GEOGEBRA SIMULATIONS OF THE MONTY HALL GAME SHOW

Günhan Caglayan

New Jersey City University

Abstract

The purpose of this exploratory note is to offer teaching/learning ideas in the exploration of the famous Monty Hall Game Show, *Let's Make a Deal*, in an introduction to probability theory class using GeoGebra spreadsheets in a computer lab in groups of 2 to 3 people.

Keywords: GeoGebra, probability, conditional events, Bayes' rule

1 Introduction

This exploratory article is based on an activity implemented by the author in the teaching of a university level *Probability Theory* class with an audience of mathematics and mathematics education majors. Upon the completion of *Conditional Probability and Independence* chapter from the course textbook, 75 minutes of class time was dedicated to the exploration of the famous *Monty Hall Paradox,* which is named after the host of the famous TV game show *Let's Make a Deal.* The problem owes its fame to a reader's letter that was sent to Parade magazine's "Ask Marilyn" column from 1990: *Suppose you're on a game show, and you're given the choice of three doors. Behind one door is a car; behind the others, goats. You pick a door, say No.1, and the host, who knows what's behind the doors, opens another door, say No.3, which has a goat. He then says to you, "Do you want to pick door No.2?" Is it to your advantage to switch your choice?*

Figure 1. Let's Make a Deal (Hevesi, 2017).

The columnist Marilyn vos Savant responded to the letter with the recommendation that the contestant should switch to the other door. Many readers disagreed with Marilyn's response that switching would be a better strategy than keeping the originally selected door. My students explored this problem with GeoGebra spreadsheets.

2 Setting the Stage Towards Simulating

Students first began by defining a list containing door numbers via the syntax `door = {1, 2, 3}` in GeoGebra's input bar. They then made a plan for how to use each column of the spreadsheet. The most common approach was to use Column A to define the prize door and Column B to define the door selected by the contestant (or vice versa) by typing `A1 = RandomElement(door)` and `B1 = RandomElement(door)` respectively. In this article, we assume that Column A represents the door that hides the prize (the car) and Column B is the contestant's selection.

3 Modeling the Game Show Host's Action

Column C is then used to model the host's opening of an unselected door containing a goat: `C1 = If(A1==B1, RandomElement(Remove(door, {A1})), RandomElement(Remove (door, {A1, B1})))`. This command might be explained as follows: When the contestant selects the door with the car behind it, the code takes a random element from the list of doors with the winning door removed. When the contestant selects a door with a goat, the code takes a random element from the list of doors with the winning door and the contestant's pick removed (in other words, the "other" goat door). At this point, I noted two different approaches among the groups. Some groups explored the outcome without conditioning (`KEEP` or `SWITCH`). With this approach, students basically verified that you either win or lose the game with equal chance. For that purpose, Column D is used to randomly select any number from `door = {1, 2, 3}` except the one shown by the host: `D1 = RandomElement(Remove(door, {C1}))`. Finally, Column E is used to determine whether the contestant actually won or lost the game: `E1 = If(A1 == D1, "W", "L")`. This way, all columns are dragged down to row 1000 to simulate the game 1000 times. Figure 2 depicts the first 20 trial-snapshot from 1000 simulations.

	A	B	C	D	E		A	B	C	D	E
1	1	1	3	2	L	11	1	2	3	2	L
2	1	3	2	1	W	12	3	3	2	1	L
3	1	1	2	1	W	13	1	3	2	1	W
4	3	3	1	3	W	14	2	3	1	2	W
5	2	1	3	1	L	15	3	3	2	1	L
6	1	1	2	1	W	16	2	3	1	2	W
7	2	3	1	3	L	17	1	2	3	2	L
8	3	2	1	2	L	18	1	1	2	3	L
9	2	1	3	2	W	19	2	1	3	2	W
10	1	2	3	2	L	20	2	2	1	2	W

Figure 2. A 20-Trial snapshot from 1000 simulations.

