Universidad Autónoma de Querétaro

Facultad de Ingeniería

Integración numérica

Análisis numérico

Autor:
David Gómez Torres

27 Octubre del 2021

ÍNDICE GENERAL

CAPÍTULO 1

INTRODUCCIÓN

1.1. La regla del trapecio

La regla del trapecio es la primera de las fórmulas cerradas de integración de Newton-Cotes. Corresponde al caso donde el polinomio de la ecuación es de primer grado:

$$I = \int_a^b f(x)dx \cong \int_a^b f_1(x)dx \tag{1.1}$$

Sabemos que una línea recta se puede representar como

$$I = \int_a^b \left[f(a) + \frac{f(b) - f(a)}{b - a}(x - a) \right] \tag{1.2}$$

El resultado de la integración

$$I = (b - a)\frac{f(a) + f(b)}{2} \tag{1.3}$$

que se denomina **regla del trapecio**.

Geométricamente, la regla del trapecio es equivalente a aproximar el área del trapecio bajo la línea recta que une $f(a)$ y $f(b)$ en la figura (1.1).

Por lo tanto, la integral aproximada se representa como

$$I \cong \text{ancho} \times \text{altura promedio} \tag{1.4}$$

o

$$I \cong (b - a) \times \text{altura promedio} \tag{1.5}$$

donde, para la regla del trapecio, la altura promedio es el promedio de los valores de la función en los puntos extremos.

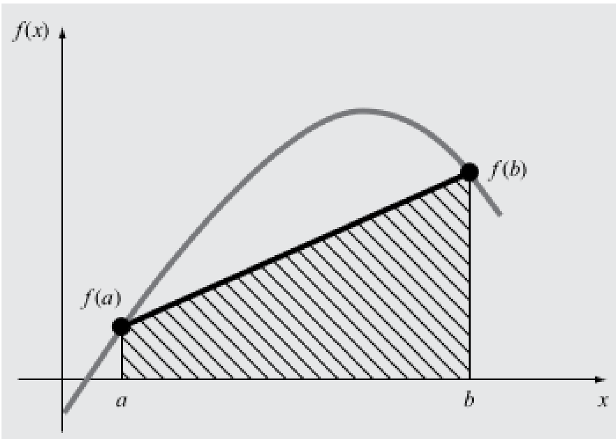


Figura 1.1: Representación de la regla de trapecio

1.1.1. Error en la regla del trapecio

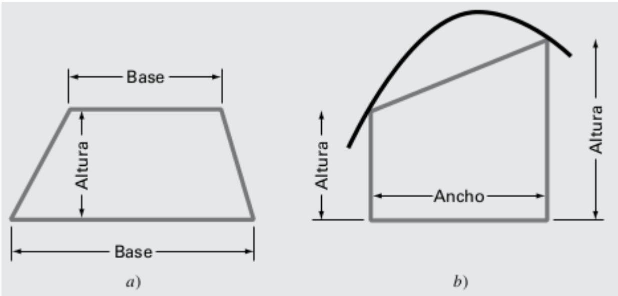


Figura 1.2: Error del método

Cuando empleamos la integral bajo un segmento de línea recta para aproximar la integral bajo una curva, obviamente se tiene un error que puede ser importante (1.2). Una estimación al error de truncamiento local para una sola aplicación de la regla del trapecio es

$$E_t = -\frac{1}{12} f''(\xi)(b-a)^3 \tag{1.6}$$

1.2. Reglas de Simpson

Una forma de obtener una estimación más exacta de una integral consiste en usar polinomios de grado superior para unir los puntos. Ejemplo, Si hay dos puntos igualmente espaciados entre f(a) y f(b), los cuatro puntos se pueden unir mediante un polinomio de tercer grado.
Las fórmulas que resultan de tomar las integrales bajo esos polinomios se conocen como **reglas de Simpson**.

