

Méthodes et simulation numériques - langage Python

IPSA - Semestre 3

H. El-Otmany

ATER & docteur en mathématiques appliquées
Département Techniques de commercialisation
IUT de Tarbes

Email personnel : hamou.elotmany@gmail.com

Email professionnel : hammou.el-otmany@iut-tarbes.fr

Web site : www.hamoelotmany.github.io

Tarbes, 12 Janvier 2022

*"Composition en caractère Cochin sur un ordinateur Apple MacBook Pro 15 à l'aide des logiciels libres suivants : $\text{\LaTeX}2_{\epsilon}$
 \pdfTeX Xfig Grace*

version : Cours_IUT_TC_20210923.tex du 9 février 2022 Copyright ©2021 by H. El-Otmany

Outline d'exposé

❶ Chapitre 1 : Intégration numérique

- Méthode rectangle
- Méthode du point milieu
- Méthode de trapèze
- Méthode de Simpson
- Mise en application en TP par Python

❷ Chapitre 2 : Introduction à la résolution d'EDO

- Méthode d'Euler
- Méthode de Range-Kutta
- Mise en application en TP par Python

❸ Chapitre 3 : Simulation de variables aléatoires

- Génération des variables suivant la fonction densité
- Simulation des variables aléatoires par la fonction inverse
- Simulation de la loi normale
- Simulation du mouvement Brownien

❹ Chapitre 4 : Algèbre linéaire et résolution des systèmes linéaires

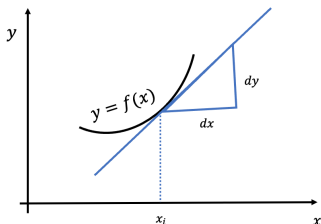
- Multiplication de matrice par vecteur
- Méthode du Pivot de Gauss
- Mise en application en TP par Python

Chapitre 1 : Intégration numérique

- ❶ Introduction et motivations
- ❷ Intégration par quadratique simple (Rectangle, Point milieu, Trapèze, Simpson)
- ❸ Méthodes composites

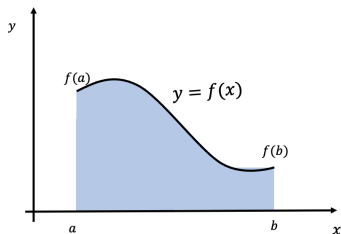
Dérivation et intégration numériques

Différentier : préciser la vitesse à laquelle une courbe change en un certain point de l'équation



Ceci revient à calculer la dérivée de f .

Intégrer : calculer l'aire de (la surface sous la courbe)



Ceci revient à calculer
$$I_{(a,b)}(f) = \int_a^b f(x)dx.$$

Introduction et position du problème

Introduction et position du problème

On s'intéresse à l'intégrale de la fonction continue f sur l'intervalle $[a, b]$:

$$I_{(a,b)}(f) = \int_a^b f(x)dx$$

☰ Problématique :

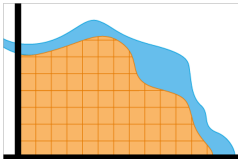
- En pratique, on ne connaît pas forcément la formule explicite de f (force source en géophysique, tomographie sismique, ...).
- Le calcul analytique est long et très compliqué (ex. $\int_0^{\pi/2} \cos(x^2)dx$).
- La plupart des fonctions n'admettent pas de primitives pouvant s'exprimer à l'aide de fonctions élémentaires (ex. $\int_0^{\pi/2} \sqrt{1 + \cos x^2}dx$).
- La primitive de f est connue mais elle n'est pas une fonction élémentaire (ex. $\int e^{-x^2} dx$).
- f n'est pas donnée par une forme explicite mais seulement par un nombre fini de points $(x_i; y_i)$, $0 \leq i \leq n$.

