Using the Excel Simulation package

The file *simulations.xls* contains a set of Visual Basic for Applications (VBA) macros and user-defined functions that extend Excel's built-in probability features.

The *simulations.xls* file is intended to be a blank template for creating simulation worksheets. To create a simulation model, open *simulations.xls*, and immediately save it with a new file name (so that you keep a "clean" copy of the original template).

Please send comments, suggestions, or error reports to the author:

Dr. Darryl Nester (nesterd@bluffton.edu).

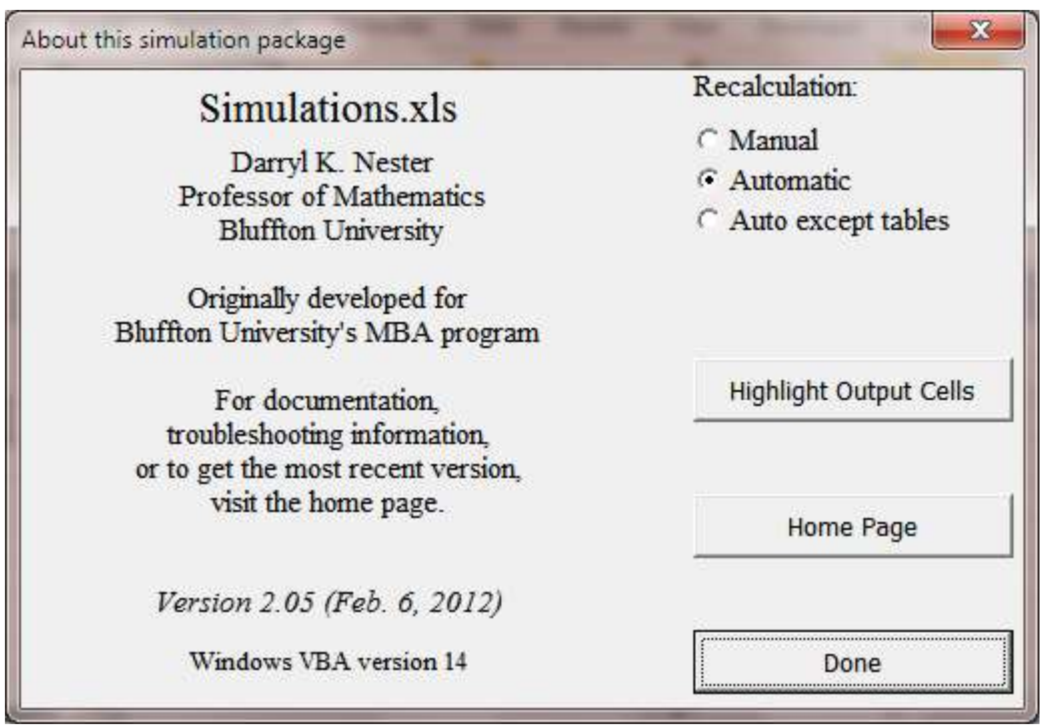
A clean (and up-to-date) copy of *simulations.xls*, as well as more documentation, can be downloaded from tinyurl.com/bluffton-mgt515. This address redirects to the less-easily memorized www.bluffton.edu/~nesterd/MGT515.

Excel will typically ask for confirmation that you want to enable the macros when opening a simulation spreadsheet. (If you do not enable macros, you will not be able to use the simulation package.) This is Excel's default setting; if your installation has been set to maximum security (no macros), you will need to change the security setting. To see how to do this for your version of Excel, consult your local help desk, or search the Internet for "excel change macro security level."

When installed, the following buttons will be available on the "Add-Ins" tab. (In Excel 2003, these buttons will appear as part of a floating toolbar.)

The "About" button

reports the version of the simulation software, and provides three other features that can be useful. Clicking "Home Page" will open the simulations home page in your browser. You can change Excel's recalculation mode (most people are used to Excel's default "Automatic" recalculation, but during the simulation process, it will occasionally get stuck in "Manual" mode). And finally, if you need to find the output cells in a model (perhaps one created by someone else), you can click "Highlight Output Cells" to color them green. (Of course, this is less helpful if there are already green cells in the notebook.)

