UNIVERSIDAD NACIONAL DE LOJA

PERIODO ACADÉMICO: ABRIL – SEPTIEMBRE 2019
PRACTICA # 9
ASIGNATURA: SIMULACIÓN
RESULTADO DE APRENDIZAJE DE LA PRÁCTICA: Integración numérica usando Python con la librería NumPy.

Nombre: Deiby Patricio Calva. **Fecha:** 24/02/2020
Ciclo: 6 “A”
Link Archivo: https://github.com/DeibyCalva/Laboratorio_en-_R_simulacion/blob/master/LABORATORIOS%20EN%20R/Practica_9DeibyCalva/class_Integrator.py

1. **TEMA: Simulación numérica: Integración.**
2. **OBJETIVOS:**
   - Comprende la forma de integrar usando el método de integración Monte Carlo.
   - Aplica la simulación para la resolución de problemas de integración.
3. **RECURSOS NECESARIOS:**
   - Python(NumPy), R.
   - Computador de Laboratorios
4. **INSTRUCCIONES:**
   - Prohibido consumo de alimentos
   - Prohibido equipo de diversión, celulares etc.
   - Prohibido jugar
   - Prohibido mover o intercambiar los equipos de los bancos de trabajo
   - Prohibido sacar los equipos del laboratorio sin autorización.
   - Ubicar los equipos y accesorios en el lugar dispuesto por el responsable del laboratorio, luego de terminar las prácticas.
   - Uso adecuado de equipos
5. **ACTIVIDADES POR DESARROLLAR:**

Aprender como escribir una clase Python a traves del link:
https://docs.python.org/2/tutorial/classes.html

Crea una clase “integrador” que integre numéricamente la función: $f(x) = x^{\&}e^{0}\sin(x)$

Debe proporcionar a la clase el valor mínimo **xMin**, el valor máximo **xMax** y el numero de pasos **N** para la integración. Luego, el proceso de integración debe llevarse a cabo de acuerdo con la siguiente información:

Suponga que: $S = \int_{xMin}^{xMax} f(x)dx \approx \sum_{i=0}^{N-1} f(x_i)\Delta x$ (tener en cuenta que la suma va hasta N-1)
$\Delta x = (xMax - xMin)/(N-1)$

$$x_i = xMin + i\Delta x$$

La clase esta compuesta de tres métodos: **_init_, integrate y show**

a. El método **_init_** debe inicializar el **xMin**, **xMax**, **N** y otros parámetros relacionados.

b. El método integrate debe realizar el proceso de integración con los parámetros dados

c. El método show debe imprimir en pantalla el resultado de la integración.

Asigne los parámetros con los valores: **xMin** =1, **xMax** =3, **N** = 200

El resultado de la integración de f(x) deben presentarlo con 5 decimales de exactitud. A continuación, se presenta la plantilla para la clase a crear.

```
import numpy as np import
math
 class Integrator:      def
__init_(self, xMin, xMax, N):
###############################

def integrate(self):
        ##################################
             def
show(self):
        ##################################

examp = Integrator(1,3,200)
examp.integrate() examp.show()
```

IMPLEMENTACIÓN

```
Practica_9DeibyCalva > class_Integrator.py > ...
import numpy as np
import math
class integrador:
    def __init__(self, xMin, xMax, N):
        self.xMin = xMin
        self.xMax = xMax
        self.N = N
        self.suma = 0.0

    def integrate(self):
        deltaX = (self.xMax-self.xMin) / (self.N-1)
        for i in range(200):
            xi = 1+(i*deltaX)
            xCuadrado = math.pow(xi, 2)
            euler = math.exp(-xi)
            senoX = math.sin(xi)
            self.suma += xCuadrado*euler*senoX*deltaX
        print(round(self.suma, 5))
    def show(self):
        return 0
examp = integrador(1,3,200)
examp.integrate()
examp.show()
```

```
0.7656
PS C:\Users\Usuario\Documents\6TO CICLO\SIMULACION\Laboratorio_en-_R_simulacion\LABORATORIOS EN R>
```

6. INVESTIGACIÓN COMPLEMENTARIA (a elaborar por el estudiante)

Construcción de la integral de Riemann.

Definición 9.1.1. Sea $I = [a,b] \subset$ R un intervalo cerrado y acotado (compacto).

