1 Overview

A trivial extension of the ziggurat algorithm which is suitable for continuous symmetric unimodal distributions will be presented. This is a summary of Marsaglia and Tsang [undated] and Doornik [1997]. Let θ denote the mode of the distribution, and $T(x)$ denote the tail function:

$$T(x) = \int_x^\infty f(t)\,dt.$$

Note that normalizing the density function (so that $T(-\infty) = 1$) is not necessary.

1.1 Monotone decreasing density

First consider the interval to the right of the mode, where the density function, f, is decreasing. The goal is to cover the density function by $n \geq 2$ boxes of the same volume and the following form:

$$\begin{aligned} B_0 &= \{(x, y) \,:\, x \geq \theta,\, 0 \leq y \leq \min(f(x_1), f(x))\}, \\ B_k &= [\theta, x_k] \times [f(x_k), f(x_{k+1})], \quad 1 \leq k \leq n-2, \\ B_{n-1} &= [\theta, x_k] \times [f(x_k), f(\theta)]. \end{aligned}$$

Note that the bottom box is the only non-rectangular.

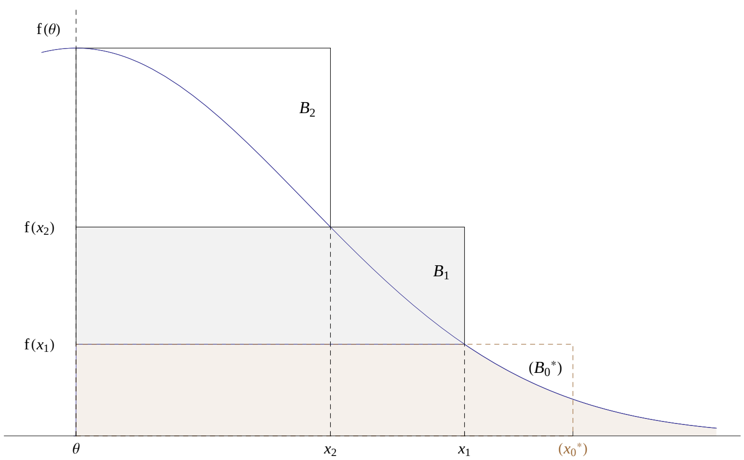

Figure 1: Illustration of one-sided ziggurat algorithm when $n = 3$.

Since the area of each box is to be the same as the area of B_0, one gets a (non-linear) system of n equations

$$\begin{cases} V &= (x_1 - \theta) f(x_1) + T(x_1) \\ V &= (x_1 - \theta)\left(f(x_2) - f(x_1)\right) \\ V &= \quad \cdots \\ V &= (x_{n-2} - \theta)\left(f(x_{n-1}) - f(x_{n-2})\right) \\ V &= (x_{n-1} - \theta)\left(f(\theta) - f(x_{n-1})\right) \end{cases} \tag{1}$$

with n unknowns. Put otherwise, $V = (x_1 - \theta) f(x_1) + T(x_1) = (x_{n-1} - \theta)\left(f(\theta) - f(x_{n-1})\right)$, and

$$x_{k+1} = f^{-1}\left(f(x_k) + \frac{V}{x_k - \theta}\right) \quad \text{for } 1 \leq k \leq n-2.$$

Proposition 1. *Due to the assumption of monotonicity, system (1) has a unique solution.*

1.2 Symmetric unimodal density

When the density is unimodal and symmetric about its mode θ, first solve (1) for the right tail to get $\{B_{n-1}, \cdots, B_0\}$ and the corresponding partition $\{\theta, x_{n-1}, \ldots, x_1\}$. One option would be to sample from the right tail, and then use an unused bit from one of the generated random numbers to determine whether to replace the generated variable z with $\theta - z$.

Alternatively, if doubling the size of lookup tables is not an issue, one could reflect the boxes about θ to get the resulting partition for $2n$ boxes $\{B_{2n-1}, \cdots, B_n, B_{n-1}, \cdots, B_0\}$:

$$\{2\theta - x_1, 2\theta - x_2, \ldots, 2\theta - x_{n-1}, \theta, x_{n-1}, \ldots, x_2, x_1\};$$

write $x_{n+i} = 2\theta - x_{n-1-i}$ for $0 \le i \le n-2$ so that each B_j (apart from the tail boxes) has x_j for either left or right abscissa. We will adopt this extended lookup table approach.

2 The ziggurat algorithm

The algorithm of Marsaglia and Tsang [undated] and Doornik [1997] is summarized below (trivially extended to the symmetric case with $2n$ boxes).