1.2.1. Regla de Simpson 1/3

La regla de Simpson 1/3 resulta cuando un polinomio de interpolación de segundo grado se sustituye en la ecuación:
Después de la integración y de las manipulaciones algebraicas, se obtiene al siguiente fórmula:

$$I \cong \frac{h}{3}[f(x_0) + 4f(x_1) + f(x_2)] \tag{1.7}$$

donde, en este caso, $h = (b-a)/2$.

La regla de Simpson 1/3 también se puede expresar en el siguiente formato:

$$I \cong \underbrace{(b-a)}_{\text{Ancho}} \underbrace{\frac{f(x_0) + 4f(x_1) + f(x_2)}{6}}_{\text{Altura promedio}} \tag{1.8}$$

donde a $= x_0$, b $= x_2$ y $x_1 =$ el punto a la mitad entre a y b, que está dado por $(b+a)/2$.

Se puede demostrar que la aplicación a un solo segmento de la regla de Simpson 1/3 tiene un error de truncamiento de

$$E_t = -\frac{1}{90} h^5 f^{(4)}(\xi) \tag{1.9}$$

$$E_t = -\frac{(b-a)^5}{2880} f^{(4)(\xi)} \tag{1.10}$$

donde ξ está en algún lugar en el intervalo de a a b.

1.2.2. Regla de Simpson de 3/8

De manera similar a la obtención de la regla del trapecio y Simpson 1/3, es posible ajustar un polinomio de Lagrange de tercer grado a cuatro puntos e integrarlo:

$$I = \int_a^b f(x)dx \cong \int_a^b f_3(x)dx$$

para obtener

$$I \cong \frac{3h}{8}[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)]$$

donde h = (b − a)/3.

Esta ecuación se llama **regla de Simpson 3/8** debido a que h se multiplica por 3/8. Ésta es la tercera fórmula de integración cerrada de Newton-Cotes. La regla 3/8 se expresa también en

$$I \cong \underbrace{(b-a)}_{\text{ancho}} \underbrace{\frac{f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)}{8}}_{alturapromedio} \tag{1.11}$$

Así los dos puntos interiores tienen pesos de tres octavos, mientras que los puntos extremos tienen un peso de un octavo. La regla de Simpson 3/8 tiene un error de

$$E_t = -\frac{3}{80}h^5 f^{(4)}(\xi) \tag{1.12}$$

$$E_t = -\frac{(b-a)^5}{6480} f^{(4)}(\xi) \tag{1.13}$$

1.3. Fórmulas de Newton-Cotes

Considere que, como en el caso de las reglas de Simpson 1/3 y 3/8, las fórmulas de cinco y seis puntos tienen el mismo orden de error. Esta característica general se satisface para fórmulas con más puntos y lleva al resultado de que las fórmulas con segmentos pares y puntos impares (por ejemplo, la regla 1/3 y la regla de Boole) usualmente son los métodos de preferencia.

Segmentos (n) height	**Puntos**	**Nombre**	**Fórmula**
2	3	Regla de Simpson 1/3	$(b-a)\frac{f(x_0) + 4f(x_1) + f(x_2)}{6}$
3	4	Regla de Simpson 3/8	$(b-a)\frac{f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)}{8}$
4	5	Regla de Boole	$(b-a)\frac{7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4)}{90}$
5	6		$(b-a)\frac{19f(x_0) + 75f(x_1) + 50f(x_2) + 50f(x_3) + 75f(x_4) + 19f(x_5)}{288}$

CAPÍTULO 2

METODOLOGÍA

2.1. Problema 21.3: Integral por distintos métodos

Problema 21.3

Evalúe la integral siguiente:

$$\int_{-2}^{4} (1 - x - 4x^3 + 2x^5)dx$$

a) en forma analítica;

b) con una sola aplicación de la regla del trapecio;

c) con la regla del trapecio compuesta, con $n = 2$ y 4;

d) con una sola aplicación de la regla de Simpson 1/3;

e) con la regla de Simpson 3/8, y

f) con la regla de Boole.