Se llama partición de I a todo conjunto de puntos $P = \{x_0,x_1,\cdots,x_n\}$ de forma que $a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b$.

Se llama norma de la partición P, y se denotar ‘a por |P|, al m´aximo de los números $x_k - x_{k-1}$, con $k = 1,...,n$.

Denotaremos por P[a, b] (o más brevemente P, si no hay confusión posible con el intervalo) al conjunto de todas las particiones de [a, b].

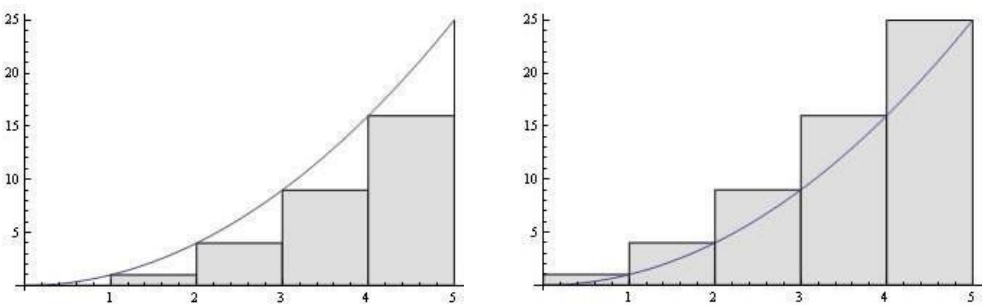

Figura 9.1: Suma inferior y superior de Riemann de la función $f(x) = x^2$ en el intervalo $I = [0,5]$ respecto de la partición $P = \{0,1,2,3,4,5\}$

Definición 9.1.2. Sea $f : [a, b] \to$ R una función acotada y sea $P \in$ P[a, b] con $P = \{a = x_0 < x_1 < \cdots < x_n = b\}$. Sean

$$m_k := \inf\{f(x) : x_{k-1} \le x \le x_k\} = \inf_{x\in[x_{k-1},x_k]} \{f(x)\}$$

$$M_k := \sup\{f(x) : x_{k-1} \le x \le x_k\} = \sup_{x\in[x_{k-1},x_k]} \{f(x)\}.$$

Se llaman, respectivamente, Suma inferior y suma superior de Riemann de la función f relativas a la partición P a las siguientes sumas:

$$L(f,P) := \sum_{k=1}^{n} m_k\,(x_k, x_{k-1}) \qquad U(f,P) := \sum_{k=1}^{n} M_k\,(x_k, x_{k-1}).$$

Nota 9.1.3. Para cada $P \in$ P y cada función f, es claro que $L(f, P) \le U(f, P)$

Definición 9.1.4. Sean $P,Q \in$ P[a, b]. Se dice que Q es más fina que P (o que P es menos fina que Q), y se denotara $P \preceq Q$, cuando $P \subset Q$.

Teorema 9.1.5. Sean $P,Q \in$ P[a, b] con $P \preceq Q$. Entonces

$$L(f, P) \le L(f, Q) \quad \text{y} \quad U(f, Q) \le U(f,P).$$

Teorema 9.1.6. Dadas P, Q ∈ P[a, b], se cumple que $L(f, P) \leq U(f, Q)$.

Corolario 9.1.7.

(a) El conjunto $\{L(f, P) : P \in P[a, b]\}$ está acotado superiormente.

(b) El conjunto $\{U(f, P) : P \in P[a, b]\}$ está acotado inferiormente.

Definición 9.1.8.

(a) Se llama integral inferior de Riemann, y se denotará por $\underline{\int_a^b} f(x)\,dx$, al supremo del conjunto de sumas inferiores.

(b) Se llama integral superior de Riemann, y se denotará por $\overline{\int_a^b} f(x)\,dx$, al ´ínfimo del conjunto de sumas superiores.

Nota 9.1.9. Se cumple que $\underline{\int_a^b} f(x)\,dx \leq \overline{\int_a^b} f(x)\,dx$.

Definición 9.1.10. Sea $f : [a, b] \to R$ una función acotada. Se dice que f es integrable Riemann en [a, b], cuando $\underline{\int_a^b} f(x)\,dx = \overline{\int_a^b} f(x)\,dx$. Al este valor común se le llamará integral de Riemann de f en el intervalo [a ,b] y se denotara por $\int_a^b f(x)\,dx$.

