Statistical Computing HW 3

Mario Ibanez

January 23, 2016

Problem 5.1)

Question

Give a method for generating a random variable having density function

$$f(x) = e^x/(e-1), \quad 0 \leq x \leq 1$$

Answer

Derivation

$$\begin{aligned} F(x) &= \int_0^x \frac{1}{e-1} e^t dt \\ &= \frac{1}{e-1} \left[e^t\right]_0^x \\ &= \frac{1}{e-1}(e^x - 1) \\ &= \frac{e^x - 1}{e-1} \end{aligned}$$

Then set $U = F(x)$ where U is a standard normal uniform random variable, and solve for x:

$$\begin{aligned} u &= \frac{e^x - 1}{e-1} \\ (e-1)u + 1 &= e^x \\ ln((e-1)u + 1) &= x \end{aligned}$$

and so we have:

$$X = ln((e-1)U + 1)$$

Algorithm

1) Generate N values from $U(0,1)$

2) Evaluate $x = ln((e-1)u + 1)$ for each of the N u values

Program

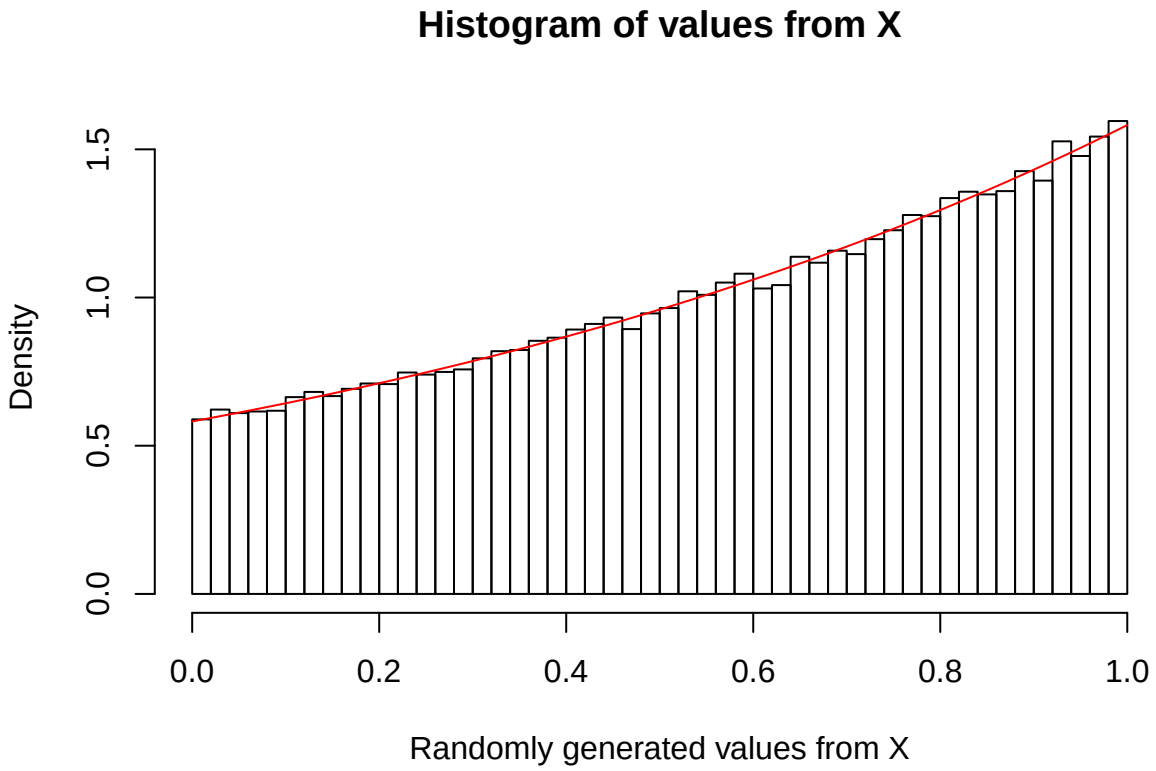
The R function *runif()* creates a random vector which can then be used to do element-wise operations with. Loops in R should be avoided when possible:

```
# 10,000 randomly generated values
N <- 10^5
random_x <- log((exp(1) - 1) * runif(N) + 1)
```

Visualization

Below is a plot to confirm our results visually:

```
# Histogram of random numbers along with curve of f(x) function
hist(random_x, freq = FALSE, breaks = seq.int(0,50)/50,
     xlab = "Randomly generated values from X",
     main = "Histogram of values from X" )
curve(exp(x)/(exp(1)-1), 0, 1, add = TRUE, col = "red")
```



Problem 5.3)

Question

Use the inverse transform method to generate a random variable having distribution function

$$F(x) = \frac{x^2 + x}{2}, \quad 0 \leq x \leq 1$$

Answer

Derivation

We have:

$$\begin{aligned} u &= \frac{x^2 + x}{2} \\ 2u &= x^2 + x \\ 0 &= x^2 + x - 2u \\ x &= \frac{-1 \pm \sqrt{1 - 4(1)(-2u)}}{2} \\ x &= \frac{-1 \pm \sqrt{1 + 8u)}}{2} \end{aligned}$$

No we take only the addition case because we know that x cannot be negative:

$$x = \frac{\sqrt{1 + 8u)} - 1}{2}$$

and so:

$$X = \frac{\sqrt{1 + 8U)} - 1}{2}$$

Algorithm

1) Generate N values from $U(0, 1)$

2) Evaluate $x = \frac{\sqrt{1+8u)}-1}{2}$ for each of the N u values

Program

The R function *runif()* creates a random vector which can then be used to do element-wise operations with. Loops in R should be avoided when possible:

```
# 10,000 randomly generated values
N <- 10^5
random_x <- (1/2)*(sqrt(1 + 8 * runif(N)) - 1)
```

Visualization

Below is a plot to confirm our results visually, with $f(x) = x + 1/2$ (note that the distribution is a trapezoid):

```
# Histogram of random numbers along with curve of f(x) function
hist(random_x, freq = FALSE, breaks = seq.int(0,50)/50,
     xlab = "Randomly generated values from X",
     main = "Histogram of values from X" )
abline(0.5, 1, col = "red")
```