# ISI-BUDS
# PROBABILITY - PART IV

Veronica J. Berrocal

July 12, 2022

# PART V PLAN:

- Practical implications of a Monte Carlo method
- Introducing 3 algorithms:
  - Inversion method
  - Transformation method
  - Reject sampling

# A KEY NEED TO IMPLEMENT MONTE CARLO METHOD

$E(g(X))$

- We have seen that the Monte Carlo estimator provides us with an approximation of the expected value of a function $\boxed{g(X)}$ of any random variable $\boxed{X}$.

$f(x)$ $F$

- We can calculate probabilities using expected values of appropriate functions of random variable (e.g. indicator functions).
- To implement Monte Carlo estimation, <u>we ought to be able to simulate random values from distributions!</u>

# HOW TO SIMULATE RANDOM VARIABLES

- How do we simulate values of a random variable $X$ that follows a certain distribution with p.d.f. $f(x)$?
- If the distribution has a name, then we can use R and the appropriate function of the form "*r+name of distribution*", such as `runif`, `rgamma`, `rnorm`, `rbinom`, `rpoiss`, `rexp`, `rbeta`, etc
- What if the random variable $X$ follows a distribution for which we don't know the name? How do we simulate values for the random variable $X$?

# INVERSION METHOD

- There are different algorithms that can be used to simulate random variales.
- The inversion method is one of the most commonly used methods to simulate random variables and perhaps the simplest to implement.

- To use this method, we ought to go back to c.d.f.'s.

# PROPERTIES OF CDF'S

- We have seen the definition of a c.d.f. starting from a random variable X:

$$F(x) = P(X \leq x).$$

- What guarantees that a function F(x) is a c.d.f.? What are the properties of a c.d.f.?

  - $F(x)$ is increasing. [Why?]
  - $F(x)$ is right continuous, that is $F(x+\varepsilon) \to F(x)$ as $\varepsilon \to 0$
  - $F(x) \to 0$ as $x \to -\infty$ while $F(x) \to 1$ as $x \to \infty$.

# THE CDF OF A RANDOM VARIABLE HAS A UNIFORM DISTRIBUTION!

- Propostion: If $F(x)$ is a continuous and strictly increasing function on R. Let $X$ be a random variable with c.d.f. $F(x)$. Suppose $F(x)$ has an inverse $F^{-1}(x)$.

  The random variable $F(X)$ follows a uniform distribution on $[0,1]$.

# GENERATING RANDOM VARIABLES FROM ANY CDF

- This result is at the basis of the Inversion method to generate random variables from a given distribution.
- Proposition: Let $F(x)$ be a continuous and strictly increasing c.d.f. on R with inverse function $F^{-1}(x)$, which maps the interval [0,1] onto R.

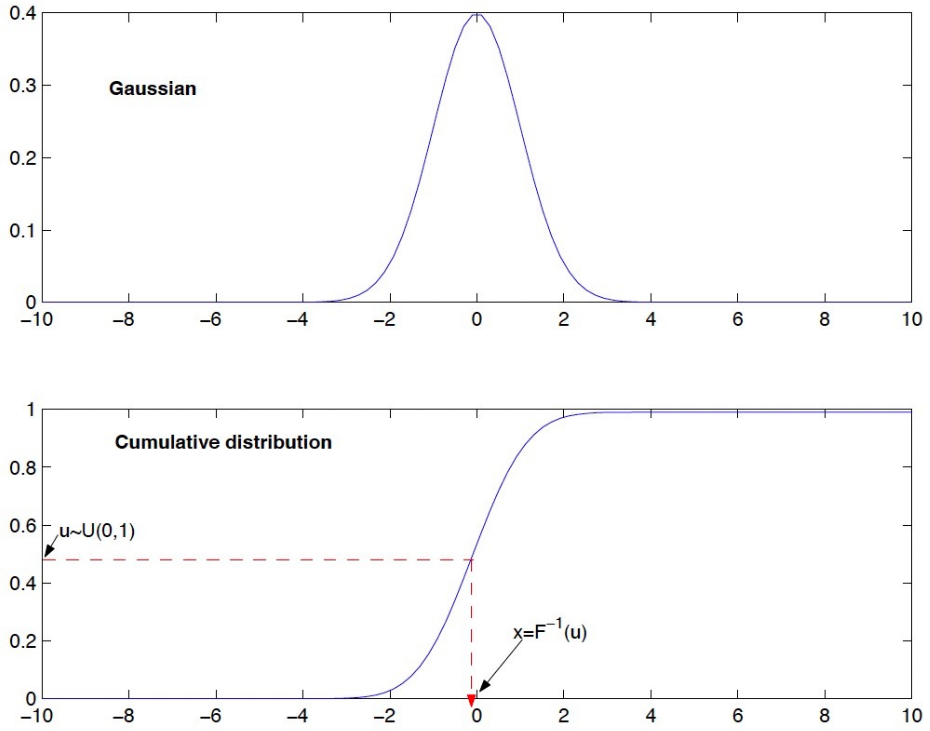
  Let $U \sim Uniform((0,1))$, that is, let U be a uniform random variable on[0,1], then $X=F^{-1}(U)$ is a random variable with c.d.f. $F(x)$.