g) Para cada una de las estimaciones numéricas de los incisos b) a f), determine el error relativo porcentual con base en el inciso a).

a) De forma analítica:

$$\begin{aligned}\int_{-2}^{4} (1 - x - 4x^3 + 2x^5)dx &= \left[x - \frac{x^2}{2} - x^4 + 2\frac{x^6}{6}\right]_{-2}^{4} \\ &= \left[(-2-4) - \frac{(-2-4)^2}{2} - (-2-4)^4 + 2\frac{(-2-4)^6}{6}\right] \\ &= 1104\end{aligned}$$

b) Regla del trapecio

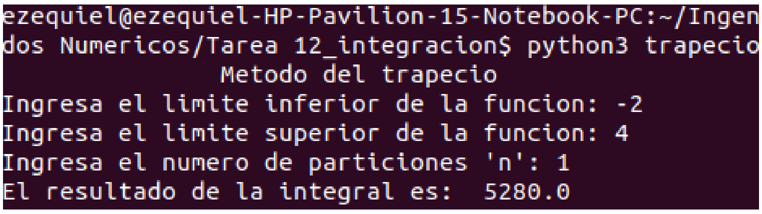
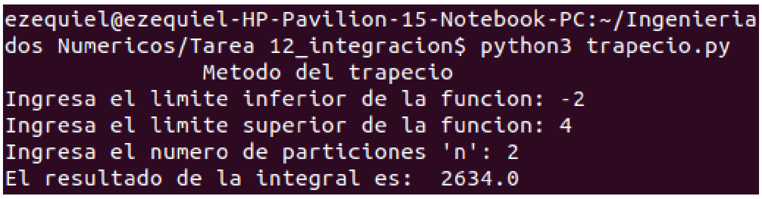


Figura 2.1: Integral con método del trapecio simple

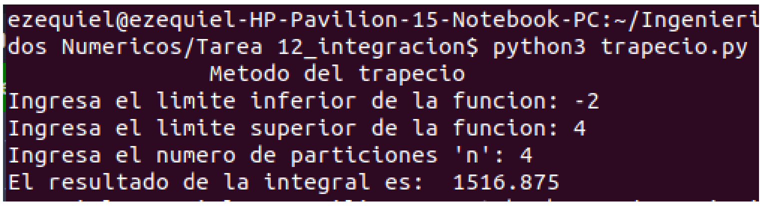
```
import math

#declarar la funcion
f = lambda x:(1-x-4*x**3+2*x**5)

#inicializar el contador i y la suma
i = 0
s = 0

cadena = "Metodo del trapecio".capitalize()
print(cadena.center(50, " "))

    #pedir al usuario "a"
a = float(input("Ingresa el limite inferior de la funcion: "))
    #pedir al usuario "b"
b = float(input("Ingresa el limite superior de la funcion: "))

def trapecio(a,b,n):
#calculo de delta x
    delta = (b-a)
    integral = delta*(f(a)+f(b))/2

#imprimir el resultado en pantalla

    print("El resultado de la integral es: ", integral)

trapecio(a,b,n)
```

c) Regla del trapecio compuesta



```
ezequiel@ezequiel-HP-Pavilion-15-Notebook-PC:~/Ingenieria
dos Numericos/Tarea 12_integracion$ python3 trapecio.py
                 Metodo del trapecio
Ingresa el limite inferior de la funcion: -2
Ingresa el limite superior de la funcion: 4
Ingresa el numero de particiones 'n': 2
El resultado de la integral es:  2634.0
```

Figura 2.2: Método con 2 iteraciones



```
ezequiel@ezequiel-HP-Pavilion-15-Notebook-PC:~/Ingenieri
dos Numericos/Tarea 12_integracion$ python3 trapecio.py
                 Metodo del trapecio
Ingresa el limite inferior de la funcion: -2
Ingresa el limite superior de la funcion: 4
Ingresa el numero de particiones 'n': 4
El resultado de la integral es:  1516.875
```

