

# 計算物理学B

## 第7回

### 数値積分

藤本 悠輝、野垣 康介  
(藤本担当回)

質問等あればメールでも受け付けます:  
[yuki.fujimoto.phys\\_at\\_niigata-u.ac.jp](mailto:yuki.fujimoto.phys_at_niigata-u.ac.jp)  
(\_at\_ を @ に変えてください)

# 講義予定

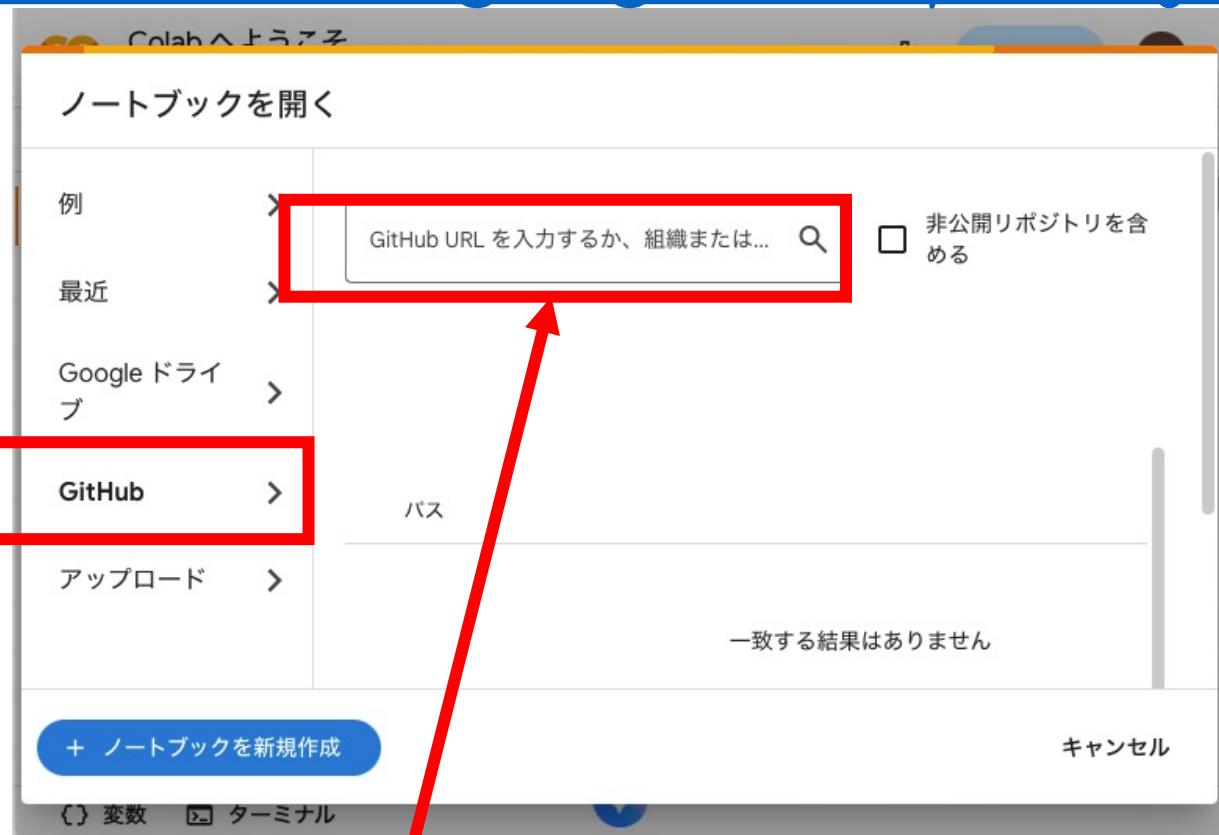
10/07	両名:四則演算	12/09	野垣:モンテカルロ1
10/14	野垣:制御文(for, if)	12/16	野垣:モンテカルロ2
10/21	野垣:関数	12/23	藤本:微分方程式1
10/28	藤本:配列(numpy)	01/13	藤本:微分方程式2
11/04	藤本:可視化(matplotlib)	01/20	藤本:微分方程式3
11/11	野垣:数値微分	01/27	野垣:最適化
11/18	藤本:数値積分 ～中間レポート～	02/03	藤本:機械学習 ～期末レポート～

あくまで予定なので変更の可能性あり

# 実習環境

まず、Google Colabを開く：

<https://colab.research.google.com/?hl=ja>



I. GitHubを選択

2. ここに今週のNotebookのURLを入力してEnter:

[https://github.com/nogaki/Computational\\_Physics\\_B/blob/main/week7/week7.ipynb](https://github.com/nogaki/Computational_Physics_B/blob/main/week7/week7.ipynb)

# 重要なお知らせ

中間レポートを出題します。

[https://github.com/nogaki/Computational\\_Physics\\_B/blob/main/report1/report1.pdf](https://github.com/nogaki/Computational_Physics_B/blob/main/report1/report1.pdf)  
に掲載予定。

【締切】12/9 23:59  
厳守のこと

# 第7回 数値積分（求積法）

- 基本的な数値積分：  
中点公式・台形公式・シンプソン公式
- 適応求積法
- 特殊な場合
- 高次の求積法・Gauss-Legendre求積法

この講義は以下を参考に準備されています：

<https://github.com/vlvoch/PHYS6350-ComputationalPhysics>

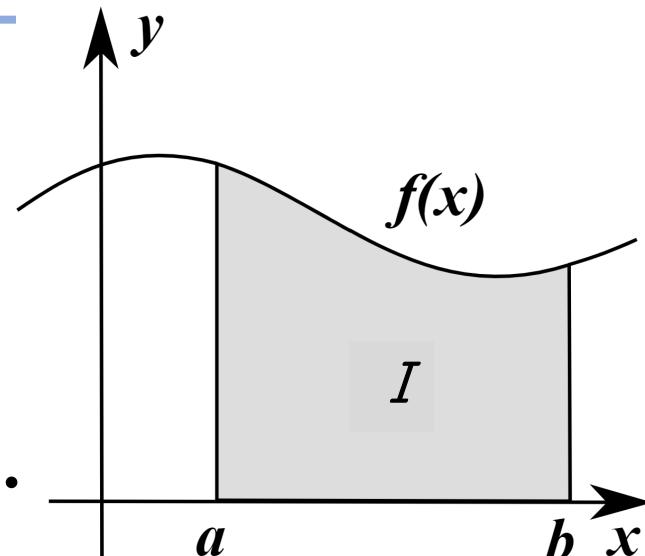
「実践計算物理学」野本拓也、是常隆、有田亮太郎 著（共立出版）

「理工学のための数値計算法」水島二郎、柳瀬眞一郎 著（数理工学社）

# 数値積分

一般的な問題: 積分の計算(求積法)

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



- 解析的に積分を実行するのが困難・不可能な場合
- 被積分関数  $f(x)$  が特定の点のみでしか見積もれない場合

以上のような場合は、**数値積分**が必要になる。

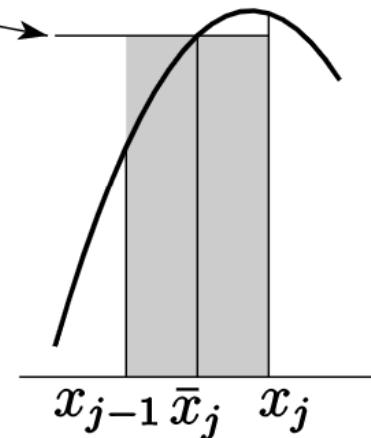
Pythonでは `scipy.integrate`という強力なライブラリがあるが、この授業ではそれを使わず、アルゴリズムを学ぶことを目標とする。

