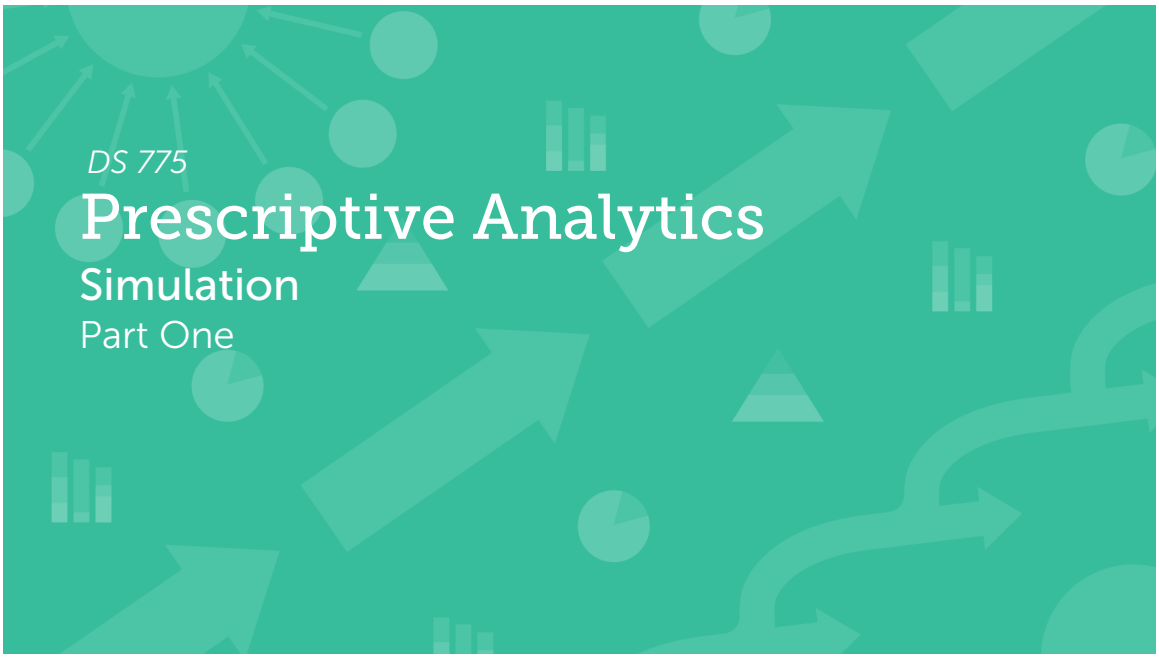

Slide 1

Simulation: What is it?

Simulation is using a computer to imitate the operation of a process or system in order to estimate its actual performance.

Components of a simulation model:

- A definition of the state of the system
- Identification of the possible states of the system
- Identification of possible events that could change the value of the system
- A simulation clock
- Methods for randomly generating events
- A way to relate the state transitions to the events that brought them about

Slide 2

Motivation for Simulation

Simulation is used for

- Mathematical models that are not tractable
- Analyzing stochastic (random/probabilistic) systems that operate indefinitely to gain insight into the behavior of the system over time
- Analyzing and designing systems that would otherwise be very time consuming and/or expensive
- Experimenting with a system without actually implementing it

Slide 3

Limitations of Simulation

- Simulations have inherent variability, so they provide only statistical estimates rather than exact results (analytical methods provide exact results when tractable)
- Simulations compare various alternative without necessarily finding the optimal one
- Even with today’s computers, some complex simulations still require large amounts of computing time and expense in programming and analysis

Slide 4

The optimal solution must be one that's feasible, that is, one that doesn't violate any of the constraints. The collection of all such possibilities is called a feasible /

More Simulation Limitations

- Simulations provide only numerical measures about the performance of a system and cause-and-effect relationships are not always evident
- Simulations results apply only to the conditions that were simulated
- Sensitivity analysis can be unwieldy in large simulations

Slide 5

Self-Assessment Question

Simulation is used for analyzing and designing systems that would otherwise be very time consuming and/or expensive.

- True
- False

Correct answers can be found at the end of this transcript.

Slide 6

Self-Assessment Question

Simulations provide exact results just like analytical methods.

