Chapter 7

More on Simulation of Random Variables

7.1 Random number generators

A *random number* is a stochastic variable with a uniform distribution over the interval $(0,1)$. A *random number generator* is a mathematical method to generate so called *pseudo random numbers* that is some sense are close to truly being independent random numbers. Note: True random numbers only exist as a mathematical abstraction - they cannot be manifactured in real life.

Starting with three non-negative integers, *the multiplicator* a, *the modulus* b, and *the seed* x_0, one can generate (pseudo) random numbers $y_1, y_2, y_3, \ldots$ with a so called *multiplicative kongruential random number generator* according to the scheme

$$\begin{cases} x_i = (a\,x_{i-1}) \pmod b \\ \qquad y_i = x_i/b \end{cases},$$

where $z \pmod b$ denotes the remainder when integer dividing z by b.

After a transient period of length $\hat{b}$ at most $b-1$, a multiplicative kongruential generator is periodic with period p at most b (as there are only b different possible values, and each random number y_i is uniquely determined by its predecessor y_{i-1}). Hence the obtained random number sequence takes the form

$$y_1, \ldots, y_{\hat{b}},\ y_{\hat{b}+1}, \ldots, y_{\hat{b}+p},\ y_{\hat{b}+1}, \ldots, y_{\hat{b}+p},\ y_{\hat{b}+1}, \ldots, y_{\hat{b}+p},\ \cdots\cdots\cdots\ .$$

Here it is desirable to select a, b and x_0 such that p is not too small as compared with the upper bound (for its possible value) b.

For a multiplicative kongruential generator it is in fact the case that $p<b$ (because if 0 is a possible value, then $p=1$). But the period $p=b$ is possible if on considers a so called *mixed kongruential generator* given by

$$\begin{cases} x_i = (a\,x_{i-1} + c) \pmod b \\ \qquad y_i = x_i/b \end{cases},$$

where the non-negative integer c is called *the increment*. Of course, also a mixed kongruential generator is periodic with period p at most b (after a transient period of length $\hat{b}$ at most $b-1$).

Theorem 7.1. (Hull–Dobell) *A mixed kongruential generator has period $p = b$ if and only if the following three conditiond hold:*

- *b and c are relatively prime (i.e., the only integer that divides both b and c is 1);*
- *$b = p_1 \cdot \ldots \cdot p_n$ where $p_1, \ldots, p_n$ are primes $\Rightarrow a\,(mod\ p_1) = \ldots = a\,(mod\ p_n) = 1$;*
- *b is divided by 4 $\Rightarrow a\,(mod\ 4) = 1$.*

Another important calss of random number generators as so called *feedback shift register methods.* These methods are related to congruential generators, but are somewhat more sophisticated.

We now state two more useful results for congruential generators:

Theorem 7.2. *Given any integer $m \geq b$, the period p for a multiplicative congruential generator is given by*

$$p = \min\left\{n \geq 1 : a^{n+m}x_0\,(mod\ b) = a^m x_0\,(mod\ b)\right\}.$$

Theorem 7.3. (Knuth) *A multiplicative congruential generator such that $b = 2^d$ with $d \geq 4$ has period $p \leq 2^{d-2}$. If in addition $a\,(mod\ 8) = 3$ and x_0 is odd, then $p = 2^{d-2}$.*

7.2 Test of random number generators

No random number generators generates true random numbers! In fact, the numbers you get are virtually always not random at all! Also, usually (as for the congruential generators we have looked at), if you know one random number x_n or y_n, and if you know what is the algorithm for the random number generator, then you can compute all future random numbers $x_{n+1}, x_{n+2}, \ldots$ $(y_{n+1}, y_{n+2}, \ldots)$ generated by the generator. It is therefore a difficult task to specify what to mean with a "good" random number generator. And it is then seldom any easier to verify that a given generator is good, in the sense one has selected to specify.

It is easy to e.g., by means of a χ^2-test check that a sequence of random numbers does not significantly deviate from a uniform distribution over $(0, 1)$.

A somewhat more sophisticated test of a generator is to gather consecutive random numbers in to n-tuples and test whether these have a uniform distribution over $(0, 1)^n$. Note that this test besides testing the uniform distribution also testes whether the random numbers are independent.

Obviously it is inappropriate to use random number generators whose period p is short in comparison with the number of random numbers one plans to generate. A second test thus consists os checking (e.g., using Theorems 7.1-7.3) that p is large enough.

Really efficient test procedures of random numbers are hard to implement and analyze theoretically. For example, such tests can make use of the fact that an infinite sequence of n-tuples of random numbers from a congruential generator is contained in a finite number of parallell hyperplanes in $\mathbb{R}^n$: The smaller the largest distance between two consecutive such planes, the better is the generator. (Obviously, for true random numbers this distance is zero.)