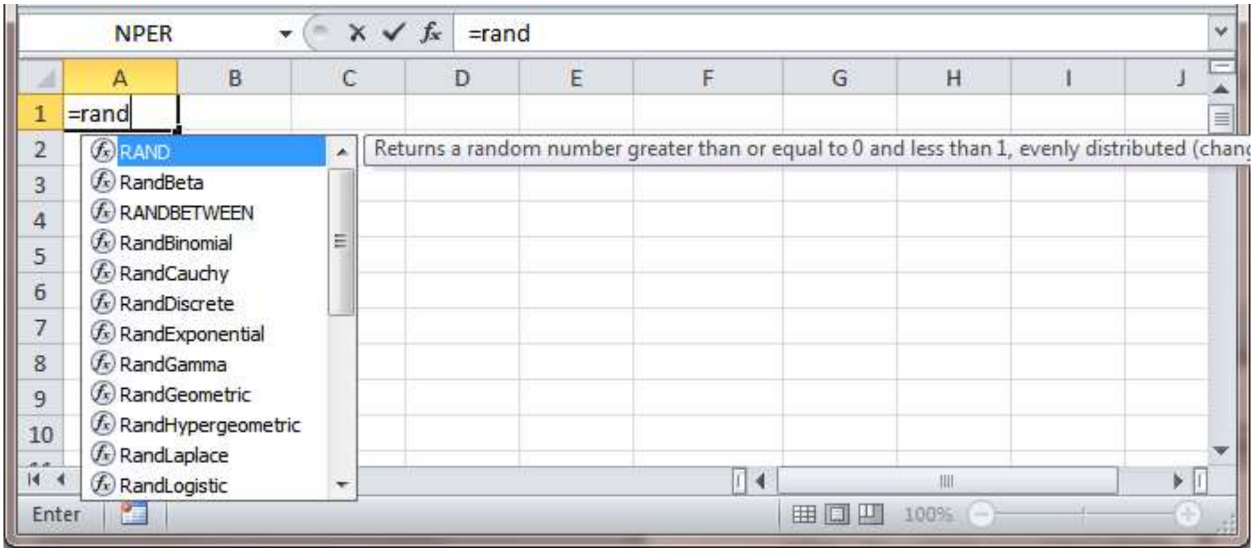


Functions defined in simulations.xls

Most of the added functions are designed to generate random numbers, and are named "Rand_____", where _____ is the name of the distribution. A partial list of available distributions is

Normal, Exponential, Gamma, Triangular, Binomial, Poisson

You will see a complete list if you type "rand" while entering a formula; RAND and RANDBETWEEN are built-in Excel functions; the other "Rand" functions are defined in simulations.xls. Details about these functions are found later in this document.



Other than RAND and RANDBETWEEN, these random functions do not (by default) return random results. Instead, they return fixed values—typically the distribution's mean or median. For example, RandNormal(100, 15) returns 100, RandBinomial(10, 0.75) returns 7.5, and RandUniform(0, 20) returns 10. This can be helpful in setting up your model, because it allows you to see a typical result of the simulation. However, if you want your formulas to return random results, you can click "Toggle random" in the toolbar. (The icon for this button switches between when functions return fixed values, and when they return random values.)

In addition to the Rand____ functions, there are several other useful functions available, including:

- `simOutput([name])` – a "dummy" function used to mark a cell as "output"; that is, if present in a cell, the value of that cell will be tabulated when a simulation is run. Note that this function always returns the value 0, so it will not change the value of the cell if added to other quantities. The optional name is a description of the cell's contents—for example, "Yield" or "Profit" or "Present value." If omitted, the cell's address is used.
- `getFormula(cell)` – returns the formula from the specified cell as a string. This can be useful because it allows the user to see the formulas used in a model as well as the results of those formulas. (Based on information from David McRitchie at http://www.mvps.org/dmcritchie/excel/formula.htm.)
- `PaybackPeriod(cashFlow)` – Given a cashflow, this computes the numbers of terms until the net balance hits 0. The cashflow (which must begin with a negative number) is given either as a range of cells or a list of numbers enclosed in braces. If the balance does not hit 0 by the end of the given flow, the function will (by default) return NA. Optionally, by adding ",TRUE" after the cashflow, the function will extrapolate the cash flow beyond the listed values, assuming that the final inflow continues indefinitely. For example, the formulas:

  `=PaybackPeriod({-100, 25, 20})`, or

  `=PaybackPeriod(A1:A3)` if cells A1:A3 contain the numbers -100, 25, and 20 would return NA because the initial balance of -$100 has changed to -$55 after incomes of $25 and $20. On the other hand, either `=PaybackPeriod({-100, 25, 20},TRUE)` or `=PaybackPeriod(A1:A3,TRUE)` would return 4.75, because the function would assume that each future term brings in $20, so that the balance is -$35, then -$15, and then (three-fourths of a term later) $0—a total of 4.75 terms.

A sample simulation model

Consider a three-year cash flow consisting of revenue, cost, and profit as shown on the right. This model is too simplistic in that revenue and costs are fixed; in a more realistic model, both revenue and costs might vary from year to year. We will construct a model in which revenue varies randomly in the range $500 ± $240 (approximately), and costs vary in the range $300 ± $150. To do this, we assume that revenue (for example) comes from a normal distribution with mean $500 and standard deviation (SD) $80; most numbers from a normal distribution fall in the range (mean) ± (3 × SD).

	Year 1	Year 2	Year 3
Revenue	$500	$500	$500
Cost	$300	$300	$300
Profit	$200	$200	$200

We'll begin by entering the mean revenue and cost in the spreadsheet, as shown in yellow. We also specify the standard deviation for each. Also in this table is a discount rate of 6%. These yellow cells represent *most* of the underlying assumptions in the model. (In addition, we make assumptions about the distributions when we construct the cash flow table.)

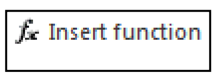
	A	B	C
1		Mean	SD
2	Revenue	$500	$80
3	Cost	$300	$50
4	Discount rate	6%	

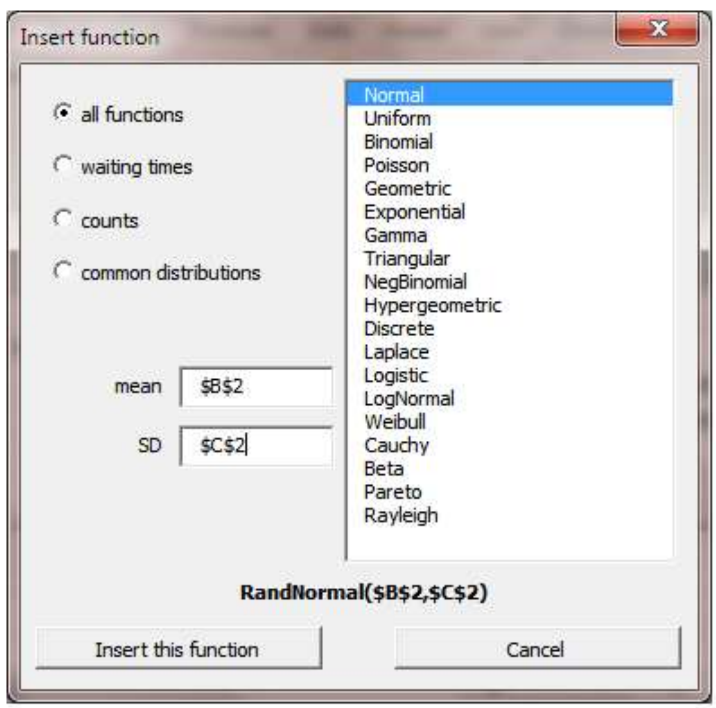
Below that, we create a table for each year's revenue/cost/profit combination. The formula for profit is what one would expect; for example, in year 1, cell B9 contains the formula `=B7-B8`. The formulas for revenue and cost, however, are:

```
=RandNormal($B$2,$C$2)
=RandNormal($B$3,$C$3)
```