4 Recording Values

To record the outcomes, a 4×4 table space (e.g., `I1:L4`) is selected on the spreadsheet. Cells `I1:L1` are labeled as Event, Win, Loss, Total, respectively. Cells `I2:I4` are labeled as Frequency, Experimental Probability, Theoretical Probability, respectively. The `CountIf` command is then used to calculate the frequency of each event: `J2 = CountIf(x == "W", E1:E1000)` and `K2 = CountIf(x == "L", E1:E1000)`. Experimental probabilities are obtained by dividing these numbers by 1000, the number of simulations: `J3 = J2 / 1000` and `K3 = K2 /1000`. The theoretical probabilities are entered in the corresponding cells `J4 = 0.5` and `K4 = 0.5`. Finally, the Sum command is used to add these frequencies and the probabilities: `L2 = Sum(J2:K2)` and `L3 = Sum(J3:K3)`.

5 Conditioning (KEEP)

Column F is used to simulate the `KEEP` conditional event. The syntax `F1 = If(A1 == B1, "W", "L")` basically reaffirms the fact that the contestant did not change their mind. Column F is then dragged down to row 1000 to simulate the game 1000 times as before. Figure 3 depicts the first 20 trial-snapshot from 1000 simulations. The `CountIf` command is used to determine the frequencies and the experimental probabilities in a similar manner as before.

	A	B	C	D	E	F		A	B	C	D	E	F
1	1	1	3	2	L	W	11	1	2	3	2	L	L
2	1	3	2	1	W	L	12	3	3	2	1	L	W
3	1	1	2	1	W	W	13	1	3	2	1	W	L
4	3	3	1	3	W	W	14	2	3	1	2	W	L
5	2	1	3	1	L	L	15	3	3	2	1	L	W
6	1	1	2	1	W	W	16	2	3	1	2	W	L
7	2	3	1	3	L	L	17	1	2	3	2	L	L
8	3	2	1	2	L	L	18	1	1	2	3	L	W
9	2	1	3	2	W	L	19	2	1	3	2	W	L
10	1	2	3	2	L	L	20	2	2	1	2	W	W

Figure 3. A 20-Trial snapshot from 1000 simulations (KEEP Conditioning—Column F).

6 Conditioning (SWITCH)

Likewise, Column G is used to simulate the `SWITCH` conditional event. The syntax `G1 = If(F1 == "W", "L", "W")` ensures that the contestant did change their mind and switched to the remaining closed door. Dragging down to Cell G1000, the game is simulated 1000 times as before. Figure 4 depicts the first 20-trial snapshot from 1000 simulations. The `CountIf` command is used to record the experimental values in a similar manner.

	A	B	C	D	E	F	G		A	B	C	D	E	F	G
1	1	1	3	2	L	W	L	11	1	2	3	2	L	L	W
2	1	3	2	1	W	L	W	12	3	3	2	1	L	W	L
3	1	1	2	1	W	W	L	13	1	3	2	1	W	L	W
4	3	3	1	3	W	W	L	14	2	3	1	2	W	L	W
5	2	1	3	1	L	L	W	15	3	3	2	1	L	W	L
6	1	1	2	1	W	W	L	16	2	3	1	2	W	L	W
7	2	3	1	3	L	L	W	17	1	2	3	2	L	L	W
8	3	2	1	2	L	L	W	18	1	1	2	3	L	W	L
9	2	1	3	2	W	L	W	19	2	1	3	2	W	L	W
10	1	2	3	2	L	L	W	20	2	2	1	2	W	W	L

Figure 4. A 20-Trial snapshot from 1000 simulations (SWITCH Conditioning—Column G).

7 Increasing the Number of Simulations

I observed that some groups performed the simulation 5,000 times and some others 10,000 times, which seemed to have produced results that appeared more in agreement with the theoretical ones (Figure 5).