7.3 Simulation of Markov chains

A sequence $\{X_n\}_{n\geq 0}$ of discrete valued random variables is called a *Markov chain* if

$$\mathbf{P}\big\{X_{n+1}=j \bigm| X_n=i_n,\ldots,X_0=i_0\big\} = \mathbf{P}\big\{X_{n+1}=j \bigm| X_n=i_n\big\} = \mathbf{P}\big\{X_1=j \bigm| X_0=i_n\big\}.$$

Markov chains constitute one of the few most important modelling tools for random timke series in science. Their importance can hardly be exagregated.

The probability for different events for a Markov chain is determined by the transition square matrix $P=(P_{i,j})$ of transition probabilities and the initial distribution column matrix $\mu=(\mu_j)$ given by

$$P_{i,j} = \mathbf{P}\big\{X_{n+1}=j\big|X_n=i\big\} \quad \text{and} \quad \mu_j = \mathbf{P}\{X_0=j\},$$

respectively.

If $\xi_0,\xi_1,\ldots$ are independent random numbers, then a Markov chain with a specified transition matrix P and initial distribution μ can be simulated according to the following scheme:

(0) Simulate the initial value x_0 for X_0 with discrete density function $f_{X_0}(j)=\mu_j$ using ξ_0;

(1) Simulate the value x_1 for X_1 with discrete density function $f_{(X_1|X_0=x_0)}(j) = P_{x_0,j}$ using ξ_1;

(2) Simulate the value x_2 for X_2 with discrete density function $f_{(X_2|X_1=x_1)}(j) = P_{x_1,j}$ using ξ_2;

$\vdots$

(n) Simulate the value x_n for X_n with discrete density function $f_{(X_n|X_{n-1}=x_{n-1})}(j) = P_{x_{n-1},j}$ using ξ_n.

$\vdots$

7.4 Simulation of expected values

For a stochastic variable X with an unknown expected value $\mathbf{E}\{X\}=\mu$ and a variance $\mathbf{Var}\{X\}=\sigma^2<\infty$, we want to find a confidence interval for the expected value μ by means of simulating independent identically distributed observations $X_1,X_2,\ldots$ of X in the computer. (It is assumed that we can create such observations of X.)

According to the central limit theorem, the sample mean $\overline{X}=\sum_{i=1}^n X_i/n$ is approximately normal $\mathrm{N}(\mu,\sigma^2/n)$ distributed for large n. Hence, if for a $p\in(0,1)$ the number $\lambda_p\in\mathbb{R}$ solves the equation

$$\mathbf{P}\{\mathrm{N}(0,1)\le\lambda_p\} = 1-p,$$

it follows that

$$\mathbf{P}\left\{\overline{X}-\frac{\lambda_{p/2}\,\sigma}{\sqrt{n}}\le\mu\le\overline{X}+\frac{\lambda_{p/2}\,\sigma}{\sqrt{n}}\right\} = \mathbf{P}\left\{-\lambda_{p/2}\le\frac{\overline{X}-\mu}{\sigma/\sqrt{n}}\le\lambda_{p/2}\right\}\approx 1-p, \qquad (\star)$$

so that an approximative $1-p$-level confidence interval for μ is given by

$$\mu\in\left(\overline{X}-\frac{\lambda_{p/2}\,\sigma}{\sqrt{n}},\overline{X}+\frac{\lambda_{p/2}\,\sigma}{\sqrt{n}}\right).$$

The approximation error for the above confidence interval goes to zero as $n\to\infty$. In fact, according to the so called Berry-Essen theorem, it holds that the difference between the probability to the left in $(\star)$ and the probability to the right $1-p$ goes to zero at the rate $(\mathbf{E}\{|X|^3\}/\sigma^3)/\sqrt{n}$ as $n\to\infty$.

If σ is unknown, which is often the case, then σ can be estimated by the sample standard deviation s given by $s^2=\sum_{i=1}^n(X_i-\overline{X})^2/(n-1)$, to obtain the approximative confidence interval

$$\mu\in\left(\overline{X}-\frac{\lambda_{p/2}\,s}{\sqrt{n}},\overline{X}+\frac{\lambda_{p/2}\,s}{\sqrt{n}}\right).$$

It can be argued that it is more accurate to use quantiles for the Student t distribution instead of nornal distributions quantiles when using the sample standard deviation to estimate the standard deviation. However, this is really a matter of taste, and usually it is the above formula that is used.

