MonteCarlito Training Exercises

Exercise 1: Running the Example 30 minutes

1. Expose the example tab.
2. Read through the boxes and explain the contents. Delete each box as you complete it.
3. Use the "Clean Formulas" button to clean up the "runtotal" cell.
4. Step the students through how the model works and how the stats at the top work.
5. Explain why the stats in the variable area are not part of the model.
6. Run the model and examine the results.
7. Talk about formatting the columns and format them accordingly.
8. Expand the model one column and set the row 2 value to 15.
9. Run the model and examine the results
10. Set the "Trials" to 1,000 and run it again and examine the results. What does the "mean" mean?
11. Set the "Trials" to 10,000 and run it again and examine the results. What does the "mean" mean?
12. Expand the model in 10 cent intervals from $14 to $15. So, row two has 14, 14.10…15. And the formulas have been copied across the model from column I to column S.
13. Run it again and note the imprecision of the statistical values.
14. Change the C.I. precision to 2 and set the decimal format to 2 for the statistics.
15. Talk about the difference between the actual total profit and the risk of playing.
16. End of exercise.

Exercise 2: Making a Dashboard 45 minutes

1. Read the following scenario:
   You have been hired by a venture capital investor to help determine how effective their investment in the start-up marketing campaign of a small business would be. You have investigated the statistics and come up with some different types. These are:

   a. *The best estimate for marketing effectiveness for this type of business is 1.5%. But the lowest anyone has seen is about .5% and the highest has been a whopping 5%.*

   b. *The venture capital company is willing to spend anywhere between $10,000 and $30,000 on the campaign.*

   c. *The market size is difficult to nail down. But it seems to be between 500,000 and 1 million customers.*

   You want to know what the cost per new customer would be so that the client can determine if the projected profit per customer is worth the marketing investment.

   You decide that the client will not tolerate the complexities of the simulation very well. So you decide to drive the simulation from a dashboard that you can capture and paste into PowerPoint.

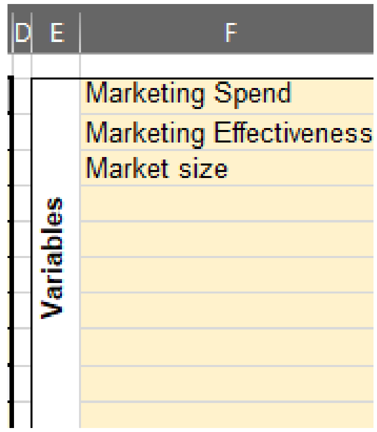
2. Discuss how we might sample the different data types.
3. Create a new simulation tab and name it "Marketing Test".
4. Create a new tab and name it "Dashboard".
5. Format the dashboard tab by adjusting the column width to make the cells square.
6. Color the whole sheet some neutral color like a soft yellow or gray.
7. Start by creating the labels for Market Size, Marketing Effectiveness, and Marketing Spend.
8. Start merging some cells to hold the appropriate values. It should look something like:

		Min	Avg	Max	Type
1					
2	**Marketing Effectiveness**	0.50%	1.50%	5.00%	TPE
3	**Marketing Spend**	$ 10,000		$ 30,000	Uniform
4	**Market size**	500,000		1,000,000	Uniform

   Note that the by removing the color fill from the value cells, the user can see the values that are meant to be changeable in the model. They can clearly see the type of distribution you are using.

   Note that yours doesn't have to be exactly like this. You could put them in any order you wish. The design of an actual dashboard may be more complex and may need to account for differences in culture and color use.

9. Return to the simulation tab you previously named "Marketing Test."
10. In the variables section, name the top three variables by these same names. Again, order doesn't matter. It should look something like:



11. “Marketing Spend” is intended to be a uniform distribution. We do sample from a uniform distribution by using the native Excel distribution, “RANDBETWEEN()”.

    In the corresponding cell in column G, type the formula, using the cells on the “Dashboard” tab as the Min and Max values. It should look something like:

    ```
    fx  =RANDBETWEEN(Dashboard!H3,Dashboard!L3)
    ```

    Note that H3 and L3 are where I stored the values in my dashboard page for this example.

