

# Tarea 3: El método Monte Carlo

Elías Alejandro García Bueno  
Bryan Alejandro Andrade Amaya  
Jesus Adalberto Mendoza Flores

15 de septiembre de 2022

## Resumen

Se llama de Montecarlo porque tiene relación con la generación de números aleatorios. Es un método estadístico por el que, a través de números aleatorios, pueden resolverse de manera muy aproximada problemas complejos. Su inventor, en 1947, fue el matemático de origen húngaro, nacionalizado estadounidense, von Newman (1903-1957). Parece ser que la idea le surgió cuando el matemático Stanislaw Ulam (1909-1984) le comentó que había resuelto un complicado solitario, a través de pruebas aleatorias. Von Newman utilizó este método para calcular a qué distancia del suelo debían explotar las bombas de Hiroshima y Nagasaki para que el efecto fuera más devastador.[1]

## 1. Introducción

El termino Monte Carlo se aplica a un conjunto de metodos matemáticos que se empezaron a usar en los 1940s para el desarrollo de armas nucleares en Los Alamos, favorecidos por la aparicion de los ordenadores digitales modernos. Consisten en resolver un problema mediante la invencion de juegos de azar cuyo comportamiento ´ simula algun fenomeno real gobernado por una distribucion de probabilidad (ej. un proceso físico) o sirve para realizar un cálculo (ej. evaluar una integral). Mas técnicamente, un Monte Carlo es un proceso estocástico numerico, es decir, una ´ secuencia de estados cuya evolucion viene determinada por sucesos aleatorios. Recordemos que un suceso aleatorio es un conjunto de resultados que se producen con cierta probabilidad. [2]

## 2. Desarrollo

A continuacion se muestra un ejemplo para opbservar como funciona el metodo montecarlo:

Ejemplo 1: Gotas de lluvia para estimar // Consideremos un circulo de radio unidad circunscrito por un cuadrado. Suponiendo una lluvia uniforme sobre el cuadrado, podemos hallar el valor de  $\pi$  a partir de la probabilidad de que las gotas caigan dentro del circulo (Figura 1):

$$P = \frac{\text{área del círculo}}{\text{área del cuadrado}} = \frac{\int_{-1}^1 dx \int_{-\sqrt{1-x^2}}^{\sqrt{1-x^2}} dy}{\int_{-1}^1 dx \int_{-1}^1 dy} = \frac{2 \int_{-1}^1 dx \sqrt{1-x^2}}{2 \cdot 2} = \frac{\pi}{4}.$$