# 数値積分: 中点公式

まず、最もシンプルな場合を考える。 $f\left(\frac{x_{j-1}+x_j}{2}\right)$

区間  $[x_{j-1}, x_j]$  での積分を、中点での  
関数値の長方形だと近似する

$$\int_a^b f(x) dx \approx (b - a) f\left(\frac{a + b}{2}\right)$$



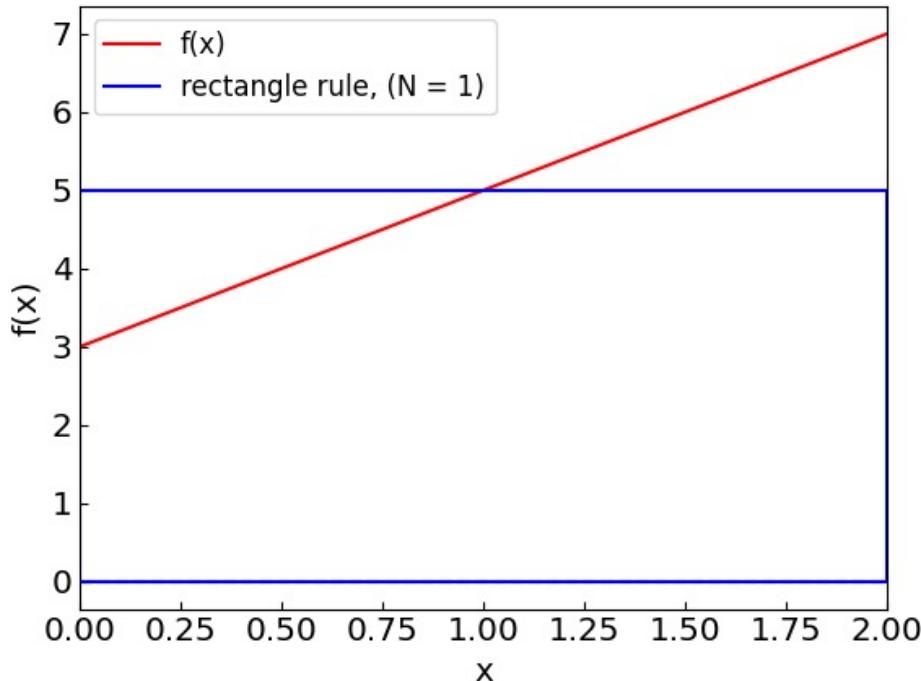
誤差:  $\int_a^b f(x)dx - (b - a) f\left(\frac{a + b}{2}\right) \approx \frac{(b - a)^3}{24} f''(a)$   
(テイラー展開より導出可)

線形関数の積分のときには中点公式は厳密

# 数値積分: 中点公式

例:

$$I = \int_0^2 (2x + 3)dx = 10$$

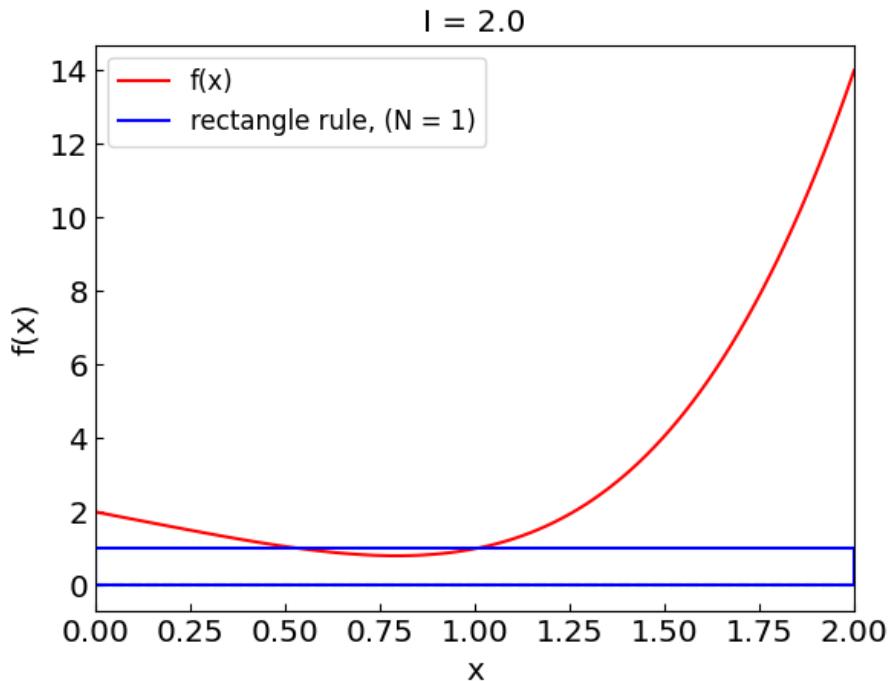


被積分関数は台形(赤線)。中点公式で与えられる長方形は、図形としては良くない近似だが、結果として誤差が相殺して厳密な結果になる

# 数値積分: 中点公式

他の例:

$$I = \int_0^2 (x^2 + 2x + 2) dx = 6.4$$

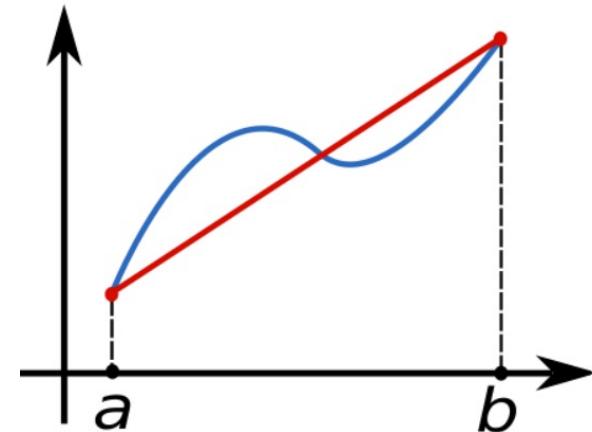


中点公式は  $I = 2.0$  となる。非常に精度が悪い。

# 数值積分：台形公式

積分を台形で近似：

$$\int_a^b f(x) dx \approx (b - a) \frac{f(a) + f(b)}{2}$$

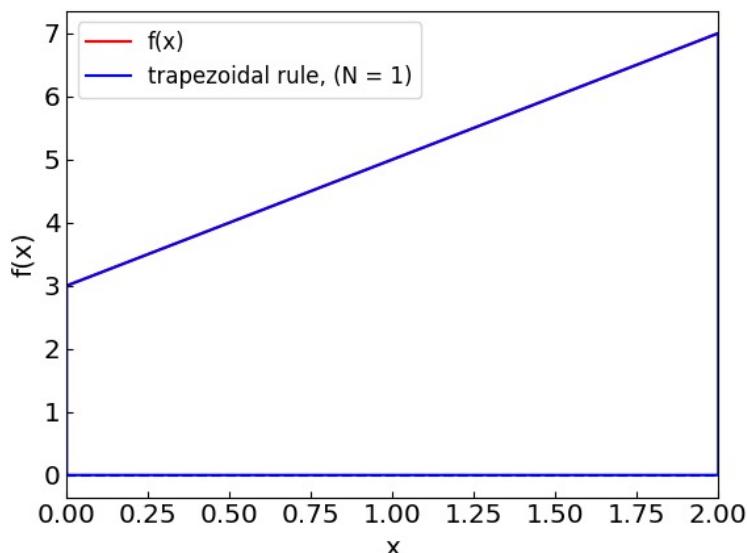


誤差： $\int_a^b f(x)dx - (b - a) \frac{f(a) + f(b)}{2} \approx -\frac{(b - a)^3}{12} f''(a)$

