Intégration

Lycée Thiers

Intégration

Lycée Thiers

1. Méthode des rectangles

2. Méthode des trapèzes

3. Fonction de bibliothèque

Intégration numérique : méthode des rectangles

- Approximations de l'intégrale d'une fonction $f(x)$ sur $[a, b]$.
- Principe : approximer l'aire sous la courbe par la somme d'aires de rectangles.
- Trois variantes courantes :
  - Bord gauche
  - Bord droit
  - Point milieu

Méthode des rectangles : bord gauche

- On utilise $f(x_i)$ pour approximer $\int_{x_i}^{x_{i+1}} f(x)dx$.
- Formule :

$$\int_a^b f(x)dx \approx h \sum_{i=0}^{n-1} f(x_i)$$

- Simple mais erreur de décalage.

```
a, b = 0, 1
n = 100
h = (b - a) / n

x = np.linspace(a, b - h, n)
integral_left = h * np.sum(f(x))
```

Méthode des rectangles : bord droit

- On utilise $f(x_{i+1})$.
- Formule :

$$\int_a^b f(x)dx \approx h \sum_{i=1}^{n} f(x_i)$$

- Décalage dans l'autre sens.

```
x = np.linspace(a + h, b, n)
integral_right = h * np.sum(f(x))
```

Méthode des rectangles : point milieu

- On utilise $f\left(\frac{x_i+x_{i+1}}{2}\right)$.
- Formule :

$$\int_a^b f(x)dx \approx h\sum_{i=0}^{n-1} f\left(x_i + \frac{h}{2}\right)$$

- Généralement plus précise que bord gauche / droit.

```
x = np.linspace(a, b - h, n)
midpoints = x + h / 2
integral_mid = h * np.sum(f(midpoints))
```

Résumé des méthodes des rectangles

- **Bord gauche** : simple, mais sous-estime ou surestime souvent.
- **Bord droit** : comportement symétrique du bord gauche.
- **Point milieu** : en général plus précis pour un coût identique.
- Toutes ces méthodes ont une erreur $\mathcal{O}(h)$, sauf le point milieu qui est $\mathcal{O}(h^2)$.

Méthode des trapèzes : principe

- Approximer $\int_a^b f(x)dx$ par la somme des aires de trapèzes.
- Sur chaque sous-intervalle $[x_i, x_{i+1}]$, on approxime $f(x)$ par une droite.
- Formule locale :

$$\int_{x_i}^{x_{i+1}} f(x)dx \approx \frac{h}{2}\left(f(x_i) + f(x_{i+1})\right)$$

Méthode des trapèzes : formule globale

- Formule globale sur $[a, b]$ avec n sous-intervalles de taille h :

$$\int_a^b f(x)dx \approx \frac{h}{2}\left(f(a) + 2\sum_{i=1}^{n-1} f(x_i) + f(b)\right)$$

- Erreur globale $\mathcal{O}(h^2) \rightarrow$ méthode plus précise que les rectangles.

Méthode des trapèzes : code Python

```
a, b = 0, 1
n = 100
h = (b - a) / n

x = np.linspace(a, b, n + 1)
y = f(x)

integral_trapz = h * (0.5 * y[0] + np.sum(y[1:-1]) + 0.5
```

Résumé : méthode des trapèzes

- Approximations par segments linéaires.
- Erreur $\mathcal{O}(h^2)$, donc plus précise que bord gauche / droit.
- Simple à implémenter, efficace.
- Très utilisée en pratique pour des intégrales approchées.

1. Méthode des rectangles
2. Méthode des trapèzes
3. Fonction de bibliothèque

scipy.integrate.quad

- Fonction d'intégration numérique adaptative générale.
- Syntaxe : `quad(f, a, b)`.
- Retourne : `(valeur, estimation_erreur)`.
- Basée sur la bibliothèque `QUADPACK`.
- Très précise, adaptée à la plupart des cas.

```python
from scipy.integrate import quad
import numpy as np

# Définir la fonction à intégrer
def f(x):
    return np.sin(x)

# Intégrer sur [0, pi]
result, error = quad(f, 0, np.pi)

print("Résultat:", result)
print("Estimation de l'erreur:", error)
```