Figura 1: Formula del ejemplo

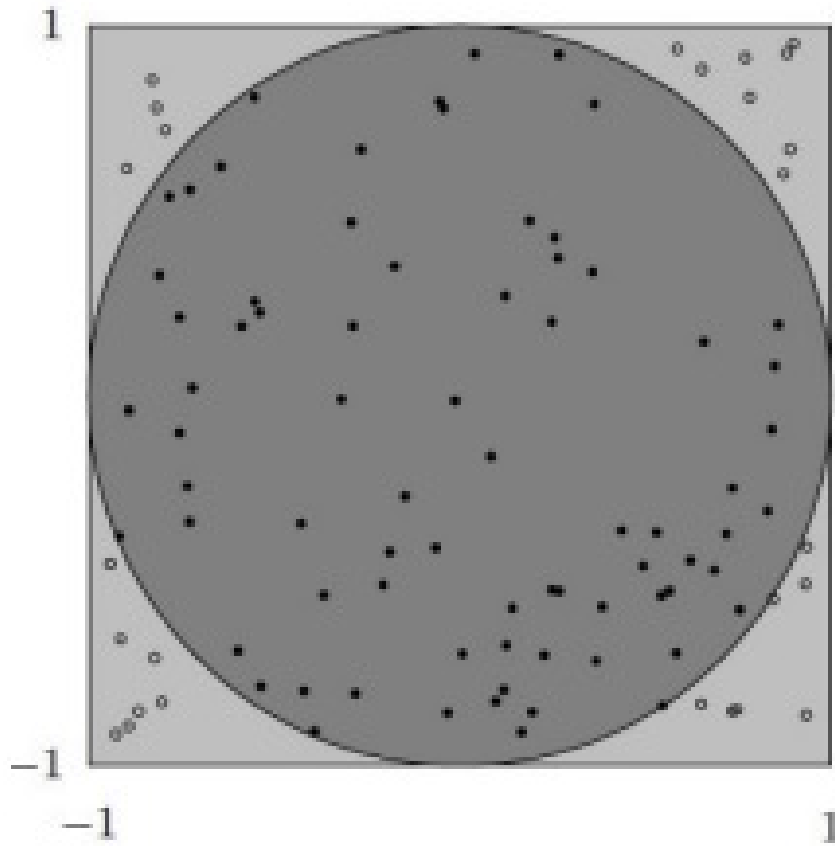


Figura 2: Experimento de las gotas de lluvia para estimar

Es decir,  $\pi \approx 4P$ . Notese que: (i) podemos simular facilmente este experimento generando aleatoriamente con un ordenador puntos de coordenadas cartesianas  $(x, y)$ ; (ii) podemos mejorar nuestra estimación de  $\pi$  aumentando el numero de puntos generados; (iii) tenemos un metodo para hallar la integral que aparece en la ecuacion (Figura 1). Ciertamente el valor de  $\pi$  puede encontrarse de forma mas rápida y precisa mediante otros metodos, pero veremos que el método Monte Carlo es el más eficiente para hallar integrales multidimensionales (Figura 2).

### 3. Simulacion

En en la figura 3 se observa la programacion que se utilizo en el IDE de python para realizar la grafica de montecarlo, y así mismo se observa en la figura 4 como quedaron los resultados de la grafica.

### 4. Conclusiones

El método de Monte Carlo permite obtener soluciones probabilísticas para ecuaciones donde no es satisfactorio utilizar un método determinístico por falta de precisión en los datos de entrada. Es muy recomendable hacer uso de un computador para aplicar el método de Monte Carlo. Éste puede realizar las cientos o miles de simulaciones necesarias en pocos segundos y con la cantidad de cifras decimales necesarias. Ahora que conocemos este metodo sera de gran utilidad para aplicarlo cuando se este realizando algun trabajo o proyecto de otra materia, asi como tambien lo podemos aplicar en nuestra vida laboral ya que es de gran utilizad.

```

Tarea 3.py > ...
3  #EQUIPO 7 - BIOMECANICA
4  import matplotlib.pyplot as plt
5  import numpy as np
6
7
8  width = 10000
9  height = width
10 radio = width
11 npuntos = 0
12 ndentro = 0
13 radio2= radio * radio
14 replicas = 100
15 promediopi = []
16
17 if __name__ == '__main__':
18     with Pool(4) as np:
19         for j in range(replicas):
20             for i in range(1,1000):
21                 x = randint(0, width)
22                 y = randint(0, width)
23                 npuntos += 1
24                 if x * x + y * y <= radio2:
25                     ndentro += 1
26                 pi = ndentro * 4 / npuntos
27                 promediopi.append(pi)
28
29 plt.xlim(0, 1000)
30 plt.ylim(2.12,5)
31 plt.plot(promediopi)
32 plt.xlabel('Numero de replicas')
33 plt.ylabel('Valor de pi')
34 plt.title('Equipo 7')
35 plt.grid()
36 plt.show()

```

Figura 3: Programacion en python para el metodo monte carlo

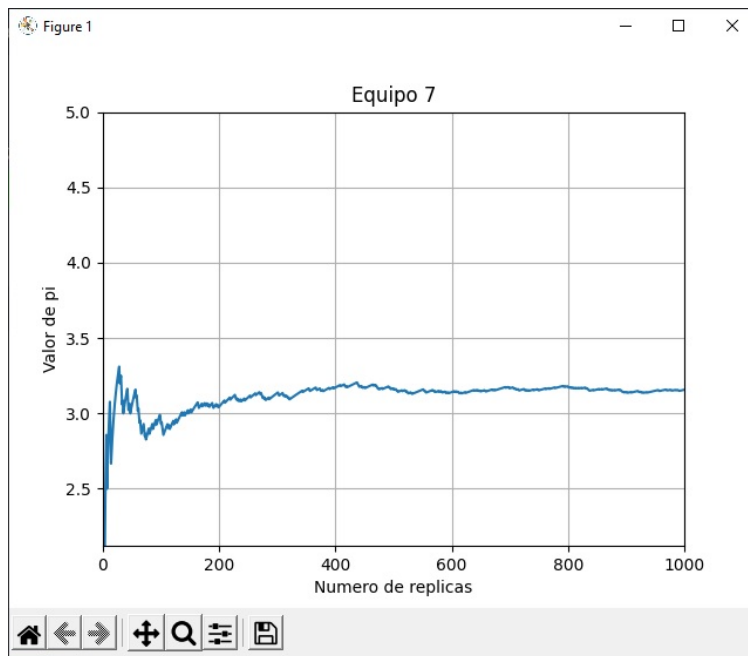


Figura 4: Programacion en python para el metodo monte carlo

## Referencias

- [1] Miguel Barreras. El método de montecarlo. *Entorno Abierto*, 6:15–16, 2015.
- [2] José Ignacio Illana. Métodos monte carlo. *Departamento de Física Teórica y del Cosmos, Universidad de Granada*, 26(01), 2013.