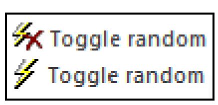
	A	B	C	D
5				
6		Year 1	Year 2	Year 3
7	Revenue	$500	$500	$500
8	Cost	$300	$300	$300
9	Profit	$200	$200	$200
10				
11	NPV	$534.60		

These formulas can be typed in like other Excel formulas, or they can be selected using the "Insert function" button on the simulations toolbar. With cursor on cell B7, click "Insert function," choose "Normal," and enter the mean and SD as shown. (These cell references need to be typed in; you cannot use the mouse to click on cells B2 and C2.) When finished, click "Insert this function."



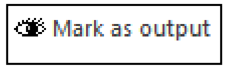


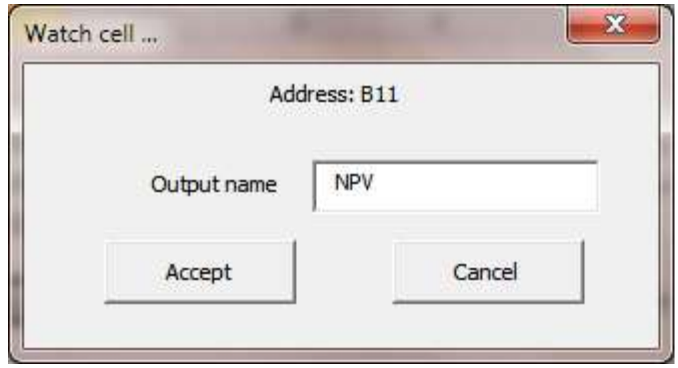
While these formulas are displayed as $500 and $300, they represent random values from a normal distribution with the specified mean and standard deviation. (To see them as random instead of fixed values, click the "Toggle random" button from the toolbar. The red X on the icon means these functions are non-random; when the X is missing, they are random.)



Copy the formulas from year 1 into years 2 and 3, and in cell B11 enter the formula

```
=NPV(B4,B9:D9)
```

