

# Monte Carlo Algorithm and its Application

## Applied Math Tutorial

Tianlu Zhu

Shanghaitech University

February 22, 2023

# Outline

# Introduction

In the presentation, I would first introduce Monte Carlo Integration, which requires enormous random sample. Then we introduce common ways of sampling from a given distribution, including Transformation method, Accept-rejection method, and Markov Chain Monte Carlo Algorithm. In each method I would show code implementation, analyse the pros and cons, and a brief proof.

# Outline

# MC Integration

Suppose we want to evaluate an integral

$$\int_D \phi(x) dx$$

for which there is no closed analytic solution. If the integration has the form

$$\phi(x) = \tilde{\phi}(x)f(x)$$

for some density function  $f$ , then the integral has the form:

$$\int_D \phi(x) dx = \int_D \tilde{\phi}(x)f(x) dx = E[\tilde{\phi}(X)]$$

where  $X$  is an RV with PDF  $f$ .

# MC Integration

If we know how to simulate realisations of  $X$ , say  $x^{(1)}, \dots, x^{(n)}$ , then we have an estimate

$$\int_D \phi(x) dx = E[\tilde{\phi}(X)] \approx \frac{1}{n} \sum_{i=1}^n \tilde{\phi}(x^{(i)}) = \hat{I}$$

Also, the variance should be

$$\begin{aligned} \text{Var}(I) &= \frac{1}{n^2} \sum_{i=1}^n \text{Var}(\tilde{\phi}(x^{(i)})) \\ &\approx \frac{1}{n(n-1)} \sum_{i=1}^n (\tilde{\phi}(x^{(i)}) - \hat{I})^2 \end{aligned}$$

So it is crucial to generate sample from a specific distribution.

# Outline

# Derivation of the Algorithm

The easiest distribution for computer to generate is  $\text{Unif}[0, 1]$ . If  $X$  owns CDF  $F(\cdot)$ , Then  $F(X) \sim \text{Unif}[0, 1]$ . By *inversion*, setting  $X \sim F^{-1}(U)$ , which  $U \sim \text{Unif}[0, 1]$ . Then From the uniform distribution sample we can generate sample follows  $F(\cdot)$ .

Consider discrete random variable, simply left

$$F^{-1}(u) = \min \{x : F(x) \geq u\}$$

And the algorithm goes on as continuous situation.



# Pros and Cons

## Pros

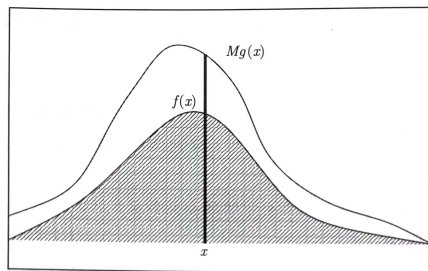
- Almost Easiest way to generate a sample
- ?

## Cons

- The CDF is not invertible, i.e. PDF  $f(\cdot) = 0$  somewhere.

# Outline

# Intuition



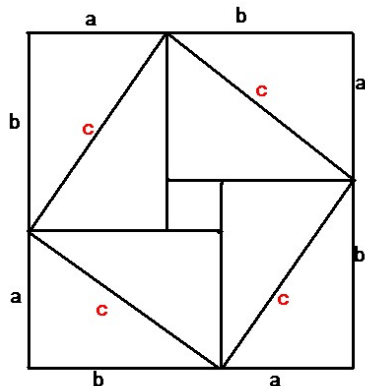
- $f(x)$ : object function
- $g(x)$ : proposal function
- $M$ : auxiliary constant
- Make sure that  $Mg(x) \geq f(x)$  for all  $x \in \Omega$

# An Implementation of the Algorithm

```
1 N <- 50000
2 M <- 3.858
3 y <- runif(N, min = 0, max = 2)
4 u <- runif(N, min = 0, max = 1)
5 gy <- 0.5
6 fy <- 6 * (y - 0.5)^2 / 7
7 x <- y[u < fy / gy / M]
8 sample <- length(x)
9 hist(x, breaks = 50, freq = FALSE, col = "#adabab",
10 main = "f(x) = 6(x - 0.5)^2/7")
11 curve(6 * (x - 0.5)^2 / 7, from = 0,
12       to = 2, add = TRUE, col = "red")
```

# Outline

# Minipage



- 1 item
- 2 another
- 3 more
  - first
  - second
  - third

# Columns

This is a text in first column.

$$E = mc^2$$

- First item
- Second item

first block

columns achieves splitting the screen

second block

stack block in columns

# Create Tables

| first | second | third |
|-------|--------|-------|
| 1     | 2      | 3     |
| 4     | 5      | 6     |
| 7     | 8      | 9     |



# Equation1

A matrix in text must be set smaller:  $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$  to not increase leading in a portion of text.

$$f(n) = \begin{cases} n/2 & \text{if } n \text{ is even} \\ -(n+1)/2 & \text{if } n \text{ is odd} \end{cases}$$

$$50apples \times 100apples = lotsofapples^2$$

# Equation2

$$\sum_{\substack{0 \leq i < m \\ 0 \leq j < n}} P(i, j) = \int_a^b \prod P(i, j)$$

$$P\left(A = 2 \left| \frac{A^2}{B} > 4 \right.\right)$$

$$(a), [b], \{c\}, |d|, \|e\|, \langle f \rangle, \lfloor g \rfloor, \lceil h \rceil, \lceil i \rceil$$

# Equation3

$$Q(\alpha) = \alpha_i \alpha_j y_i y_j (x_i \cdot x_j)$$

$$Q(\alpha) = \alpha^i \alpha^j y^{(i)} y^{(j)} (x^i \cdot x^j)$$

$$\Gamma = \beta + \alpha + \gamma + \rho$$

# Outline

# End

The last page.