1 Acceptance Rejection Method

Introduced by Von-Neumann, this method is among the most widely applicable mechanism for generating random samples. This method **generates samples from a target distribution**, say F, by first **generating candidates from a more convenient distribution**, say G and then **rejecting some samples** generated from G and **accepting the rest**. The rejection mechanism is designed so that the accepted samples are indeed distributed according to the target distribution.

Suppose we wish to generate samples from a PDF f. Let g be a PDF and we know the technique to generate samples from it. Also, assume that the following condition holds. For some real constant $c \geq 1$,

$$f(x) \leq cg(x) \quad \text{for all } x \in \mathbb{R}.$$

In the acceptance rejection method, we **generate a sample** X from g and **accept** the sample **with probability** $f(X)/cg(X)$. This can be implemented by sampling U from $U(0,1)$. **If** X is **rejected**, a **new** candidate **is sampled** from g and the **acceptance test applied again**. The process **repeats until the acceptance** test is passed and the accepted value is returned as a sample from f. Thus, we have Algorithm 1. The Theorem 1 provides the justification of the algorithm.

Algorithm 1 Acceptance Rejection Method

```
1: repeat
2:     generate X from distribution g.
3:     generate U from U(0,1).
4: until U ≤ f(X)/cg(X)
5: return X
```

Theorem 1. *Let f and g be two PDFs such that*

$$f(x) \leq cg(x) \quad \textit{for all } x \in \mathbb{R} \textit{ and for some } c \geq 1.$$

Then X generated by Algorithm 1 has PDF f.

Proof: Fix $x \in \mathbb{R}$. Let Y be the random variable having PDF g. Now, the CDF of X is

$$\begin{aligned}
P(X \leq x) &= P\left(X \leq x, U \leq \frac{f(Y)}{cg(Y)}\right) + P\left(X \leq x, U > \frac{f(Y)}{cg(Y)}\right) \\
&= P\left(X \leq x, cg(Y)U \leq f(Y)\right) + P\left(X \leq x | cg(Y)U > f(Y)\right) P(cg(Y)U > f(Y)) \\
&= \int_{-\infty}^{\infty} P\left(X \leq x, cg(Y)U \leq f(Y) | Y = t\right) g(t)dt + P\left(X \leq x\right) P\left(cg(Y)U > f(Y)\right).
\end{aligned}$$

Now, the probability of acceptance is

$$
\begin{aligned}
P\left(U \leq \frac{f(Y)}{cg(Y)}\right) &= \int_{-\infty}^{\infty} P\left(cg(Y)U \leq f(Y) | Y = t\right) g(t)dt \\
&= \int_{-\infty}^{\infty} P\left(U \leq \frac{f(t)}{cg(t)} | Y = t\right) g(t)dt \\
&= \int_{-\infty}^{\infty} P\left(U \leq \frac{f(t)}{cg(t)}\right) g(t)dt \\
&= \int_{-\infty}^{\infty} \frac{f(t)}{cg(t)} \times g(t)dt \\
&= \frac{1}{c}\int_{-\infty}^{\infty} f(t)dt \\
&= \frac{1}{c}.
\end{aligned}
$$

Thus, we have

$$
\frac{1}{c}P\left(X \leq x\right) = \int_{-\infty}^{x} P\left(U \leq \frac{f(t)}{cg(t)}\right) g(t)dt = \frac{1}{c}\int_{-\infty}^{x} f(t)dt \implies P(X \leq x) = \int_{-\infty}^{x} f(t)dt.
$$

Hence, using the definition of PDF, it is clear that the PDF of X is f. $\square$

Example 1 (**Gamma Distribution**). Consider the Gamma distribution with **shape** parameter $\alpha > 0$ and **scale** parameter $\beta > 0$. The PDF of the distribution is

$$
f(x) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x} \quad \text{for } x > 0.
$$

Note that $X \sim Gamma(\alpha, 1)$, then $Y = \frac{X}{\beta} \sim Gamma(\alpha, \beta)$. Therefore, if we **can generate** from $Gamma(\alpha, 1)$, we can **easily generate** from $Gamma(\alpha, \beta)$, and hence, in this example, we will assume that $\beta = 1$. Thus, our target PDF is

$$
f(x) = \frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x} \quad \text{for } x > 0.
$$

Now, we will consider three cases based on the values of α.
<u>Case I: $0 < \alpha < 1$</u>
Note that $f(x)$ approaches to ∞ as $x \to 0$. Also, $f(x) \to 0$ as $x \to \infty$. Here, we try to **bound** f using a **integrable function into two parts**. For $0 < x < 1$,

$$
f(x) \leq \frac{1}{\Gamma(\alpha)} x^{\alpha-1}.
$$

For $x \geq 1$,

$$
f(x) \leq \frac{1}{\Gamma(\alpha)} e^{-x}.
$$

Thus, we take

$$g(x) = \begin{cases} \frac{x^{\alpha-1}}{A} & \text{if } 0 < x < 1 \\ \frac{e^{-x}}{A} & \text{if } x \geq 1, \end{cases}$$

where $A = \frac{1}{\alpha} + \frac{1}{e}$ and $c = \frac{A}{\Gamma(\alpha)}$. Then,

$$f(x) \leq cg(x),$$

where g is a PDF as given above. The corresponding CDF is

$$G(x) = \begin{cases} \frac{x^{\alpha}}{\alpha A} & \text{if } 0 < x < 1 \\ 1 - \frac{e^{-x}}{A} & \text{if } x \geq 1. \end{cases}$$

Now, using inversion method, we can very easily generate random numbers from g. Note that G^{-1} is given by

$$G^{-1}(u) = \begin{cases} (\alpha A u)^{1/\alpha} & \text{if } 0 < u < \frac{1}{\alpha A} \\ -\ln A - \ln(1-u) & \text{if } \frac{1}{\alpha A} \leq u < 1. \end{cases}$$

Thus, we have Algorithm 2 to generate random number from $Gamma(\alpha, 1)$ distribution.

