MTH 511a - 2020: Lecture 8

Instructor: Dootika Vats

1 Generating continuous random variables

1.1 The Box-Mueller transformation: for $N(0,1)$.

A classical method to generate samples from $N(0,1)$ is the Box-Mueller transformation method. Here, we will draw random variables (R^2, Θ) from a certain distribution and then use a transformation so that $h(R^2, \Theta) \sim N(0,1)$. First, we will need some theory for this.

Let X and Y be independent and identically distributed $N(0,1)$. The joint density of (X, Y) is

$$f(x,y) = \frac{1}{2\pi} e^{-x^2/2} e^{-y^2/2}\,.$$

Let (R^2, Θ) denote the polar coordinates of (X, Y) so that $X = R\cos\Theta$ and $Y = R\sin\Theta$. Then,

$$R^2 = X^2 + Y^2 \qquad \tan\Theta = \frac{Y}{X}\,.$$

For the transformation, let $d = x^2 + y^2$ and $\theta = \tan^{-1}(y/x)$. We know that the density for (d, θ) can be found by

$$f(d,\theta) = |J| f(x,y) \qquad \text{where } J = \begin{vmatrix} \frac{\partial x}{\partial d} & \frac{\partial y}{\partial d} \\ \frac{\partial x}{\partial \theta} & \frac{\partial y}{\partial \theta} \end{vmatrix}$$

Solving for J,

$$J = \begin{vmatrix} \frac{\partial \sqrt{d}\cos\theta}{\partial d} & \frac{\partial \sqrt{d}\sin\theta}{\partial d} \\ \frac{\partial \sqrt{d}\cos\theta}{\partial \theta} & \frac{\partial \sqrt{d}\sin\theta}{\partial \theta} \end{vmatrix} = \frac{1}{2}\,.$$

Since $d = x^2 + y^2$, the joint density of (R^2, Θ) is $f(d, \theta)$ with

$$\begin{aligned} f(d,\theta) &= \frac{1}{2}\frac{1}{2\pi}e^{-d/2} \quad 0 < d < \infty, 0 < \theta < 2\pi \\ &= \underbrace{\frac{1}{2\pi}}_{U(0,2\pi)} I(0 < \theta < 2\pi) \; \underbrace{\frac{1}{2}e^{-d/2}}_{\text{Exp}(2)} I(0 < d < \infty) \end{aligned}$$

This is a separable density, so R^2 and Θ are independent, and $\Theta \sim U[0, 2\pi]$ and $R^2 \sim \text{Exp}(2)$.

To generate from Exp(2), we can use an inverse transform method. If $U \sim U(0,1)$, then by the inverse transform method, $-2\log U \sim \text{Exp}(2)$ (verify for yourself). To generate from $U(0, 2\pi)$, we know if $U \sim U(0,1)$, then $2\pi U \sim U(0, 2\pi)$. The Box-Mueller algorithm then is given in Algorithm 1 which produces X and Y from $N(0,1)$ indendently.

Algorithm 1 Box-Mueller algorithm for $N(0,1)$

1: Generate U_1 and U_2 from $U[0,1]$ independently
2: Set $R^2 = -2\log U_1$ and $\Theta = 2\pi U_2$
3: Set $X = R\cos(\Theta) = \sqrt{-2\log U_1}\cos(2\pi U_2)$
4: and $Y = R\sin(\Theta) = \sqrt{-2\log U_1}\sin(2\pi U_2)$.

1.2 Ratio-of-Uniforms

Ratio-of-uniforms is a powerful, however not so popular method to generate samples for a continuous random variables.

Theorem 1. *Let $f(x)$ be a target density with distribution function F. Define set*

$$C = \left\{(u,v) : 0 \leq u \leq \sqrt{f\left(\frac{v}{u}\right)}\right\}.$$

Let (U, V) be uniformly distributed over the set C, then $V/U \sim F$.