	I	J	K	L		I	J	K	L		I	J	K	L
1	Event	W	L	Total	6	Cond.Keep	W	L	Total	11	Cond.Switch	W	L	Total
2	Freq.	2517	2483	5000	7	Freq.	1683	3317	5000	12	Freq.	3317	1683	5000
3	Exp. Prob.	0.503	0.497	1	8	Exp. Prob.	0.337	0.663	1	13	Exp. Prob.	0.663	0.337	1
4	Th. Prob.	0.5	0.5	1	9	Th. Prob.	0.333	0.667	1	14	Th. Prob.	0.667	0.333	1

	I	J	K	L		I	J	K	L		I	J	K	L
1	Event	W	L	Total	6	Cond.Keep	W	L	Total	11	Cond.Switch	W	L	Total
2	Freq.	4997	5003	10000	7	Freq.	3346	6654	10000	12	Freq.	6654	3346	10000
3	Exp. Prob.	0.5	0.5	1	8	Exp. Prob.	0.335	0.665	1	13	Exp. Prob.	0.665	0.335	1
4	Th. Prob.	0.5	0.5	1	9	Th. Prob.	0.333	0.667	1	14	Th. Prob.	0.667	0.333	1

Figure 5. Recorded values for (a) $N = 5,000$ (b) $N = 10,000$ simulations.

8 Theoretical Probability of Winning

W, L, K, S denote the events of Winning, Losing, Keeping, and Switching, respectively. Using the sample space approach (Figure 6), students determined the actual conditional probabilities $P(W|K) = (\frac{3}{9}) = (\frac{1}{3})$ and $P(W|S) = (\frac{6}{9}{=}\frac{2}{3})$, in agreement with the experimental results. I observed that students were always questioning the validity of the Column D simulation (Figure 2) and the previously stated "obvious fact," *you either win or lose the game with equal chance.* This method of questioning led students to probe further into the actual winning and losing probabilities.

Door Hiding the Prize	Contestant's Selection	Door Opened by the Host	Contestant's Revised Decision based on **KEEP**	Contestant's Revised Decision based on **SWITCH**
1	1	2 or 3	1	2 or 3
2	2	1 or 3	2	1 or 3
3	3	1 or 2	3	1 or 2
1	2	3	2	1
1	3	2	3	1
2	1	3	1	2
2	3	1	3	2
3	1	2	1	3
3	2	1	2	3

Figure 6. Theoretical conditional probabilities $P(W|K) = 3/9$ and $P(W|S) = 6/9$.

9 Questioning the "Keep" and "Switch" Events

Because $W \cap K$ and $W \cup S$ are mutually exclusive events whose union yields W, via the *law of total probability*, we obtain the probability of Winning by conditioning on Keeping or Switching as follows:

$$P(W) = P(W \cap K) + P(W \cap S) = P(K)P(W|K) + P(S)P(W|S) = \frac{1}{3}P(K) + \frac{2}{3}P(S)$$

The probability of winning, in a sense, can be viewed as the weighted average of $P(W|K)$ (i.e., the conditional probability of winning given that `KEEP` strategy is used), and $P(W|S)$ (i.e., the conditional probability of winning given that `SWITCH` strategy is used). Students then showed that $P(W) = \frac{1}{2} \Leftrightarrow P(K) = P(S) = \frac{1}{2}$ which is in agreement with the Column D simulation (Figure 5). But where did the assumption $P(K) = P(S) = \frac{1}{2}$ arise from? This was actually a crucial assumption; in reality, are the events of `KEEP` or `SWITCH` equally likely? What makes the contestant keep their original selection? What makes them switch? Students in different groups provided diversity of factors that could influence the contestant's decision of keeping or switching, such as the game show host's attitude, the reaction of the audience, or even the contestant's mood at that particular moment in time. With these concerns in mind, students derived a general formula for the winning probability $P(W)$ and the losing probability $P(L)$ as a function of keeping probability $P(K) = p$ with the restriction that $P(S) = 1 - p$ as follows:

$$P(W) = \frac{1}{3}P(K) + \frac{2}{3}P(S) = \frac{1}{3}p + \frac{2}{3}(1-p) = \frac{(2-p)}{3}$$

$$P(L) = P(K)P(L|K) + P(S)P(L|S) = \frac{2}{3}P(K) + \frac{1}{3}P(S) = \frac{2}{3}p + \frac{1}{3}(1-p) = \frac{(1+p)}{3}$$

Students further verified that with for $p = \frac{1}{2}$, the winning and losing probabilities would reduce to $P(W) = \frac{2-p}{3} = \frac{1.5}{3} = \frac{1}{2}$ and $P(L) = \frac{1+p}{3} = \frac{1.5}{3} = \frac{1}{2}$, respectively, in agreement with the experimental results (Figure 5).