Algorithm 2 Generation from $Gamma(\alpha, 1)$ for $0 < \alpha < 1$

```
repeat
    generate U_1 from U(0, 1)
    Set X = G^{-1}(U_1)
    generate U_2 from U(0, 1)
until cg(X)U_2 ≤ f(X)
return X
```

Case II: $\alpha \geq 1$ and α is an **integer**

Notice that $Y_1 + Y_2 + \ldots + Y_n \sim Gamma(n, 1)$ if $Y_1, Y_2, \ldots, Y_n \overset{i.i.d.}{\sim} Exp(1)$. Thus, we will set $n = \alpha$ and have the Algorithm 3.

Algorithm 3 Generation from $Gamma(\alpha, 1)$ when α is a positive integer

```
n = α
Y = 0
while n ≠ 0 do
    generate U from U(0, 1)
    Set X = − ln(U)
    Y = Y + X
    n = n − 1
end while
return Y
```

Case II: $\alpha > 1$ and α is **not an integer**

Note that we can write $\alpha = \lfloor \alpha \rfloor + \{\alpha\}$, where $\{x\}$ denotes the fractional part of the positive real number x. Also, $X + Y \sim Gamma(\alpha_1 + \alpha_2,\, \beta)$ if $X \sim Gamma(\alpha_1,\, \beta)$, $Y \sim Gamma(\alpha_2,\, \beta)$, and X and Y are independent. Thus, we have the Algorithm 4.

Algorithm 4 Generation from $Gamma(\alpha,\, 1)$ when $\alpha > 1$ and α is not an integer

```
1: generate Y from Gamma({α}, 1) using Algorithm 2
2: generate X from Gamma(⌊α⌋, 1) using Algorithm 3
3: Z = X + Y
4: return Z
```

$\|$

Example 2 (Beta Distribution). The PDF of Beta distribution with parameters $\alpha_1,\, \alpha_2 > 0$ is given by

$$f(x) = \frac{1}{B(\alpha_1, \alpha_2)} x^{\alpha_1 - 1}(1 - x)^{\alpha_2 - 1} \quad 0 < x < 1,$$

where

$$B(\alpha_1, \alpha_2) = \int_0^1 x^{\alpha_1 - 1}(1 - x)^{\alpha_2 - 1} dx = \frac{\Gamma(\alpha_1)\Gamma(\alpha_2)}{\Gamma(\alpha_1 + \alpha_2)}.$$

Varying the parameters α_1 and α_2 results in a variety of shapes, making this a versatile family of distribution. For example, the case, $\alpha_1 = \alpha_2 = 1/2$ is the **arcsine distribution**. If $\alpha_1 = \alpha_2 = 1$, we have **uniform distribution** on $(0,\, 1)$. If $\alpha_1, \alpha_2 \geq 1$ and at least one of the parameter exceeds 1, the beta density is **unimodal** and achieves its maximum at $\dfrac{\alpha_1 - 1}{\alpha_1 + \alpha_2 - 2}$. Let c be the value of the density f at this point, *i.e.*,

$$c = f\left(\frac{\alpha_1 - 1}{\alpha_1 + \alpha_2 - 2}\right).$$

Then $f(x) \leq c$ for all $x \in \mathbb{R}$. For the purpose of **acceptance rejection method**, we may choose g to be the PDF of **uniform distribution** over $(0,\, 1)$, *i.e.*,

$$g(x) = 1 \quad \text{for } 0 < x < 1.$$

Therefore, we can use the Algorithm 5 to generate random numbers from a Beta distribution with parameters $\alpha_1 \geq 1$ and $\alpha_2 \geq 1$, where $\alpha_1 + \alpha_2 > 2$.

Algorithm 5 Generation from $Beta(\alpha_1,\, \alpha_2)$ with $\alpha_1 \geq 1$, $\alpha_2 \geq 1$ and $\alpha_1 + \alpha_2 > 2$.

```
1: repeat
2:     generate U1 and U2 from U(0,1)
3: until cU2 ≤ f(U1)
4: return U1
```

Note that this choice of g will **not work** if $\alpha_1 < 1$ or $\alpha_2 < 1$. In this case the PDF of Beta distribution is **unbounded** either at $x = 0$ (for $\alpha_1 < 1$) or at $x = 1$ (for $\alpha_2 < 1$). $\|$

Example 3 (Beta Distribution). Note that if $X \sim Gamma(\alpha_1, \beta)$, $Y \sim Gamma(\alpha_2, \beta)$ and X and Y are independent, then $\dfrac{X}{X+Y} \sim Beta(\alpha_1, \alpha_2)$ (Why?). Thus, Algorithm 6 can be used to generate from $Beta(\alpha_1, \alpha_2)$ distribution.

Algorithm 6 Generation from $Beta(\alpha_1, \alpha_2)$ distribution

1: generate X from $Gamma(\alpha_1, \beta)$
2: generate Y from $Gamma(\alpha_2, \beta)$
3: $Z = \dfrac{X}{X+Y}$
4: **return** Z

Unlike the previous example, there is no extra restriction on the parameters α_1 and α_2 here. Also, we can take any value for $\beta > 0$ to implement this algorithm. ||