ; such that each door has a single prize behind it. On this game show you are informed of the potential prizes and also their distribution behind the doors. A particular random distribution of the prizes behind doors has been selected by the game show host Monty Hall before you begin to play the game. The prizes behind each door will not move during the game.

Now you are allowed to choose a single door  $D_* \in \{D_1, D_2, D_3\}$ . At the end of the game you will receive the prize behind your chosen door. However, after you choose the door  $D_*$  but before the game concludes, Monty Hall (who knows what's behind each door), opens another door  $D_M \in \{D_1, D_2, D_3\} \setminus D_*$  revealing the prize behind  $D_M$  to be a goat  $\mathcal{G}$ . Let the remaining unopened door which you did not choose be called  $D_{\neq} \notin \{D_*, D_M\}$ . Monty Hall then offers you a choice,

*“Do you want the unknown prize behind your chosen door  $D_*$ , or do switch your selection to the other unopened door  $D_{\neq}$  and receive its prize?”*

Is it advantageous to switch your choice?

# INSTRUCTIONS

Load `Monty_Hall.R` in RStudio. Look at the code beginning at line 110. The last 25 lines of the file are what you will work with.

You will experiment with different how the decisions you make, as the contestant of *Let's Make a Deal*, effect the outcome of the game show. There are *two* decisions you make through the course of the game show. The first is which door to choose. The second is whether or not to switch doors after Monty Hall has opened a door containing a goat.

There are four distributions for the first decision:

- `Randomly_Select_Door()`
- `Always_Select_Door(1)`
- `Always_Select_Door(2)`
- `Always_Select_Door(3)`

Independently, there are three distributions for the second decision:

- `Randomly_Respond()`
- `Always_Respond_With(FALSE)`
- `Always_Respond_With(TRUE)`

All the code is written for you, you will only need to make very minor changes to the variable bindings of:

- `how_to_choose_door` on line 112
- `how_to_choose_swap` on line 113

You will use these decision distributions to simulate the *Let's Make a Deal* game show 1000 times and observe the aggregated outcomes. From the observations of how different decisions influence the game show's outcome, we will determine what decisions are most advantageous to a contestant.

## **Problem 1:**

**Set the following variable bindings:**

```
how_to_choose_door <- Randomly_Select_Door()
```

```
how_to_choose_swap <- Randomly_Respond()
```

**Run the entire R script.**

**Observe the output generated in the “Console” and “Plots” frames.**

1 (a): “ Include the bar-plot figure that was produced. ”

1 (b): “ How many times did each outcome (Car! and Goat) occur? ”

## **Problem 2:**

**Set the following variable bindings:**

```
how_to_choose_door <- Always_Select_Door(1)
```

```
how_to_choose_swap <- Randomly_Respond()
```

**Run the entire R script.**

**Observe the output generated in the “Console” and “Plots” frames.**

2 (a): “ Include the bar-plot figure that was produced. ”

2 (b): “ How many times did each outcome (Car! and Goat) occur? ”

### **Problem 3:**

**Set the following variable bindings:**

```
how_to_choose_door <- Randomly_Select_Door()
```

```
how_to_choose_swap <- Always_Respond_With(TRUE)
```

**Run the entire R script.**

**Observe the output generated in the “Console” and “Plots” frames.**

3 (a): “ Include the bar-plot figure that was produced. ”

3 (b): “ How many times did each outcome (Car! and Goat) occur? ”

## **Problem 4:**

**Set the following variable bindings:**

```
how_to_choose_door <- Randomly_Select_Door()
```

```
how_to_choose_swap <- Always_Respond_With(FALSE)
```

**Run the entire R script.**

**Observe the output generated in the “Console” and “Plots” frames.**

4 (a): “ Include the bar-plot figure that was produced. ”

4 (b): “ How many times did each outcome (Car! and Goat) occur? ”

### **Problem 5:**

Think about the four simulation conditions you ran in Problems 1 – 4. Consider trying other combinations of the decision distributions listed on the Instructions section.

*From these simulations, what if any, decision procedure should a contestant take to maximize their expectation of winning the car?*