- True
- False

Correct answers can be found at the end of this transcript.

Slide 7

VIDEO

Slide 8

Simulations come in all shapes and sizes. It takes a lot of creativity, programming skills, and ability to model physical processes with random variables and functions and equations. And put it all together so you that you can capture the important statistics on each simulation run and summarize those in some way so that you can gain insight into the process or system that you are simulating.

The details of this game being played can be found in your textbook so I won't take up time here. I don't want this video to be any longer than it has to be. I just wanted to show you a few basic things in this particular simulation.

One is generating uniform random numbers, the RAND function does that. They are generated on a uniform distribution, which is continuous from zero to one. And in this simulation, the result is set up such that using and if statement, if that random number is less than 0.5, we're going to say that it's heads. If it's not, then we're going to list the result as tails.

The if statement is set up in such a way that you have some logical statement in the beginning. If that value is true, then the first thing that occurs. If that value is false, then the second thing is what takes place.

Now if I want to conduct this simulation one time, all I have to do on any of these random outcomes that are up here, if I just put my cursor up there and hit Return, it will rerun the game and you can see, in this case, it took five flips.

Since we are paid $8 at the end of each game but we pay $1 for every flip, we win a total of $3, the stop sign being over here.

Since we're talking about if statements, this is an example of a nested if statement. The best way to understand these is from the inside out. And remember, this simulation comes with range names on these various cells. And so that's why these are given in here.

What it says is if the absolute value of the total heads minus total tails is greater than or equal to the required difference, that means the difference between heads and tails is greater than or equal to three. If that's a true statement, then put the word stop in that cell. If that's false, put a blank in that cell. So that's the inner if statement.

Going on the outer if statement, if the particular cell before that one-- notice my cursor is in cell G17 but this is referring to G16-- if G16 is blank, then we go to this first if statement. That's in the true state spot for the outer if statement. If it's false, then we're going to print NA there. Because basically, once we stop the simulation, we need to just say these cells after that are not applicable.

But we also need to have something to fill them up because the number of flips is obtained by this count blank statement. It's counting the number of blanks in the range name titled stop with the question mark, which is really G13 through G62. This is telling me-- one, two, three, four-- there are four blanks. And then we're going to add one because that was the final flip. And that's why this is counting the five flips.

And then the winnings are simply cash at end of game, which is the $8 here minus the number of flips since it was $1 per flip. So eight minus five is three, pretty simple.

VIDEO

Slide 9

Now, this simulation simply shows one trial of the game when we run it. But the important thing of a simulation is to be able to replicate that a large number of times. In the textbook, they show a what-if analysis, building a table of 14 observations and showing the results of that-- the average winnings and the average number of flips. But 14 isn't very big. That's really not very useful.

Then they go on to build a table of 1,000. Well, again, that's kind of cumbersome to actually show the results of all 1,000 of the simulations, although it's kind of nice to be able to do that. But there is a very quick way to get to the results using the Analytic Solver Platform.

And using the simulation dialog on the side, the thing we're most interested in here are the winnings at the end of the game. That's really our uncertain function that we're very interested in. So I'm going to list that as an uncertain function. So if I highlight over here, click Plus, that lists the winnings of the coin-flipping game as an uncertain function.

So that's important to do. And you'll notice that when I clicked on that, maybe you didn't see this before, but plus PSI output, that was added on there. So that's the way of tagging that cell as what we're going to gather up and collect on every simulation that's run.

I'm just going to put a little text in this cell as a label saying that I'm going to grab the mean of the winnings and put them in cell F8. I don't want a range

name on that. I'm just adding to this file that's here. But this is going to gather up this statistic for every trial of the simulation and then give me a report on it.

So I want to highlight Statistics Function over here on the right, say Plus. Now, here's where you have to be careful. You have to choose the right things. So under Select a Category, these are the ones called the PSI things, and we want PSI Statistics. And underneath here, we get lots of options. I think the mean would be a good thing for us to look at here. So PSI Mean is the one we want. So we want to find the average number of winnings in the long run. That'll tell us something important.