12. Do exactly the same thing for “Market Size.”

13. For “Marketing Effectiveness”, we will use the MonteCarlito function, “TRIDIST().” In column G, on the “Marketing Effectiveness” row, type the formula, we want to use the reference cells on the “Dashboard” page for the Min, Avg, and Max values.

    > *NOTE: Sometimes we want to use very small values (values between 0 and 1) in our models. However, the distribution we are using doesn’t want to work for decimal only values. To get around this, we use a multiplier to generate the data, then divide the result by the same multiplier.*
    >
    > *What we are doing amounts to: (3 X 1,000) + (5 X 1,000) = 8,000*
    > *Then when we back it out: 8,000/1,000 = 8.*
    >
    > *The scale of our multiplier will be determined by how many places beyond the decimal place (significant digits) we want to see in our result. In this case, we are using percentages and want to see some thing like 0.00% in our result. For this we need to use 1,000 as our multiplier.*

    We will need to do this for the triangular distribution (i.e. Three Point Estimate or TPE) that we are using for “Marketing Effectiveness.” It should look something like:

    ```
    fx  =tridist(Dashboard!H2*1000,Dashboard!J2*1000,Dashboard!L2*1000)/1000
    ```

    Once we set the format of the cells, we should have something that looks like:

E	F	G
Variables	Marketing Spend	11,972
	Marketing Effectiveness	2.00%
	Market size	21,741

14. Now, we are ready to build our model. We will generate histograms for every formula in this model. To do this we create the following labels in row 12, beginning in column G.

    “Market Spend<H>”
    “Market Reach<H>”
    “Cost/Customer<H>”

15. The “Market Spend” will be equal to the value in our variable table. So, in that cell (G14), you will enter “=G2”. Or whatever cell you put that reference to the dashboard page in. We do this so that we can preserve whatever values are generated for the marketing spend.

    Note that a number 1 appears in G13 and the size of the simulation increments to 1 in the Parameters table to the left.

16. The “Market Reach” will be a function of the “Market Size” and the “Marketing Effectiveness”. So we need an equation that does:

    Market Reach = Market Size X Marketing Effectiveness

    In H14, enter the formula “=G3*G4”. Replacing G3 and G4 with whatever cells you stored “Market Size” and “Marketing Effectiveness” in.

    Note once again the incrementation of the model size.

17. The cost per new customer in this simple model will be the Market Spend / Market Reach.

    > *Note that I have simplified this model considerably from a standard marketing estimation model. This is to permit the example to be easily workable. In reality, Market Reach would be additionally factored by Response Rate, and there would be time element introduced that would pace the responses and show the market size decreasing with each spend by the number of customers “taken” during that marketing effort.*

    In cell I14, enter the formula, “=G14/H14”.

18. We can now run the simulation. To do so, we need to set up the parameters. Adjust your parameter table to match this:

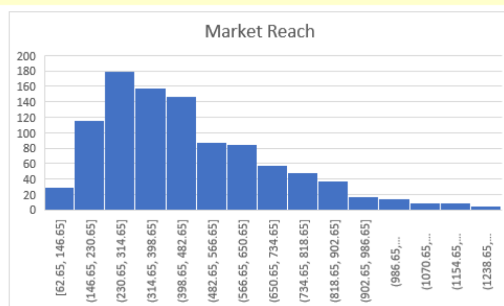
Parameters	
Run Parameters	
Trials	1000
Windows State	Progress
Histograms	This Book
Create Run Table?	Y
Show 95% C.I.?	Y
C.I. Precision	2
Statistic Parameters	
Standard error	
Median	
Standard deviation	
Variance	
Skewness	
Kurtosis	
Size	3

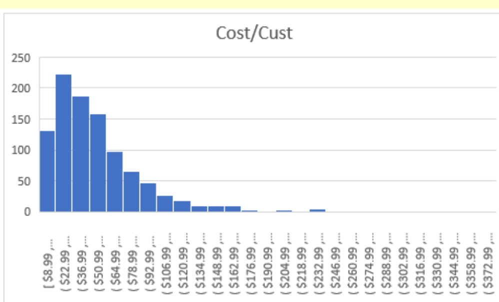
19. Activate the “MonteCarlito” ribbon tab and click the “Run Simulation” button.