10 Delving Further into Conditional Events

The last part of the class time was dedicated to retrieving conditional probabilities $P(K|W)$, $P(S|W)$, $P(K|L)$, $P(S|L)$ from the spreadsheet data and comparing them with the theoretical ones. In order to determine these conditional probabilities, students created four more columns (H, I, J, K) by defining the cells:

```
H1 = If(A1 == B1 ∧ E1 == "W", 1, 0);
I1 = If(A1 ≠ B1 ∧ E1 == "W", 1, 0);
J1 = If(A1 == B1 ∧ E1 == "L", 1, 0);
K1 = If(A1 ≠ B1 ∧ E1 == "L", 1, 0);
```

dragging these down to H10000; I10000; J10000; K10000, respectively, as usual. Figure 7 depicts the first 30-trial snapshot from 10,000 simulations. Once again, the `CountIf` command is used to determine the frequencies and the experimental probabilities in a similar manner as before:

```
N17 = CountIf(x == 1, H1:H10000);
O17 = CountIf(x == 1, I1:I10000);
N22 = CountIf(x == 1, J1:J10000);
O22 = CountIf(x ==1, K1:K10000)
```
(See Figure 10).

	A	B	C	D	E	F	G	H	I	J	K
1	2	3	1	2	W	L	W	0	1	0	0
2	3	3	1	3	W	W	L	1	0	0	0
3	1	3	2	1	W	L	W	0	1	0	0
4	1	2	3	2	L	L	W	0	0	0	1
5	1	2	3	1	W	L	W	0	1	0	0
6	1	3	2	1	W	L	W	0	1	0	0
7	3	3	2	1	L	W	L	0	0	1	0
8	3	1	2	3	W	L	W	0	1	0	0
9	2	3	1	2	W	L	W	0	1	0	0
10	3	1	2	1	L	L	W	0	0	0	1
11	1	2	3	2	L	L	W	0	0	0	1
12	2	2	1	2	W	W	L	1	0	0	0
13	3	1	2	3	W	L	W	0	1	0	0
14	2	1	3	1	L	L	W	0	0	0	1
15	2	1	3	1	L	L	W	0	0	0	1
16	3	2	1	3	W	L	W	0	1	0	0
17	3	3	2	3	W	W	L	1	0	0	0
18	3	1	2	3	W	L	W	0	1	0	0
19	1	1	2	3	L	W	L	0	0	1	0
20	2	3	1	3	L	L	W	0	0	0	1
21	3	3	1	2	L	W	L	0	0	1	0
22	1	1	3	1	W	W	L	1	0	0	0
23	1	1	2	1	W	W	L	1	0	0	0
24	3	2	1	2	L	L	W	0	0	0	1
25	3	1	2	3	W	L	W	0	1	0	0
26	2	1	3	1	L	L	W	0	0	0	1
27	2	2	3	2	W	W	L	1	0	0	0
28	1	2	3	2	L	L	W	0	0	0	1
29	1	3	2	3	L	L	W	0	0	0	1
30	3	1	2	3	W	L	W	0	1	0	0

Figure 7. Depicting a 30-Trial Snapshot from 10,000 Simulations

	M	N	O	P
1	**Event**	**W**	**L**	**Total**
2	**Freq.**	516	484	1000
3	**Exp. Prob.**	0.516	0.484	1
4	**Th. Prob.**	0.5	0.5	1

	M	N	O	P
6	**Cond.Keep**	**W**	**L**	**Total**
7	**Freq.**	357	643	1000
8	**Exp. Prob.**	0.357	0.643	1
9	**Th. Prob.**	0.333	0.667	1

	M	N	O	P
11	**Cond.Switch**	**W**	**L**	**Total**
12	**Freq.**	643	357	1000
13	**Exp. Prob.**	0.643	0.357	1
14	**Th. Prob.**	0.667	0.333	1

	M	N	O	P
16	**Cond.Win**	**K**	**S**	**Total**
17	**Freq.**	187	329	516
18	**Exp. Prob.**	0.362	0.638	1
19	**Th. Prob.**	0.333	0.667	1