Figura 2.3: Método con 4 iteraciones

```python
import math

#declarar la funcion
f = lambda x:(1-x-4*x**3+2*x**5)
#inicializar el contador i y la suma
i = 0
s = 0

cadena = "Metodo del trapecio".capitalize()
print(cadena.center(50, " "))

a = float(input("Ingresa el limite inferior de la funcion: "))
b = float(input("Ingresa el limite superior de la funcion: "))
n = int(input("Ingresa el numero de particiones 'n': "))

def trapecio(a,b,n):
    delta = (b-a)/n #calculo de delta x
    sumatoria = (f(a)+f(b))/2
    #sumatoria del metodo usan un ciclo for
    for i in range(1,n):
        sumatoria += f(a + i*delta)

#calculo del metodo completo
        integral = delta * sumatoria
#imprimir el resultado en pantalla
    print("El resultado de la integral es: ", integral)

trapecio(a,b,n)
```

d) Regla de Simpson 1/3

```
ezequiel@ezequiel-HP-Pavilion-15-Notebook-PC:~/Ingenieria
dos Numericos/Tarea 12_integracion$ python3 simpson13.py
               Regla de simpson 1/3
Ingrese el limite inferior de integracion: -2
Ingrese el limite superior de integracion: 4
Numero de iteraciones 'n': 1
El resultado de la integral es:  3520.0
```

Figura 2.4: Método con 1 sola iteraciones

```python
import math

cadena = "Regla de Simpson 1/3".capitalize()
print(cadena.center(50, " "))

a = float(input("Ingrese el limite inferior de integracion: "))
b = float(input("Ingrese el limite superior de integracion: "))
n = int(input("Numero de iteraciones 'n': "))

def simpson(a,b,n):
    sum = 0
    int = 0
    h = (b-a)/n #delta

#regla de simpson1/3
    for i in range(n+1):
        x = (a + (i*h))
#funcion
        f = 1-x-4*x**3+2*x**5
#casos
        if (x == a or x == b):
            sum += f
        elif (i % 2 != 0):
            f = 4*f
            sum += f
        elif (i % 2 == 0):
            f = 2*f
            sum += f

    int = sum*(h/3)

    print("El resultado de la integral es: ", int)

simpson(a,b,n)
```

e) Regla de Simpson 3/8

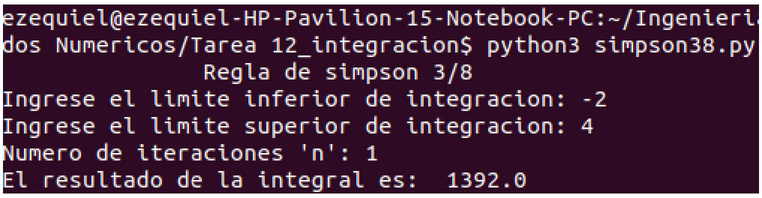
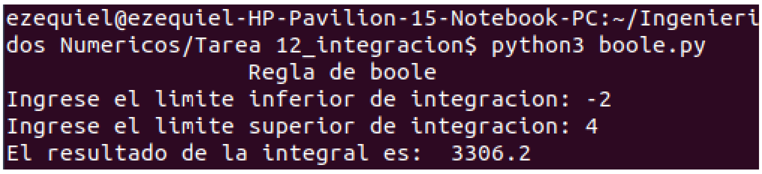