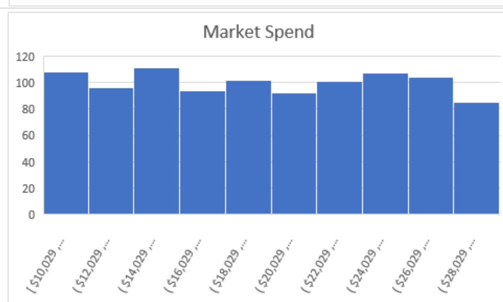
    Once the simulation is complete, we will need to do two things: 1. Arrange the histograms on our dashboard page. 2. Create references to the statistics on the dashboard page.

20. On the dashboard page, arrange the histograms so that they are in an informative position on the page. How you do this will be determined by personal preference, cultural standards, and user preference. Here’s my example:

	Min	Avg	Max	Type
Marketing Effectiveness	0.50%	1.50%	5.00%	TPE
Marketing Spend	$ 10,000		$ 30,000	Uniform
Market size	500,000		1,000,000	Uniform





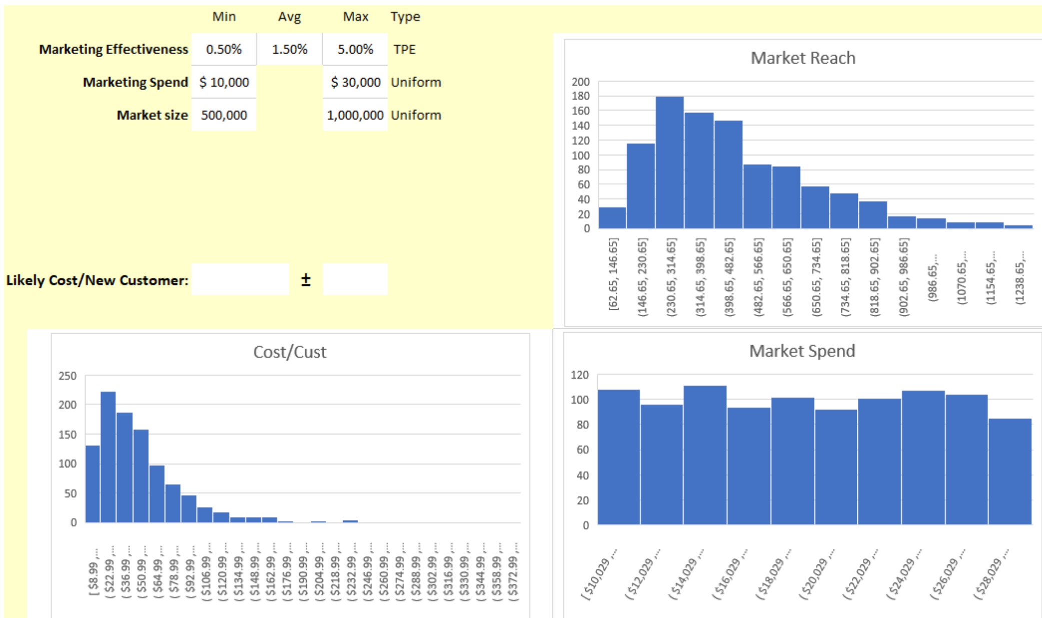


Note that you can format the histograms in any normal way because they are standard Excel graphics.

21. Finally, we need to add the cost/new customer metric on the page. Let’s group a couple of cells and add a label. Here’s how mine looks:

Likely Cost/New Customer: ±

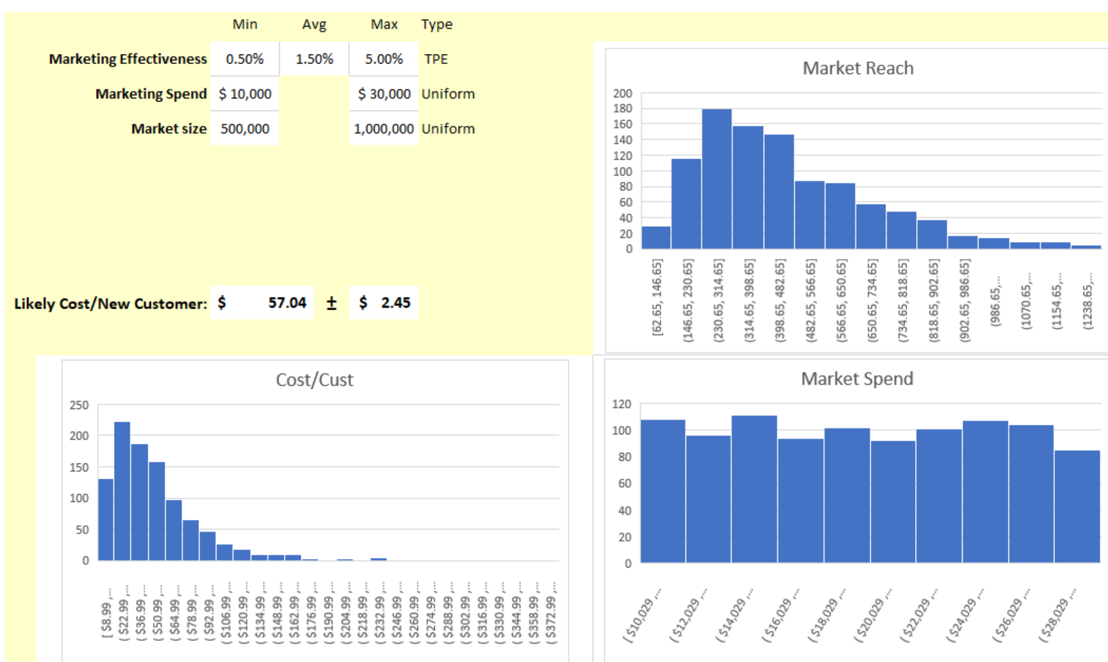
This provides a space to link to the run results. In context:



Note that we are here expressing the 95% Confidence Interval in terms of the mean and interval bounds. This is standard practice.

22. In the cell right after the “Likely Cost/New Customer:” label, type the = sign, then click the “Marketing Test” tab and select cell I15. This puts the mean of the run into this cell.

23. In the cell after the ± sign, enter the formula that subtracts the lower bounds of the 95% Confidence Interval from the mean: “='Marketing Test'!I15-'Marketing Test'!I23”.



This sheet can be captured as an image and pasted into PowerPoint.

Note that every time you run the simulation, these numbers are subject to change.

Note that so far, we have only run 1000 trials.

24. Return to the simulation tab, change the Trials value to 10,000 and run it again.

    Notice what difference this makes in the values and the width of the Confidence Intervals.

    Note that the W dashboard page. You will need to do that every time you run it.

    You may want to move them to another page and place them in a horizontal arrangement so that you can label the row and compare runs.

    You can also do this by duplicating the dashboard page and renaming the duplicate page to any name that makes sense to you. If you always maintain the original dashboard page, you can simply adjust the variables and run the simulation again. This method preserves the provenance of the runs and the decisions you made to arrive at your final recommendation.

    Remember: You can only have 1 tab named “Dashboard” in any simulation workbook.

25. End of exercise

Exercise 3: Horizontal Dependency <u>15 minutes</u>

The goal of this exercise is to see how to make a simulation that can have part of the current trial (iteration) dependent on the results of a previous trial.

1. Create a new simulation tab in a new workbook.
2. Set G14: “=H14+1”
3. Set H14 to random number between 10 and 100.
4. Set I14 to the same as G14, that is, “=H14+1”
5. Set your parameter table to look like:

Parameters	
Run Parameters	
Trials	10
Windows State	Progress
Histograms	None
Create Run Table?	Y
Show 95% C.I.?	N
C.I. Precision	
Statistic Parameters	
Standard error	
Median	
Standard deviation	
Variance	
Skewness	
Kurtosis	
Size	3

6. Run the simulation. You should get a table of scaling numbers that looks like this:

Run Table	16	35	36
	36	42	43
	43	57	58
	58	61	62
	62	95	96
	96	91	92
	92	91	92
	92	20	21
	21	87	88
	88	74	75

Notice the pattern in the numbers from iteration to iteration.

7. While this works, we would like the first value in the table to be 0, not some random value. To accomplish this we will use the “SimRunning” feature. Adjust the formula in G14 to:

   =IF(SimRunning=1,IF(H14=0,0,H14+1),0)

   And the formula in H14 to:

   =IF(SimRunning="",0,RANDBETWEEN(10,100))

   Talk yourself through these formulas so you understand what each is doing. Review the note on the Example simulation to understand what we are doing with “SimRunning”.