☰ Objectifs et solution au problème :

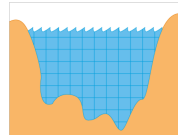
- approcher de façon numérique la valeur d'intégrale $I_{(a,b)}(f)$ (c'est-à-dire calculer numériquement la surface à partir d'un nombre fini d'appels à la fonction f).
- cas particulier lorsque l'on a accès à un échantillonnage régulier ou uniforme.

Applications

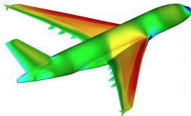
- Calculer l'aire d'un champ limité par une rivière et deux routes



- Calculer l'aire de la coupe transversale d'une rivière pour calculer le débit



- Calculer l'aire d'une aile d'avion pour simuler son profil



- Calculer le volume de la conduite pour avoir le débit volumique d'un fluide

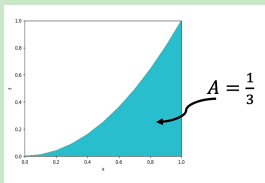
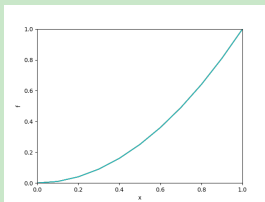


Exemple

Soit f une fonction continue sur $[0; 1]$ telle que $f(x) = x^2$.

On a facilement

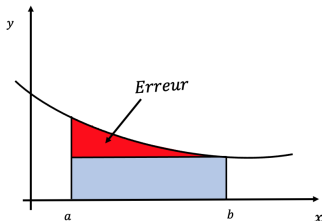
$$I_{(0,1)}(f) = \int_0^1 x^2 dx = \frac{1}{3} \approx 0.3333\dots$$



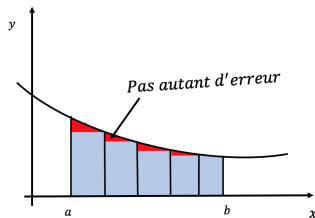
```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
@author: H.El-Otmany
"""
import numpy as np
import matplotlib.pyplot as plt
#define of integral bounds
a = 0
b = 1.1
increment = 0.1
# values of x
x = np.arange(a,b,increment)
# function f(x)
f = x**2
#plotting function
plt.plot(x, f)
plt.xlabel('x')
plt.ylabel('f')
plt.axis([0, 1, 0, 1])
plt.fill_between(x,f)
plt.show()
```


Mise en oeuvre de l'intégration numérique

- Décomposition du domaine en morceaux (un intervalle en sous-intervalles contigus).
- Intégration approchée de la fonction sur chaque morceau.
- Sommation des résultats numériques ainsi obtenus.
- Utilisation de polynômes pour approcher la fonction sur chaque morceau



Division de la courbe en parties plus petites



Raffinement des subdivisions pour minimiser l'erreur

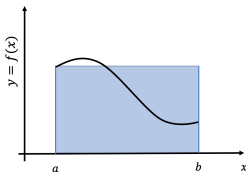
Chapitre 1 : Méthodes par quadratique simple

- ❶ Méthode du rectangle
- ❷ Méthode du point milieu
- ❸ Méthode du trapèze
- ❹ Méthode de Simpson

Méthode du rectangle

Pour approcher l'intégrale $I_{(a,b)}(f)$, on considère le rectangle (blue) dont

- la largeur est sur l'axe des abscisses d'extrémités $(a; 0)$ et $(b; 0)$
- la hauteur est sur l'axe des ordonnées $f(a)$ (méthode du rectangle gauche) ou $f(b)$ (méthode rectangle droite).



