Due Date for Assignment 2

The due date for Assignment 2 is midnight on Monday night, July 20.

Note: Before you open any new simulation spreadsheet, you should close all other spreadsheet files you may have open on your computer. If two or more simulation files are open at the same time, Excel recalculates everything on all of them whenever it recalculates anything. This recalculation of multiple spreadsheets can take a very long time! So be sure you have only one spreadsheet open at a time.

Building the spreadsheet simulation model

You can think of building any spreadsheet model as having six steps. The list of steps is given below. The rest of this study guide discusses certain aspects of these steps in more detail.

1. Enter the known information.

   Place all the known quantities together in a section of the spreadsheet (usually at the top). Name the ranges if you so desire, but it’s not required. Subsequent formulas should refer to the cells in this section of data, and alternatives can be tested just by changing these input values. The point here is: Keep data separate from formulas. Doing so allows you to construct formulas more quickly with cell references, and to change or expand your model much more easily.

2. Construct the formulas of the model.

   The model should be a logical flow of formulas based on the known or assumed mathematical relationships and dependencies in the data. Use cell references to the data section above; don’t type in numbers. With cell references, the spreadsheet will update automatically if a piece of input information is changed.
3. Define the random input cells and choose the distributions.
   Identify which quantities are uncertain. In the simulations, each of these uncertain quantities will be a random value from a particular probability distribution. Choose the probability distribution (Uniform? Normal distribution? Triangular?) for each random input cell.

4. Define the output cells to forecast.
   Identify the quantity or quantities of interest. These are the cells containing the computations of the values you are interested in: the total profit, or the total time required, or the NPV, or the maximum number of … whatever. These will be the output values calculated by the simulations. There can be more than one output cell of interest.

5. Run the simulation.
   Decide how many iterations to run. Construct the list of simulation formulas. Recalculate the spreadsheet a few times to be sure random values have been selected.

6. Interpret the results. Calculate a frequency table and other statistics from the simulations. Create a histogram or other appropriate chart.
   You should always create frequency tables from the simulations for the main desired output cells. Make 8 to 12 bin ranges. Along with the frequency table, a histogram (column chart) of the frequencies or the percent frequencies is a good idea. In addition, you will typically compute the mean, standard deviation, maximum and minimum values for the output cells.
   The frequency tables and other statistics allow you to draw some useful conclusions from your simulation runs. You can answer questions such as, “How likely is a given value for the outcome?” or “How likely is an outcome less than x or greater than y?” The answers to these questions are simply percent (probability) calculations from your frequency table. The question, “What is the highest possible outcome under this set of assumptions?” is answered by the maximum value in your series of simulations.
   Other, more advanced sensitivity analysis techniques will be dealt with later on.

**Running more complex simulations:**
**The Excel Data Table trick**

We can simulate random values in a cell by using Excel’s RAND function, usually with an inverse probability function such as NORM.INV. That’s what you did in Assignment 1. But that procedure works best for a single cell. The simulation’s output cell of interest may, however, be a formula, not a random value. The formula may refer to other cells containing RAND.
For example: The simulation of an income statement might use a random number for sales revenue (within a certain range), a random number for cost of goods sold, a random number for salary and wage expense, a random number for the interest rate to be used to compute interest expense, and a random number for that year’s average tax rate.

The value of interest is net income. But the net income cell is simply a sum function. Copying that sum function down for several thousand iterations won’t work. Making lots of copies of a cell which just refers to other cells containing RAND functions returns the same number in every iteration. There is no range of outcomes.

Alternatively, we could try to construct a row of formulas for our model. Each cell in the row could contain a RAND function to generate a number for sales revenue, costs, etc. And then this line of formulas could be summed across to a net income number. The model is contained all in one row of formulas. To run 10,000 simulations, we could just copy that line of formulas down into 10,000 rows.

Sounds reasonable, but it doesn’t actually work very well. The main reason is calculation time. The more RAND functions a worksheet has, the longer it takes Excel to calculate that worksheet. If we have 10,000 rows of formulas, and each row contains several RAND functions, it would take a very long time for Excel to complete each recalculation.

Instead, we need to construct the model and its RAND functions one time, in one place on the worksheet. Then we’ll ask Excel to recompute that one model several thousand times. In other words, we’ll ask Excel for thousands of iterations, or simulations, of the one model. Excel will make a separate set of computations for each iteration or simulation, and record each result for us – thousands of times.

The way to do that is to use an Excel Data Table. But a special little trick is required.

An Excel Data Table makes the same calculation multiple times, using a different value for one of the inputs each time. For example, suppose you want to compute the net present value of a series of cash flows, using several possible discount rates. The Excel function for net present value is =NPV(rate, future cash flows). To test different discount rates, you set up a Data Table as shown below. The formula for the NPV uses a cell reference (this is critical) for the discount rate. So the formula in cell D9 below is =NPV(D7,D5:H5).

Try recreating this example on your own. It’s good practice.
We will construct the Data Table in columns C and D, beginning in row 12. The Data Table will compute the NPV using values from 5% to 11% shown in column C.

To create this Data Table, designate a range in two adjacent columns. In the left column, enter the list of values to be substituted. Then place a cell reference to the desired formula at the top of the right column, one row above the list of trial values. You can see this structure in the example above.

Now highlight the entire Data Table. Click on the menu item for a Data Table. In Excel for Windows 2007, 2010 or 2013, or in Mac Excel 2011, activate the Data ribbon and click on What-If Analysis | Data Table. In Mac Excel 2008 or Windows Excel 2003, click on Data | Table (open the Data menu and click on Table).

