Probability, Computation and Simulation Homework 2

Brian Cervantes Alvarez
October 7, 2024

Problem 4

Give a method for generating a random variable having the distribution function

$$F(x) = 1 - \exp(-\alpha x^{\beta}), \quad 0 < x < \infty$$

This describes a Weibull random variable.

Pseudocode

1. Generate a random number u from a $Uniform(0, 1)$.
2. Compute the inverse of the Weibull CDF: $X = \left(-\frac{\log(1-u)}{\alpha}\right)^{\frac{1}{\beta}}$.
3. Output the value of X.

Putting it in practice

```r
set.seed(202425)
alpha <- 2
beta <- 3
u <- runif(1)
X <- (-log(1-u)/alpha)^(1/beta)
X
```

```
[1] 0.9606969
```

Problem 6

Let X be an exponential random variable with mean 1. Simulate a random variable whose distribution is the conditional distribution of X given $X < 0.05$. Its density function is:

$$f(x) = \frac{e^{-x}}{1 - e^{-0.05}}, \quad 0 < x < 0.05$$

Pseudocode

1. Generate an exponential random variable X with rate 1.
2. If $X < 0.05$, accept it. Otherwise, repeat.
3. Repeat this process 1000 times to estimate the expected value.

Putting it in practice

```r
set.seed(202425)
N <- 1000
accepted <- replicate(N, {
  accept <- FALSE
  while(!accept) {
    X <- rexp(1)
    if(X < 0.05) accept <- TRUE
  }
  X
})
mean(accepted)
```

```
[1] 0.02463227
```

Problem 10

A casualty insurance company has 1000 policyholders, each presenting a claim with probability 0.05. The claims are exponentially distributed with a mean of $800. Estimate the probability that the total claims exceed $50,000 using simulation.

Pseudocode

1. Set N as the number of simulations.
2. For each simulation:
   - Generate 1000 Bernoulli random variables to model claims.
   - For each nonzero claim, generate an exponential random variable with mean 800.
   - Sum the claim amounts.
3. Estimate the probability by counting the proportion of times the sum exceeds $50,000$.

Putting it in practice

```r
set.seed(202425)
N <- 10000
mean_claim <- 800
prob <- 0.05
threshold <- 50000

exceed_prob <- replicate(N, {
  claims <- rbinom(1000, 1, prob)
  total_claim <- sum(rexp(sum(claims), mean_claim))
  total_claim > threshold
})
mean(exceed_prob)
```

```
[1] 0
```

Problem 19

Show how to generate a random variable with the distribution function

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

Part A: Inverse Transform Method

Pseudocode

1. Solve $F(x) = u$ to obtain x.
2. Rearrange to get $x = \sqrt{2u - 1} + 1$ for u from $Uniform(0, 1)$.
3. Output x.

Putting it in practice

```r
set.seed(202425)
u <- runif(1)
x <- (sqrt(2*u - 1) + 1)
x
```

```
[1] 1.812691
```

Problem 24

In Example 5f, we simulated a normal random variable using the rejection technique with an exponential distribution. Show that the number of iterations is minimized when $\lambda = 1$.

Pseudocode

1. Let $g(x) = \lambda e^{-\lambda x}$.
2. To minimize iterations, compare acceptance rates across different λ values.
3. Use simulation to show that when $\lambda = 1$, the number of iterations is lowest.

Putting it in practice

```
set.seed(202425)
lambda <- 1
accept <- FALSE
iterations <- 0
while(!accept) {
  X <- rexp(1, rate = lambda)
  u <- runif(1)
  if(u <= exp(-(X^2)/2)) accept <- TRUE
  iterations <- iterations + 1
}
iterations
```

```
[1] 3
```