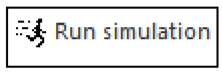


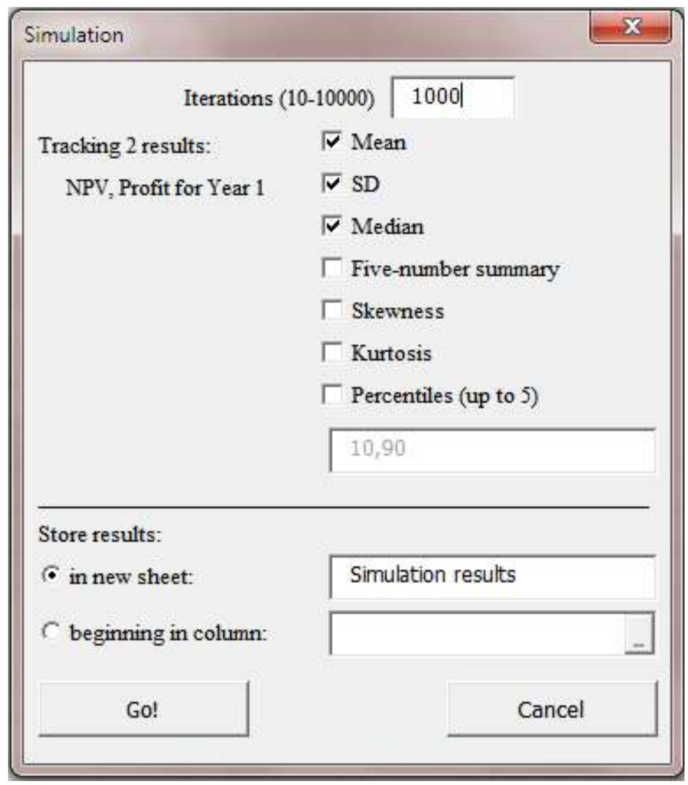


Every model should have one or more cells to "watch" – that is, formulas which represent the output of interest in the model. In this example, we probably want to track B11, the NPV of the cash flow. For each such cell, select it and click "Mark as output." This will bring up a window asking for a name for this output. The default name will be the address of the cell, or an "educated guess" for the name (if there is a label in the cell to the left of or above the selected cell). You can change this name if you wish, then click "Accept." This will add the function "simOutput" as part of the formula in the current cell, so that (for example) cell B11 now contains the formula

```
=simOutput("NPV")+NPV(B4,B9:D9)
```

If you click "Mark as output" for a cell that was previously marked for output, you can change the output name.



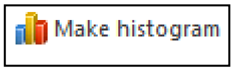


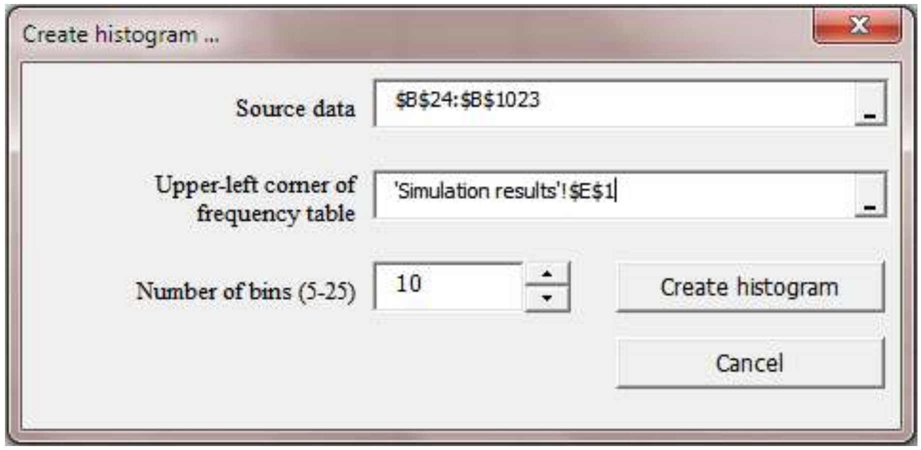
Once the model is complete and output cells have been marked, click "Run simulation." This will repeatedly generate random numbers for each of the Rand___ functions in the model, and make a list of the values in the output cells. You will be prompted for the number of simulated trials, and then the results of the output cells will be saved in separate sheet (the default), or within the current worksheet. It will also compute a selected set of summary statistics for the output values. If you run the simulation again, those results can be saved in yet another separate sheet. In the screen shown on the right, cell B9 ("Profit in Year 1") has been marked as output, along with the NPV.

Shown on the right is an example of the simulation results. (The actual simulated values are stored beginning in row 24.) Based on this report, we can see (for example) that in 1000 simulated cash flows, the NPV averaged about \$530—similar to the "nonrandom" NPV of \$534.60—but it varied fairly substantially, roughly (though not exactly) in the range (mean) ± (3 × SD).

	A	B	C
1	Simulation results for Sheet1		
2	1000 iterations		
3		NPV	Profit for Year 1
4			
5	Mean	$529.89	$199
6	SD	$148.74	$96
7	Min	$65.93	($100)
8	Q1	$431.98	$136
9	Median	$533.75	$199
10	Q3	$627.00	$271
11	Max	$987.24	$486