中点公式と逆符号！

例： $I = \int_0^2 (2x + 3)dx = 10$

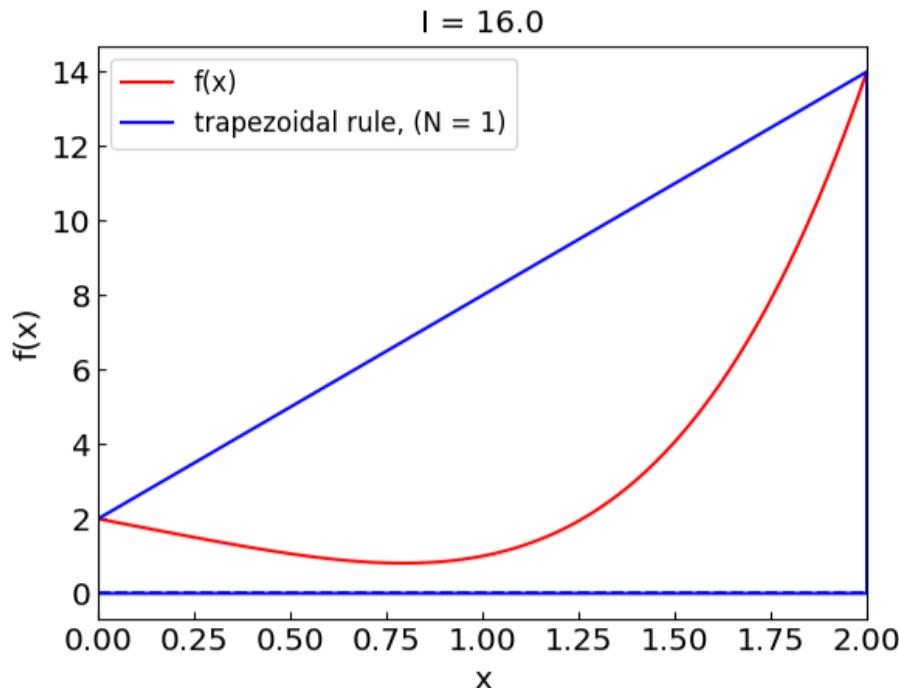
線形関数の積分のときには  
台形公式は厳密



# 数値積分: 台形公式

他の例:

$$I = \int_0^2 (x^2 + 2x + 2) dx = 6.4$$



台形公式は  $I = 16$  となる。精度が悪い。  
中点公式は積分値を過小評価していた一方、  
台形公式は過大評価。ズレが逆方向。

# 中点・台形公式の改善

中点・台形公式の誤差は、それぞれ  $\frac{(b-a)^3}{24}f''(a)$ ,  $-\frac{(b-a)^3}{12}f''(a)$   
 $b-a$  が大きい場合には近似が悪くなる。

単純な解決策：

1. 合成中点・台形公式

区間  $[a,b]$  を微小区間に分割

2. 高次の近似

積分公式の誤差が  $(b-a)$  の高次になるようにする。

# 中点・台形公式の改善

中点・台形公式の誤差は、それぞれ  $\frac{(b-a)^3}{24} f''(a)$ ,  $-\frac{(b-a)^3}{12} f''(a)$   
 $b-a$  が大きい場合には近似が悪くなる。

単純な解決策：

I. 合成中点・台形公式

区間  $[a,b]$  を微小区間に分割

2. 高次の近似

積分公式の誤差が  $(b-a)$  の高次になるようにする。

# 合成中点公式

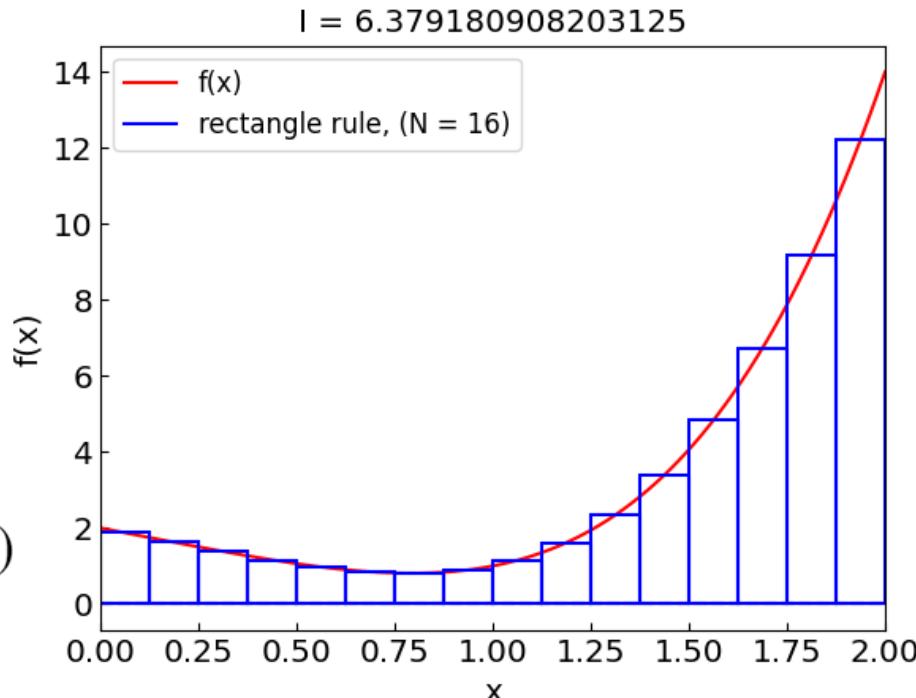
積分区間をN個の微小区間に分割し、各区間で中点公式 (rectangular rule) を適用：