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
@author: H. El-Otmany
"""
#rectangle method
def rectangleMethod(f, a, b):
    A = (b-a)*f(a)
    return A
```

$$I_{(a,b)}(f) = (b-a)f(a) + \mathcal{O}\left(\frac{(b-a)^2}{2}f'(\xi)\right) \text{ ou}$$

$$I_{(a,b)}(f) = (b-a)f(b) + \mathcal{O}\left(\frac{(b-a)^2}{2}f'(\xi)\right), \xi \in [a; b]$$

Exemples

$$*I_{(0,1)}(3) = 3; \quad *I_{(0,1)}(x^2) \approx 0 \neq \frac{1}{3}; \quad *I_{(0,1)}(4\sqrt{1-x^2}) \approx 4 \neq \pi$$

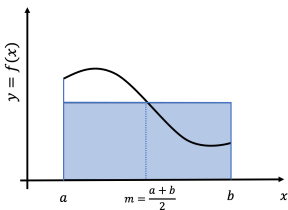
⇒ Méthode d'ordre 0, exacte pour les fonctions constantes.

⇒ * erreur pour les fonctions non-constantes ?

↪ Raffiner pour minimiser l'erreur

Méthode du point milieu

Pour approcher l'intégrale $I_{(a,b)}(f)$, on subdivise le rectangle en deux subdivisions, c'est-à-dire on considère trois points sur l'axe des abscisses $(a; 0)$, $(b; 0)$ et $\left(\frac{a+b}{2}, 0\right)$



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
@author: H. El-Otmany
"""

#Mid point method
def midPointMethod(f, a, b):
    A = (b-a)*f((a+b)/2)
    return A
```

$$I_{(a,b)}(f) = (b-a)f\left(\frac{a+b}{2}\right) + \mathcal{O}\left(\frac{(b-a)^3}{24}f'''(\xi)\right), \xi \in [a; b]$$

Exemples

$$\begin{aligned} *I_{(0,1)}(x) &\approx (1-0)\frac{1+0}{2} = \frac{1}{2}; & *I_{(0,1)}(x^2) &\approx (1-0)\frac{1}{4} = \frac{1}{4} \neq \frac{1}{3}; & *I_{(0,1)}(4\sqrt{1-x^2}) &\approx \\ (1-0)4\sqrt{1-\left(\frac{1}{2}\right)^2} &= 2\sqrt{3} \neq \pi. \end{aligned}$$

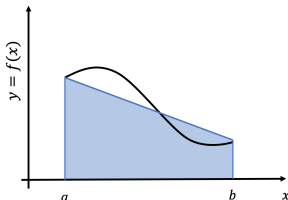
⇒ Méthode d'ordre 1, exacte pour les fonctions linéaires.

⇒ * erreur pour les fonctions non-linéaires? ⇝ Raffiner pour minimiser l'erreur

Méthode du trapèze

Pour approcher l'intégrale $I_{(a,b)}(f)$, on considère le trapèze où

- les bases s'appuient sur les abscisses $x = a$ et $x = b$
- les deux côtés sont l'axe des abscisses et le segment reliant les points $(a; f(a))$ et $(b; f(b))$.



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
@author: H. El-Otmany
"""

#Trapezoidal method
def trapezeMethod(f, a, b) :
    A = (b-a) * (f(a)+f(b)) / 2
    return A
```

$$I_{(a,b)}(f) = (b-a) \frac{f(a) + f(b)}{2} + \mathcal{O}\left(-\frac{(b-a)^3}{12} f''(\xi)\right), \xi \in [a; b]$$

Exemples

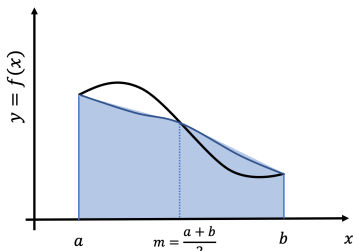
$$* I_{(0,1)}(x) = (1-0) \left(\frac{0+1}{2}\right) = \frac{1}{2}; \quad * I_{(0,1)}(x^2) \approx (1-0) \frac{0+1}{2} = \frac{1}{2} \neq$$

$$\frac{1}{3}; \quad * I_{(0,1)}(4\sqrt{1-x^2}) \approx (1-0) \frac{4\sqrt{1-0^2} + 4\sqrt{1-1^2}}{2} = 2 \neq \pi$$

⇒ Méthode d'ordre 1, exacte pour les fonctions linéaires.

Méthode de Simpson

Pour approcher l'intégrale $I_{(a,b)}(f)$, on subdivise le rectangle en deux subdivisions, c'est-à-dire on considère trois points sur l'axe des abscisses $(a, 0)$, $(b, 0)$ et $\left(\frac{a+b}{2}, 0\right)$



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
@author: Hammou El-Otmany
"""
#Simpsonmethod
def simpsonMethod(f, a, b) :
    m = (a+b) / 2
    A = (b-a) * (f(a) + 4*f(m) + f(b)) / 6
    return A
```

