Dam Truyen Duc
NUID: 061801876

# Special Mathematics Lecture
## Introduction to Probability (Spring 2021)
## Report: Random number sampling Box-Muller transform.

It is of great interest in Brownian Dynamics simulations to study how to generate Gaussian random number $R_G$ with variance of 1. This report introduce the Box Muller method with a corresponding C++ code. I also briefly introduce the Marsaglia polar methods as an acceleration technique for the Box Muller method.

## 1 Box Muller Method

Given $U_1$, $U_2$ random numbers uniformly distributed in the range [0; 1], then the following $X_1$ and $X_2$ have independent Gaussian distributions with variance 1:

$$X_1 = \sqrt{-2\ln U_1}\cos(2\pi U_2) \quad (1)$$
$$X_2 = \sqrt{-2\ln U_1}\sin(2\pi U_2) \quad (2)$$

Proof:

+ From (1) and (2) $\Rightarrow \begin{cases} \ln(U_1) = -\frac{1}{2}(X_1^2+X_2^2) \Rightarrow U_1 = \exp\left[-\frac{x_1^2+x_2^2}{2}\right] = U_{1(x_1,x_2)} \\ \tan(2\pi U_2) = \frac{X_2}{X_1} \Rightarrow U_2 = \frac{1}{2\pi}\tan^{-1}\left(\frac{X_2}{X_1}\right) = U_{2(x_1,x_2)} \end{cases}$

+ Let $P_{(x_1,x_2)}$ be the joint pdf of $x_1$ and $x_2$

$\bar{P}_{(U_1,U_2)}$ be the joint pdf of $U_1$ and $U_2$

By the Jacobian formula (6.51)

$\Rightarrow P_{(x_1,x_2)} = \bar{P}(U_{1(x_1,x_2)}, U_{2(x_1,x_2)})\ |J_{(x_1,x_2)}|$

- Since $U_1$ and $U_2$ are independent uniform random variables in [0; 1]

$\rightsquigarrow \bar{P}_{(U_1,U_2)} = \bar{P}_{(U_1)}\bar{P}_{(U_2)} = 1.1 = 1 \quad (3)$

- $U_{1(x_1,x_2)} = \exp\left(-\frac{x_1^2+x_2^2}{2}\right) \rightsquigarrow \frac{\partial U_1}{\partial x_1} = -x_1 U_1 \quad ; \quad \frac{\partial U_1}{\partial x_2} = -x_2 U_1$

$U_{2(x_1,x_2)} = \frac{1}{2\pi}\tan^{-1}\left(\frac{x_2}{x_1}\right) \rightsquigarrow \frac{\partial U_2}{\partial x_1} = -\frac{\cos^2(2\pi U_2)x_2}{2\pi x_1^2} \quad ; \quad \frac{\partial U_2}{\partial x_2} = +\frac{\cos^2(2\pi U_2)}{2\pi x_1}$

$\rightsquigarrow |J_{(x_1,x_2)}| = \left|\det\begin{bmatrix} \frac{\partial U_1}{\partial x_1} & \frac{\partial U_2}{\partial x_1} \\ \frac{\partial U_1}{\partial x_2} & \frac{\partial U_2}{\partial x_2}\end{bmatrix}\right| = \left|\frac{\partial U_1}{\partial x_1}\frac{\partial U_2}{\partial x_2} - \frac{\partial U_1}{\partial x_2}\frac{\partial U_2}{\partial x_1}\right|$

$\rightsquigarrow |J_{(x_1,x_2)}| = \left|\frac{1}{2\pi}x_1 U_1\frac{\cos^2(2\pi U_2)}{x_1} + \frac{1}{2\pi}x_2 U_1\frac{\cos^2(2\pi U_2)x_2}{x_1^2}\right|$ with $1+\left(\frac{x_2}{x_1}\right)^2 = \frac{1}{\cos^2(2\pi U_2)}$

$= \frac{1}{2\pi}|U_1| = \frac{1}{2\pi}\exp\left[-\frac{x_1^2+x_2^2}{2}\right]$

+ substitute to (3)

$$\Rightarrow P_{(x_1,x_2)} = 1 \cdot \frac{1}{2\pi} \exp\left[-\frac{x_1^2+x_2^2}{2}\right]$$

$$= \underbrace{\left(\frac{1}{\sqrt{2\pi}} e^{-\frac{x_1^2}{2}}\right)}_{\tilde{P}_{(x_1)}} \underbrace{\left(\frac{1}{\sqrt{2\pi}} e^{-\frac{x_2^2}{2}}\right)}_{\tilde{P}_{(x_2)}}$$

$\Rightarrow P_{(x_1,x_2)} = \tilde{P}_{(x_1)} \tilde{P}_{(x_2)}$ where $\tilde{P}_{(x_i)}$ is pdf of random variables $x_i$ with gaussian distribution of mean 0 an variance 1

$\Leftrightarrow$ $X_1$ and $X_2$ are independen random variables with gaussian distribution of mean 0 and variance 1 ☒

## ② An acceleration technique for Box Muller method: Marsaglia polar method

In computational simulations, calculations of trigometric functions can rapidly become calculation load. Thus, I introduce a Box Muller method without using trigometric functions explicitly.

**Implementation procedure for the efficient Box Muller method**

$\boxed{1}$ Set $\mathbb{R} = (u_1, u_2)$, where $u_1, u_2$ are uniform random numbers distributed in $[-1; 1]$

We adopt only $\mathbb{R}$ inside the unit circle centered at $(0;0)$ and dismiss all other data

$\boxed{2}$ $R^2 = u_1^2 + u_2^2$ uniformly distributed in $[0;1]$

Set $\mathcal{U}_1 = R^2$ and calculate $\sqrt{-2\ln(\mathcal{U}_1)}$

$\boxed{3}$ as $\mathbb{R}$'s deflection angle is random

$\rightsquigarrow$ $\cos(2\pi\mathcal{U}_2) = \frac{u_1}{R}$ and $\sin(2\pi\mathcal{U}_2) = \frac{u_2}{R}$ can be obtained without calculating trigometric function explicitly.

I design the beside example code in C++ language, adopting the Marsaglia polar method for Box Muller transformation. A function for generating a random number is also presented as below.

```
double gaussian_rand(void)
{
  static double iset = 0;
  static double gset;
  double fac, rsq, v1, v2;

  if (iset == 0) {
    do {
      v1 = unif_rand(-1, 1);
      v2 = unif_rand(-1, 1);
      rsq = v1*v1 + v2*v2;
    } while (rsq >= 1.0 || rsq == 0.0);
    fac = sqrt(-2.0*log(rsq)/rsq);

    gset = v1*fac;
    iset = 0.50;
    return v2*fac;
  } else {
    iset = 0;
    return gset;
  }
}
```

```
double unif_rand(double left, double right)
{
  return left + (right - left)*rand()/RAND_MAX;
}
```