Note that you don’t have to type “SimRunning.” You can simply select cell E13.

8. Run the simulation again. You should get a table that looks like:

Run Table	0	50	51
	51	92	93
	93	79	80
	80	70	71
	71	50	51
	51	41	42
	42	63	64
	64	16	17
	17	83	84
	84	54	55

Note that we have the same number pattern, but with an initial starting point of 0.

9. End of exercise.

Exercise 4: Generate a TPE **30 minutes**

The purpose of this exercise is to demonstrate how to create a Three Point Estimate based on known factors. This is useful for those cases in which the max and min of known factors might be combined to create an estimate when no other estimate is available.

Additionally, this exercise will demonstrate the value of high iteration counts.

We will be developing a probability distribution for throwing 2 dice. Imagine rolling 2 dice and obtaining a score that is the sum of the faces. The lowest value you could get would be 2. The highest value you could get would be 12.

We could use a uniform distribution to estimate the results, but that would introduce some significant error. The reason is that there is only one way to obtain the values of 2 and 12. But there are many ways to obtain the other values. Some values can be obtained by more ways than others.

If we all remembered our statistics class, we might remember how to calculate the probability of obtaining every value between 2 and 12. Fortunately, MonteCarlito can help us without having to recall everything about probability and statistics.

1. In a new workbook, create a new simulation tab and name it "Test TPE Creation."

2. Let's think about how we might simulate this. Here's a good approach:
   a. Roll one D6. *D6 is shorthand for a 6 sided die.*
   b. Roll a second D6. *This will ensure that the rolls are independent.*
   c. Add the values of the two dice.
   d. Tally up every time it is a 2 and every time it is a 3 and…12
   e. Count the tally marks.
   f. It would be good to see it in a graph.

3. We will need 14 columns for this simulation. Let's set it up like this:

Label => => => =>	Die 1	Die 2	Total	2	3	4	5	6	7	8	9	10	11	12

4. Then we need a formula to roll a single D6. It will look like this:
   "=trunc(randbetween(1,6)0)"
   This will generate a random number between 1 and 6, then truncate any digits after the decimal point. We will need this to be the formula below Die 1 and Die 2. It should look like:

Equations => =>	=TRUNC(RANDBETWEEN(1,6),0)	=TRUNC(RANDBETWEEN(1,6),0)

   This is showing the formulas. Yours will show the actual integer values.

5. Write the next formula to sum the two dice. Your simulation should look something like this:

Label => => => =>	Die 1	Die 2	Total
Do not Change =>	1	2	3
Equations => =>	1	2	3

6. Now, let's make the rest of the formulas easy. In the next formula, type: "=IF(I14=J12,1,0)". This formula looks at the total of the dice and makes a tally mark (1) every time the total is 2. Otherwise, it marks a 0.

7. Let's copy that formula all the way across from 2 to 12. Your simulation should resemble this:

Label => => => =>	Die 1	Die 2	Total	2	3	4	5	6	7	8	9	10	11	12
Do not Change =>	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Equations => =>	4	5	9	0	0	0	0	0	0	0	1	0	0	0

8. To get the frequency at which the simulation generates values between 2 and 12, we need to count the tally marks for each column. To do this we will write the formula "=RunTotal()" in the

last line of the variable section, above the labels 2 through 12. Your simulation should look something like this:

Variables															
					0	0	0	0	0	0	0	0	0	0	0
	Label => => => =>	Die 1	Die 2	Total	2	3	4	5	6	7	8	9	10	11	12
	Do not Change =>	1	2	3	4	5	6	7	8	9	10	11	12	13	14
	Equations => =>	1	2	3	0	1	0	0	0	0	0	0	0	0	0

9. Set your parameter table for 100 iterations, Fast, Run Table = Y, Histograms = None, C.I. = N. It should look like this:

Parameters	
Run Parameters	
Trials	100
Windows State	Fast
Histograms	None
Create Run Table?	Y
Show 95% C.I.?	N
C.I. Precision	
Statistic Parameters	
Standard error	
Median	
Standard deviation	
Variance	
Skewness	
Kurtosis	
Size	14

Now we could run the simulation, but we couldn't visualize the results very well. So, let's set up a page to create a visual.

