Práctica 0: Python

Objetivo: Toma de contacto con Python.

1. Descripción de la práctica

En esta primera práctica has de implementar un algoritmo de integración numérica basado en el método de Monte Carlo.

Dada una función real e integrable de una sola variable $f(x)$, y su integral $F(x)$, la integral definida de $f(x)$ entre a y b viene dada por la expresión

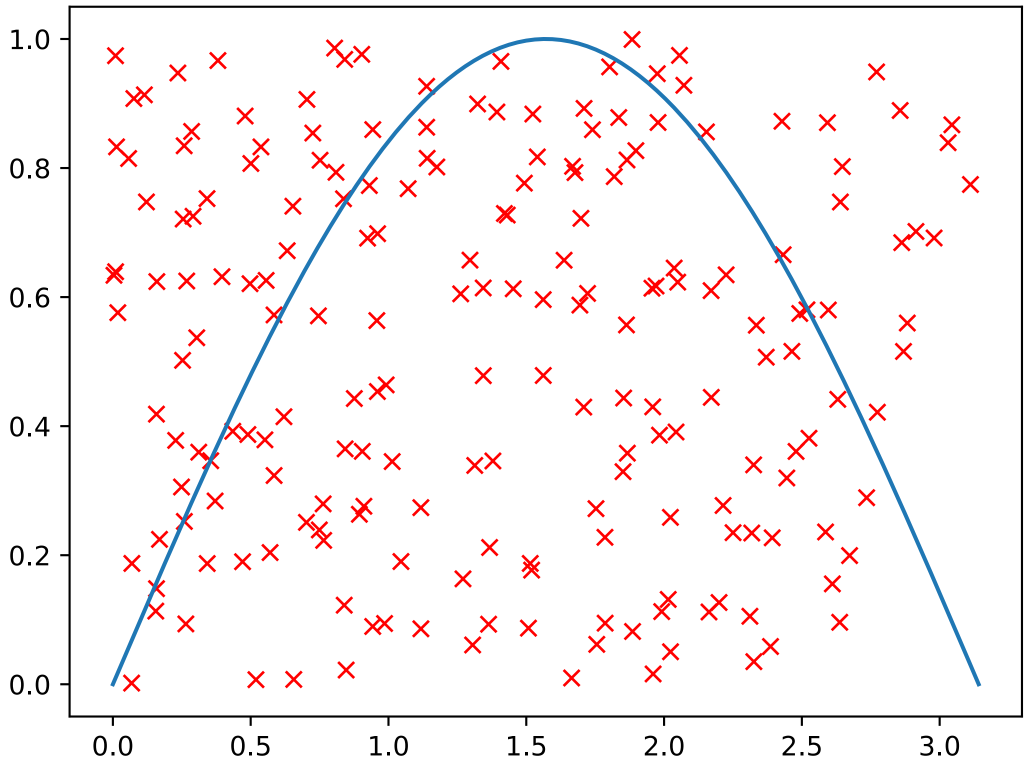
$$I = \int_a^b f(x)dx = F(b) - F(a)$$

como el cálculo simbólico de la integral $F(x)$ puede ser muy difícil, se utilizan métodos numéricos que aproximan su valor utilizando la interpretación geométrica de la integral definida que se corresponde con el área bajo la curva $f(x)$ entre a y b.

Dada una función $f(x)$ positiva en el intervalo $x \in [a; b]$ cuyo valor máximo es M dentro de ese intervalo, podemos definir un rectángulo de área $(b - a) \times M$ como el que se muestra en la figura para el intervalo $[0; 2]$. El método de Monte Carlo para el cálculo de la integral consiste en generar aleatoriamente puntos (en rojo en la figura) dentro de ese rectángulo y aproximar el valor de la integral por el porcentaje de puntos que caen por debajo de la función en cuestión:

$$I \approx \frac{N_{debajo}}{N_{total}}(b - a)M$$

donde N_{debajo} es el número de puntos (x, y) generados aleatoriamente cuya coordenada y es menor que el valor de la función $f(x)$ para ese valor de x y N_{total} es el número total de puntos generados aleatoriamente dentro del rectángulo.



Implementa en Python una función con la siguiente cabecera

```
def integra_mc(fun, a, b, num_puntos=10000)
```

que calcule la integral de fun entre a y b por el método de Monte Carlo antes descrito, generando para ello num_puntos aleatoriamente. Puedes comprobar la corrección del resultado obtenido, comparándolo con el de aplicar la función *scipy.integrate.quad* de Python.

Debes implementar dos versiones del algoritmo, una iterativa que realice num_puntos iteraciones para calcular el resultado, y otra que utilice operaciones entre vectores en lugar de bucles, comparando los tiempos de ejecución obtenidos con ambas versiones.