# GENERAL INVERSION METHOD ALGORITHM

A general inversion method algorithm then would be structured as follows:

- Given a c.d.f $F(x)$, calculate $F^{-1}(x)$
- Simulate independent $U_i \sim \text{Uniform}(0,1)$
- Return $X_i = F^{-1}(U_i)$

# ILLUSTRATIVE EXAMPLE




Top: pdf of a Gaussian r.v., bottom: associated cdf.

# EXPONENTIAL DISTRIBUTION EXAMPLE

- Let $\lambda > 0$. The exponential CDF is given by

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

- Generate values of a random variable following an exponential distribution!

1. Set $F(x) = y$

$$1 - e^{-\lambda x} = y$$

2. Solve it for $x$

$$1 - y = e^{-\lambda x}$$

$$\log(1-y) = -\lambda x$$

$$x = -\frac{\log(1-y)}{\lambda} = F^{-1}(y)$$

Algorithm:

$n = 1{,}000$

1. Simulate $n$ independent $U_i \sim \text{unif}(0,1)$

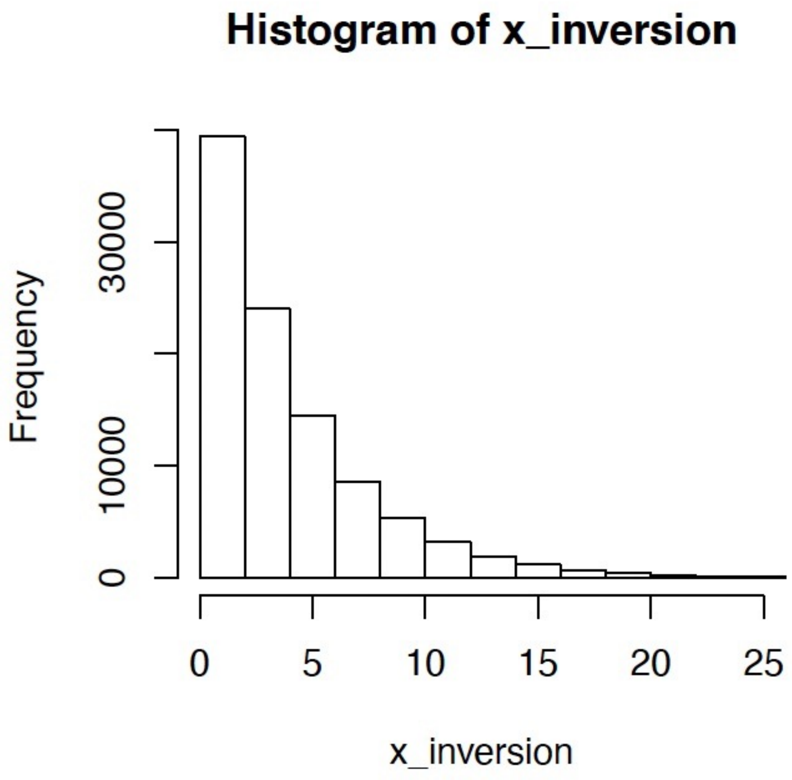
2. Calculate $X_i = -\frac{\log(1-U_i)}{\lambda}$

