8 Random Numbers

Examples for Random numbers used in computer simulations:

- Instances with quenched disorder, e.g. spin glasses (interactions are random)
- Simulation at finite temperatures using Monte Carlo algorithms
- Randomized algorithms (deterministic algorithms made random)

Literature: [1, 2].

8.1 Generating random numbers

Computers are deterministic → no true randomness possible.

Randomness created by user (time intervals between keystrokes): not controllable.

Pseudo random numbers: generated deterministically but look random, i.e. have many properties of random numbers: uniform distribution, low correlations.

Linear congruential generator: generates sequence $I_1, I_2, \ldots$ between 0 and $m-1$, starting from given $I_0$.

\[ I_{n+1} = (aI_n + c) \mod m \]  \hspace{1cm} (1)

Random numbers $r$ uniformly in interval $[0, 1)$: $r_n = I_n/m$. Arbitrary distributions (see below).

Task: choose parameters $a, c, m$ (and $I_0$) such that generator is “good” → test criteria needed. Attention: several times results for simulations were wrong because of bad random number generators [3].

Example: $a = 12351, c = 1, m = 2^{15}$ and $I_0 = 1000$ (and dividing by $m$) is “uniformly” distributed in $[0, 1)$ (Fig. 1).
Figure 1: Distribution of random numbers in the interval [0, 1). They are generated using a linear congruential generator with the parameters \( a = 12351, c = 1, m = 2^{15} \).

But they have correlations. Study: \( k \)-tuples of \( k \) successive random numbers \((x_i, x_{i+1}, \ldots, x_{i+k-1})\). Low correlations: \( k \)-dim space uniformly filled. LCGs: points lie on \((k-1)\)-dimensional planes, number is at most \( O(m^{1/k}) \). Above case: few planes, see Fig. 2

Figure 2: Two point correlations \( x_{i+1}(x_i) \) between successive random numbers \( x_i, x_{i+1} \). Linear congruential generator with the parameters \( a = 12351, c = 1, m = 2^{15} \).

Better: \( a = 12349 \) instead, Fig. 3.

“Good generator”: \( a = 7^5 = 16807, m = 2^{31} - 1, c = 0 \). Note: more than 32-bit arithmetic needed, see Ref. [1].

Low-order bits much less random than the high-order bits \( \rightarrow \) numbers in an interval \([1, N]\):
Figure 3: Two point correlations $x_{i+1}(x_i)$ between successive random numbers $x_i, x_{i+1}$. Linear congruential generator with the parameters $a = 12349$, $c = 1$, $m = 2^{15}$.

$$r = 1 + (\text{int}) \left( N \ast (1_n) / m \right);$$

instead of using the modulo

8.2 Inversion Method

Given: `drand()` generating uniformly distributed random numbers in $[0, 1)$.

Aim: random numbers $Z$ according pdf $p(z)$ with distribution

$$P(z) \equiv \text{Prob}(Z \leq z) \equiv \int_{-\infty}^{z} dz'p(z')$$

(2)

target: find a function $g(X)$, such that after the transformation $Z = g(U)$.

Assume $g$ is strongly monotonically increasing i.e. can be inverted →

$$P(z) = \text{Prob}(Z \leq z) = \text{Prob}(g(U) \leq z) = \text{Prob}(U \leq g^{-1}(z))$$

(3)

Since $\text{Prob}(U \leq u) = F(u) = u$ for $U$ uniformly in $[0, 1)$ and with identifying $u$ with $g^{-1}(z)$, we get $P(z) = g^{-1}(z)$, hence $g(z) = P^{-1}(z)$. Works if $P$ can be calculated (eventually numerically) and can be inverted.

Example:

Exponential distribution: probability density $p(z) = \lambda \exp(-\lambda z)$ with $P(z) = 1 - \exp(-\lambda z)$. Hence generate uniformly distributed random numbers $U$ and choose $Z = -\ln(1 - U)/\lambda$.

8.3 Rejection Method

For non-integrable pdfs of non-invertible distribution functions, if distribution $p(z)$ fits into a box $[x_0, x_1] \times [0, p_{\text{max}})$, i.e. $p(z) = 0$ for $z \notin [x_0, x_1]$ and $p(z) \leq p_{\text{max}}$.

Basic idea: generate random pairs $(x, y)$, which are distributed uniformly in $[x_0, x_1] \times [0, p_{\text{max}}]$ and accept only those values $x$ where $y \leq p(x)$ holds, i.e. the pairs which are located below $p(x)$, see Fig. 5.
Figure 4: Histogram of random numbers generated according to an exponential distribution ($\lambda = 1$) compared with the probability density (straight line) in a logarithmic plot.

**algorithm** rejection\_method($p_{\text{max}}, x_0, x_1, p$)

begin
    $found := \text{false}$;
    while not $found$ do
        begin
            $u_1 := \text{random number in } [0, 1)$;
            $x := x_0 + (x_1 - x_0) \times u_1$;
            $u_2 := \text{random number in } [0, 1)$;
            $y := p_{\text{max}} \times u_2$;
            if $y \leq p(x)$ then
                $found := \text{true}$;
            end;
            return(x);
        end
end

Drawback: Many random numbers needed, some are even thrown away.

