Método de Monte Carlo y Aplicación en la Física Estadística

Javier Alejandro Salcedo Castañeda

Física estadística

Departamento de Física – Facultad de Ciencias

Universidad de Los Andes

Mérida, Venezuela

Agosto, 2025

Objetivo General:

En el corazón de la ciencia y la física, a menudo nos enfrentamos a problemas tan complejos que resolverlos de forma exacta se vuelve una tarea imposible. El **método de Monte Carlo** emerge como una ingeniosa solución a este dilema. Su magia no radica en la precisión matemática de un solo cálculo, sino en la **probabilidad y la aleatoriedad**, utilizando la capacidad de las computadoras para realizar miles y miles de intentos. Nuestro objetivo es adentrarnos en esta técnica y demostrar su poder para estimar valores numéricos que, de otro modo, serían inalcanzables. A través de la programación en **Python**, convertiremos la teoría en algo tangible, usando el caos para encontrar el orden y resolviendo dos desafíos prácticos: la estimación del valor de Pi y el cálculo de una integral definida.

Objetivos Específicos:

1. Comprender la lógica del método: Explicar cómo el muestreo aleatorio se puede utilizar para obtener resultados precisos y confiables, desmitificando la idea de que la aleatoriedad es opuesta a la precisión.
2. Aplicar el método para la estimación de Pi: Usar la simulación de ”lanzamiento de dardos” para demostrar cómo una simple relación de áreas puede llevarnos a estimar uno de los números más fascinantes de las matemáticas.
3. Resolver una integral definida: Mostrar la versatilidad del método al utilizarlo para calcular el área bajo la curva de una función compleja, un problema común en la física y la ingeniería.
4. Visualizar los resultados: Utilizar gráficos para representar la forma en que los datos aleatorios se comportan y cómo la estimación del resultado converge hacia el valor esperado a medida que aumenta el número de puntos.
5. Conectar con la Física Estadística: Establecer una clara relación entre las simulaciones que realizamos y el concepto de los ensambles en la física estadística, explicando cómo se utiliza esta técnica para explorar y entender sistemas con un número enorme de partículas.

A. Fundamento Físico

El método de Monte Carlo debe su nombre a la famosa capital del juego, y no por casualidad. Aunque los principios matemáticos detrás del método existen desde hace siglos, su verdadera aplicación y nombre moderno surgieron durante el ultrasecreto Proyecto Manhattan en la década de 1940. En ese momento, científicos como Stanislaw Ulam y John von Neumann necesitaban resolver problemas complejos relacionados con la difusión de neutrones, algo que era casi imposible de calcular de forma analítica.

Ulam tuvo la idea de usar un proceso aleatorio, similar a los juegos de azar, para simular el comportamiento de las partículas. Von Neumann, al percatarse del potencial, lo implementó en la primera computadora moderna, el ENIAC. Así, el método de Monte Carlo nació de una necesidad de la física nuclear, demostrando que el azar, lejos de ser un obstáculo, puede ser una poderosa herramienta para desentrañar los secretos de la naturaleza.

B. Fundamento Estadístico

El método de Monte Carlo se apoya en dos pilares fundamentales de la estadística: la Ley de los Grandes Números y la probabilidad de variables aleatorias.

B- α. La Ley de los Grandes Números y el Concepto de Promedio

Esta ley establece que, si repetimos un experimento aleatorio un número suficientemente grande de veces, el promedio de los resultados obtenidos se aproximará cada vez más al valor esperado o promedio teórico. En términos sencillos, el caos de los resultados individuales se diluye, revelando un patrón subyacente que se acerca a la ”verdad” estadística.

Ejemplo del lanzamiento de una moneda: La probabilidad de que caiga ”cara” es de 0.5. Si lanzas la moneda 10 veces, es posible que obtengas 8 caras. Sin embargo, si la lanzas 100,000 veces, la proporción de caras se acercará mucho más al 50 %.

El método de Monte Carlo aplica esta misma lógica a problemas físicos y matemáticos complejos. En lugar de resolver la ecuación directamente, la convierte en un experimento de ”lanzamiento de moneda” a gran escala. Cada punto aleatorio que generas es un intento, y el promedio de todos los intentos te da la respuesta que buscas.