Figura 2.5: Método con 1 sola iteraciones

```
import math

cadena = "Regla de Simpson 3/8".capitalize()
print(cadena.center(50, " "))

a = float(input("Ingrese el limite inferior de integracion: "))
b = float(input("Ingrese el limite superior de integracion: "))
n = int(input("Numero de iteraciones 'n': "))

def simpson38(a,b,n):
#funcion
    #f = lambda x:(4*x-3)**3
    f = lambda x:1-x-4*x**3+2*x**5
#delta
    h = (b-a)/n
#puntos medios
    m1 = (2*a+b)/3
    m2 = (a+2*b)/3
#regla de simpson
    integral = (b-a)/8 * (f(a) + 3*f(m1)+ 3*f(m2) + f(b))
    for i in range(n+1):
        sum = 0
        b = a+h
        area = integral
        sum += integral
        a = b

    print("El resultado de la integral es: ", integral)

simpson38(a,b,n)
```

f) Regla de Boole



```
ezequiel@ezequiel-HP-Pavilion-15-Notebook-PC:~/Ingenieri
dos Numericos/Tarea 12_integracion$ python3 boole.py
                  Regla de boole
Ingrese el limite inferior de integracion: -2
Ingrese el limite superior de integracion: 4
El resultado de la integral es:  3306.2
```

Figura 2.6: Método con 1 sola iteraciones

```python
import math

cadena = "Regla de Boole".capitalize()
print(cadena.center(50, " "))

a = float(input("Ingrese el limite inferior de integracion: "))
b = float(input("Ingrese el limite superior de integracion: "))

def boole(a,b):

#funcion
    f = lambda x:1-x-4*x**3+2*x**5
#puntos medios
    m1 = (2*a+b)/4
    m2 = (a+2*b)/4
    m3 = a+b/2
#regla de boole
    integral = 2*(b-a)/45 * (7*f(a) + 32*f(m1)+ 12*f(m2) +
    32*f(m3) + 7*f(b))

    print("El resultado de la integral es: ", integral)

boole(a,b)
```

Error en la estimación

1. Método de trapecio simple

```
                Metodo del trapecio
Ingresa el limite inferior de la funcion: -2
Ingresa el limite superior de la funcion: 4
El resultado de la integral es:  5280.0
El error relativo porcentual es: 41.76 %
```

2. Método de trapecio con $n = 2$

```
                Metodo del trapecio
Ingresa el limite inferior de la funcion: -2
Ingresa el limite superior de la funcion: 4
Ingresa el numero de particiones 'n': 2
El error relativo porcentual es: 14.49 %

El error relativo porcentual es: 14.43 %

El error relativo porcentual es: 68.1 %

El resultado de la integral es:  7914.0
```

3. Método del trapecio con $n = 4$

```
               Metodo del trapecio
Ingresa el limite inferior de la funcion: -2
Ingresa el limite superior de la funcion: 4
Ingresa el numero de particiones 'n': 4
El error relativo porcentual es: 1.725 %

El error relativo porcentual es: 1.7540625 %

El error relativo porcentual es: 1.7240625 %

El error relativo porcentual es: 3.69375 %

El error relativo porcentual es: 30.52875 %

El resultado de la integral es:  4156.875
```

4. Regla de Simpson 1/3

```
                Regla de simpson 1/3
Ingrese el limite inferior de integracion: -2
Ingrese el limite superior de integracion: 4
Numero de iteraciones 'n': 4
El error relativo porcentual es: 11.33 %

El error relativo porcentual es: 11.2525 %

El error relativo porcentual es: 11.2925 %

El error relativo porcentual es: 6.04 %

El error relativo porcentual es: 11.85 %

El resultado de la integral es:  1144.5
```

5. Regla de Simpson 3/8

```
                Regla de simpson 3/8
Ingrese el limite inferior de integracion: -2
Ingrese el limite superior de integracion: 4
Numero de iteraciones 'n': 4
El error relativo porcentual es: 2.88 %

El error relativo porcentual es: 2.88 %

El error relativo porcentual es: 2.88 %

El error relativo porcentual es: 2.88 %

El error relativo porcentual es: 2.88 %

El resultado de la integral es:  1392.0
```