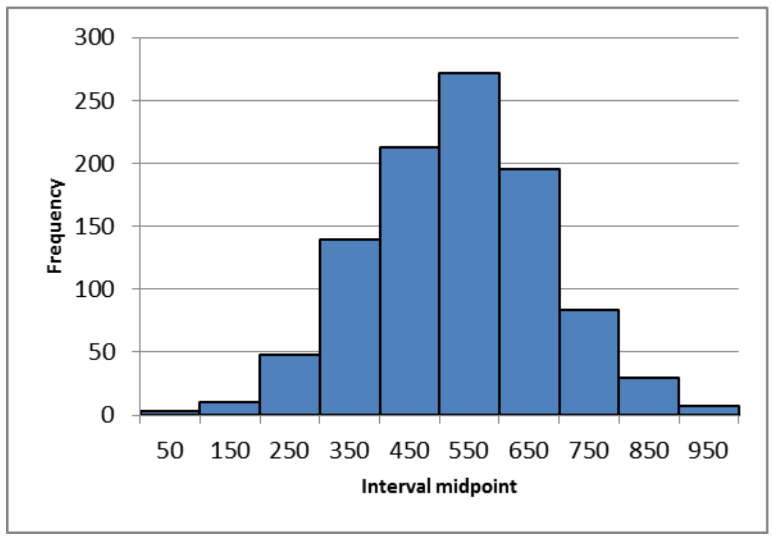
To get a more detailed view of the distribution of simulated results, one can create a frequency table and histogram. The "Make histogram" button automates this process: Simply specify the location of the data, and where the frequency table should be placed. (The mouse can be used for both of these selections.) One can also specify how many "bins" (intervals) should be used for the frequency table. The settings on the right could be used to show the distribution of the simulated NPV results.





The bin endpoints of the frequency table will typically be kind of "messy" (like those below, left), but they can be adjusted by changing the specified value of "Min" and "Step." Shown (below, right) is the histogram that was produced by setting Min = 0 and Step = 100.

Min		65.9338
Step		92.22289076
Total count		1000
Bin Label	Bin Max	Frequency
112.0452454	158.1566908	9
204.2681361	250.3795815	18
296.4910269	342.6024723	79
388.7139177	434.825363	155
480.9368084	527.0482538	219
573.1596992	619.2711446	248
665.3825899	711.4940353	164
757.6054807	803.7169261	73
849.8283715	895.9398168	27
942.0512622	988.1627076	8



Additional notes:

- In theory, you can have as many models as you want in a single notebook, although you should have each model contained in its own sheet. (When you click "Run simulation," only the model in the visible sheet will be run.) However, if you find your simulation is not working well, it may help to limit yourself to one model per notebook.
- Running a simulation for a complex model might be fairly slow. If the program estimates that a 1000-step simulation will take more than 30 seconds, it asks for confirmation before proceeding.
- Occasionally, when you open a notebook containing a model, the toolbar will not appear. Should that happen, close the notebook and open it again.
- Conversely, from time to time, the toolbar will remain after the notebook is closed. If this happens, you can (with Windows) right-click or Ctrl-click on any button and choose "Delete Custom Toolbar" to get rid of it, or (with a Mac) click on the toolbar's red "close dot."

A complete list of the simulation (Rand____) functions

Arguments in square brackets [] are optional; if omitted, they take the specified default value. If "Toggle random" is "off," the function returns the specified value (typically the mean). Valid parameter values are given; invalid values result in a #NUM error.

Miscellaneous random functions:

`RandDiscrete(values, [weights])`

A random number from a discrete distribution with a specified set of possible values. If weights are not given, all possible values are equally likely; if they are given, they determine the relative probability of each possible value.
Return value: the median of the distribution (in the order given in the list of values)
Valid parameter values: values is either a cell range (e.g. A1:A10) containing a list of numbers, or a list of numbers enclosed in braces—e.g., {1,2,3,4,5}. If given, weights should be a cell range or a list, the same length as values, and should contain no negative numbers.