Quite often it is desired to decide a sample size n such that a certain specified width b of the confidence interval is obtained, given by

$$b=\frac{2\,\lambda_{p/2}\,\sigma}{\sqrt{n}}\approx\frac{2\,\lambda_{p/2}\,s}{\sqrt{n}} \quad\Rightarrow\quad n=\left(\frac{2\,\lambda_{p/2}\,\sigma}{b}\right)^2\approx\left(\frac{2\,\lambda_{p/2}\,s}{b}\right)^2.$$

Here it is common that the values of σ and s are unknown before the simulation has been carried out. This means that one has to carry out an initial preparatory simulation based on a rather small and not too time consuming to create sample $\tilde{X}_1,\ldots,\tilde{X}_m$ of observations of X that is used to estimate σ by means of the sample standard deviation $\tilde{s}$ of that initial sample. This estimate $\tilde{s}$ is inserted instead of s in the above formula for n to find an appropriate sample size n for the main simulation which can then be carried out.

Given a desired confidence interval width b, the required sample size n to obtain that width grows in a quadratic manner as a function of σ. To save computyer time it is therefore desirable to try to decrease σ. This in turn can often be accomplished by means of the methods for variance reduction discussed in Section 3 of Chapter 4. These methods use the oiginal sample $X_1,X_2,\ldots$ to create a new sample $Z_1,Z_2,\ldots$ such that

$$\mathbf{E}\{Z_i\}=\mathbf{E}\{X_i\}=\mu \quad\text{and}\quad \mathbf{Var}\{Z_i\}<\mathbf{Var}\{X_i\}=\sigma^2.$$

Then μ is estimated by $\overline{Z}$, for which a smaller n is required to obtain a certain specified precision [see $(\star)$] than if the original estimator $\overline{X}$ had been used.

7.5 Simulation of $\mathbb{R}^2$-valued random variables

The problem to simulate an $\mathbb{R}^n$-valued random variable can be solved by a starightforward extension of the following theorem for 2 dimensions to higher dimensions:

Theorem 7.4. *Let (X,Y) be an $\mathbb{R}^2$-valued random variable and define*

$$F^{(y)}(x) = \mathbf{P}\{X \le x \,|\, Y = y\} \quad \text{for } (x,y) \in \mathbb{R}^2.$$

Given independent random numbers ξ and η, put $\zeta = F_Y^{\leftarrow}(\eta)$. Then we have that

$$\big((F^{(\zeta)})^{\leftarrow}(\xi), \zeta\big) \quad \text{and} \quad (X,Y) \quad \text{are equally distributed.}$$

Proof. Suppose for simplicity that Y is continuously distributed. We have that ζ and Y are equally distributed (see Section 4 of Chapter 4). As ζ and ξ are independent it follows that

$$\begin{aligned}
\mathbf{P}\big\{(F^{(\zeta)})^{\leftarrow}(\xi) \le x,\, \zeta \le y\big\} &= \int_{-\infty}^{y} \mathbf{P}\big\{(F^{(\zeta)})^{\leftarrow}(\xi) \le x \big| \zeta = z\big\}\, f_\zeta(z)\, dz \\
&= \int_{-\infty}^{y} \mathbf{P}\big\{(F^{(z)})^{\leftarrow}(\xi) \le x\big\}\, f_Y(z)\, dz \\
&= \int_{-\infty}^{y} \mathbf{P}\{X \le x \,|\, Y = z\}\, f_Y(z)\, dz \\
&= \mathbf{P}\{X \le x, Y \le y\}. \qquad \Box
\end{aligned}$$

7.6 Simulation of uniform distributions over regions in $\mathbb{R}^n$

We have the following natural extension of the concept of uniform distribution:

Definition 7.1. *An $\mathbb{R}^n$-valued stochastic variable η is* uniformly distributed *over the region $O \subseteq \mathbb{R}^n$ if it holds that*

$$\mathbf{P}\{\eta \in o\} = \frac{\mathit{Volume}(o)}{\mathit{Volume}(O)} \quad \text{for } o \subseteq O.$$

To obtain a stochastic variable ξ that is uniformly distributed over a rectangle $R = [a_1, b_1] \times \ldots \times [a_n, b_n] \subseteq \mathbb{R}^n$ we can just use $\xi = (\xi_1, \ldots, \xi_n)$ where $\xi_1, \ldots, \xi_n$ are independent stochastic variables such that ξ_i is uniformly distributed over $[a_i, b_i]$ for $i = 1, \ldots, n$. Building on this in turn, we have the follwing recipe to simulate a stochastic variable with a uniform distribution over a bounded region $R \subseteq \mathbb{R}^n$:

Theorem 7.5. (Rejection Sampling) *Let $R \subseteq \mathbb{R}^n$ be a rectangle such that $O \subseteq R$. If $\xi_1, \xi_2, \ldots$ are independent stochastic variables that are uniformly distributed over R and $J = \min\{j : \xi_j \in O\}$, then ξ_J is uniformly distributed over O.*