So what is this cell that we want to find the mean of? Well, it's the winnings. So I can click over here and grab that one, and it puts the range names. I could just have easily have typed D8. And that's all I need for now. That's all I want, and I'm going to just say OK. Well, for now it says that's the one. Cell F8 is the statistic function that's going to gather up. Right now I haven't run the simulation, so there's nothing in there.

But when I do that, I know I can do that by coming to this green arrow. Let's go ahead and run the simulation. And there is a nice summary of what it's giving me. It tells me that the mean winnings is negative $1, standard deviation is $7, and give me some percentiles. And actually, this is useful too-- a frequency distribution histogram showing me that this distribution is very skewed to the left, and here's 0.

So a lot of times, I am going to win money, up to $5. But I could lose big on a small number of turns, in fact, so much so that in the long run I'm going to lose $1. And in the textbook they tell us that analytically this is solvable, though it's not easy. And so we could have found out that we would lose $1 in the long run analytically as well. Over here in these statistics, note that the minimum, in this case, is $41.

Now, the question is, this doesn't tell me how many observations did I run here in this simulation. Well, I'm going to get this out of the way. We can find out by looking at Platform, for example. Trials per Simulation is 1,000. In fact, this is one place I can go to change that if I want to do 5,000 or 10,000. Another place I can go to is under the Analytic Solver Platform ribbon at the top, under Options and under the Simulation tab again, 1,000 is the default. But you can come in here and change this as you wish.

One other thing we can do under Analytic Solver Platform and Reports and Simulation, over to Simulation Reports, this will create a summary table of the simulation, which shows some important things, namely the trials per simulation. And if I scroll down just a little bit here, you can see that for my

1,000 values, actually it wasn't 1. It was rounded to 1 in the output. It was really negative 1.268 and a standard deviation of 7.17.

Now, this is going to change every time I run the simulation because these are random outcomes. So in this case, the analytic result, that's going to be a constant at negative 1. But every time I run this simulation, this average for the winnings ought to be around negative 1, but it will vary because it's a simulation. Coming back to the spreadsheet, here's the cell that we put the PSI mean in, the mean of the winnings, negative 1.268. So we could track that there too.

There's one last thing I want to show you with this particular simulation. Let's do a sensitivity analysis on the cash at the end of the game so we can decide whether it's worthwhile playing or not, or maybe we want to adjust this in some way. So under Analytic Solver Platform, very similar to as we've done before, but Parameters, this time under Simulation, and we'll say, what's the lower value? I don't know, let's just say $5, and the upper value is, oh, maybe $12, so we can evaluate this.

OK. Well, it starts it out on the lower end. But I want a report on this. Go to Reports, Simulation, and Parameter Analysis, and it brings up this box. The mean is something we can look at, but there's quite a few other things. You may want to look at the standard deviation or some other measurement there, confidence intervals for those things. For now, we're going to get a mean for the cash at the end of the game, for the payment at the end of the game, between $5 and $12.

And we can pick how many major access points. Maybe we want 10 of those. That means it's going to conduct this simulation 10 times, and currently we still have 1,000 runs per simulation. But that's still really isn't very big for a simulation size, but we're going to leave it at that for now. Say OK, and it crunches its way through all of these simulations in not too big of a time and shows me the spreadsheet here.

Notice cash at the end of the game. When we had $8, as we had before, pretty close to negative 1 there and all of these negatives that are below that. So we wouldn't want to play the game if we were the player. If we're the ones conducting the game, maybe we do, because we'll be winning that much.

Even at $9 per coin toss, still losing money. It's not till it's $10 or more that that's-- in fact, that looks like, yeah, it's actually $9.67 there. These are just rounding them to the nearest whole number, $10.44. So there at $9.67, we're finally in the positive on this one.

VIDEO

Slide 10

Hopefully you've taken some time to get familiar with problem 20.6-3 from the textbook. This is a spreadsheet that addresses that problem and answers the question. Well, there are four relays, each characterized by a uniform distribution from 1,000 to 2,000 hours for their time to failure.

Now, generating a random number from that distribution, if you have the ASPE add-in you can use what's called a PsiUniform. And that simply selects a random number from a uniform distribution with the lower boundary 1,000 and upper boundary 2,000. I simply refer to those cells to pick those numbers.

