SIMULATION OF RANDOM VARIABLES

We want a random number generator to give a sequence $U_1, U_2, ...$ of random variables which are independent and uniform on $[0,1]$. An obvious problem is that $[0,1]$ is an entire interval but a computer only has finite precision. However, if we can generate random integers $Y_1, Y_2, ...$ such that each Y_k is discrete uniform on $\{0,1,...,m\}$, i.e. $P(Y_k = j) = 1/(m+1)$ for $j = 0,1,...,m$ and let $U_k = Y_k/m$, then U_k is uniform on the lattice $[0, 1/m, ..., (m-1)/m, 1]$ which is a good enough approximation if m is large.

The most common algorithms to generate such Y_k are the so called *congruential* random number generators (or *power residue* random number generators). These start with a value Y_0, the *seed* and generates a sequence of integers by computing the next from the previous according to the formula

$$Y_{n+1} = aY_n + b(\mathrm{mod}(m+1))$$

where a, b and m are fixed integers. Note that this is of course a deterministic sequence, we just want it to "look random". Sometimes such number are therefore called *pseudo-random*. Also note that the sequence is periodic.

Example. With $m = 19$, $a = b = 1$ and $Y_0 = 0$ we get

$Y_1 = 1 \cdot Y_0 + 1(\text{mod } 20) = 1$
$Y_2 = 1 \cdot Y_1 + 1(\text{mod } 20) = 2$
$\vdots Y_{19} = 1 \cdot Y_{18} + 1(\text{mod } 20) = 19$
$Y_{20} = 1 \cdot Y_{19} + 1(\text{mod } 20) = 20(\text{mod } 20) = 0$

which gives the sequence $0,1,2,...,19,0,1,2,...,19,0,1,2,...$ which has each number in the right proportion in the long run, but certainly does not look random. For example, if we generate three observations with this algorithm, then at least two of these will always be consecutive regardless of the seed Y_0. This is clearly not desirable for a sequence that is supposed to be random.■

Example. With $m = 19$, $a = 5, b = 3$ and $Y_0 = 0$ we get
$Y_1 = 5 \cdot 0 + 3(\text{mod } 20) = 3$
$Y_2 = 5 \cdot 3 + 3(\text{mod } 20) = 18$
$Y_3 = 5 \cdot 18 + 3(\text{mod } 20) = 93(\text{mod } 20) = 13$

$Y_4 = 5 \cdot 9 + 3(\text{mod } 20) = 8$
$Y_5 = 5 \cdot 8 + 3(\text{mod } 20) = 3$

which gives the sequence $0, 3, 18, 13, 8, 3, 18, 13, 8, 3, \ldots$ where the pattern $3, 18, 13, 8$ is repeated indefinitely. This sequence looks more random than the previous but instead does not have the right proportions. The period is too short to generate all numbers $0, 1, \ldots 19$. ■

There are theoretical results how to choose m, a and b to avoid these problem and get a sequence which looks random. For practical purposes, m of course must be much larger than in the two examples above.

Assuming thus that we can generate our independent, uniform $[0, 1]$ random variables $U_1, U_2, \ldots$, how can we use this to generate observations from other distributions? The following theorem is very helpful.

Theorem. Let F be a continuous and strictly increasing cdf. Let $U \sim$unif$[0, 1]$ and let $Y = F^{-1}(U)$. Then Y has cdf F.

Proof. Start with the cdf of Y:

$$F_Y(x) = P(Y \leq x) = P(F^{-1}(U) \leq x) = P(U \leq F(x)) = F_U(F(x)) = F(x)$$

since $F_U(u) = u$ (U uniform on $[0, 1]$). ■

Hence if we generate $U_1, U_2, \ldots$ uniform $[0, 1]$ and let $Y_k = F^{-1}(U_k$, then the sequence $Y_1, Y_2, \ldots$ is a sample from the cdf F. This is called the *inverse transformation method*

Example. Generate observations from an exponential distribution with parameter λ. Here $F(x) = 1 - e^{-\lambda x}$ which has inverse $F^{-1}(x) = -\frac{1}{\lambda}\log(1 - x)$ so if $U \sim$unif$[0, 1]$, then the random variable $Y = -\frac{1}{\lambda}\log(1 - U)$ is $\exp(\lambda)$. ■

Example. Simulate (X, Y) uniform on the triangle $\{(x, y) : 0 \leq x \leq 1, 0 \leq y \leq 1 - x\}$, i.e. the triangle with corners in $(0, 1), (0, 0)$ and $(1, 0)$. The joint pdf is $f(x, y) = 2$ which gives the marginal pdf for X

$$f_X(x) = \int_0^{1-x} f(x,y)dy = 2\int_0^{1-x} dy = 2(1-x)$$

which in its turn gives the cdf

$$F_X(x) = \int_0^x 2(1-t)dt = 2x - x^2, 0 \le x \le 1.$$

This has inverse $F^{-1} = 1 - \sqrt{1-x}$ (note that when you solve the quadratic you get two solutions but only the one with "-" is correct between 0 and 1).

Further, the conditional pdf of Y given $X = x$ is

$$f_Y(y|x) = \frac{f(x,y)}{f_X(x)} = \frac{1}{1-x}$$

i.e. $Y|X = x \sim$unif $[0, 1-x]$.

The inverse transformation method thus gives that if U and V are independent uniform $[0,1]$, then

$X = 1 - \sqrt{1-U}$
$Y = V(1-X)$

gives a pair (X,Y) which is uniform on the triangle. The following sequence of Matlab commands simulates and plots 1000 observations on (X,Y):

```
u=random('unif',0,1,1,1000);
v=random('unif',0,1,1,1000);
x=1-sqrt(1-u);
y=v.*(1-x);
plot(x,y,'*')
```

Note that, since X and Y are dependent, you must pair each X-value with its corresponding Y-value. ∎

Example. Simulate (X,Y) uniform on the unit disk $\{(x,y) : x^2 + y^2 \le 1\}$. Use polar coordinates (R, Θ). These are independent and such that Θ is uniform on $[0, 2\pi]$ and R has pdf $f_R(r) = 2r$. This gives cdf $F_R(r) = r^2$ which has inverse $F^{-1}(r) = \sqrt{r}$ and if U, V are independent uniform $[0, 1]$, then

$$\Theta = 2\pi U$$
$$R = \sqrt{V}$$

gives a point which is uniform on the disk. Note that because of independence, you can generate observations on Θ and R independently of each other and pair them any way you want. ■

A good textbook on simulation is "Simulation: A Modeler's Approach" by James R. Thompson.