First, introduce an auxiliary *rectangular* box (B_0^*) corresponding to B_0, so that it has the same height and volume, and denote

$$x_0 \triangleq \theta + V/f(x_1), \qquad x_{2n-1} \triangleq \theta - V/f(x_1) = 2\theta - x_0.$$

2.1 Continuous generators

Pre-compute the following properties for the $2n$ boxes:

(i) Box width w (signed: positive for right tail and negative for left), height h, and bottom ordinate b:

$$w_i = x_i - \theta, \quad h_i = \big|f(x_i) - f(x_{i+1})\big|, \quad b_i = f(x_i)$$

for $0 \le i \le 2n-1$. In the case of tail boxes, w, h and b lose their intuitive meaning; but the former is mathematically convenient, whereas the latter two are plainly not used.

(ii) Probabilities q of "simple coverage":

$$\begin{aligned} q_i &= w_{i+1}/w_i \quad \text{for} \quad 0 \le i \le n-1, \\ q_i &= w_i/w_{i+1} \quad \text{for} \quad n \le i \le 2n-1. \end{aligned}$$

We are now ready to present the main algorithm.

1. Generate the zero-based box index $I \sim \text{Uniform}(\{0, 1, \cdots, 2n-1\})$.
2. Generate $U \sim \text{Uniform}([0, 1))$, and let $z = \theta + U \cdot w_I$.
3. If $U < q_I$ accept z.
4. (a) If $I = 0$, accept a v from the right tail (in the normal case see, e.g., Marsaglia [1964]); for the left tail, i.e., when $I = 2n-1$, accept $2\theta - v$.
   (b) If $1 \le I \le 2n-2$, generate $V \sim \text{Uniform}([0, 1))$, and if $b_I + V \cdot h_I < f(z)$ accept z.
5. Otherwise return to step 1.

This is a sample-rejection type algorithm; note that step 4a is essential because (B_0^*) does *not* cover the tail. Note also, that $w_j = -w_{2n-1-j}$ and $(\cdot)_j = (\cdot)_{2n-1-j}$ for $(\cdot) \in \{h, b, q\}$, $0 \le j \le n-1$.

2.2 Discrete generators

In practice one usually has to deal with m-bit integer generators. The main algorithm can be adjusted to take advantage of integer arithmetic in the following way:

1. Generate the zero-based box index $I \sim \text{Uniform}(\{0, 1, \cdots, 2n-1\})$.
2. Generate $U \sim \text{Uniform}(\{0, \cdots, 2^m - 1\})$, and let $z = \theta + U \cdot (2^{-m}\, w_I)$.
3. If $U < \lfloor 2^m\, q_I \rfloor$ accept z.
4. (a) If $I = 0$, accept a v from the right tail (in the normal case see, e.g., Marsaglia [1964]); for the left tail, i.e., when $I = 2n-1$, accept $2\theta - v$.
   (b) If $1 \le I \le 2n-2$, generate $V \sim \text{Uniform}(\{0, \cdots, 2^m - 1\})$, and if $b_I + V \cdot (2^{-m}\, h_I) < f(z)$ accept z.
5. Otherwise return to step 1.

Thus, instead of storing w and h we can store $2^{-m}\, w$ and $2^{-m}\, h$, and instead of q—integer-valued $\lfloor 2^m\, q \rfloor$; b stays intact. To summarize, we store:

(i) Down-scaled box width $\tilde{w}$, down-scaled height $\tilde{h}$, and bottom ordinate b:

$$\tilde{w}_i = 2^{-m}\,(x_i - \theta), \quad \tilde{h}_i = 2^{-m}\,\big|f(x_i) - f(x_{i+1})\big|, \quad b_i = f(x_i)$$

for $0 \le i \le 2n-1$.

(ii) Up-scaled probabilities $\hat{q}$ of "simple coverage":

$$\begin{aligned} \hat{q}_i &= \lfloor 2^m\ \ w_{i+1}/w_i \rfloor \quad \text{for} \quad 0 \le i \le n-1, \\ \hat{q}_i &= \lfloor 2^m\ \ w_i/w_{i+1} \rfloor \quad \text{for} \quad n \le i \le 2n-1. \end{aligned}$$

Note that $\lceil \log_2(2n) \rceil$ bits are used for I; thus the remaining $2^m - \lceil \log_2(2n) \rceil$ bits may be used to enhance U—or may they?

References

Jurgen A. Doornik. An improved ziggurat method to generate normal random samples. *Communications in Statistics – Theory and Methods*, 26(5):1253–1268, December 1997.

George Marsaglia. Generating a variable from the tail of the normal distribution. *Technometrics*, 6(1): 101–102, February 1964. ISSN 00401706.

George Marsaglia and Wai Wan Tsang. The ziggurat method for generating random variables. *Journal of Statistical Software*, 5(i08), undated.