If you weren't sure where to find this, as long as you have the ASPE add-in you can just start typing equals and Psi, and you'll see lots of different Psi functions. If you're going to use PsiUniform, then you would type a U next. And that's the only one in the U's, so double clicking that. And I could refer to these cells as I did before, or it is possible just simply to type your lower bound and upper bound that you want for that continuous uniform distribution. I got one too many 0's in there.

OK. I have the first three generated that way. But I put one of them in here in case you don't have that as an add-in. If you just have access to the Rand function, which generates a uniform random variable from 0 to 1, you can still simulate from a uniform distribution from 1,000 to 2,000 according to this function. So we're going to add to the 1,000 starting point the uniform random number times 2,000 minus 1,000, and that will generate a uniform random number between 1,000 and 2,000.

The time to first failure is simply the minimum of these 4 relay times. The textbook tells us that it's a 2-hour repair time in replacing all 4 relays. And the total time to the end of the shutdown was just simply the sum of these 2 times.

The total cost for the repair is $1,000 for the 2 hours that the system is down plus $200 for each of 4 relays that are replaced. That gives a cost per hour of total cost divided by the time to the end of the shutdown, in this particular case, $2.23. Well, that's just one simulation of it. The current cost is $3.19.

Well, what we would like to know is, if I simulate this a large number of times, we'll say 1,000 times, for example, then we'll compute the mean cost per hour for all of those simulations to gain insight into that system. So I'm going to open the Analytic Solver Platform menu over here so we can see how it's loaded up.

The Uncertain Variables are these 4 relay times. Those are variables that are involved in the simulation. The Uncertain Function is the cost per hour. And the statistic that I want to gather up is the mean cost per hour. So those have all been put in here, as you've seen before.

So to run that simulation, I will click on the green button. And here are some results, lots of results here. But I'm going to get this screen out of the way just so we can see that that put the $2.37 as the average cost per hour for this way of doing things, whereas if you replace each relay individually, it was $3.19 per hour. And therefore, this is a better way to do it. Just replace all 4 relays when the first one fails. That's what this simulation shows.

Generating Random Numbers

Random Number Generator - An algorithm that produces sequences of numbers that follow a specified probability distribution and possess the appearance of randomness

Uniform (0,1) - Continuous and equally likely between 0 and 1

Inverse Transformation Method

1. Generate a uniform random number r between 0 and 1
2. Set $F(x) = r$ and solve for x, which is a random observation from the distribution with cumulative distribution function $F(x)$

Slide 11

Random number generation is an integral part of any simulation using a computer. Random variables are just about inescapable. So it's crucial to be able to realistically simulate the behavior of any variables involved in a simulation. Generally speaking, a random number generator is an algorithm that produces sequences of numbers that follow a specified probability distribution and possess the appearance of randomness.

Random numbers can be generated by physical devices, such as a spinning disk with the spacing of sections marked on the disk corresponding to probability, rolling dice, or even something like the machine that tumbles numbered ping-pong balls until one is selected, which you may have seen when they're choosing lottery numbers.

These physical devices are limited in the probability distributions they can draw random values from, not to mention the time needed to use them. There are also printed tables of random numbers that can be used, but again, these are not practical for large-scale simulations. Computers are definitely the way to generate random numbers today.

Hopefully you're familiar with both discrete and continuous probability distributions. A good starting point for random number generation is either with a discrete uniform distribution, which consists of random integer numbers that are equally likely between any two specified integers, and the continuous uniform distribution, which includes all real numbers between a lower and upper bound with each equally likely to be selected.

The continuous uniform distribution on the interval from 0 to 1 is particularly useful in the generation of random numbers from other probability distributions using the inverse transformation method. Step one of the inverse transformation method is to first generate a random number from a uniform distribution on the interval from 0 to 1.

And then step two is to set the Cumulative Distribution Function, or CDF, of the distribution you want a random value from equal to the uniform random number and then solve for x, which will then be a random observation from that distribution. The randomly generated variable from the uniform distribution serves as the cumulative probability associated with a particular value in the chosen distribution.