### 8.4 The Gaussian Distribution

Gaussian distribution with mean $m$ and width $\sigma$ (most commonly used distribution in simulations), pdf: $\sigma$ is (see also Fig. 6)

$$p_G(z) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{(z - m)^2}{2\sigma^2}\right)$$

(4)

Here: $z$ according normal distribution $(m = 0, \sigma = 1)$. General case: use $\sigma z + m$

Neither inversion nor rejection method works here. 3 possibilities

- Work artificially boxed, e.g. in $[-3, 3] \to$ no large values.
Figure 5: The rejection method: points \((x, y)\) are scattered uniformly over a bounded rectangle. The probability that \(y \leq p(x)\) is proportional to \(p(x)\).

- Use central limit theorem: sum of \(N\) independently distributed random variables \(u_i\) (with mean \(m\) and variance \(v\)) converge to a Gaussian distribution with mean \(Nm\) and variance \(Nv\). Use \(u_i\) \((m = 0.5, v = 1/12)\) uniformly in \([0,1)\), \(N = 12\), then \(Z = \sum_{i=1}^{12} u_i - 6\) as desired. Drawback: 12 numbers needed and boxed in \([-6,6]\).

- **Box-Müller method:** take two values drawn from uniformly in \([0,1)\) distributed random variables and set.

\[
\begin{align*}
n_1 &= \sqrt{-2 \log(1 - u_1)} \cos(2\pi u_2) \\
n_2 &= \sqrt{-2 \log(1 - u_1)} \sin(2\pi u_2)
\end{align*}
\]

Proof [1, 2]: Write \(n_1, n_2\) in polar coordinates \((r, \theta)\), i.e. \((r, \theta) = f(n_1, n_2)\), the inverse is:

\[
\begin{align*}
n_1 &= r \cos(\theta) \\
n_2 &= r \sin(\theta)
\end{align*}
\]

We want to obtain the pdfs for \((r, \theta)\). For the general case \((W, Z) = f(X, Y)\) with \(p_{X,Y}\) being the (joint) pdf of \((X, Y)\) we have \(p_{W,Z}(w, z) = p_{X,Y}(f^{-1}(w, z))|\mathbf{J}^{-1}|\) with \(|\mathbf{J}^{-1}|\) being the Jacobi determinant of the inverse Transformation.

Using

\[
|\mathbf{J}^{-1}| = \left| \begin{array}{cc}
\frac{\partial n_1}{\partial r} & \frac{\partial n_1}{\partial \theta} \\
\frac{\partial n_2}{\partial r} & \frac{\partial n_2}{\partial \theta}
\end{array} \right| = \begin{vmatrix}
\cos(\theta) & -r \sin(\theta) \\
\sin(\theta) & r \cos(\theta)
\end{vmatrix} = r \cos^2(\theta) + r \sin^2(\theta) = r \quad (5)
\]

we get

\[
p_{R,\Theta}(r, \theta) = \frac{r}{2\pi} e^{-\frac{r^2}{2}} \quad \text{for } \theta = 2\pi u_2 \quad \text{and } p_R(r) = re^{-r^2/2} \quad (*)
\]

The distribution factorizes in \(r\) and \(\theta\). Hence \(\theta\) can be taken uniformly distributed in \([0,2\pi]\) (i.e. \(\theta = 2\pi u_2\)) and \(p_R(r) = re^{-r^2/2} \quad (*)\). Now it remains
For this, consider $X$ exponentially distributed with parameter $\lambda$, i.e. $p_X(x) = \lambda e^{-\lambda x}$. Let’s take $Y = \sqrt{X}$. We want to obtain the pdf $p_Y(y)$. For the general case $Y = H(X)$ ($H$ strongly monotonic) it is $p_Y(y) = p_X(H^{-1}(y)) \left| \frac{1}{H'(H^{-1}(y))} \right|$. Here with $H(x) = \sqrt{x}$, i.e. $H^{-1}(y) = y^2$ and $H'(x) = 1/2\sqrt{x}$, we get $p_Y(y) = 2\lambda ye^{-\lambda y^2}$. Comparing with (*), we see that taking $X$ exponentially distributed with $\lambda = 0.5$ (i.e. $x = -2\log(1 - u_1)$) and then taking $r = \sqrt{x}$ we get the desired distribution for $r$. QED.

- Simulation of particles in a box is explained in Ref. [4]

![Figure 6: Gaussian distribution with zero mean and unit width. The circles represent a histogram obtained from $10^4$ values drawn with the Box-Müller method.](image)

### 8.5 Exercise

Write a program that generates $10^4$ random numbers for a Gaussian distribution boxed in $[-6, 6]$ using the rejection method and record a normalized histogram of bin width 0.1. Draw the histogram and the Gaussian distribution in gnuplot.

### References


