Stat 361 - Recitation 3

Generating Random Variates

Orçun Oltulu

20 / 02 / 2020

Inverse Transform Method:

Consider a continuous random variable having distribution function F. A general method for generating such a random variable is called the inverse transformation method which is based on the following proposition.

Proposition: Let U be a Uniform (0,1) random variable. For any continuous distribution function F, the random variable X defined by

$$X = F^{-1}(U)$$

has distribution F. [$F^{-1}(u)$ is defined to be that value of x such that $F(x) = u$.]

Proof: Let F_x denote the distribution function of $X = F^{-1}(U)$. Then,

$$F_X = P(X \leq x) = P(F^{-1}(U) \leq x)$$

(Comes from Probability Integral Transformation)

Now, since F is a distribution function it follows that $F(x)$ is a monotone increasing function of x and so the inequality $a \leq b$ is equivalent to the inequality $F(a) \leq F(b)$. Hence, from previous equation we see that

$$F_X(x) = P[F(F^{-1}(U)) \leq F(x)]$$
$$F_X(x) = P[U \leq F(x)], \quad \text{Since } [F(F^{-1}(U)) = U]$$
$$F_X(x) = F(x), \quad \text{Since U is Uniform(0,1)}$$

The above proposition thus shows that we can generate a random variable X from the continuous distribution function F by generating a random number U and then setting $X = F^{-1}(U)$.

Theorem (Probability Integral Transformation):

The inverse transform method of generating random variables applies the probability integral transformation. Define the inverse transformation

$$F_X^{-1} = \inf(x : F_X(x) = u), \quad 0 < u < 1$$

If $U \sim$ Uniform(0,1), then for all $x \in R$

$$P[F_X^{-1}(U) \leq x] = P[\inf(t : F_X(t) = U) \leq x] = P(U \leq F_X(x))$$
$$\text{then,} \quad P[F_X^{-1}(U) \leq x] = F_U(F_X(x)) = F_X(x)$$

and therefore $F_X^{-1}(U)$ has the same distribution as X. Thus, to generate a random observation X, first generate a Uniform(0,1) variate u and deliver the inverse value $F_X^{-1}(U)$. The method is easy to apply, provided that $F_X^{-1}(U)$ is easy to compute. The method can be applied for generating continuous or discrete random variables.

This method can be summarized as follows:

- Derive the expression for the inverse distribution function $F^{-1}(U)$.
- Generate a uniform random number U. (by using runif function)
- Obtain the desired X from $X = F^{-1}(U)$.

Continuous Case:

Question 1:

Generate random variables from exponential distribution $(\lambda = 2)$ by using inverse transform method.

PDF of the exponential distribution with parameter λ given by

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

The cumulative distribution function for an exponential random variable is

$$F(X) = 1 - e^{-\lambda x} \quad x > 0$$

Now apply the procedure;

$$U = F(X) = 1 - e^{-\lambda x}$$

$$\begin{aligned} e^{-\lambda x} &= 1 - U \\ -\lambda x &= log(1 - U) \\ x &= -\frac{1}{\lambda} log(1 - U) \end{aligned}$$

Since we will be generating U from a Uniform(0,1), 1-U and U have the same distribution. Therefore, we can generate exponential random variables with parameter λ using the transformation

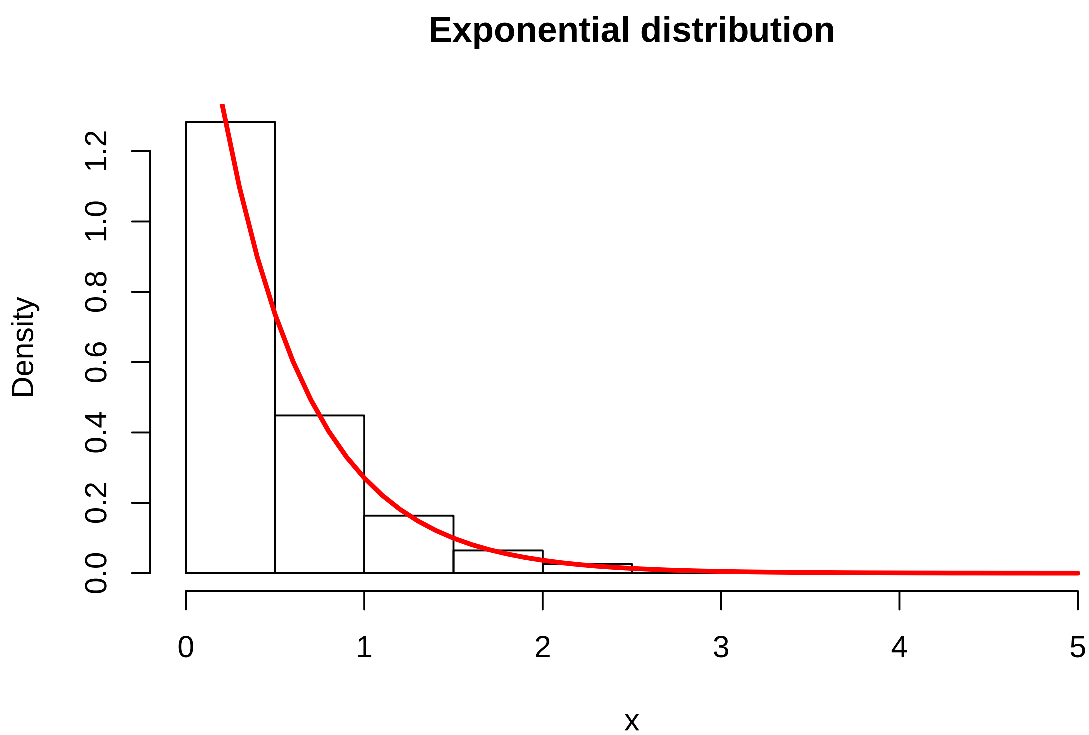
$$x = -\frac{1}{\lambda} log(U)$$

```r
lambda <- 2

set.seed(361)
u <- runif(n = 10000)

x <- -log(u) / lambda
```