6. Regla de Boole simple

```
                   Regla de boole
Ingrese el limite inferior de integracion: -2
Ingrese el limite superior de integracion: 4
El error relativo porcentual es: 22.022 %

El resultado de la integral es:  3306.2
```

Figura 2.7: Método con 1 sola iteraciones

2.2. Problema 21.5: Regla de la cadena

Problema 21.5

Integre la función siguiente en forma tanto analítica como con la regla de Simpson, con $n = 4$ y 5. Analice los resultados.

$$\int_{-3}^{5} (4x-3)^3 dx$$

a) De forma analítica:

Aplicando la sustitución de $u = 4x - 3$

$$\begin{aligned}
\int_{-3}^{5} (4x-3)^3 dx &= \int_{-15}^{17} \frac{u^3}{4} du \\
&= \frac{1}{4} \left. \frac{u^4}{4} \right|_{15}^{17} \\
&= \frac{1}{4}(8224) = 2056
\end{aligned}$$

b) Con $n = 4$:

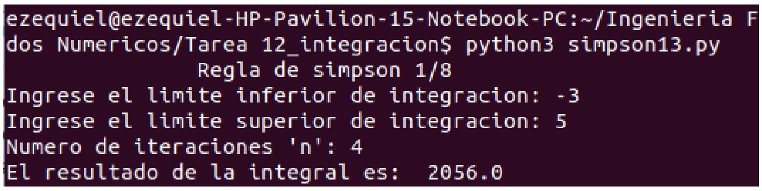

```
ezequiel@ezequiel-HP-Pavilion-15-Notebook-PC:~/Ingenieria F
dos Numericos/Tarea 12_integracion$ python3 simpson13.py
                Regla de simpson 1/8
Ingrese el limite inferior de integracion: -3
Ingrese el limite superior de integracion: 5
Numero de iteraciones 'n': 4
El resultado de la integral es:  2056.0
```

Figura 2.8: Método con 4 iteraciones

c) Con $n = 5$:

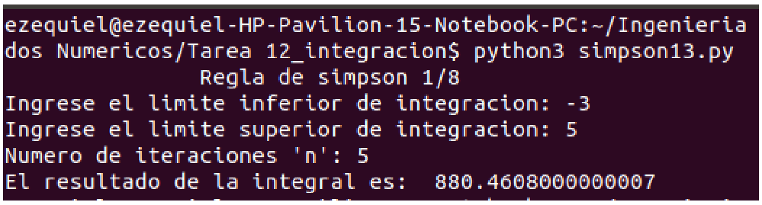

```
ezequiel@ezequiel-HP-Pavilion-15-Notebook-PC:~/Ingenieria
dos Numericos/Tarea 12_integracion$ python3 simpson13.py
                Regla de simpson 1/8
Ingrese el limite inferior de integracion: -3
Ingrese el limite superior de integracion: 5
Numero de iteraciones 'n': 5
El resultado de la integral es:  880.4608000000007
```

Figura 2.9: Método con 5 iteraciones

Analizando la gráfica, vemos que la función crece muy rápidamente, en este caso con 4 iteraciones, llegamos al resultado mientras que con 5 iteraciones, el método hace una subestimación del resultado.