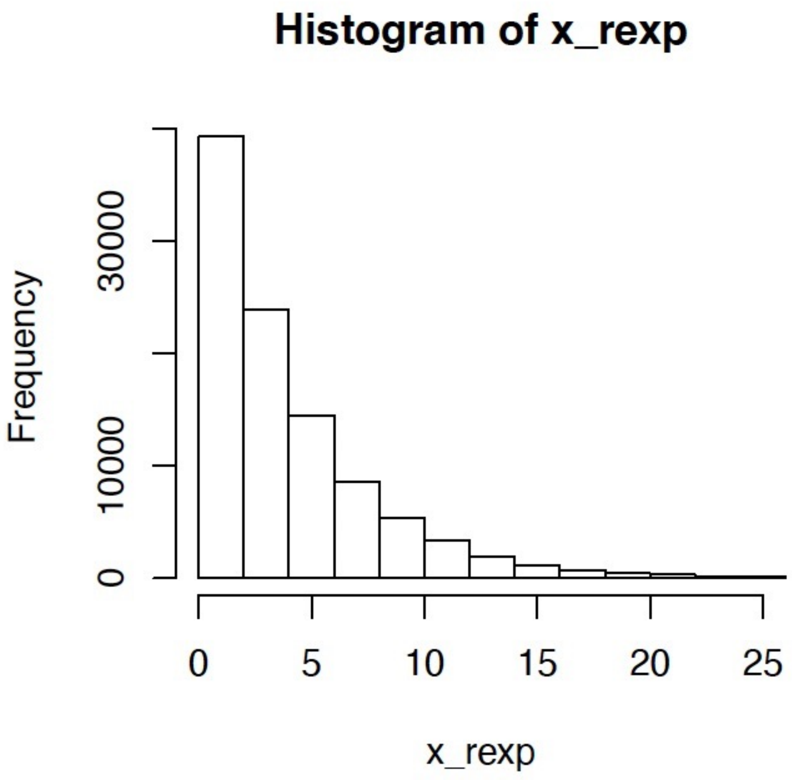
# EXPONENTIAL R.V.'S USING THE INVERSION METHOD

```r
# Generating values following an exponential distribution
lambda <- 0.25
n <- 10000
u <- runif(n)
x.inv <- -log(1-u)/lambda
x.rexp <- rexp(n,rate=lambda)

# Checking if the two samples are different
wilcox.text(x.inv,x.rexp)$p.value
```

# EXPONENTIAL R.V.'S USING THE INVERSION METHOD



# ANOTHER EXAMPLE: LOGISTIC DISTRIBUTION

- The logistic distribution has p.d.f. and c.d.f. given respectively by:

$$f(x) = \frac{exp(-x)}{(1+exp(-x))^2} \qquad F(x) = \frac{1}{1+exp(-x)}$$

- How would we simulate a random sample from a logistic distribution?

# HOW TO HANDLE DISCRETE R.V.'S?

- Simulating discrete random variables with the method of inversion requires slight more care.
- Why?

# GENERALIZED INVERSE

- For discrete random variables, we don't apply the inverse of the c.d.f. to a sample of values drawn from the uniform distribution. Rather, what we use is the generalized inverse to the c.d.f.
- Let F(x) be a c.d.f. on R. We define the generalized inverse F-1: [0, 1] -> R as the function

$$F^{-1}(u) = inf\{x \in \mathbb{R}: F(x) = u\}.$$

Then, if U~Uniform(0,1), $X=F^{-1}(U)$ has F(x) as c.d.f.

# DISCRETE RANDOM VARIABLES

- If X is a discrete random variable with p.d.f. $P(X = k) = p(k)$, then the c.d.f. of X is defined as:

$$F(x) = \sum_{k=0}^{\lfloor x \rfloor} P(x = k) = \sum_{k=0}^{\lfloor x \rfloor} p(k)$$

and its generalized inverse is defined as follows: $F^{-1}(u)$ is $x \in \mathbb{N}$ such that

$$\sum_{k=0}^{x-1} p(k) < x \leq \sum_{k=0}^{x} p(k)$$

with the LHS equal to 0 if x=0.

# EXAMPLE

```
p <- c(0.5, 0.3, 0.2) ## pmf
p_norm <- c(0, cumsum(p)) ## 0.0 0.5 0.8 1.0
m <- length(p)
n <- 100000
u <- runif(n)
x <- array(NA, n)
for(i in 1:n) {
    for(j in 1:m) {
        if ((p_norm[j] < u[i]) & (u[i] <= p_norm[j + 1])) {
            x[i] <- j
        }
    }
}
sum(is.na(x)) ## 0
table(x)
##     1     2     3
## 50227 30105 19668
```

# TRANSFORMATION METHOD

- Suppose we have a random variable Y that follow some distribution with c.d.f. $F_Y$ and defined on some space $S_Y$, which we know how to simulate.
- Suppose that there is a random variable X that follows some distribution with c.d.f. $F_X$ and defined on some space $S_X$, that we wish to simulate.
- If we can find a function $\boldsymbol{\varphi : S_Y \to S_X}$, with the property that if Y follows the distribution with c.d.f. $F_Y$, then $\boldsymbol{X = \varphi(Y)}$ follows the distribution with c.d.f. $F_X$, we can simulate the random variable X, by simulating first Y and then applying the transformation $\boldsymbol{\varphi}$ onto the sampled values for Y.

