CRN in DES script

Hi everyone, I’m Hanxuan Yu, a research analyst from Vanderbilt Health Policy, and I’m applying for Phd in health policy and decision science for next year.

We declare no conflicts of interest for this project.

The stochastic noises are always annoying for us when we build simulation models. However, common random numbers are here to help!

For the definition, it is the coordinated use of random numbers that we are applying the same random number for the same event among all the runs. For disease modeling, this technique has been applied in microsimulation, but it is under-explored in discrete event simulation.

To know how to use random numbers in discrete event simulation, we give a simple Healthy-Sick-Death DES model here. A person can die of background mortality, for example, a car accident, or get sick and die of the disease. We will generate random numbers for the time to background death, the time to get sick, and the time to die of the disease after getting sick.

When it goes to the way to sampling the time, we use inverse transform sampling for cumulative distribution. First, we generate a random number between 0 and 1 from a uniform distribution, which will work as a quantile, and then put u into the cumulative distribution function to get the corresponding time t. For the same u, distributions with different rates will produce different times.

Go back to our Healthy-Sick-Death example, assuming the usual way to do the simulation: set seed which helps to produce the same random number sequence between runs, but do not use common random numbers. For the status quo, the rate of getting sick from Healthy is 0.1. So, we generate the random number for background death and the number for getting sick. Compare the two numbers for time, the next event is getting sick. Then generate the number of dying of the disease, so the next event is background death, happening in the 21st year. The life expectancy will be 21 years.

For patient 2, the time to background death is 30 years, lower than the time to get sick. For patient 3, the time to get sick adding the time to die of the disease is 16 years, lower than the time to background death. The life expectancy will be 16 years.

assuming an intervention reduces the rate of healthy to sick. With a lower incidence, we will expect a higher life expectancy. For the first person, with the same u, the time to sick changes to 22 years, which led to the person going to background death directly, still in the 21st year.

Because person 1 only uses 2 instead of 3 numbers, For the next person, the following simulated events will all change. It is like the Domino effect. The numbers for him will all change. He got sick in the 11th year and died of disease in the 27th year. Person 3 gets one more year when getting sick. The random number misalignment will influence the following simulated events, and for person 2, a lower incident led to a lower life expectation, the stochastic noise is misleading here.

But if we use common random numbers here, we pre-sample and distribute the random number for each event and each person, like these, the numbers will not change along the runs, and the lower incidence will only lead to an improved life expectation.

Here we see common random numbers reduce stochastic noises, which will lead to fewer trajectories to converge totally and help with sensitivity analysis. I’ll show you the improvement of efficiency in the following case study.

On the other side, it will also make programming more complicated. Also, as we need to generate all the random numbers for all possible events and people ahead of time, this may take some time to compute.

Let’s look further into the case study for the net health benefit of using statins to reduce the risk of ASCVD in the US population based on NHANES.

When doing the one-way sensitivity analysis for the relative risk of ASCVD, a higher relative risk should lead to a lower incremental net health benefit between statins versus no treatment. The blue line is the smooth trend line, and the ribbon is the 95% confidence interval. Using common random numbers significantly reduces the noise from the scatter plot.

On the other hand, when comparing the two strategies, the x-axis here is the sample size we used for simulation. common random numbers also contribute to faster stabilization of results around the true value with a smaller sample size.

One of my mentors, John published a paper indicating DES is more efficient and accurate than other simulation models. The common random number can make it even better. We’ll need it in lots of computational-intensive tasks.

Thanks for listening and welcome for questions!