Proof. We will show that the density of $Z = V/U$ is $f(z)$. Note that by definition,, the joint density of (U, V) is

$$f_{(U,V)}(u,v) = \frac{1}{\int\int_C du\, dv} I((u,v) \in C).$$

Consider transformation $(U, V) \mapsto (U, Z)$ with $Z = V/U$. Then $U = U$ and $V = UZ$. The Jacobian for this transformation is U. So

$$f_{(U,Z)}(u,z) = \frac{u}{\int\int_C du\, dv} I\left\{0 \leq u \leq f^{1/2}(z)\right\}.$$

Finding the marginal distribution of $Z = V/U$, we integrate out U,

$$\begin{aligned} f_Z(z) &= \int \frac{u}{\int\int_C du\, dv} I\left\{0 \le u \le f^{1/2}(z)\right\} du \\ &= \frac{1}{\int\int_C du\, dv} \int_0^{f^{1/2}(z)} u du \\ &= \frac{f(z)}{2\int\int_C du\, dv}\,. \end{aligned}$$

Since $f_Z(z)$ and $f(z)$ are both densities, this implies that

$$1 = \int f_Z(z)dz = \frac{\int f(z)dz}{2\int\int_C du\, dv} = \frac{1}{2\int\int_C du\, dv} \Rightarrow \int\int_C du\, dv = \frac{1}{2}$$

This implies $f_Z(z) = f(z)$. □

Thus, V/U has the desired distribution.

So if we can draw $(U, V) \sim \text{Unif}(C)$, then $V/U \sim F$. But C looks quite complicated, so how do we uniformly draw from C?

Think back to the AR technique used to draw uniformly from a circle! If we enclose C in a rectangle, we can use accept-reject! Find $U[0, a] \times [b, c]$ such that

$$0 \le u \le a \quad b \le v \le c\,.$$

First, note that if $\sup_x f^{1/2}(x)$ exists, then

$$0 \le u \le f^{1/2}\left(\frac{v}{u}\right) \le \sup_x f^{1/2}(x) := a\,.$$

Note now that if $x = v/u \Rightarrow v/x = u \le f^{1/2}(x)$. This implies that

$$\frac{v}{x} \le f^{1/2}(x)\,.$$

For:

$$\begin{aligned} x \le 0: &\quad v \ge x f^{1/2}(x) \ge \inf_{x\le 0} x f^{1/2}(x) := b \\ x \ge 0: &\quad v \le x f^{1/2}(x) \le \sup_{x\ge 0} x f^{1/2}(x) := c\,. \end{aligned}$$

Note that if $\sqrt{f(x)}$ or $x^2 f(x)$ are unbounded, then C is unbounded, and the method cannot work.

Algorithm 2 Ratio-of-Uniforms

1: Generate $(U, V) \sim U[0, a] \times U[b, c]$

2: If $U \leq \sqrt{f(V/U)}$, then set $X = V/U$.

3: Else go to 1.

Steps 1 and 2 in Algorithm 2 are implementing an Accept-Reject to sample uniformly from C. To understand how effective this algorithm will be, we can calculate the probability of acceptance for the AR. First, note that

$$\sup_{(u,v)\in C} \frac{f(u,v)}{g(u,v)} = \sup_{(u,v)\in C} \frac{\frac{I((u,v)\in C)}{\int_C dudv}}{\frac{I((u,v)\in(0,a)\times(b,c))}{a*(c-b)}} = 2a(c-b)$$

Thus,

$$\Pr\left(\text{Accepting for AR in RoU}\right) = \frac{1}{2a(c-b)}\,.$$

So if a is large and/or $(c-b)$ is large, the probability is small.