`RandSum(count, x)`
`RandProduct(count, x)`

Computes the sum (or product) of count terms, each using the expression in x. count is an integer expression (possibly random), while x should be either a cell reference (to a cell containing a numeric value or formula) or a string containing a valid numeric formula. For example, `RandSum(5,"rand()")` would compute the sum of 5 random numbers returned by the Excel function `RAND()`. `RandProduct(RandPoisson(10),A5)` would evaluate the formula in cell A5, multiplying the result a random (Poisson) number of times. *Note:* These two functions are not very inefficient; if you can avoid using them, do so. For example, `RandSum(5,"rand()")` is evaluated more slowly than the formula `=RAND()+RAND()+RAND()+RAND()+RAND()`.
Possible values: depends on the values of count and x
Return value: $\text{count} \times x$ (`RandSum`) or x^{count} (`RandProduct`)

`RandPermutation(list)`

Given a list of numbers (either in a specified range of cells, or entered as a list like "{1,2,3,4,5}"), this function returns that list in scrambled order. Because this function returns an array (multiple values, rather than just one value), it must be entered by highlighting a range of cells, typing the formula, and then pressing CTRL-SHIFT-ENTER (or Control-Shift-Return).

For example, suppose the range A1:A10 contains a set of numbers. Highlight cells B1:B10, type the formula `=RandPermutation(A1:A10)`, and then press CTRL-SHIFT-ENTER. Now B1:B10 will contain a permutation (shuffled copy) of cells A1:A10. Or, if you highlight cells A1:E1, the formula `=RandPermutation({1,2,3,4,5})` would result in a random ordering of the numbers from 1 to 5.

Note that unlike the other Rand____ functions, `RandPermutation` always returns a random result regardless of the state of "Toggle random," but it will only produce new random results when "Toggle random" is "on."

Continuous distributions:

`RandBeta(mean, SD, [low=0], [high=1])`
`RandBetaAB(alpha, beta, [low=0], [high=1])`

A random number from a beta distribution with a given mean and standard deviation SD, optionally rescaled from low to high, OR a beta distribution with parameters alpha (α) and beta (β), optionally rescaled, which then has mean = $\text{low} + \frac{\alpha}{\alpha+\beta}$ and $SD^2 = \frac{\alpha\beta}{(\alpha+\beta)(\alpha+\beta+1)}$.
Possible values: (low, high)
Return value: the mean
Valid parameter values: low < mean < high, 0 < SD and $SD^2 < (\text{mean} - \text{low})(\text{low} + 1 - \text{mean})$
OR alpha > 0, beta > 0, and low < high

```
RandCauchy([center=0], [scale=1])
```

A random number from a Cauchy distribution with the specified center and scale parameters.
Possible values: $(-\infty, +\infty)$
Return value: center
Valid parameter values: $\text{scale} > 0$ (but no error is returned if $\text{scale} \leq 0$)

```
RandExponential(mean)
```

A random number from an exponential distribution with the specified mean.
Possible values: $(0, +\infty)$*
Return value: mean
Valid parameter values: $\text{mean} \neq 0$ (*if $\text{mean} < 0$, the possible values are negative)

```
RandGamma(mean, SD)
RandGammaAB(alpha, beta)
```

A random number from a gamma distribution with the specified mean and standard deviation, OR the specified parameter values alpha (α) and beta (β), which then has mean $\alpha\beta$ and $SD^2 = \alpha\beta^2$.
Possible values: $(0, +\infty)$
Return value: the mean (that is, $\alpha\beta$)
Valid parameter values: $\text{mean} > 0$ and $\text{SD} > 0$, OR $\text{alpha} > 0$ and $\text{beta} > 0$

```
RandLaplace([center=0], [spread=1])
```

A random number from a Laplace (double-exponential) distribution with specified center and spread.
Possible values: $(-\infty, +\infty)$
Return value: center
Valid parameter values: $\text{spread} > 0$ (but no error is returned for invalid spread)

```
RandLogistic([center=0], [spread=1])
```

A random number from a logistic distribution with specified center and spread.
Possible values: $(-\infty, +\infty)$
Return value: center
Valid parameter values: $\text{spread} > 0$ (but no error is returned invalid scale)

```
RandLogNormal(mean, SD, [low=0])
```