$$I_{(a,b)}(f) = (b-a) \frac{f(a) + f\left(\frac{a+b}{2}\right) + f(b)}{6} + \mathcal{O}\left(-\frac{(b-a)^5}{90 \times 2^5} f^{(4)}(\xi)\right), \xi \in [a; b]$$

Exemples

$$\begin{aligned} * I_{(0,1)}(x^2) &\approx (1-0) \left[0^2 + \frac{1}{4} + 1^2 \right] / 6 = \frac{1}{3}; & * I_{(0,1)}(4\sqrt{1-x^2}) &\approx \\ (1-0) \left[4\sqrt{1-0^2} + 4\sqrt{1-\frac{1}{4}} + 4\sqrt{1-1^2} \right] &= 4 + 2\sqrt{3} \neq \pi \end{aligned}$$

⇒ Méthode d'ordre 2, exacte pour les fonctions quadratiques et cubiques.

Bilan à retenir

- ▣ Toutes les méthodes (formules) apparaissent le produit de la largeur d'intervalle $[a; b]$ par la moyenne pondérée des valeurs prises par f en différents points x_i .
- ▣ Toutes les méthodes ont une erreur sous la forme d'une puissance de $b - a$.
- ▣ Le tableau ci-dessous récapitule le nombre de points, les poids et l'ordre de la méthode ($\xi \in [a; b]$) :

Méthode	Nombre de points	poids des points	Ordre	Erreur
Rectangle gauche	1	$(1, 0, 0)$	0	$\frac{(b-a)^2}{2} f'(\xi)$
Rectangle droite	1	$(0, 0, 1)$	0	$\frac{(b-a)^2}{2} f'(\xi)$
Point Milieu	1	$(0, 1, 0)$	1	$\frac{(b-a)^3}{24} f''(\xi)$
Trapèze	2	$\left(\frac{1}{2}, 0, \frac{1}{2}\right)$	2	$-\frac{(b-a)^3}{12} f''(\xi)$
Simpson	3	$\left(\frac{1}{6}, \frac{4}{6}, \frac{1}{6}\right)$	3	$-\frac{(b-a)^5}{90 \times 2^5} f^{(4)}(\xi)$

- ▣ En utilisant d'autres subdivisions de l'intervalle $[a; b]$, et d'autres coefficients, ce qui conduira à autant d'autres méthodes avec leurs forces et leurs faiblesses.

↪ À la fin, nous comparerons la précision obtenue avec les méthodes de quadrature simple.

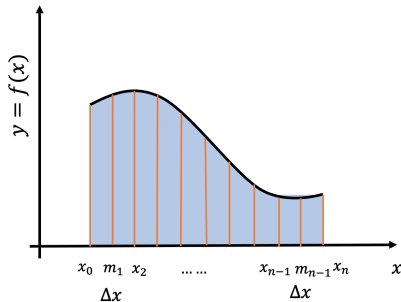
Chapitre 2 : Méthodes composites d'intégration numérique

- ❶ Méthode du rectangle composite
- ❷ Méthode du point milieu composite
- ❸ Méthode du trapèze composite
- ❹ Méthode de Simpson composite