B- β. El Muestreo como Herramienta de Aproximación

La clave de la eficiencia de Monte Carlo es el muestreo aleatorio. En lugar de explorar todo el espacio de posibles soluciones (lo cual es inviable para la mayoría de los problemas de física), el método selecciona una muestra representativa de puntos. La exactitud de la estimación depende únicamente del número de puntos muestreados, no de la dimensionalidad del problema. Esto lo hace excepcionalmente útil en problemas de alta dimensión, donde los métodos numéricos tradicionales fallan.

C. Ecuaciones y Metodología del Método de Monte Carlo

El método de Monte Carlo no es una simple fórmula, sino una poderosa metodología computacional que transforma problemas complejos en una serie de experimentos aleatorios. Su esencia radica en el principio de que las propiedades de una población (o un sistema físico) pueden ser inferidas a partir de una muestra aleatoria lo suficientemente grande. La clave es convertir un problema matemático o físico, que podría ser imposible de resolver de forma analítica, en un problema de probabilidad que se pueda solucionar con la generación de números al azar.

La metodología general se puede desglosar en los siguientes pasos, que actúan como la columna vertebral de cualquier simulación de Monte Carlo:

1. Definir el Dominio de Entrada: Este es el primer paso crítico. Consiste en establecer claramente el espacio de parámetros o el rango de variables que son relevantes para el problema. En un problema de física, esto podría ser el rango de temperaturas o el volumen del sistema. En un problema matemático, es el intervalo o la región en la que se busca una solución.
2. Generar Entradas Aleatorias: Una vez definido el dominio, se procede a generar una gran cantidad de puntos de datos aleatorios e independientes dentro de ese espacio. La calidad de estos números aleatorios es crucial para la precisión del método. Estos puntos actúan como ”sondas” que exploran de manera no sesgada el dominio del problema.
3. Realizar el Cálculo o Simulación: Para cada punto aleatorio generado, se realiza un cálculo o se ejecuta una simulación para determinar un resultado. Este paso es el corazón de la computación. Aunque el cálculo para un solo punto puede ser simple, la ejecución repetida miles o millones de veces es lo que genera una visión estadística del problema.
4. Promediar los Resultados: Finalmente, se analizan los resultados obtenidos de todos los puntos aleatorios. Se suman y se promedian para obtener una estimación del valor que se busca. La Ley de los Grandes Números garantiza que, a medida que el número de puntos aleatorios aumenta, esta estimación converge hacia el valor real del sistema.

Esta metodología es un pilar de la Física Estadística, donde el ”estado” de un sistema (como un gas o un imán) es una variable aleatoria. El método de Monte Carlo permite explorar las configuraciones más probables del sistema sin tener que calcular explícitamente cada una de ellas, lo cual sería computacionalmente imposible.

D. Simulaciones

Aquí es donde la teoría del método de Monte Carlo cobra vida a través de la programación en Python. Demostraremos su poder con dos ejemplos concretos que ilustran su versatilidad para resolver problemas matemáticos complejos. Cada simulación no solo nos dará un resultado numérico, sino que también nos proporcionará una visualización intuitiva del proceso.

D- α. Simulación 1: Estimación del Valor de Pi

Este ejemplo es la piedra angular para entender el método. Nuestra simulación de "lanzamiento de dardos" calcula el valor de Pi de manera probabilística.

D- α-I. El Ejercicio Numérico: El objetivo es estimar Pi lanzando una gran cantidad de puntos aleatorios ($NumeroParticulas$) dentro de un cuadrado de $2x2$. Al contar cuántos de esos puntos caen dentro del círculo inscrito, podemos usar la proporción de aciertos para aproximar el valor de Pi. En tu código, la variable $puntos_e n_c irculo$ es la que lleva la cuenta de los "aciertos", y la fórmula $4 * (puntos_e n_c irculo/num_p untos)$ aplica la relación de áreas.

D- α-II. El Código ($monte\_carlo\_pi$): El script (Fig. 1) genera coordenadas (x, y) uniformemente entre -1 y 1. La condición $(x**2 + y**2)**0{,}5 <= 1$ es la que determina si un punto está dentro del círculo. El código no solo calcula el valor, sino que también prepara los datos para su visualización.