	M	N	O	P
21	**Cond.Lose**	**K**	**S**	**Total**
22	**Freq.**	170	314	484
23	**Exp. Prob.**	0.351	0.649	1
24	**Th. Prob.**	0.333	0.667	1

Figure 8. Recorded Values for $N = 1,000$ Simulations

	M	N	O	P
1	**Event**	**W**	**L**	**Total**
2	**Freq.**	2469	2531	5000
3	**Exp. Prob.**	0.494	0.506	1
4	**Th. Prob.**	0.5	0.5	1

	M	N	O	P
6	**Cond.Keep**	**W**	**L**	**Total**
7	**Freq.**	1730	3270	5000
8	**Exp. Prob.**	0.346	0.654	1
9	**Th. Prob.**	0.333	0.667	1

	M	N	O	P
11	**Cond.Switch**	**W**	**L**	**Total**
12	**Freq.**	3270	1730	5000
13	**Exp. Prob.**	0.654	0.346	1
14	**Th. Prob.**	0.667	0.333	1

	M	N	O	P
16	**Cond.Win**	**K**	**S**	**Total**
17	**Freq.**	822	1647	2469
18	**Exp. Prob.**	0.333	0.667	1
19	**Th. Prob.**	0.333	0.667	1

	M	N	O	P
21	**Cond.Lose**	**K**	**S**	**Total**
22	**Freq.**	908	1623	2531
23	**Exp. Prob.**	0.359	0.641	1
24	**Th. Prob.**	0.333	0.667	1

Figure 9. Recorded values for $N = 5,000$ simulations.

	M	N	O	P
1	**Event**	**W**	**L**	**Total**
2	**Freq.**	5055	4945	10000
3	**Exp. Prob.**	0.506	0.495	1
4	**Th. Prob.**	0.5	0.5	1

	M	N	O	P
6	**Cond.Keep**	**W**	**L**	**Total**
7	**Freq.**	3395	6605	10000
8	**Exp. Prob.**	0.34	0.661	1
9	**Th. Prob.**	0.333	0.667	1

	M	N	O	P
11	**Cond.Switch**	**W**	**L**	**Total**
12	**Freq.**	6605	3395	10000
13	**Exp. Prob.**	0.661	0.34	1
14	**Th. Prob.**	0.667	0.333	1

	M	N	O	P
16	**Cond.Win**	**K**	**S**	**Total**
17	**Freq.**	1658	3397	5055
18	**Exp. Prob.**	0.328	0.672	1
19	**Th. Prob.**	0.333	0.667	1

	M	N	O	P
21	**Cond.Lose**	**K**	**S**	**Total**
22	**Freq.**	1737	3208	4945
23	**Exp. Prob.**	0.351	0.649	1
24	**Th. Prob.**	0.333	0.667	1

Figure 10. Recorded values for $N = 10,000$ simulations.

11 Using Bayes' Rule

The class concluded with the derivation of the theoretical conditional probabilities $P(K|W)$, $P(S|W)$, $P(K|L)$, $P(S|L)$ in terms of $P(K) = p$.

$$\begin{aligned} P(K|W) &= \tfrac{P(W|K)P(K)}{P(W)} = \tfrac{\frac{1}{3}\cdot p}{\frac{2-p}{3}} = \tfrac{p}{2-p} \\ P(S|W) &= \tfrac{P(W|S)P(S)}{P(W)} = \tfrac{\frac{2}{3}\cdot(1-p)}{\frac{2-p}{3}} = \tfrac{2-2p}{2-p} \\ P(K|L) &= \tfrac{P(L|K)P(K)}{P(L)} = \tfrac{\frac{2}{3}\cdot p}{\frac{1+p}{3}} = \tfrac{2p}{1+p} \\ P(S|L) &= \tfrac{P(L|S)P(S)}{P(L)} = \tfrac{\frac{1}{3}\cdot(1-p)}{\frac{1+p}{3}} = \tfrac{1-p}{1+p} \end{aligned}$$