$$\int_a^b f(x) \approx h \sum_{k=1}^N f(x_k), \quad k = 1, \dots, N$$

$$I = \int_0^2 (x^2 + 2x + 2) dx = 6.4$$

$$x_k = a + \frac{2k - 1}{2} h .$$

$$h = (b - a)/N$$

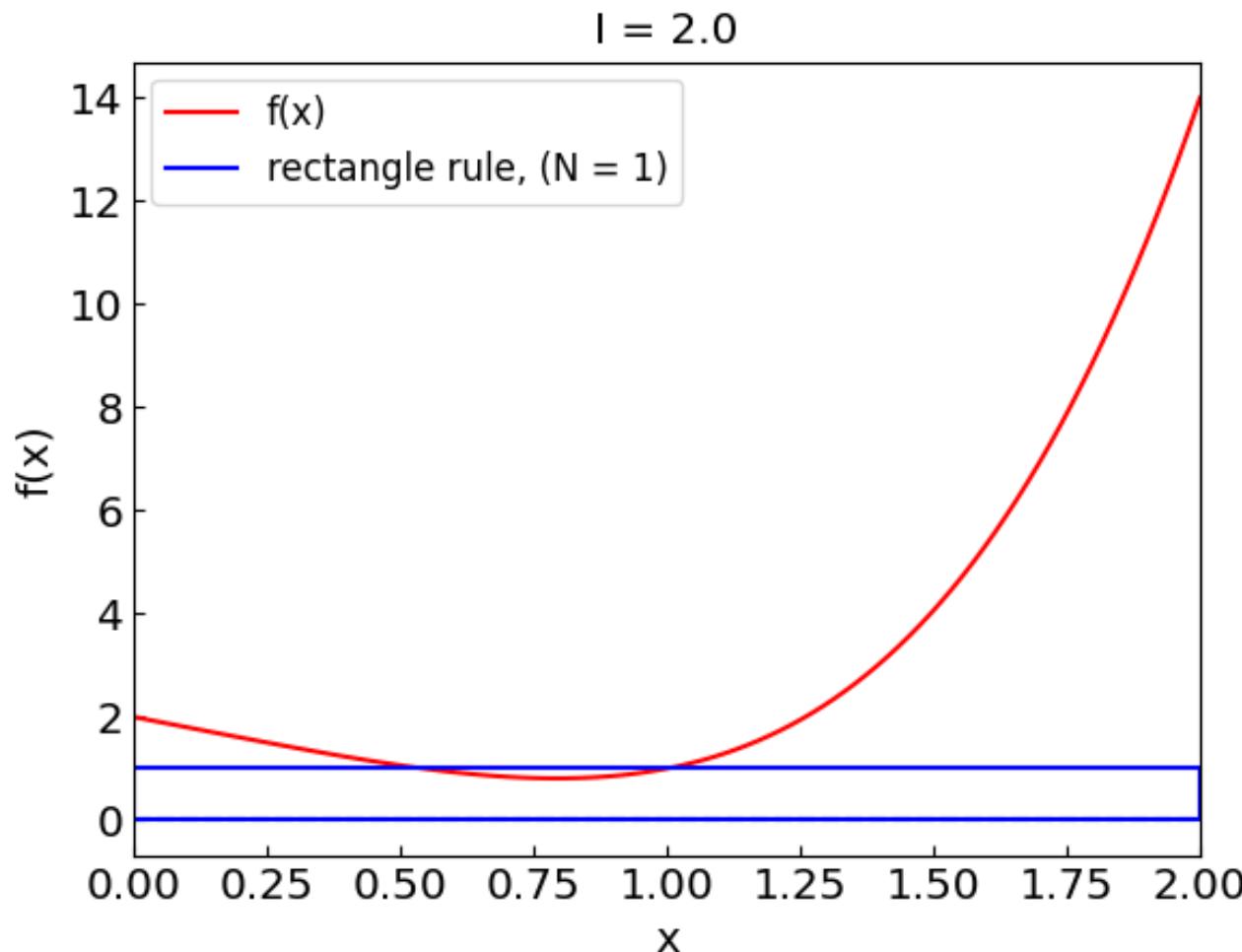


誤差：

$$I - I_{\text{rect}} = (b - a) \frac{h^2}{24} f''(a) + \mathcal{O}(h^4)$$

# 合成中点公式

$$I = \int_0^2 (x^2 + 2x + 2) dx = 6.4$$



# 合成台形公式

積分区間をN個の微小区間に分割し、各区間で台形公式 (trapezoidal rule) を適用：

$$\int_a^b f(x) \approx h \sum_{k=0}^N \frac{f(x_k) + f(x_{k+1})}{2}, \quad i = 0, \dots, N$$

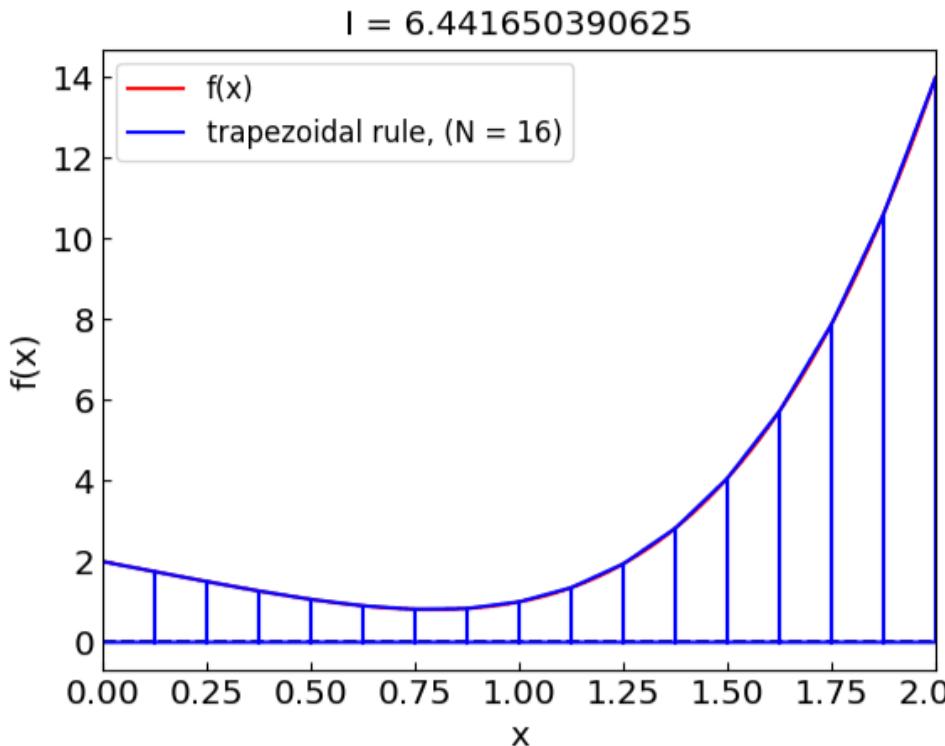
$$I = \int_0^2 (x^2 + 2x + 2) dx = 6.4$$