D- α-III. Gráficos y Análisis: La visualización del proceso es fundamental. El gráfico (Fig. 4) muestra los puntos generados, diferenciando con colores $('blue' y 'red')$ los que cayeron dentro y fuera del círculo. Este gráfico demuestra visualmente cómo el método "llena" el espacio y cómo la estimación se vuelve más precisa a medida que más puntos se lanzan. A mayor número de puntos, la distribución se vuelve más densa y la aproximación de la proporción de áreas es más exacta, llevando a una estimación de Pi más cercana a la real. En esta parte se obtuvo una **discrepancia del** $0{,}001\,\%$

D- β. Simulación 2: Integración Definida de una Función Compleja

Este segundo ejemplo va un paso más allá, mostrando cómo el método de Monte Carlo puede resolver integrales definidas de funciones que son difíciles o imposibles de integrar analíticamente.

D- β-I. El Ejercicio Numérico: El objetivo es estimar el valor de la integral de la función $f(x) = 1/(1+sinh(2x)*(log(x))^2)$ en el intervalo de 0.8 a 3. Esto se logra promediando el valor de la función en miles de puntos aleatorios generados en ese intervalo. La fórmula $(lim\_sup - lim\_inf) * suma/cant\_num$ realiza esta operación, multiplicando el valor promedio de la función por el ancho del intervalo para obtener el área total.

D- β-II. El Código ($IntegralDefinida$): El script (Fig. 3) utiliza la función $unif$ para generar $cant\_num$ puntos aleatorios uniformemente distribuidos en el intervalo $[0{,}8, 3]$. Para cada punto, la función $f(x)$ es evaluada, y la suma de estos valores se utiliza para la estimación.

D- β-III. Gráficos y Análisis: El gráfico (Fig. 4) de esta simulación es especialmente revelador. Muestra la curva de la función a integrar y un histograma de los puntos aleatorios utilizados para el cálculo. Esto ilustra cómo el método "muestrea" el área bajo la curva. Los picos en el histograma de los puntos indican que el muestreo es uniforme. La visualización demuestra cómo el método de Monte Carlo

```python
import random
import matplotlib.pyplot as plt

def monte_carlo_pi(num_puntos):

    puntos_en_circulo = 0
    puntos_x = []
    puntos_y = []
    colores = []

    for _ in range(num_puntos):
        x = random.uniform(-1, 1)
        y = random.uniform(-1, 1)
        puntos_x.append(x)
        puntos_y.append(y)

        distancia = (x**2 + y**2)**0.5

        if distancia <= 1 and distancia >= -1:
            puntos_en_circulo += 1
            colores.append('blue')
        else:
            colores.append('red')

    pi_estimado = 4 * (puntos_en_circulo / num_puntos)

    plt.figure(figsize=(6, 6))
    plt.scatter(puntos_x, puntos_y, c=colores, s=1)
    plt.title(f'Particulas : {num_puntos}, Estimación de Pi: {pi_estimado}')
    plt.xlabel('x')
    plt.ylabel('y')
    plt.xlim(-1.1, 1.1)
    plt.ylim(-1.1, 1.1)
    plt.gca().set_aspect('equal', adjustable='box')
    plt.show()

    return pi_estimado

NumeroParticulas = 100000000
pi_calculado = monte_carlo_pi(NumeroParticulas)
print(f"El valor estimado de Pi con {NumeroParticulas} puntos es: {pi_calculado}")
```

Fig. 1. Código de Simulación 1 : Estimación del Valor de Pi.