Proof. For $o \subseteq O$ we have that

$$\begin{aligned}
\mathbf{P}\{\xi_J \in o\} &= \sum_{j=1}^{\infty} \mathbf{P}\{\xi_J \in o \,|\, J=j\}\, \mathbf{P}\{J=j\} \\
&= \sum_{j=1}^{\infty} \mathbf{P}\big\{\xi_j \in o \big| \xi_1,\ldots,\xi_{j-1} \notin O,\ \xi_j \in O\big\}\, \mathbf{P}\{J=j\} \\
&= \sum_{j=1}^{\infty} \frac{\mathbf{P}\big\{\xi_j \in o,\ \xi_1,\ldots,\xi_{j-1} \notin O\big\}}{\mathbf{P}\big\{\xi_1,\ldots,\xi_{j-1} \notin O,\ \xi_j \in O\big\}}\, \mathbf{P}\{J=j\} \\
&= \sum_{j=1}^{\infty} \frac{\mathbf{P}\{\xi_j \in o\}}{\mathbf{P}\{\xi_j \in O\}}\, \mathbf{P}\{J=j\} \\
&= \frac{\mathbf{P}\{\xi_1 \in o\}}{\mathbf{P}\{\xi_1 \in O\}} \\
&= \frac{\mathrm{Volume}\,(o)}{\mathrm{Volume}\,(O)}. \qquad \square
\end{aligned}$$

7.7 On the approach to simulation problems

A schematical description of the different steps in solving a simulation problem could be as follows:

Step 1. Selection of mathematical model. Here everything that is going to be simulated is expressed in terms of a mathematical (mathematical statistical) language as quantities that can be simulated. For example, the distribution is specified for stochastic variables that are involved in the simulation. Often some parameters are left as unspecified to be specified later to obtain sufficient flexibility for the model.

Step 2. Selection of parameter values. Before the simulation can be carried out one must specify the values for parameter values that have previously been left unspecified. Here some parameters can often be estimated using some real world information avaiable for the actual real world problem under consideration. Still other parameter values might be determined by an iterative process where different parameter values are tried out until the simulation gives results that are judged to be sufficiently correct.

Most simulations involve repititive scenarios where the number of repetitions is a trade off between available/acceptable computer time and the precision of the result of the simulation. In Step 2 the number of such repetoitions is also determined. Sometimes a preparatory initial simulation is carried out to find the appropriate number of repetitions (see also Section 4 above) before the main simulation is begun.

Step 3. Evaluation of the simulation. Before starting to draw conclusions from the results of the simulation it is important to conduct some self-critic to evaluate the simulation:

Does the mathematical model employed give a sufficiently accurate model of the real world phenomena that is studied?

Is the real world information that is used to decide values for parametrars sufficiently reliable/accurate? Note that if non-robust methods are involved it can happen that a single erroneous data value completely spoils the result of the whole simulation!

Have numerical methods and approximations that have been involved been sufficiently accurate for the simulation to give results that have a sufficient accuracy?

Has the simulation been correcvtly programmed in the computer? In the 1950's computers were programmed by means of coupling together vacuum tubes in different patterns by wires. In the 1970's computers were programmed by means of hole cards (=hålkort). Obviously both these historical methods involved a lot of physical labour, and it was vital to make a more or less correct program already from the beginning. This was even more so for the reason that computers weher rather slow and computer time very expensive. Todays interactive software give answers to commands more or less immediately and it is vital to check these answers in all possible ways to make sure that really the right commands are used, as it is otherwise easy to be fooled into trusting answers from the software just because they are answers.

Step 4. Usage of the results of the simulation. Now it is time to draw conclusions from the results of the simulation. Often these results can be used to obtain improved understanding of the real world problem under consideration in such a way that the mathematical model in Step 1 can be improved. This leads to an iterative approach with a model and simulation results that are improved in each iterative step. But be aware of the issue of mass significance, that is, if chance is given too many opportunities to create a non-representative result of a simulation, then it eventually will.

Remark. The history of science can be viewed as a long sequence of iteratively improved models for the functions of nature. In each iteration some newly discovered aspect of the phenomena that is modelled showed that the present model was insufficient, and made necessary a revised model. There is absolutely no reason to belive that the models of science today are completely correct. Rather, they are just sufficiently accurate descriptions of the real world as we are able to observe it today. Conclusion: A simulation model need not be perfect, and never can be. It is quite sufficient that the model is a sufficiently good approximation of the real world phenomena under consideration with respect to the issue that shall be investigated by means the simulation.