$$x_k = a + kh .$$

$$h = (b - a),$$

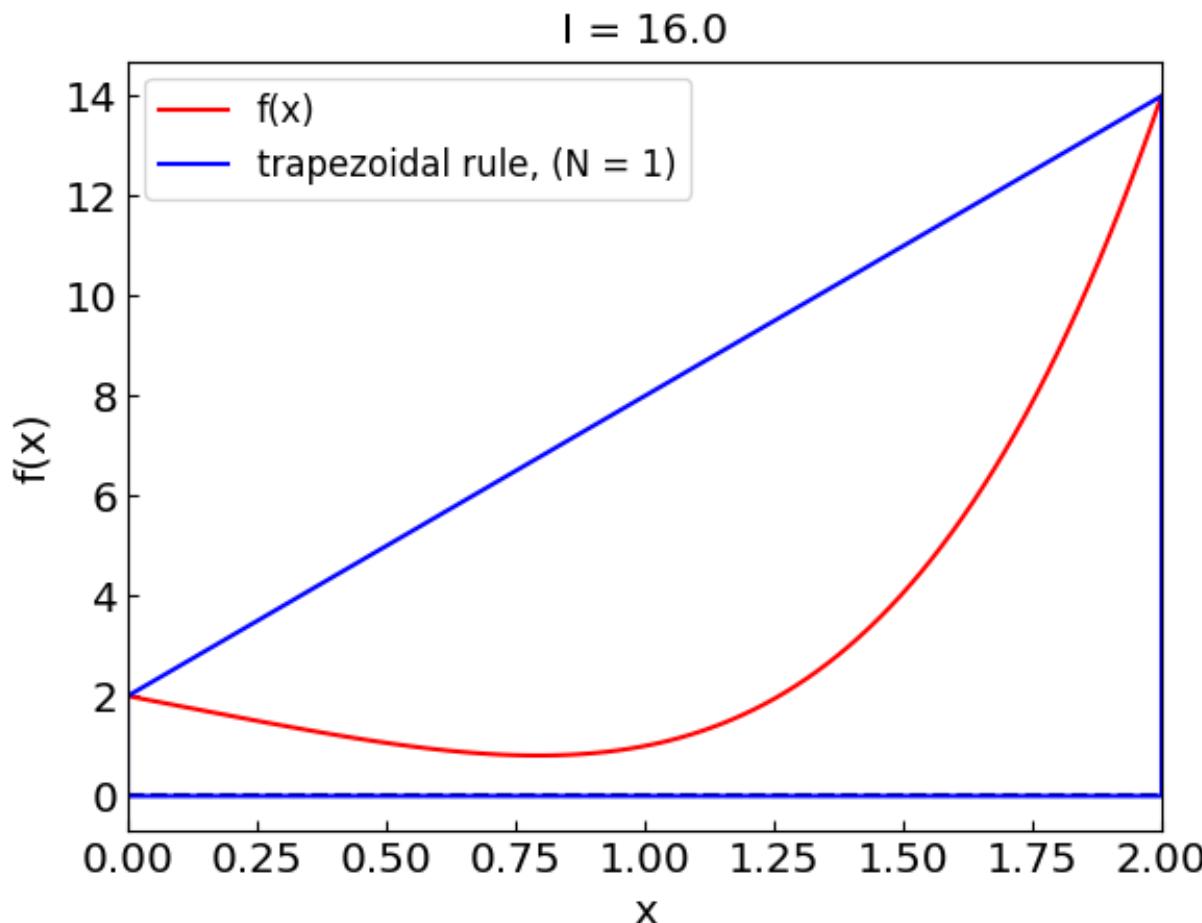
誤差：

$$I - I_{\text{trap}} = -(b - a) \frac{h^2}{12} f''(a) + \mathcal{O}(h^4)$$



# 合成台形公式

$$I = \int_0^2 (x^2 + 2x + 2) dx = 6.4$$



# 中点・台形公式の改善

中点・台形公式の誤差は、それぞれ  $\frac{(b-a)^3}{24} f''(a)$ ,  $-\frac{(b-a)^3}{12} f''(a)$   
 $b-a$  が大きい場合には近似が悪くなる。

単純な解決策：

I. 合成中点・台形公式

区間  $[a,b]$  を微小区間に分割

2. 高次の近似

積分公式の誤差が  $(b-a)$  の高次になるようにする。

# 数値積分：Simpson公式

中点則と台形則の誤差：

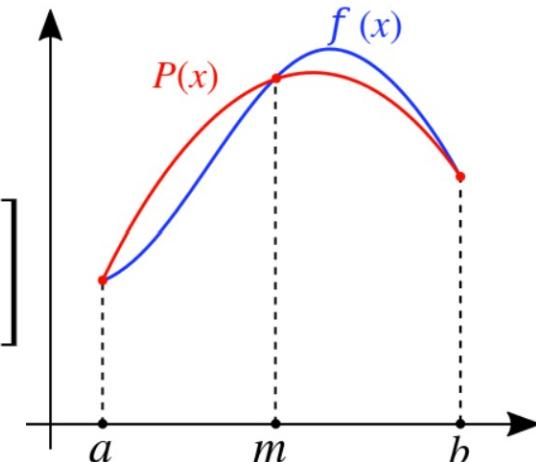
$$I - I_{\text{rect}} = (b - a) \frac{h^2}{24} f''(a) + \mathcal{O}(h^4) \quad I - I_{\text{trap}} = -(b - a) \frac{h^2}{12} f''(a) + \mathcal{O}(h^4)$$

組み合わせて  $\mathcal{O}(h^2)$  の誤差を相殺

$$I_S = \frac{2I_{\text{rect}} + I_{\text{trap}}}{3}$$

つまり、

$$\int_a^b f(x) dx \approx \frac{(b - a)}{6} \left[ f(a) + 4f\left(\frac{a + b}{2}\right) + f(b) \right]$$

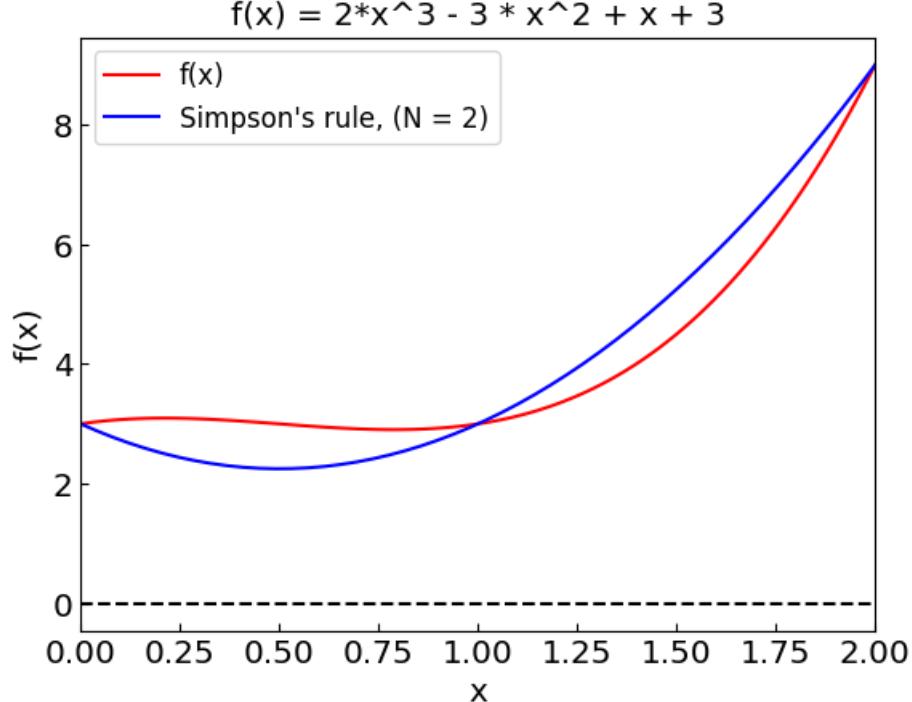
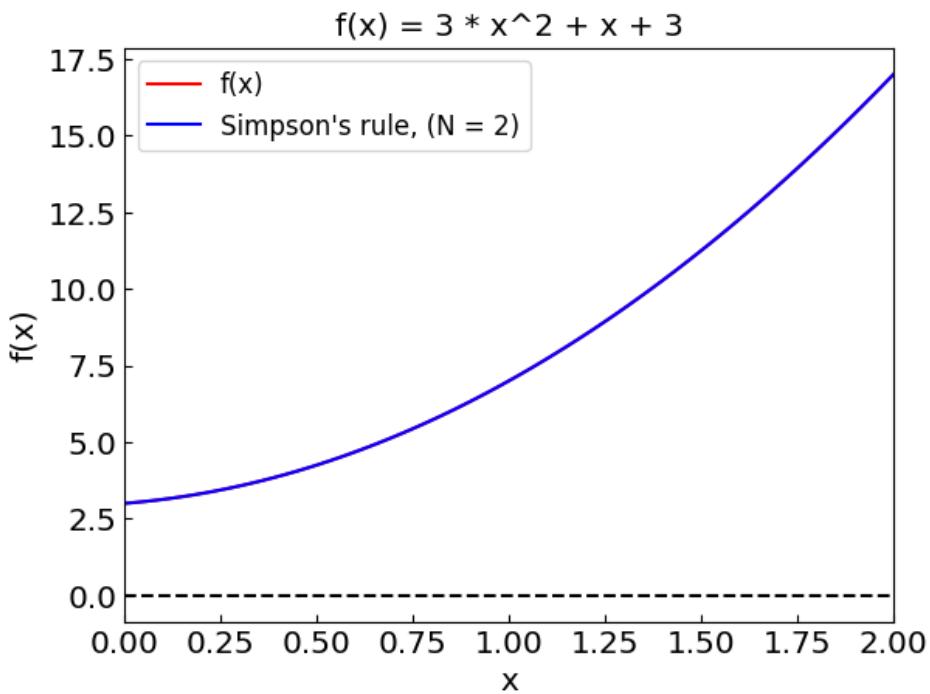