10. On a blank tab, label A1 as "Value", B1 as "Freq" and C1 as "Probability."

11. In cells A2:A12, type in the values 2, 3,…,12. It should look like this:

Value	Freq	Probability
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		

12. Now we need a reference to each of the tally totals in column B. Select B2, type the "=", then select the simulation tab and click on cell J11. This is the tally total cell for the value of 2.

13. Repeat step 12 for each of the other values. When you are done, you may need to return to the simulation tab and click the "Clean Formula" button to restore the RunTotal formulas. Your new tab should look something like:

Value	Freq	Probability
2	0	
3	0	
4	0	
5	0	
6	0	
7	0	
8	0	
9	0	
10	0	
11	0	
12	0	

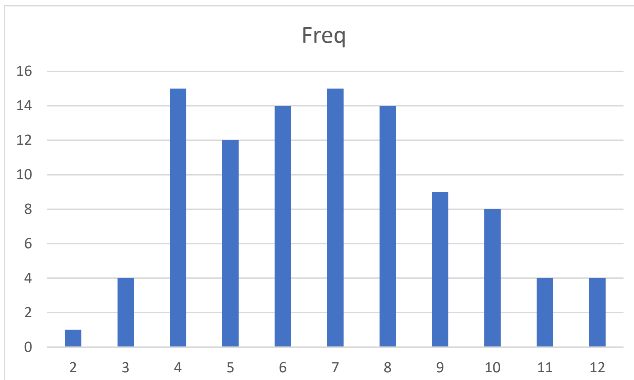
14. Now we want to know the relative frequency of each value. To do this, we will divide by the total number of iterations (Trials). In cell C2 on your new tab, type "=B2/" , then without hitting enter, click the simulation tab and select the trials value cell (C4) in the Parameters table. Then hit enter. Now copy that formula all the way down to C12. Return to the simulation tab and click the "Clean Formula" button if you need to. Your table should look like:

Value	Freq	Probability
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0
11	0	0
12	0	0

15. Return to the simulation tab and run the simulation. After it completes, return to the new tab. Your table should look something like:

Value	Freq	Probability
2	1	0.01
3	4	0.04
4	15	0.15
5	12	0.12
6	14	0.14
7	15	0.15
8	14	0.14
9	9	0.09
10	8	0.08
11	4	0.04
12	4	0.04

16. Now, let's make a chart. Select the first two columns of your table and click Insert, and select the "Clustered Column Chart". You should get a chart that looks like:



Yours will be different due RANDOMNESS!

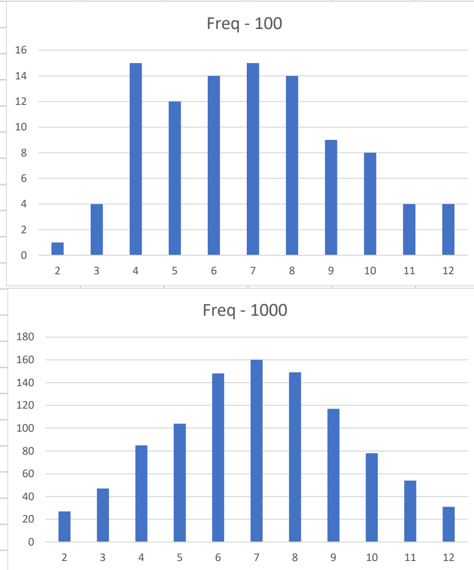
17. 100 iterations are not very many. We want to test more. And we want to be able to compare the results. Return to cell B2 and replace its contents with: "=CONCATENATE("Freq - ", 'Test TPE Creation'!Trials)" This will make the cell read: "Freq – 100". It will also make the title in the chart match. Be sure return to the simulation tab and click "Clean Formula" whenever you need to.

18. We want to preserve the results of this experiment.
    a. Select the chart then click on the "Home" ribbon tab.
    b. Select the down arrow next to copy, and select Copy as Picture.
    c. Hit enter.
    d. Move the chart out of the way and select the cell D2.
    e. Click Ctrl+v to paste the image on the page.