As I had emphasized in the last section of the *Conditional Probability and Independence* chapter from the course textbook, students completed their exploration of the Monty Hall Problem by verifying that the conditional probability $P(\square|W)$ itself is a probability by showing $P(K|W) + P(S|W) = \frac{p}{2-p} + \frac{2-2p}{2-p} = \frac{2-p}{2-p} = 1$ and that that the conditional probability $P(\square|L)$ itself is a probability by showing $P(K|L) + P(S|L) = \frac{2p}{1+p} + \frac{1-p}{1+p} = \frac{1+p}{1+p} = 1$. Students further verified that with the $P(K) = p = \frac{1}{2}$ assumption, the conditional probabilities $P(K|W) = \frac{p}{2-p}$, $P(S|W) = \frac{2-2p}{2-p}$, $P(K|L) = \frac{2p}{1+p}$, and $P(S|L) = \frac{1-p}{1+p}$ would reduce to $P(K|W) = \frac{0.5}{1.5} = \frac{1}{3}$, $P(S|W) = \frac{1}{1.5} = \frac{2}{3}$, $P(K|L) = \frac{1}{1.5} = \frac{2}{3}$, and $P(S|L) = \frac{0.5}{1.5} = \frac{1}{3}$, respectively, in agreement with the experimental results.

12 Concluding Remarks

12.1 Why the Monte Hall Problem?

With regard to the study of the Monte Hall problem, the significance of our exploration is threefold. The activity illustrates (1) how a popular game show can motivate students; (2) how the notion of conditional probability $P(\square|F)$ can be visualized as a probability itself by comparing the actual conditional probabilities with the experimental results, in coordination with the sample space approach and the total law of probability; and (3) how conditional probability simulations can be implemented in GeoGebra for prior and posterior probabilities in coordination with the Bayesian analysis.

12.2 Why GeoGebra?

While one can model the Monte Hall scenario using other software—for instance, an Excel spreadsheet—GeoGebra offers a number of advantages. For one, the GeoGebra scripting environment enables teachers and students to extend the functionality of applets using GeoGebra commands or Javascript. Secondly, GeoGebra offers an integrated mathematics environment—with graphical, algebraic, geometric, and statistical representations dynamically linked. The capacity to link representations offers valuable pedagogical advantages that build upon students' previous mathematical understandings in ways that Excel (or Minitab or SPSS or SAS) cannot hope to do. Thirdly, GeoGebra is familiar. Its flexibility makes it useful across courses in our department and in various contexts within my course. Thus, when exploring the Monte Hall problem with the software, I didn't need to spend time preparing my students to use GeoGebra—we hit the ground running and used familiar tools immediately in service of our investigation. In all, exploring the Monte Hall problem using GeoGebra with my students was a winning combination.

REFERENCES

Hevesi, D. (2017, September 30). Monty Hall, Co-Creator and Host of 'Let's Make a Deal,' Dies at 96. Retrieved from https://www.nytimes.com/2017/09/30/obituaries/monty-hall-dead-lets-make-a-deal.html?mcubz=3

Ross, S. M. (2015). *A first course in probability* (9th ed.). Boston: Pearson.

Vos Savant, M. (1990). Ask Marilyn. *Parade Magazine*, (16).

Günhan Caglayan, GCaglayan@njcu.edu teaches mathematics at New Jersey City University. His main interests are mathematical visualization and student learning through modeling and visualization.

APPENDIX – GEOGEBRA COMMAND SYNTAX GUIDANCE

Retrieved from wiki.geogebra.org

Command	Functionality	Example
CountIf(<Condition>, <List>)	Counts the number of elements in the list satisfying the condition.	**CountIf(x < 3, {1, 2, 3, 4, 5})** gives you the number 2.
If(<Condition>, <Then>, <Else>)	Yields a copy of object *Then* if the condition evaluates to true, and a copy of object *Else* if it evaluates to false.	**If(n==3, x + y = 4, x - y = 4)** yields line x + y = 4 when n = 3, and line x - y = 4 for all n not equal to 3.
RandomElement(<List>)	Returns randomly chosen element from the list (with uniform probability).	**RandomElement ({3,2,-4,7})** yields one of {-4,2,3,7}.
Remove(<List>, <List>)	Removes objects from the first list each time they appear in the second list.	**Remove({1,3,4,4,9},{1,4,5})** yields list {3,4,9}.