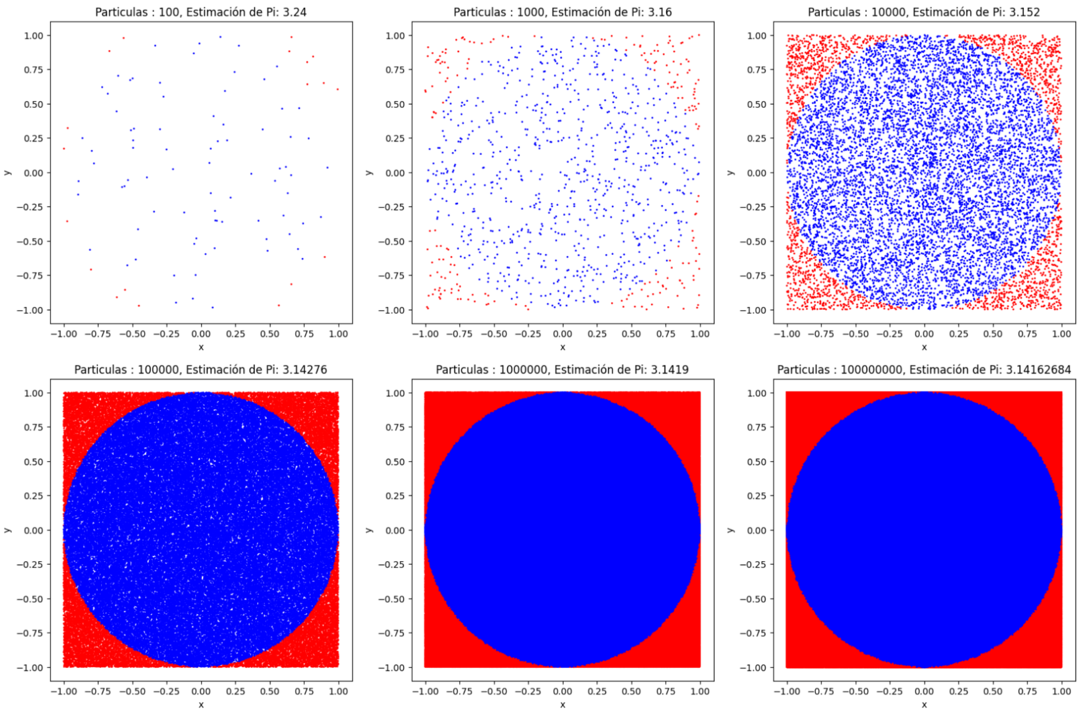

Fig. 2. Gráfica de Simulación 1 : Múltiples Valores para Diferentes Cantidades de Partículas.

estima el área total al promediar los valores de la función en toda la región, proporcionando una solución robusta y confiable. En esta parte se obtuvo una **discrepancia del** $0{,}033\,\%$

E. Relevancia y Aplicaciones

La relevancia del método de Monte Carlo radica en su capacidad para resolver problemas que desafían los métodos matemáticos tradicionales, especialmente aquellos con un gran número de variables o alta dimensionalidad. A diferencia de las soluciones determinísticas que requieren un cálculo exacto, Monte Carlo ofrece una solución probabilística que puede ser sorprendentemente precisa, eficiente y robusta.

Su importancia se extiende mucho más allá de las matemáticas puras, convirtiéndose en una herramienta esencial en la ciencia, la ingeniería y las finanzas. Sus aplicaciones abarcan desde la física hasta la inteligencia artificial, demostrando la versatilidad de la simulación aleatoria.

A continuación, se presentan algunas de las áreas más destacadas donde se aplica el método de Monte Carlo:

E- α. Física Estadística y Mecánica Cuántica:

Es fundamental para simular el comportamiento de sistemas con muchas partículas, como el **modelo de Ising** para el magnetismo o el estudio de la dinámica de fluidos. Permite calcular propiedades termodinámicas (energía, magnetización) sin tener que resolver las complejas ecuaciones de la mecánica cuántica o estadística de forma exacta.

```python
import numpy as np
import matplotlib.pyplot as plt
from numpy.random import uniform as unif

def IntegralDefinida(lim_inf, lim_sup, cant_num):
    x = unif(lim_inf, lim_sup, cant_num)

    suma = 0.0

    for val in x:
        suma = suma + 1 / (1 + np.sinh(2 * val) * (np.log(val))**2)

    resultado = (lim_sup - lim_inf) * suma / cant_num

    c = np.linspace(lim_inf, lim_sup, 1000)
    f = 1 / (1 + np.sinh(2 * c) * (np.log(c))**2)

    plt.figure(figsize=(10, 6))
    plt.plot(c, f, 'b-', label='Función a integrar')
    plt.hist(x, bins=50, density=True, alpha=0.6, color='g', label='Histograma de puntos')
    texto_area = f'Área ≈ {resultado:.5f}'
    plt.text(lim_inf + 0.1, max(f)*0.9, texto_area, fontsize=12, color='red', bbox=dict(facecolor='white', alpha=0.8))
    plt.xlabel('x')
    plt.ylabel('y')
    plt.title('Visualización del Método de Monte Carlo para Integración')
    plt.legend()
    plt.grid(True)
    plt.show()

    return print('El resultado de la integral es:', resultado)

IntegralDefinida(0.8, 3, 10000)
```

Fig. 3. Código de Simulación 2 : Integración Definida de una Función Compleja

E- β. Física Nuclear:

Su uso original en el **Proyecto Manhattan** sigue siendo relevante. Se utiliza para simular el transporte de neutrones y fotones en reactores nucleares y para el diseño de blindajes contra la radiación.