A random number from a log-normal distribution with specified mean and SD, offset by the amount low. The associated normal distribution has variance $\sigma^2 = \ln(1 + (\text{SD}/(\text{mean} - \text{low}))^2)$ and mean $\mu = \ln(\text{mean} - \text{low}) - \sigma^2/2$.
Possible values: $(\text{low}, +\infty)$
Return value: mean
Valid parameter values: $\text{mean} > \text{low}$ and $\text{SD} > 0$

```
RandNormal([mean=0], [SD=1])
```

A random number from a normal distribution with specified mean and SD.
Possible values: $(-\infty, +\infty)$
Return value: mean
Valid parameter values: $\text{SD} > 0$ (but no error is returned if $\text{SD} \leq 0$)

```
RandPareto([low=1], [alpha=1])
```

A random number from a Pareto distribution with shape parameter alpha.
Possible values: $(\text{low}, +\infty)$
Return value: the median $= \text{low} \times \sqrt[\text{alpha}]{2}$
Valid parameter values: $\text{low} > 0$ and $\text{alpha} > 0$

`RandRayleigh([sigma=1])`

A random number from a Rayleigh distribution with parameter sigma.
Possible values: $(0, +\infty)$
Return value: the median $=$ sigma$\sqrt{\ln(4)}$
Valid parameter values: sigma > 0

`RandT(df)`

A random number from a *t* distribution with degrees of freedom df.
Possible values: $(-\infty, +\infty)$
Return value: 0
Valid parameter values: df > 0

`RandTriangular(low, high, [peak=(low+high)/2])`

A random number from a triangular distribution ranging from low to high, with specified peak value.
Possible values: $(\text{low}, \text{high})$
Return value: peak
Valid parameter values: low < high, and low ≤ peak ≤ high

`RandUniform([low=0], [high=1])`

A random number from a continuous uniform distribution ranging from low to high.
Possible values: $(\text{low}, \text{high})$
Return value: the mean = (low + high) / 2.
Valid parameter values: low < high (but no error is returned for invalid values)

`RandWeibull(lambda, k)`

A random number from a Weibull distribution with scale parameter lambda and shape parameter k.
Possible values: $(0, +\infty)$
Return value: the mean = lambda × Γ(1 + 1 / k)
Valid parameter values: lambda > 0 and k > 0

Discrete distributions:

`RandBinomial(trials, probSuccess)`

A random number from a binomial distribution: The number of successes in trials independent attempts, each with probability of success probSuccess.
Possible values: $\{0,1,2,\ldots,\text{trials}\}$
Return value: the mean $= \text{trials} \cdot \text{probSuccess}$
Valid parameter values: trials is an integer > 0; 0 < probSuccess < 1

`RandGeometric(probSuccess)`

A random number from a geometric distribution: The number of failures before the first success, where each independent attempt has probability of success probSuccess.
Possible values: $\{0,1,2,\ldots\}$
Return value: the mean $= (1 - \text{probSuccess}) / \text{probSuccess}$
Valid parameter values: 0 < probSuccess < 1

`RandNegBinomial(probSuccess, numSuccesses)`

A random number from a negative binomal distribution: The number of failures before succeeding numSuccesses times, where each independent attempt has probability of success probSuccess.
Possible values: $\{0,1,2,\ldots\}$
Return value: the mean $= \text{numSuccesses} \times (1 - \text{probSuccess}) / \text{probSuccess}$
Valid parameter values: 0 < probSuccess < 1, and numSuccesses > 0

`RandHypergeometric(nPop, nSample, r)`

A random number from a hypergeometric distribution: The number of "special" items in a sample of size nSample from a population of size nPop, or which r are special.
Possible values: $\{\max(0, \text{nSample} + \text{r} - \text{nPop}), \ldots, \min(\text{r}, \text{nSample})\}$
Return value: the mean $= \text{nSample} \times \text{r} \,/\, \text{nPop}$
Valid parameter values: 0 < nSample < nPop, 0 < r < nPop, all integers

`RandPoisson(mean)`

A random number from a Poisson distribution.
Possible values: $\{0,1,2, \ldots\}$
Return value: the mean
Valid parameter values: mean > 0

Darryl Nester, March 31, 2015