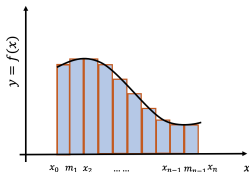
Principe des méthodes composites

Pour approcher l'intégrale $I_{(a,b)}(f)$, on procède comme suit

- On approche f par des fonctions de plus en plus complexes (notamment des polynômes d'ordre de plus en plus élevés)
- On découpe l'intervalle $[a; b]$ en n sous intervalles
- On applique les méthodes d'intégration simple sur les petits intervalles.
- On multiplie les méthodes par $\Delta x = \frac{b-a}{n}$ (la largeur d'intervalle)
- On utilise le découpage simple dont les points d'échantillonnage régulier sont :
$$x_i = x_0 + i\Delta x, \quad m_i = x_0 + \left(i + \frac{1}{2}\right) \Delta x \text{ pour } 0 \leq i \leq n.$$
- On retrouve bien $x_0 = a$ et $x_n = b$.



Méthode du rectangle composite



$$I_{(a,b)}^{RG}(f) \approx \frac{b-a}{n} \sum_{i=0}^{n-1} f(x_i) + \mathcal{O}\left(\frac{(b-a)^2}{2n} f'(\xi)\right),$$

$$I_{(a,b)}^{RD}(f) \approx \frac{b-a}{n} \sum_{i=0}^{n-1} f(x_{i+1}) + \mathcal{O}\left(\frac{(b-a)^2}{2n} f'(\xi)\right),$$

$$\xi \in [a; b]$$

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
```

```
@author: H. El-Otmany
"""
```

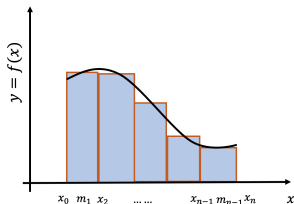
```
import math
```

```
#composite Rectangle method
```

```
def rectCompositeMethod(f, a, b, n) :
    pas = (b-a)/n
    #define a list to storage the sum's
    liste = []
    #compute the function at x_i
    for i in range(n) :
        liste.append( f(a + i*pas))
    #compute the sum of different terms
    A = pas * math.fsum(liste)
    return A
```

⇒ * erreur numérique? ~> Raffiner pour minimiser l'erreur

Méthode du point milieu composite

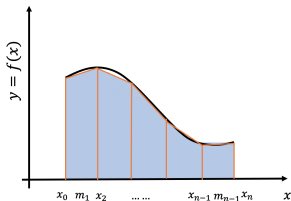


```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
@author: H. El-Otmany
"""
import math
#Composite midpoint method
def midPointCompositeMethod(f, a, b, n)
    pas = (b-a)/n
    liste = []
    #compute the function at x_i
    for i in range(n) :
        liste.append( f(a + (i+0.5)*pas) )
    #compute the sum of different terms
    A = pas * math.fsum(liste)
    return A
```

$$I_{(a,b)}(f) = \frac{b-a}{n} \sum_{i=0}^n f(m_i) + \mathcal{O}\left(\frac{(b-a)^3}{24n^2} f''(\xi)\right), \xi \in [a; b]$$

⇒ * erreur numérique ? ↪ Raffiner pour minimiser l'erreur

Méthode du trapèze composite

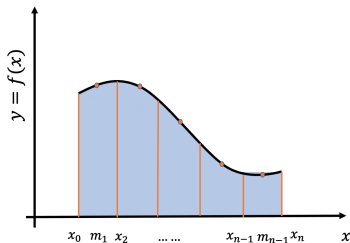