等価な方法として、被積分関数を2次関数で内挿することでもSimpson公式が得られる。

# 数値積分:Simpson公式

$$\int_a^b f(x) dx \approx \frac{(b-a)}{6} \left[ f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right]$$

誤差:  $I - I_S = C h^4 + \mathcal{O}(h^6)$

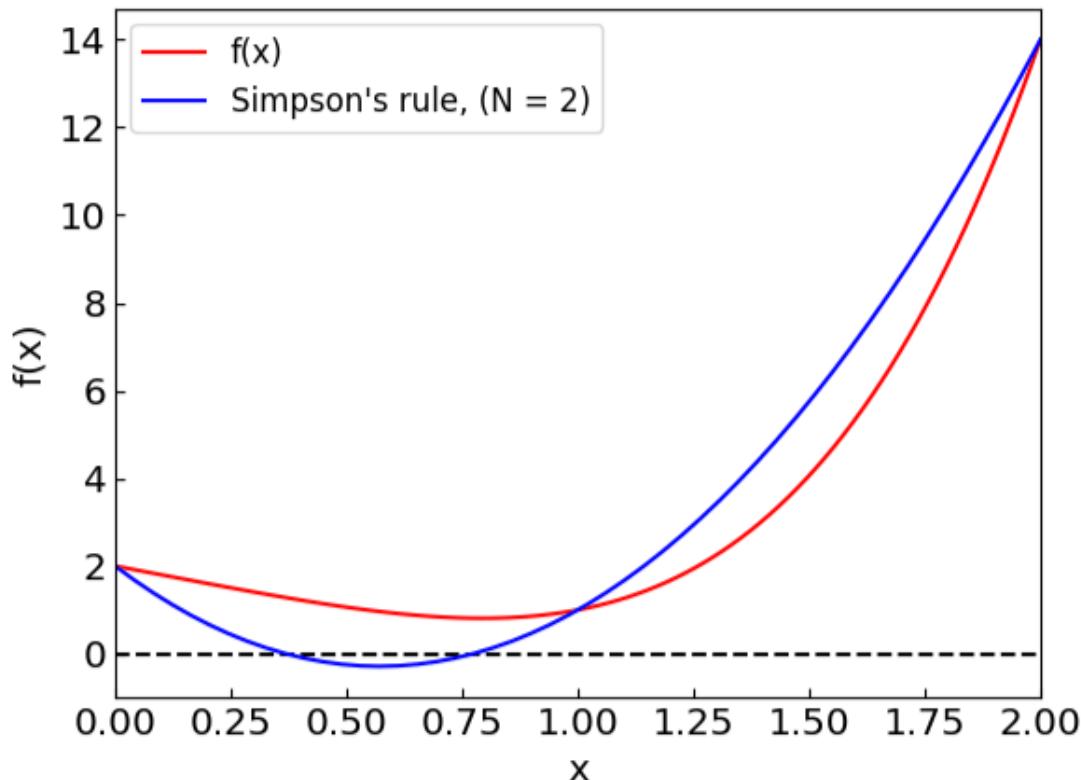


3次以下の関数の積分のときにはSimpson公式は厳密

# 数值積分：Simpson公式

例： $I = \int_0^2 (x^4 - 2x + 2)dx = 6.4$

$I = 6.666666666666666$



3点だけのSimpson公式  $I_S = 6.67$ 、悪くない近似

# 合成Simpson公式

$$\int_a^b f(x) \approx \frac{h}{3} \left[ f(x_0) + 4 \sum_{k=1}^{N/2} f(x_{2k-1}) + 2 \sum_{k=1}^{N/2-1} f(x_{2k}) + f(x_N) \right], \quad i = 0, \dots, N$$

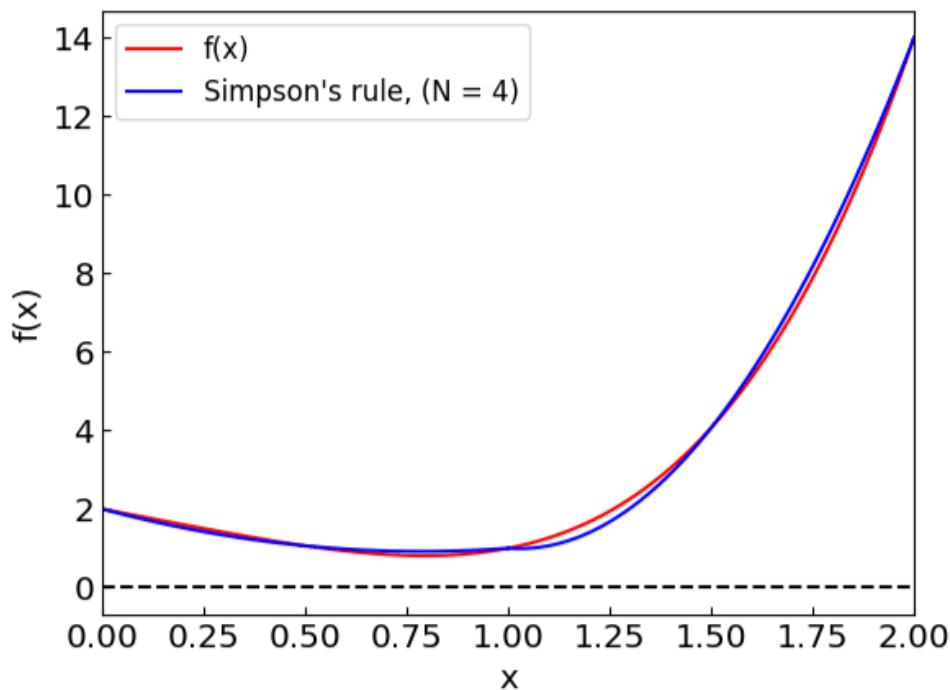
$$h = (b - a)/N \quad \text{ただし } N \text{ は偶数}$$