# TRANSFORMATION METHOD GENERAL ALGORITHM

- The transformation method general algorithm to generate samples from a given distribution will be as follows:

1. Find $Y \sim F_Y$ that you can simulate from, and a function $\varphi$ such that $X = \varphi(Y) \sim F_X$.
2. Simulate independently $Y_i \sim F_Y$.
3. Return $X_i = \varphi(Y_i) \sim F_X$.

# EXAMPLE: EXPONENTIAL TO GAMMA

- Note: Let $Y_i$, i=1,2,..$\alpha$ be independent random variables that follow an Exponential distribution with parameter $\lambda = 1$, that is: $Y_i \sim Exp(1)$.
- Denote by $X = \frac{1}{\beta}\sum_{i=1}^{\alpha} Y_i$, then $X \sim Gamma(a, \beta)$.

- Use this result to devise an algorithm to sample values from a Gamma(7,10) distribution.

# REJECTION SAMPLING: MAIN IDEA

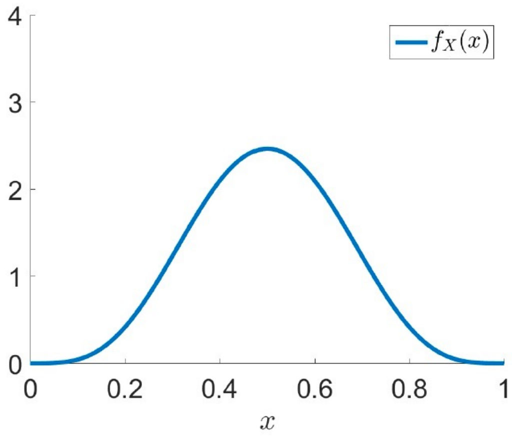
- Another algorithm to simulate random variables is based on a rejection sampling idea.
- The method consists into drawing random variables from a larger space defined by a proposal p.d.f., and "reject" those values that are not in the region defined by the target p.d.f.

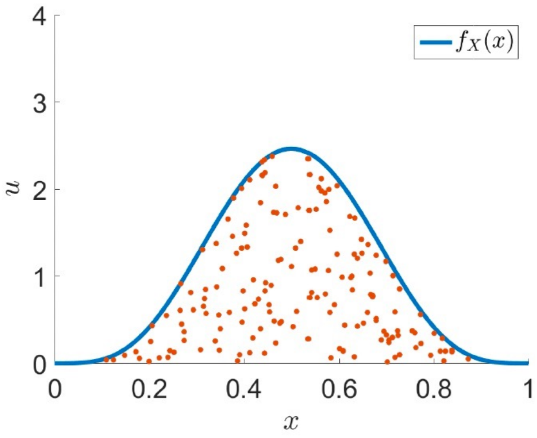
# FUNDAMENTAL THEOREM OF SAMPLING

**Theorem (Fundamental Theorem of simulation)**

*Let $X$ be a rv on $\Omega$ with pdf or pmf $f_X$. Simulating $X$ is equivalent to simulating*

$$(X, U) \sim \mathrm{Unif}(\{(x, u) | x \in \Omega, 0 < u < f_X(x)\})$$



# REJECTION SAMPLING IDEA

- Direct sampling of $(X, U)$ uniformly over the set $\mathcal{A}$ is in general challenging
- Consider some superset $\mathcal{S}$ such that $\mathcal{A} \subseteq \mathcal{S}$, such that simulating uniform rv on $\mathcal{S}$ is easy
- Therefore, a uniform distribution on $\mathcal{A}$ can be obtained by drawing from a uniform distribution on $\mathcal{S}$, and *rejecting* samples in $\mathcal{S}$ not in $\mathcal{A}$
- Rejection sampling technique:
  1. Simulate $(Y, V) \sim \mathrm{Unif}(\mathcal{S})$, with simulated values $y$ and $v$
  2. if $(y, v) \in \mathcal{A}$ then stop and return $X = y$,$U = v$,
  3. otherwise go back to 1.
- The resulting rv $(X, U)$ is uniformly distributed on $\mathcal{A}$
- $X$ is marginally distributed from $f_X$