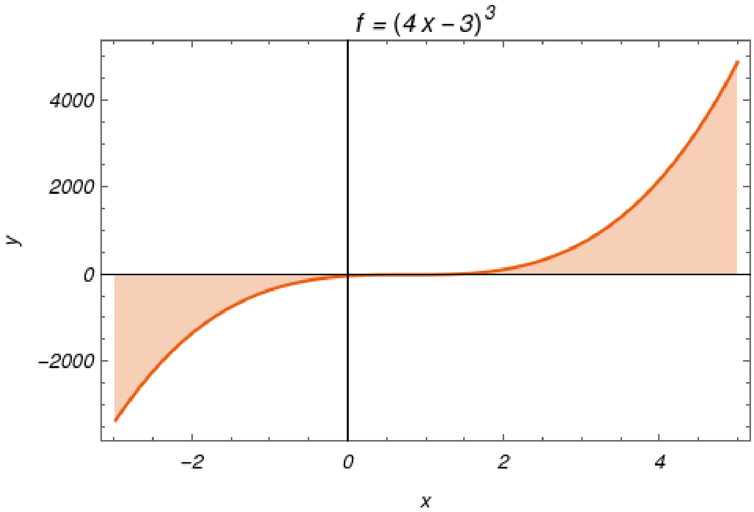


Figura 2.10: Gráfica de la integral

```python
import math

a = float(input("Ingrese el limite inferior de integracion: "))
b = float(input("Ingrese el limite superior de integracion: "))
n = int(input("Numero de iteraciones 'n': "))

def simpson(a,b,n):
    sum = 0
    int = 0
    h = (b-a)/n #delta
#regla de simpson1/3
    for i in range(n+1):
        x = (a + (i*h))
#funcion
        f = (4*x-3)**3
#casos
        if (x == a or x == b):
            sum += f
        elif (i % 2 != 0):
            f = 4*f
            sum += f
        elif (i % 2 == 0):
            f = 2*f
            sum += f
    int = sum*(h/3)
    print("El resultado de la integral es: ", int)

simpson(a,b,n)
```

2.3. Problema 21.11: Integral por tabla de datos

Problema 21.11

Evalúe la integral de los datos que se tabula en seguida, con

a) a) la regla del trapecio y

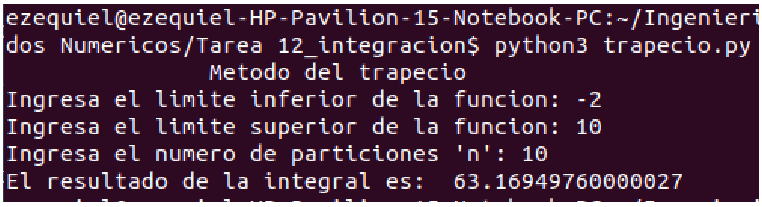
b) las reglas de Simpson

x	-2	0	2	4	6	8	10
f(x)	35	5	-10	2	5	3	20

Si le interpolamos un polinomio de Lagrange a los datos, tenemos

$$-3x^6/1280 + 25x^5/384 - 113x^4/192 + 143x^3/96 + 64x^2/15 - 73x/4 + 5$$

a) Regla del trapecio:



```
ezequiel@ezequiel-HP-Pavilion-15-Notebook-PC:~/Ingenieri
dos Numericos/Tarea 12_integracion$ python3 trapecio.py
               Metodo del trapecio
Ingresa el limite inferior de la funcion: -2
Ingresa el limite superior de la funcion: 10
Ingresa el numero de particiones 'n': 10
El resultado de la integral es:  63.16949760000027
```

Figura 2.11: Integral con método del trapecio

```
import math

#declarar la funcion
f = lambda x:-3*x**6/1280 + 25*x**5/384 - 113*x**4/192 +
143*x**3/96 + 64*x**2/15 - 73*x/4 + 5
#inicializar el contador i y la suma
i = 0
s = 0

cadena = "Metodo del trapecio".capitalize()
print(cadena.center(50, " "))

    #pedir al usuario "a"
a = float(input("Ingresa el limite inferior de la funcion: "))
    #pedir al usuario "b"
b = float(input("Ingresa el limite superior de la funcion: "))

    #pedir al usuario "n"
n = int(input("Ingresa el numero de particiones 'n': "))

def trapecio(a,b,n):
#calculo de delta x
    delta = (b-a)/n
    sumatoria = (f(a)+f(b))/2

    #sumatoria del metodo usan un ciclo for
    for i in range(1,n):

        sumatoria += f(a + i*delta)

#calculo del metodo completo
        integral = delta * sumatoria

#imprimir el resultado en pantalla

    print("El resultado de la integral es: ", integral)

trapecio(a,b,n)
```

b) Regla de Simpson:

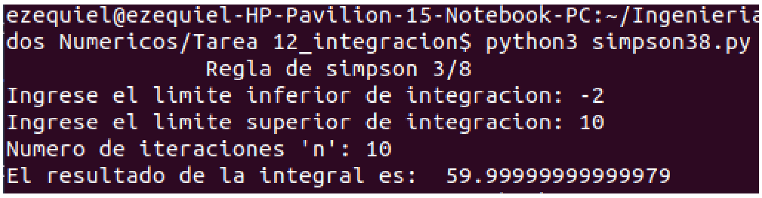


Figura 2.12: Integral con método de Simpson 3/8

```python
import math

cadena = "Regla de Simpson 3/8".capitalize()
print(cadena.center(50, " "))

a = float(input("Ingrese el limite inferior de integracion: "))
b = float(input("Ingrese el limite superior de integracion: "))
n = int(input("Numero de iteraciones 'n': "))

def simpson38(a,b,n):
#funcion
    f = lambda x:-3*x**6/1280 + 25*x**5/384 - 113*x**4/192 +
    143*x**3/96 + 64*x**2/15 - 73*x/4 + 5
#delta
    h = (b-a)/n
#puntos medios
    m1 = (2*a+b)/3
    m2 = (a+2*b)/3
#regla de simpson
    integral = (b-a)/8 * (f(a) + 3*f(m1)+ 3*f(m2) + f(b))
    for i in range(n+1):
        sum = 0
        b = a+h
        area = integral
        sum += integral
        a = b

    print("El resultado de la integral es: ", integral)

simpson38(a,b,n)
```

2.4. Problema 21.19: aplicación de la integral

Problema 21.19

Se recolectaron los siguientes datos para una sección transversal de un río (y = distancia de una ribera, H = profundidad y U = velocidad):

y, m	0	1	3	5	7	8	9	10
H, m	0	1	1.5	3	3.5	3.2	2	0
U, m/s	0	0.1	0.12	0.2	0.25	0.3	0.15	0

Use integración numérica para calcular

a) la profundidad promedio,

b) el área de la sección transversal,

c) la velocidad promedio y

d) el caudal.

Observe que el área de la sección transversal (A_c) y el caudal (Q) se pueden calcular como

$$A_c = \int_0^y H(y)dy, \qquad Q = \int_0^y H(y)U(y)dy$$

a) la profundidad promedio,

```
              Metodo del trapecio
Ingresa el limite inferior de la funcion: 0
Ingresa el limite superior de la funcion: 10
Ingresa el numero de particiones 'n': 10
El resultado de la profundidad es:  25.396428571415065
```

b) el área de la sección transversal,

```
              Metodo del trapecio
Ingresa el limite inferior de la funcion: 0
Ingresa el limite superior de la funcion: 10
Ingresa el numero de particiones 'n': 20
El resultado de seccion transversal es:  21.168048967562125
```

c) la velocidad promedio y

```
                Metodo del trapecio
Ingresa el limite inferior de la funcion: 0
Ingresa el limite superior de la funcion: 10
Ingresa el numero de particiones 'n': 10
El resultado de la velocidad es:  1.3602678571395284
```

d) el caudal.

```
                Metodo del trapecio
Ingresa el limite inferior de la funcion: 0
Ingresa el limite superior de la funcion: 10
Ingresa el numero de particiones 'n': 10
El resultado del caudal es: 5.013641
```

CAPÍTULO 3

BIBLIOGRAFÍA

1. Chapra, S. C., Canale, R. P. (2011). *Numerical methods for engineers* (Vol. 1221). New York: Mcgraw-hill.