Random numbers generated by computers are often referred to as pseudo-random numbers because the sequence generated is not infinite in length and so will eventually repeat. And when you look behind the scenes, they're actually predictable and reproducible. This little fact is helpful because a particular simulation could be reproduced exactly if the random number generation begins with the same seed. The seed is the starting point from which the random number sequence is generated.

Self-Assessment Question

According to the Inverse Transformation Method, the random observation (rounded to 3 decimal places) generated by the uniform (0,1) random number $r = 0.731$ for a distribution with $F(x) = 1 - exp(-x^2/5)$ and $x > 0$.

- 2.562
- 1.252
- 1.146
- 0.512
- 6.565

Correct answers can be found at the end of this transcript.

Slide 12

VIDEO

Slide 13

As stated before, the computer is a good way to generate random values in a simulation, and we have a few options to do that. In fact, there are quite a few options out there if you look around. But since we have the benefit of the Analytic Solver Platform in this course, we're going to take a look at that first. But I will show you a few others as well in this video.

Well, clicking on the Analytic Solver Platform ribbon, we can see over here a menu called Distributions. And if I click there, this menu comes up. And there are a variety of choices, a menu that says Common Distributions, and here are some examples. As you get more familiar with these, you'll find out what might be a good distribution for various situations.

For example, beta distributions are continuous, and they are bound between 0 and 1. So oftentimes they're used to model probabilities. Normal distributions are good for things that might be symmetric and mound-shaped, exponential, gamma, log normal.

Well, we have a couple of different log normal options here and Weibull. Those are good distributions to use for lengths of time, maybe a time to failure or time to a specified event, repair time, recovery time, things like that. Time often turns out to be something that's skewed to the right like that.

The triangular distribution is used in a few examples in the textbook. There are more advanced continuous distributions. Discrete distributions-- Bernoulli, binomial, geometric, and these things. And as you learn more about these

maybe in other venues, you'll find out when's a good time to use each of these.

Well, let's just see how to generate random variables here. So let's say I want to generate something from this gamma distribution. So I click on that, and it brings me this. And over here, there are a lot of different things about that particular gamma distribution. In this case, the default is that the shape parameter's 4, and the scale parameter's 1.

But you could change these, maybe play around with them. If I change that to 3 and click Save, you see how it changes what it looks like. If I click Save, that just puts that PsiGamma function into that cell. So there's one random observation. And you wonder, why did it pick this value? Well, if I come up here and hit Return again, it'll generate a new value. Or let's say I copy several of these, they're all randomly generated values.

Another way to access the random number generator through the ASPE is to open the Model window and select Uncertain Variable. And instead of clicking the big plus, click the little dropdown next to it. Say we want to Add an Uncertain Variable, and this same dialog box pops up.

But if you don't want a normal distribution, you can click in that cell and you have the choice of all the same options we saw before. So instead, say we would like a Weibull distribution, and there you have it. Also, you see in here a Seed that we could set with any of our random variables. And just choosing a particular seed could give us that ability to reproduce an exact simulation, starting that sequence of pseudo-random numbers at the same spot.

VIDEO

Slide 14

If you have only the basic Excel package without something like an ASPE add-in, you still have options. Now, to enter a random variable, because it's a number you would start with an equal sign. An option for discrete uniform distribution is given by RANDBETWEEN, selecting this option. And you get a little help, Bottom and Top.

So if I wanted to go between 20 and 50, Enter, and there's my random number between 20 and 50. If I want a uniform random number, it's simply RAND with some empty parentheses. That generates a uniform random number between 0 and 1.

Now, there are some other distributions, inverse distribution functions, in the basic Excel. You can use the inverse transformation method principle with those and the uniform distribution from 0 to 1 with the RAND function to generate those. For example, from the normal distribution, I can pick NORM.INV for norm inverse and parenthesis.

Now, the probability comes first here. So that's where I would enter the RAND statement comma and whatever mean and standard deviation I'd like to generate from. So there's a random value from a normal distribution, as you saw with a mean of 100 and a standard deviation of 20. If I click in here again and hit Enter, it'll generate a new value.