19. Let's test this simulation with more iterations. Return to the simulation tab, change the Trials value to 1000 and run the simulation again.

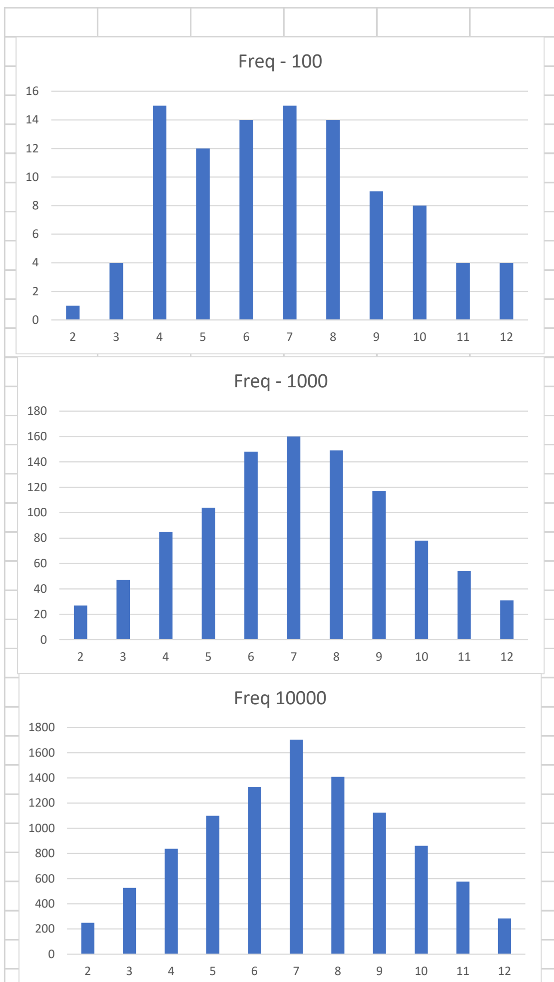
20. Return to the new tab and repeat step 18 to preserve the chart from this run. Paste it below the chart for 100. Your charts should look something like this, with differences for randomness:

Value	Freq - 1000	Probability
2	27	0.027
3	47	0.047
4	85	0.085
5	104	0.104
6	148	0.148
7	160	0.16
8	149	0.149
9	117	0.117
10	78	0.078
11	54	0.054
12	31	0.031



21. Notice the following:
    a. How much longer it took to run the simulation with 10X more trials.
    b. How much more triangular the plot looks like at 1000 iterations. More random trials gets us closer to reality.

22. Repeat steps 19 and 20 for 10,000 trials. This will take a while to run. The results should look something like this, with differences for randomness:



23. Notice how perfectly triangular this is! We could now formulate a TPE for throwing two D6 as "TRIDIST(2,7,12)". This estimate would represent the likelihood of actual values from reality.

24. Notice:
    a. We might be tempted to think the likelihood of values 2 through 12 could be represented by a uniform distribution of "Randbetween(2,12)". But we now know that this would drastically overestimate the high and low values.

    b. From a limited number of trials (100), we might be tempted to think the likelihood of getting a value near the high or low values is much greater than it actually is. In my example above, it appears greater at the low values (Freq – 100). Yours is probably different.

    c. The middle chart might tempt us to think the distribution of values is bell shaped. But this would lead us to overestimate values on either side of the actual mean of 7. For instance, values of 5, 6, 8 and 9 are much closer to 7 than in reality.

    d. No matter which if us runs the simulation, by the time we reach the 10,000 iteration level, the differences in our charts disappear. The table below shows the probability (%) of my last 10,000 iteration run and the calculated probability (%), along with the error. Note how consistently the experiment approaches reality.

Value	Sim Results	Calculated	Error
2	3.000%	2.778%	0.222%
3	5.320%	5.556%	0.222%
4	8.600%	8.333%	0.222%
5	10.760%	11.111%	0.222%
6	13.590%	13.889%	0.222%
7	16.550%	16.667%	0.222%
8	13.380%	13.889%	0.222%
9	11.430%	11.111%	0.222%
10	8.790%	8.333%	0.222%
11	5.590%	5.556%	0.222%
12	2.990%	2.778%	0.222%

25. What do you think might happen if we ran 100,000 iterations?

26. End of Exercise.