Let's check the distribution of the generated x

```r
hist(x, prob = TRUE,
     main = 'Exponential distribution')
# Add theoritical density line
```

```r
y = seq(0,5,0.1)
lines(y,lambda*exp(-lambda*y),col="red",lwd = 2.5)
```



Now, calculate the sample mean to compare with the expectation of exponential distribution.

```r
mean(x)
```

```
## [1] 0.4990323
```

Expectation of exponential distribution is $\frac{1}{\lambda} = \frac{1}{2}$

Question 2:

Use the inverse transformation method to simulate a random sample from Pareto(9,2) distribution.

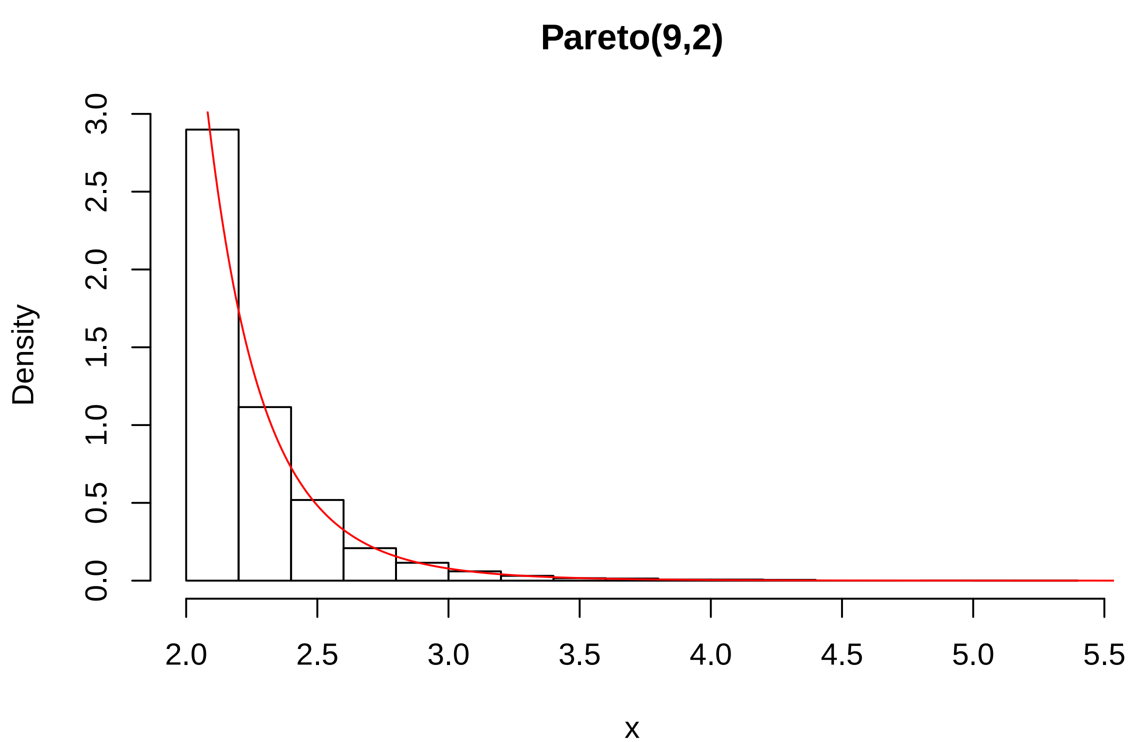
CDF of Pareto Distribution is

$$F(X) = 1 - (\frac{b}{x})^a, \quad x \geq b > 0, a > 0$$

To solve x,

$$u = F(X) = 1 - (\frac{b}{x})^a$$
$$x = b * (1-u)^{-\frac{1}{a}}$$

```
# Set up parameters
n = 10000
a = 9
b = 2

#Generate random variables
u = runif(n)
x = b*(1-u)^(-1/a)
hist(x, prob = TRUE,
     main ="Pareto(9,2)")
y <- seq(1, 40, 0.01)
lines(y, a*b^a/y^(a+1),col="red")
```




```
summary(x)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.000   2.066   2.156   2.253   2.337   5.398
```