```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
@author: H. El-Otmany
"""
import math
#Composite Trapezoidal method
def trapCompositeMethod(f, a, b, n) :
    pas = (b-a)/n
    #define a list to storage different
    liste = [ f(a)/2, f(b)/2 ]
    #Which add the sum's terms
    x = a + pas
    for i in range(n-1) :
        liste.append( f(x) )
        x = x + pas
    A = pas * math.fsum(liste)
    return A
```

$$I_{(a,b)}(f) = \frac{b-a}{n} \left[\frac{f(a)+f(b)}{2} + \sum_{i=1}^{n-1} f(x_i) \right] + \mathcal{O} \left(\frac{(b-a)^3}{12n^2} f''(\xi) \right), \xi \in [a;b]$$

⇒ * erreur numérique? ~> Raffiner pour minimiser l'erreur

Méthode de Simpson composite



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
@author: H. El-Otmany
"""
import math
#Composite Simpson method
def simpsonCompositeMethod(f, a, b, n) :
    pas = (b-a)/(2*n)
    #define a list to storage different com
    liste = [ f(a), f(b) ]
    #Which add the sum's terms
    x = a + pas
    for i in range(2*n-1) :
        if i%2 == 1 :
            liste.append(2 * f(x))
        else :
            liste.append(4 * f(x))
        x = x + pas
    A = pas * math.fsum(liste) / 3
```

$$I_{(a,b)}(f) = \frac{b-a}{6n} \left[f(a) + f(b) + 2 \sum_{i=1}^{n-1} f(x_i) + 4 \sum_{i=0}^{n-1} f(m_i) \right] + \mathcal{O} \left(\frac{(b-a)^5}{180 \times (4n)^4} f^{(4)}(\xi) \right), \xi \in [a; b]$$

Bilan et exemples d'applications

Bilan à retenir

- ▢ Toutes les méthodes (formules) apparaissent le produit de la largeur $\frac{b-a}{n}$ par la moyenne pondérée des valeurs prises par f en différents points x_i .
- ▢ Toutes les méthodes ont une erreur sous la forme d'une puissance de $\frac{b-a}{n}$.
- ▢ Les méthodes traitées dans ce cours sont prédéfinies dans le package "scipy.integrate" de Python. Pour plus d'informations, consulter le lien <https://docs.scipy.org/doc/scipy/reference/integrate.html>

Exemples

- ❶ Le tableau ci-dessous donne les valeurs du profil de température lors d'une expérience d'émulsion. Estimer $\int_0^6 f(x)dx$ en appliquant les différentes méthodes d'intégrations avec trois sous-intervalles égaux.

x	0	1	2	3	4	5	6
$f(x)$	9.3	9.0	8.3	6.5	2.3	7.6	10.5

- ❷ Estimer l'aire de la région bornée par la courbe de $y = \sin(x)$ et l'axe des abscisses avec x allant de 0 jusqu'à π en utilisant trois sous-intervalles égaux.
- ❸ Supposons que $\int_4^9 \sqrt{x}dx = \frac{38}{3}$. En utilisant les transformations, calculer

$$\int_9^4 \sqrt{t}dt; \quad \int_4^9 (\sqrt{x} + 3)dx; \quad \int_9^{14} \sqrt{x-5}dx; \quad \int_4^9 (\sqrt{x}dx$$

Bilan et exemples d'applications

- 4 Le tableau ci-dessous donne les valeurs du profil de température lors d'une expérience d'émulsion. Estimer $\int_1^{9.6} f(x)dx$ en appliquant les différentes méthodes d'intégrations avec trois sous-intervalles égaux.

x	1	2.5	4	6	8	8.8	9.6	10.4
$f(x)$	4	3	1	3	5	6	4	7

- 5 Estimer l'aire de la région bornée par la courbe de $y = \sin(x)$ et l'axe des abscisses avec x allant de 0 jusqu'à π en utilisant trois sous-intervalles égaux.
- 6 On considère le piston d'une voiture (voir le schéma) contenant $n = 10^3 \text{ mol}$ de gaz sous une température constante $T = 300\text{K}$ durant le processus. Sachant que $V_1 = 1\text{m}^3$ et $V_2 = 5\text{m}^3$, estimer le travail W produit dans ce piston. On rappelle la loi des gaz parfaits : $PV = nRT$ où R est la constante universelles des gaz, $R = 8.314\text{kJ/kmol} \cdot \text{K}$.