E- γ. Finanzas Cuantitativas:

Se aplica para modelar y predecir el comportamiento de los mercados, como la valoración de opciones financieras y la gestión de riesgos. Los modelos de Monte Carlo permiten simular miles de posibles escenarios de precios para estimar el riesgo de una inversión.

E- δ. Gráficos por Computadora:

Se utiliza en la industria del cine y los videojuegos para el renderizado realista. El trazado de rayos con Monte Carlo simula el camino de la luz de manera aleatoria, creando efectos de iluminación, sombras y reflejos increíblemente naturales.

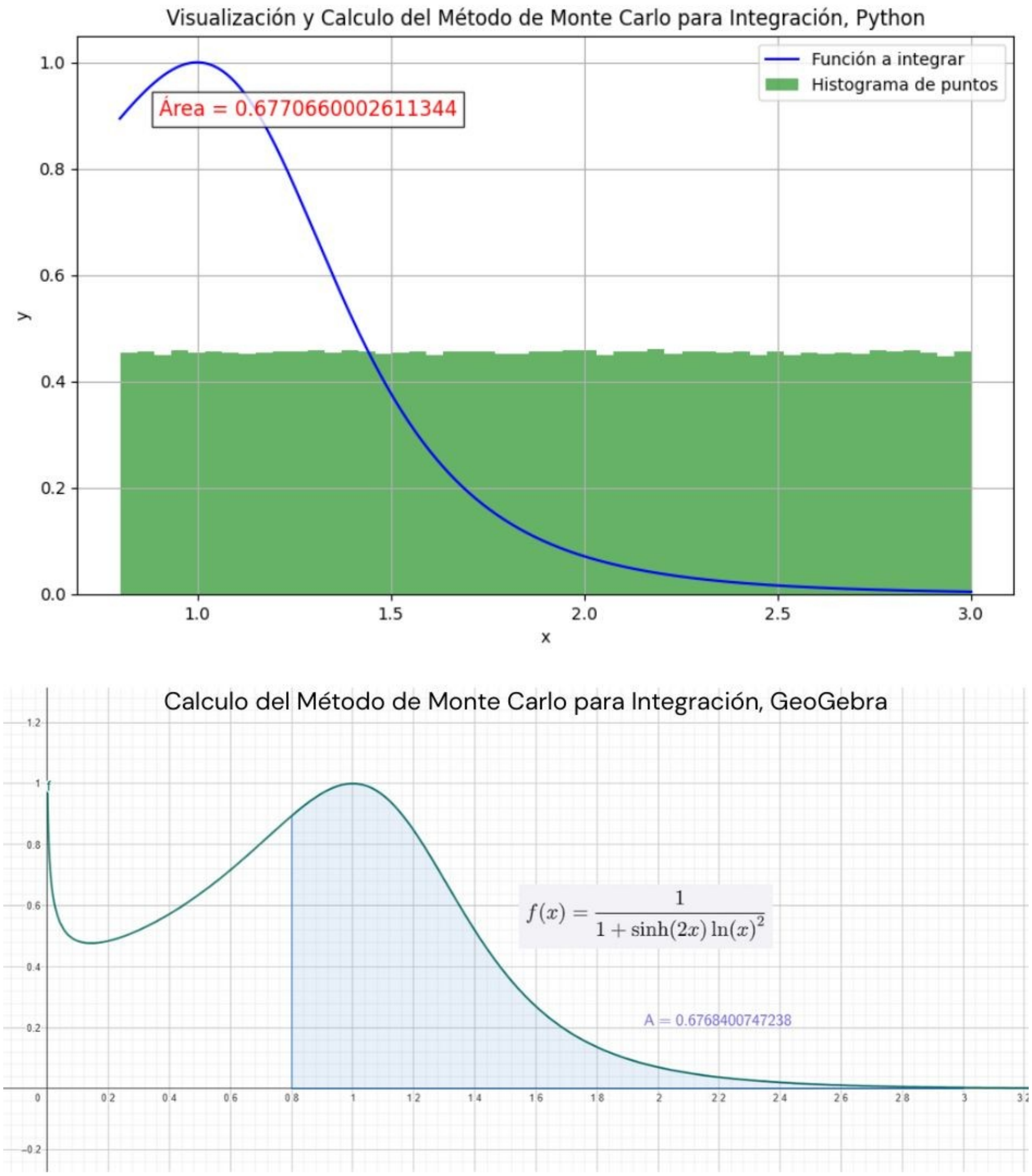

Fig. 4. Gráfica de Simulación 2 : Calculo del área bajo la curva en Python y en GeoGebra.

E- ϵ. Inteligencia Artificial y Aprendizaje Automático:

Es un componente clave en algoritmos de aprendizaje por refuerzo, donde se utilizan simulaciones para enseñar a una IA a tomar decisiones óptimas en entornos inciertos. Un ejemplo es su uso en programas de ajedrez y Go, como AlphaGo.

F. ¿Cómo se relaciona con los ensambles?

Para entender la relación entre el **método de Monte Carlo** y los **ensambles**, primero debemos recordar la meta de la Física Estadística: predecir las propiedades macroscópicas de un sistema (como su temperatura, presión o magnetización) a partir del comportamiento de sus innumerables partículas microscópicas.

Un **ensamble estadístico** es una colección de todas las configuraciones microscópicas posibles de un sistema que son consistentes con sus propiedades macroscópicas. Por ejemplo, en el **ensamble canónico**, todas las configuraciones posibles de un sistema a una temperatura y volumen constantes se consideran. Para calcular la magnetización promedio de un material, la física teórica nos dice que debemos promediar la magnetización de todas las configuraciones microscópicas posibles, lo cual es, en la práctica, imposible.

Aquí es donde entra el método de Monte Carlo como una solución ingeniosa. En lugar de recorrer exhaustivamente todas las configuraciones, el método realiza un **muestreo inteligente y aleatorio**.