Al conjunto de todas las funciones integrables Riemann en un intervalo [a, b] se le denotara por R[a, b].

Ejemplo 9.1.11. Sea $f : [0,1] \to R$ dada por

$$f(x) := \chi_{[0,1]\cap\mathbb{Q}}(x) = \begin{cases} 1 & \text{si } x \in [0,1] \cap Q \\ 0 & \text{si } x \notin [0,1] \cap Q \end{cases}$$

Entonces $f \notin R[0,1]$.

Definición 9.1.12. Sea $f : [a, b] \to R$ una función acotada, $f \in R[a, b]$ con $f(x) \geq 0$ $\forall x \in [a, b]$. Consideremos el conjunto

$$S := \{(x, y) \in R^2 : a \leq x \leq b, 0 \leq y \leq f(x)\}.$$

Se define el área de S como $A(S) = \int_a^b f(x)dx$.

Si fuese $f(x) \leq 0$ $\forall x \in [a, b]$, entonces, por simetría, el ´área de S (sustituyendo f(x) por −f(x)) sería $A(S) = -\int_a^b f(x)\,dx$.

Teorema 9.1.13 (Condición de integrabilidad de Riemann)**.** Sea $f : [a, b] \to R$ acotada. $f \in R[a, b]$

$$\iff \forall \varepsilon > 0, \exists P \in P[a,b] \text{ tal que } U(f,P) - L(f,P) < \varepsilon.$$

Corolario 9.1.14. Si $f \in R[a,b]$, su integral es el único número real que cumple lo siguiente

$$L(f,P) \leq \int_a^b f(x)\,dx \leq U(f,Q) \qquad \forall P,Q \in P[a,b].$$

Teorema 9.1.15. Sea $f : [a, b] \to R$ acotada. Entonces, $f \in R[a,b]$ si y solo si existe una sucesión $\{P_n\}_n \subset P[a, b]$ tal que $\lim_{n\to\infty} [U(f, P_n) - L(f, P_n)] = 0$.

Además, en ese caso, se cumple además que

$$\int_a^b f(x)\,dx = \lim_{n\to\infty} U(f,P_n) = \lim_{n\to\infty} L(f,P_n).$$

Nota 9.1.16.

(a) En la practica, se suele tomar $P_n \in P[a, b]$ la partición del intervalo $[a, b]$ en n partes iguales, es decir, $P_n = \left\{a + \frac{k}{n}(b-a) : \ k = 0,1,\dots,n\right\}$.

(b) Además, si para cada $k = 1,\cdots,n$ seleccionamos $t_k \in [x_{k-1},x_k]$, se cumple que

$$\lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} f(t_k) = \frac{1}{b-a} \int_a^b f(x)\,dx$$

7. DISCUSIÓN

En esta práctica desarrollada para el aprendizaje de algoritmos de simulación basados en conceptos de integración, una de actividades que se desarrolló es la investigación de como crear clases en Python, ya que es lenguaje de programación seleccionado, facilita portabilidad y el acceso a las librerías necesarias para la representación de cálculos. Las principal librería de Python usada en este ejercicio es: **NumPy** ya que es una extensión de Python, que le agrega mayor soporte para vectores y matrices, constituyendo una biblioteca de funciones matemáticas de alto nivel para operar con esos vectores o matrices..

8. CONCLUSIONES

- El método Montecarlo es útil para establecer probabilidades y definir escenarios de actuación.
- El objetivo de este método no es el de brindar decisiones sino apoyar a la toma de estas.
- Es una técnica cuantitativa utilizada para obtener la respuesta más probable de un evento por medio de la simulación de un modelo matemático.

9. RECOMENDACIONES

- Se debe conocer bien la definición y las fórmulas de la integración de numérica, para poder diseñar el programa correctamente.
- Tener conocimiento de las librerías y funciones que Python ofrece para poder implementarlas eficientemente.

BIBLIOGRAFÍA:

- BARTLE et al. *Introducción al Análisis Matemático de una Variable* (*Introduction to Real Analysis*), trad., ed. Limusa S.A. 2009.
- KURTZ et al.*Theories of Integration The Integrals of Riemann, Lebesgue, Henstock-Kurzweil and McShane*, ed. World Scientific Publishing Co. Pte. Ltd. 2004.