Discrete Case:

The inverse transform method can also be applied to discrete distributions. If X is a discrete random variable that has a probability mass function given by

$$P(X_i = x_i) = p_i, \quad x_0 < x_1 < ..., \sum_{i=1} p_i = 1$$

the inverse transformation is $X = x_i$, if $F(x_{i-1}) < U < F(x_i)$

The following algorithm can be used;

- Define a pmf for $x_i, \quad i = 1, ..., k$
- Generate a uniform random number U.
- If $U \leq p_0$ deliver $X = x_0$
- else if $U \leq p_0 + p_1$ deliver $X = x_1$
- else if $U \leq p_0 + p_1 + p_2$ deliver $X = x_2 \ldots$

Question 3:

Let X be a discrete random variable that has probability mass function given by

$$P(X = 0) = 0.3, \;\; P(X = 1) = 0.2, \;\; P(X = 2) = 0.5$$

Generate random variable for X by using discrete inverse transform method for 100 variates from the desired probability mass function.

Cumulative distribution function of X is

$$F(x) = \begin{cases} 0 & x < 0 \\ 0.3 & 0 \leq x < 1 \\ 0.5 & 1 \leq x < 2 \\ 1 & 2 \leq x \end{cases}$$

Now, generate random variable for X according to the following scheme by using discrete inverse transform method for 1000 variates from the desired probability mass function.

$$X = \begin{cases} 0 & U \leq 0.3 \\ 1 & 0.3 < U \leq 0.5 \\ 2 & 0.5 < U \leq 1 \end{cases}$$

```r
n <- 1000
# Set up storage space for the generated variables.
x <- numeric(n)
# These are the probability masses.
prob <- c(0.3, 0.2, 0.5)

# Generate n rv's from the desired distribution.

set.seed(361)
for (i in 1:n){

  u <- runif(1)

  if(u <= prob[1]){
    x[i] <-  0
  }else if(u <= sum(prob[1:2])){
    x[i] <- 1
  }else{
    x[i] <- 2
  }
}

# In order to confirm, we look at the relative frequency of each x.
# For this, find the proportion of each number.
x0 <- length(which(x==0))/n
x1 <- length(which(x==1))/n
x2 <- length(which(x==2))/n

est.prob <- c(x0,x1,x2)
est.prob
```

```
## [1] 0.318 0.197 0.485
```

Question 4:

Consider a balanced coin tossing example where $P_H = 0.6$. You are tossing that balanced coin 4 times and let the random variable Y is the number of Heads occured.

A) Find the pmf of this distribution of Y and from that distribution, generate a random sample of size 1000 by using inverse transformation method.

The distribution of this random variable is simply a Binomial(n = 4, p = 0.6). PMF and CDF of this distribution are shown as

y	0	1	2	3	4
p	0.0256	0.1536	0.3456	0.3456	0.1296

$$F(y) = \begin{cases} 0 & y < 0 \\ 0.0256 & 0 \le y < 1 \\ 0.1792 & 1 \le y < 2 \\ 0.5248 & 2 \le y < 3 \\ 0.8704 & 3 \le y < 4 \\ 1 & 4 \le y \end{cases}$$

$$Y = \begin{cases} 0 & U \le 0.0256 \\ 1 & 0.0256 < U \le 0.1792 \\ 2 & 0.1792 < U \le 0.5248 \\ 3 & 0.5248 < U \le 0.8704 \\ 4 & 0.8704 < U \le 1 \end{cases}$$

```r
n <- 1000
y <- numeric(n)
prob <- c(0.0256, 0.1536, 0.3456, 0.3456, 0.1296)
i <- 0

set.seed(361)
while( i <= n ){
  u <- runif(1)

  y[i] <- ifelse(u <= prob[1],0,
                 ifelse(u <= sum(prob[1:2]),1,
                 ifelse(u <= sum(prob[1:3]),2,
                 ifelse(u <= sum(prob[1:4]),3,4))))
  i <- i + 1
}
```

```r
#Estimated Probabilities.
table(y)/n
```

```
## y
##     0     1     2     3     4
## 0.033 0.154 0.351 0.335 0.127
```

B) Compute the mean and variance of the generated sample and compare the results with theoretical expectatian and variance.

```r
#domain of y
d <- seq(0,4)

theoretical.mean = sum(prob*d)
est.mean = mean(y)

theoretical.var = sum(prob*d^2)-(sum(prob*d))^2
est.var = var(y)

vec <- round(c(theoretical.mean,est.mean,theoretical.var,est.var),4)
out <- matrix(vec,nrow=2)
rownames(out) <- c("Theoretical", "Estimated")
colnames(out) <- c("Mean","Variance")
out
```

```
##              Mean Variance
## Theoretical 2.400   0.9600
## Estimated   2.369   0.9938
```