誤差:

$$I - I_S = C h^4 + \mathcal{O}(h^6)$$

$$I = \int_0^2 (x^4 - 2x + 2) dx = 6.4$$

$$I = 6.416666666666666$$

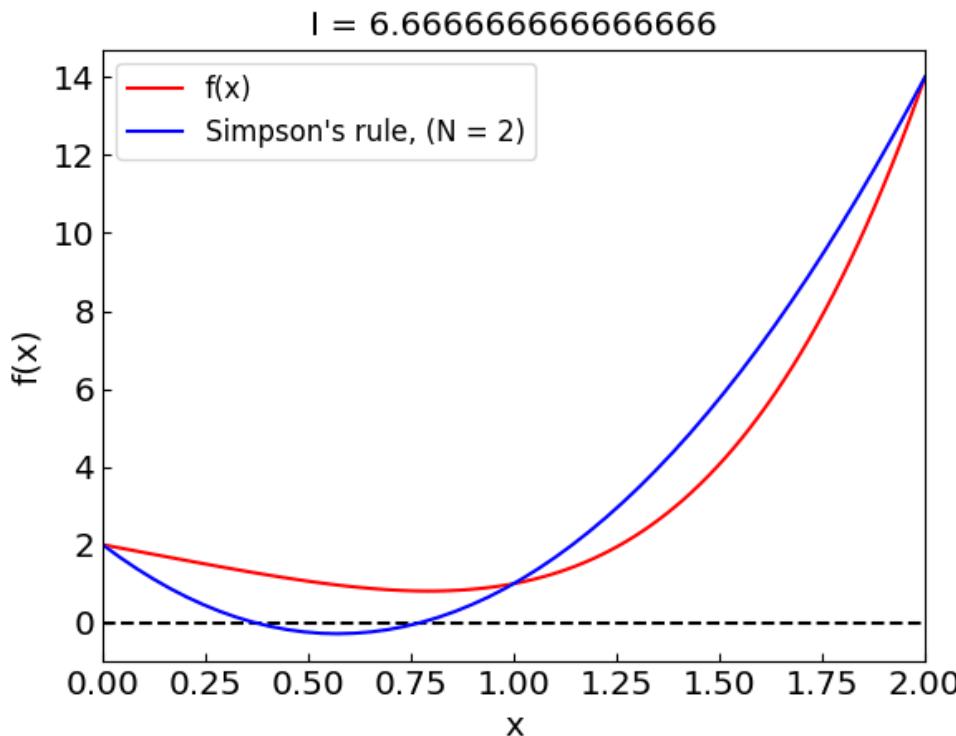


# 合成Simpson公式

$$\int_a^b f(x) \approx \frac{h}{3} \left[ f(x_0) + 4 \sum_{k=1}^{N/2} f(x_{2k-1}) + 2 \sum_{k=1}^{N/2-1} f(x_{2k}) + f(x_N) \right], \quad i = 0, \dots, N$$

$h = (b - a)/N$  ただし  $N$  は偶数

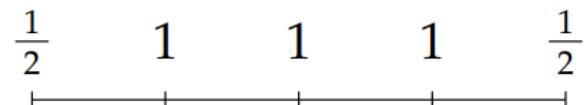
$$I = \int_0^2 (x^4 - 2x + 2) dx = 6.4$$



# 適応求積法 (adaptive quadrature)

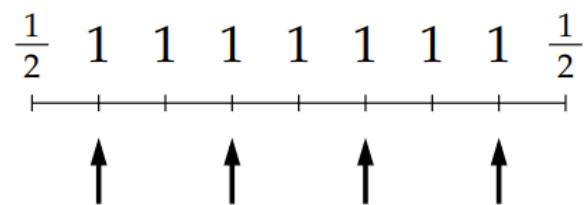
積分の誤差を制御したい。

- 1ステップごとに積分区間の分割数を2倍にする。
- 誤差評価を追跡。



中点・台形則の誤差は  $O(h^2)$  :

$$I - I_{\text{trap}} \approx ch^2$$



$k$ ステップ目では  $h_k = h_{k-1} / 2$

$$I - I_{\text{trap}}^k \approx ch_k^2, \quad I - I_{\text{trap}}^{k-1} \approx 4ch_k^2,$$

よって、 $k$ ステップ目の誤差は  $\varepsilon_k \simeq (I_{\text{trap}}^k - I_{\text{trap}}^{k-1})/3$

# 適応求積法 (adaptive quadrature)

さらなる改善: Romberg積分 (レポート問題)

$k$ ステップ目の積分値は

$$I = R_{k,1} = I_{\text{trap}}^k + \frac{I_{\text{trap}}^k - I_{\text{trap}}^{k-1}}{3} + \mathcal{O}(h^4)$$

... 級数の  $k$  番目とみなせる。

級数の収束を加速する手法 (Richardson加速) と組み合わせることによって、誤差を改善 (Romberg積分)