F- α. Exploración del Espacio de Fases:

El método de Monte Carlo no simula la evolución real del tiempo del sistema (como un simulador de dinámica molecular), sino que ”salta” de una configuración a otra dentro del ensamble. Cada salto es un movimiento aleatorio, pero que sigue ciertas reglas para asegurar que las configuraciones más probables del sistema sean visitadas con más frecuencia.

F- β. Cálculo de Promedios:

Al ”visitar” las diferentes configuraciones, el método calcula la propiedad que nos interesa (por ejemplo, la magnetización). Al final, se promedia el valor de esta propiedad sobre todas las configuraciones muestreadas. Según la **Ley de los Grandes Números**, este promedio se aproxima al promedio de todo el ensamble.

Por lo tanto, el método de Monte Carlo actúa como un **explorador eficiente** del espacio de fases de un ensamble. Permite sortear la inmensidad del problema y obtener resultados fiables sobre las propiedades de un sistema, haciendo posible el estudio de fenómenos como las transiciones de fase y el comportamiento de materiales a diferentes temperaturas, que son el pan de cada día de la Física Estadística.

Referencias

[1] P. D. Moral, *Advanced Monte Carlo Methods*, ser. Lecture Notes in Statistics. New York, NY: Springer-Verlag, 2004, vol. 146.

[2] D. P. Landau and K. Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*, 5th ed. Cambridge University Press, 2021.

[3] E. Delgado, “Integrando con monte carlo | tutorial python,” YouTube, 2021, video en línea, disponible en: https://www.youtube.com/watch?v=WJjDr67frtM.

[4] D. un Voltio, “¿en qué consiste el método montecarlo?” YouTube, 2015, video en línea, disponible en: https://www.youtube.com/watch?v=AofM_lkdd50.

[5] Derivando, “Ruletas y bombas atómicas: El mÉtodo montecarlo,” YouTube, 2024, video en línea, disponible en: https://www.youtube.com/watch?v=m4X94Sq1Q4M.