This brings up the dialog box shown below.
This Data Table dialog box is not very helpful, is it? It’s not obvious what you should do next. But it’s in this dialog box that you tell Excel where the values in the left column of the table are to be substituted. In this case, we want them to be substituted, one by one, into cell D7 for the discount rate. And the trial values in the table are in a column (they’re all in column C), so enter D7 in the “Column input cell” field.

The phrase “Column input cell” in the dialog box is asking for the address where a column of trial values is to be substituted. These are discount rates, so they are to be substituted into the discount rate cell, which is D7. That’s our “Column input cell.”

If the list of trial values to be substituted were all in a row, instead of a column, then we would type the input cell address into the “Row input cell” field. But we normally use columns.

Then when you click OK, the Data Table is created, as shown below. Make sure you **press F9 if necessary** (Cmd = on a Mac), to have Excel calculate all the cells.

As you can see below, Excel has calculated the NPV at each of the discount rates and recorded the results next to each trial value.
The trick for running many, many simulations instantly

That’s how you create a simple Data Table, to test multiple values for an input. But if you want Excel to compute thousands of simulations for you, on a model which includes random values, you use a little trick. It goes like this:

In the left column of the Data Table, just list the simulation sequence numbers (1, 2, 3, etc.)

Then when the dialog box comes up, place a reference to any blank cell in the “Column input cell” box.

A Data Table for simulating the roll of a die is shown below. To handle the way Excel rounds, the formula in cell D2 (the top-right cell of the Data Table) is:

\[ =\text{ROUND}(0.5+\text{RAND}()\times6,0) \]

This generates a random number from 1 to 6, and rounds it to whole number.

I entered the sequence numbers for 20 simulations into Column C. Then at the top of the next column, Column D, I placed that formula. So the Data Table is in Columns C and D.
To create these simulations, I selected the entire Data Table and called up the Data Table dialog box. As the column input cell, I designated cell C1, but it could be any blank cell.

This little dice Data Table contains only 20 simulations. But one nice thing about Excel is that you can make your Data Table almost as big as you want. Just run your sequence numbers down through, say, 10,000. Highlight all 10,000 rows (plus the top cell where the formula is) for your Data Table. Enter a reference to a blank cell in the Column Input Cell field. Click OK, and presto! You have simulated rolling a dice 10,000 times.

Excel can compute 10,000 iterations of a single-column Data Table like that in just a couple of seconds. It’s making all the computations on just one set of RAND functions, in one construction of the model. Therefore, the Data Table procedure is far, far faster than computing thousands of rows of RAND functions.

Just remember to press the recalc key to force the Data Table to be calculated all the way down.

I have put together some other examples and video demonstrations. These have been posted on the K516 course materials pages in Canvas.

---

<table>
<thead>
<tr>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>11</td>
<td>5</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
</tr>
<tr>
<td>13</td>
<td>5</td>
</tr>
<tr>
<td>14</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>3</td>
</tr>
<tr>
<td>16</td>
<td>5</td>
</tr>
<tr>
<td>17</td>
<td>5</td>
</tr>
<tr>
<td>18</td>
<td>4</td>
</tr>
<tr>
<td>19</td>
<td>3</td>
</tr>
<tr>
<td>20</td>
<td>3</td>
</tr>
<tr>
<td>21</td>
<td>6</td>
</tr>
<tr>
<td>22</td>
<td>1</td>
</tr>
</tbody>
</table>

---

**General Tips for running simulations with Excel Data Tables**

1. Only open **one simulation spreadsheet** at a time. Otherwise the recalc time can be annoyingly long. Excel won’t finish calculating one worksheet until all Data Tables on all open worksheets have been recalculated. And the more open worksheets you have, the more RAND functions you may have. Too many unnecessary RAND functions will slow Excel down.
2. In Excel’s Options, go to the Formulas section. In the Calculation options, select the option for Automatic except tables. This option keeps Excel from trying to recalculate the entire Data Table each time you do something else somewhere in the spreadsheet.

3. Construct your model neatly and logically, all in one place on the worksheet. Use RAND functions to pull random values from whatever distribution you have decided upon for each number in the model which has a range of possible values. Of course, some of the values in that range may be more likely than others, and that’s why we generally use probability distributions and those inverse transformation functions like NORM.INV.

   a. The numbers which can take on a range of possible values are sometimes called “uncertain variables.” But “uncertain” may be too strong a word. If you have a range of possible values for that number, then you have a pretty good idea of the maximum and minimum value that number can take. And you probably have a pretty good idea of how likely are certain values within that range. So the number isn’t completely “uncertain.” It will come from a defined range with defined probabilities in that range.

4. Identify the output value or values. These are the calculated cells in your model which depend on the uncertain variables or the numbers with a range of possible values. In the example above, the output value was net income. It is calculated from all the numbers which have ranges.

5. Use an Excel Data Table to run thousands of iterations (simulations) of your model and record the output values for each iteration. Create a Data Table to compute the output values and enter the address of any blank cell as the Column Input Cell.

6. Allow adequate cooling for your computer. Particularly if you’re using a laptop to do these simulations, it can get hot. The more complicated simulations we will perform later in K516 are very computationally intensive. Some of them will require Excel to crunch for more than a minute. All that time, your computer’s processor and hard disk are creating lots of heat. Prop up one edge, or in some other way give the poor machine access to cool air.