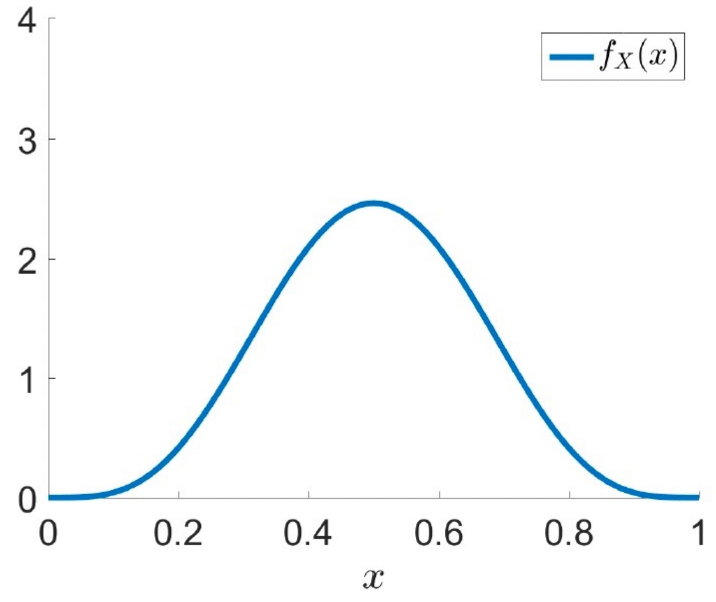
# EXAMPLE: BETA DENSITY

- Let $X \sim Beta(\alpha = 5, \beta = 5)$, then the p.d.f. is given by:

$$f(x) = \frac{1}{B(\alpha, \beta)} x^{\alpha - 1} \cdot (1 - x)^{\beta - 1}$$

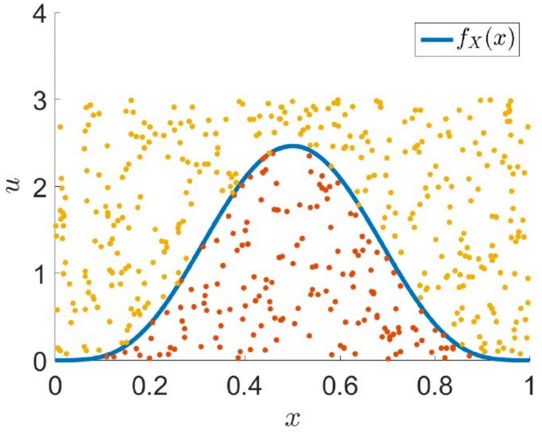
for $0<x<1$ and with $\alpha = 5, \beta = 5$.

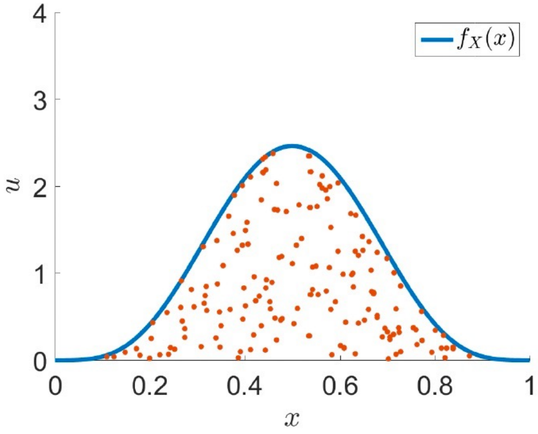
Then, $f(x)$ is bounded above by 3 on [0, 1].



# EXAMPLE: BETA DENSITY

- Let $\mathcal{S} = \{(y, v) | y \in [0, 1], v \in [0, 3]\}$
  1. Simulate $Y \sim \mathcal{U}([0, 1])$ and $V \sim \mathcal{U}([0, 3])$, with simulated values $y$ and $v$
  2. If $v < f_X(x)$, return $X = x$
  3. Otherwise go back to Step 1.
- Only requires simulating uniform random variables and evaluating the pdf pointwise



# REJECTION SAMPLING MORE PRECISELY

- In general: consider a random variable X defined on a space $S_X$ with p.d.f. f(x), the target distribution.
- We want to sample from f(x) using a proposal p.d.f. q(x) from which can sample easily.
- We find a constant M such that $\frac{f(x)}{q(x)} \leq M$ for all $x \in S$.
- The following rejection algorithm, returns samples for a random variable X with p.d.f. f(x).

# REJECTION SAMPLING GENERAL ALGORITHM

1. Identify a proposal c.d.f. Q(x) from which it is **easy to sample** with p.d.f. q(x). Find M such that $\frac{f(x)}{q(x)} \leq M$ for all $x \in S$.

2. Simulate $Y_i \sim Q$ and $U_i \sim Unif((0,1))$.

3. If $U_i \leq {\left( {}^{f(Y_i)}/_{q(Y_i)} \right)}/_{M}$, then return $X_i = Y_i$, otherwise do not return anything.