Some other options are GAMMA.INV the inverse gamma function. And again, the probability comes first, and then alpha and beta are the shape and scale

parameters. So you would have to know a little bit something about what you were doing as far as the shape that you were generating from and the type of distribution. I just picked a few values for example.

The other options are log normal. Once you type log and you see LOGNORM.INV, then you can double click that. And you could get a little help here as far as where to put things. Let's choose a different mean and standard deviation, whatever is appropriate for the situation.

Another one for a continuous distribution is the chi-square, so chi-square inverse. And we can type RAND and some degrees of freedom to generate a random value there. That's a random value generated from a chi-square distribution with 7 degrees of freedom.

The other option is F dot inverse, or another option I should say. We're somewhat limited. And here you have a numerator and a denominator degrees of freedom to control that distribution. A t-distribution is another option, T inverse. And in this case, it's the probability and then the degrees of freedom, and we get a random value there.

And the last one is the beta distribution, the last one that's part of the standard package. So beta inverse, and the probability comes first. And then the alpha and the beta are the parameters of the beta distribution that you are generating from. And you'll get random values there.

So that's how you would manage it. You're a lot more limited, but you do have options for generating random numbers for other distributions using the concept of the inverse transformation method in the basic Excel.

VIDEO

Slide 15

In the basic stats package in R, you can generate random observations from a variety of distributions. And one way to see what's available is to type the question mark and distributions to see what is there. So in my help page over here, I can see lots of different distribution functions. Here are some of the ones we saw in Excel-- beta, binomial-- and some we didn't see over there. But chi-square, exponential, F, gamma, geometric, hypergeometric, you have a lot of options.

So let's use the normal distribution. That one we tend to see from time to time. Now, it says dnorm, but if I click on that, it goes to that page that tells me all about it. And actually, all of those distributions have a similar design. Well, the d is for the density function. The p is for the distribution function. The q is for the quantile function. And what we're interested in is rnorm to generate a random number.

So on the left side, if I type rnorm and some sample of, say, size 10 and a mean of 75 and a standard deviation of 3 and return, I get 10 random observations. If you see some of the other distributions that are in here, but you're not quite sure what they look like, you could generate a sample and then make a histogram of it to see what the shape of it looks like.

So rgamma, for example, and I get a little help again. So maybe a sample of, well, let's generate 1,000, and let's store this as some object. And 3 and 2-- just pick a shape and a scale parameter. So if I make a histogram of that, I can see what that looks like. Oh, it's skewed to the right. It starts at 0, goes out to

about 5 on these maximum. Just to get a feel for, is that going to match up with the random variable that you're trying to model?

Now, something else that's very interesting is set seed-- oops-- is set seed. For random number generation, the seed is basically the starting point for it. Watch this, watch this. If we do rnorm again and just generate one observation with a mean of 30 and a standard deviation of 4, just for example, I get this value. And then if I do it again, I get another random value and another random value and another random value, and they're all different.

But if I set the seed to really any number, I could just pick a number, say, 12. So I set the seed and I generate the rnorm, I get a value. And I set the seed again and I generate a value, boom, it's the same one. That's what that seed does.

This will also work for an entire sample of numbers. Maybe I don't want just one. Maybe it is 10 that I want, and I get that. And I reset the seed at 12, generate another 10, but they're the same 10. So you see the special thing about setting that seed for the random number generator.

Self-Assessment Question

Simulation results can be reproduced exactly by running the simulation again using the same _____.

- seed.
- computer.
- plant.
- method of random number generation.
- simulation clock.

Correct answers can be found at the end of this transcript.

Slide 16

Self-Assessment Quiz Question Answers

Question 1
Simulation is used for analyzing and designing systems that would otherwise be very time consuming and/or expensive.

Correct Answer:
True

Question 2:
Simulations provide exact results just like analytical methods.

Correct Answer:
False

Question 3:
According to the Inverse Transformation Method, the random observation (rounded to 3 decimal places) generated by the uniform (0,1) random number $r = 0.731$ for a distribution with
$F(x) = 1- exp(-x^2/5)$ and $x > 0$.

Correct Answer:
2.562

Question 4:
Simulation results can be reproduced exactly by running the simulation again using the same _____.

Correct Answer:
Seed