Example 1 (Exponential(1)).

$$f(x) = e^{-x} \quad x \geq 0$$

Here,

$$C = \left\{(u, v) : 0 \leq u \leq e^{-v/2u}\right\}.$$

Recall that the set $a = \sup_x e^{-x/2} = 1$, since that is a decreasing function. Additionally,

$$b = \inf_{x\leq 0} xe^{-x/2} = 0 \qquad \text{since suppose is } x \geq 0$$

and

$$c = \sup_{x\geq 0} xe^{-x/2} \Rightarrow c = 2e^{-1} \qquad \text{show for yourself}\;.$$

So we sample from $U[0, 1] \times [0, 2/e]$ and then implement accept-reject.

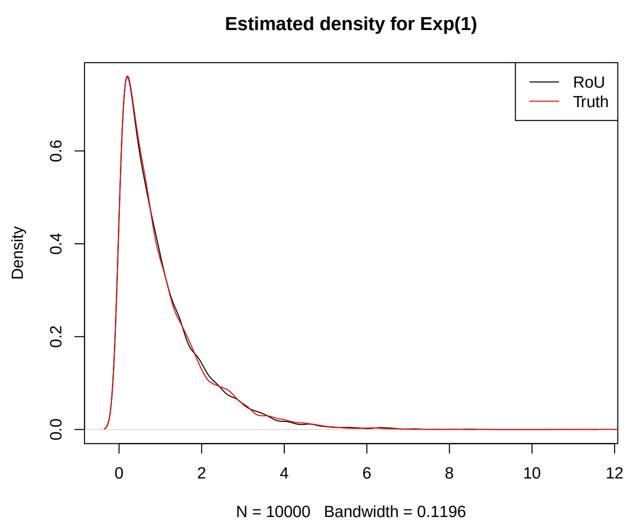
```
#####################################################
### Ratio of Uniforms for Exp(1)
#####################################################
set.seed(1)
# function to sample from the rectangle
drawFromRect <- function(a, b, c)
{
  u <- runif(1, min = 0, max = a)
  v <- runif(1, min = b, max = c)
  return(c(u,v))
}
# sqrt f function
```

```r
sqrt.f <- function(x) exp(-x/2)

# Starting the process for Exp(1)
a <- 1
b <- 0
c <- 2*exp(-1)
prob.of.acceptance <- 1/(2*a*(c-b)) # true prob. of acceptance for AR

N <- 1e4 # number of samples
samp <- numeric(length = N)
i <- 1
counter <- 0 # to check acceptance
while(i <= N)
{
  counter <- counter + 1
  prop <- drawFromRect(a = a, b = b, c = c)
  vbyu <- prop[2]/prop[1]
  if( prop[1] < sqrt.f(vbyu))
  {
    samp[i] <- vbyu
    i <- i + 1
  }
}
```

```r
plot(density(samp), main = "Estimated density for Exp(1)")
lines(density(rexp(1e4, 1)), col = "red")
legend("topright", col = c("black", "red"), lty = 1, legend = c("RoU",
   "Truth"))
```



```
(prob.of.acceptance)
# [1] 0.6795705

N/counter # very close
# [1] 0.6796248
```

Example 2 (Normal(0,1)). The target density is:

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} .$$

The set C is

$$C = \left\{ (u, v) : 0 \leq u \leq \left(\frac{1}{2\pi} \right)^{1/4} e^{-v^2/4u^2} \right\}$$

To find the bounds:

$$a = \sup_{x \in \mathbb{R}} (2\pi)^{-1/4} e^{-x^2/4} = (2\pi)^{-1/4}$$

$$b = \inf_{x \leq 0} (2\pi)^{-1/4} x e^{-x^2/4} \overset{\text{at } x = -\sqrt{2}}{=} -(2\pi)^{-1/4} \sqrt{2} e^{-\sqrt{2}^2/4} = -(2\pi)^{-1/4} \sqrt{2e^{-1}}$$

$$c = -b$$

All that needs to be done now is to implement Algorithm 2 with these values of a, b, c etc.

1.3 Questions to think about

1. Can you do a similar polar coordinate construction to sample from a Cauchy distribution?
2. Construct a similar RoU sampler for Cauchy distribution.
3. Why does RoU fail when C is unbounded?