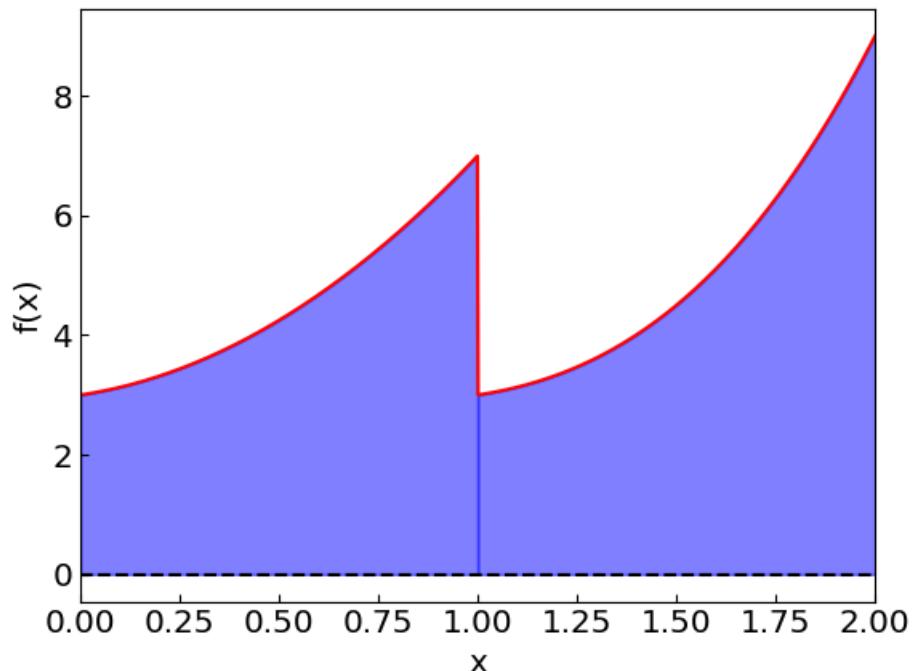
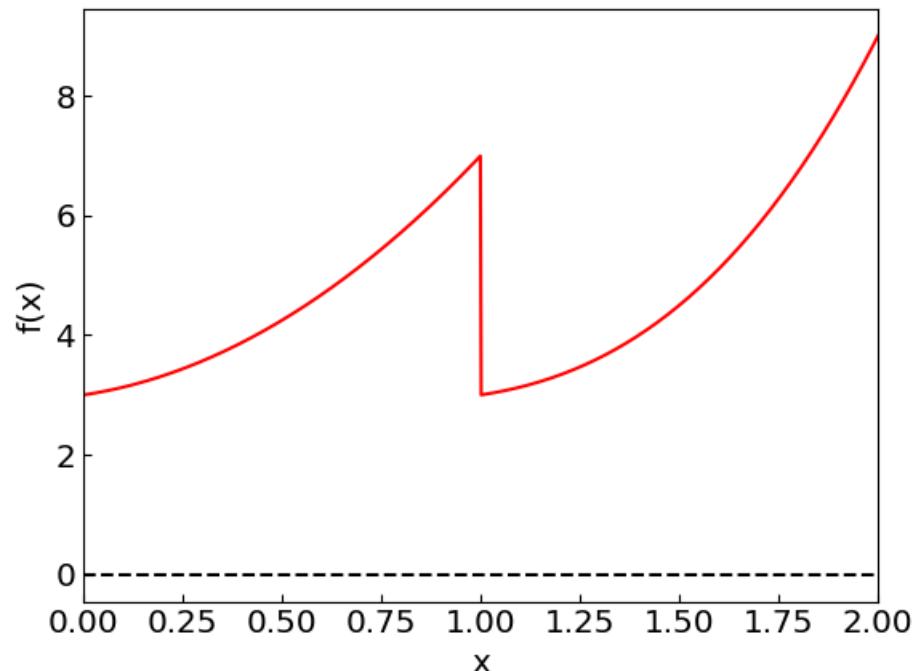
# 特殊な場合: 不連続関数の積分

例として次の関数を考える:

$$f(x) = 3x^2 + x + 3 \quad \text{for } x < 1,$$

$$f(x) = 2x^3 - 3x^2 + x + 3. \quad \text{for } x > 1,$$

$$I = \int_0^2 f(x) = 9.5$$



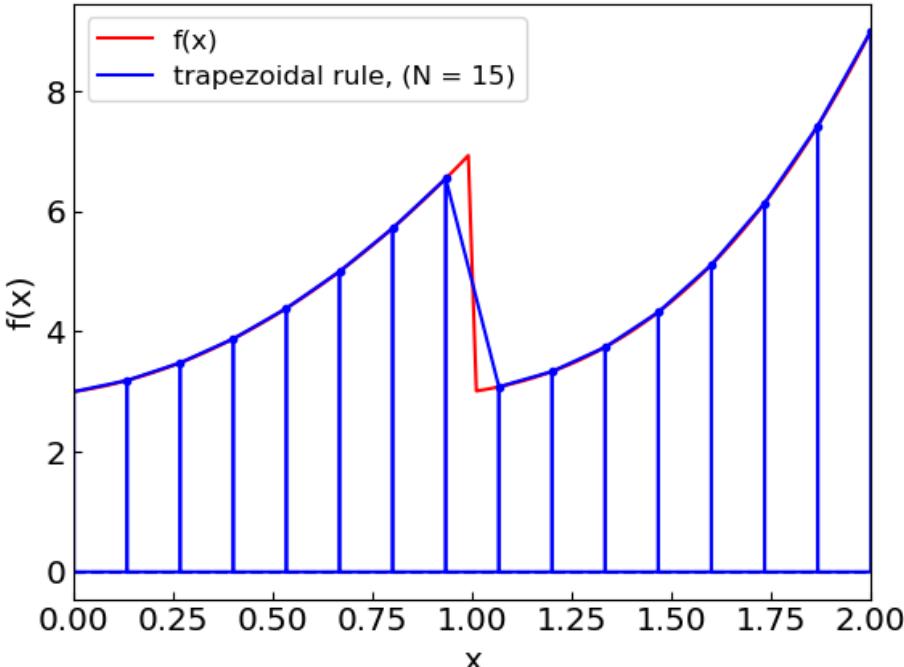
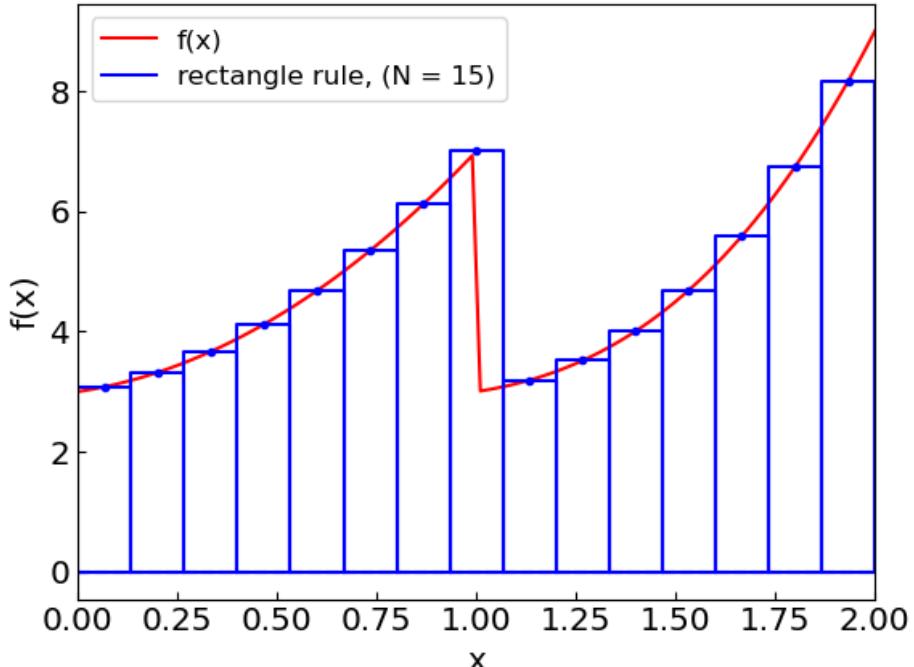
# 特殊な場合: 不連続関数の積分

中点則・台形則で問題なく積分できる

$$f(x) = 3x^2 + x + 3 \quad \text{for } x < 1,$$

$$f(x) = 2x^3 - 3x^2 + x + 3. \quad \text{for } x > 1,$$

$$I = \int_0^2 f(x) = 9.5$$



積分区間を分けた方がベター

# 特殊な場合: 端点で特異性

被積分関数に特異点を含む  
(通常、積分範囲の端点に現れる)

$$\int_0^1 \frac{1}{\sqrt{x}} dx = 2\sqrt{x} \Big|_0^1 = 2$$

対処法:

- 端点の値を参照しない方法を使う(例: 中点則)
- 特異性を取り除くための変数変換(例:  $x = t^2$ )

# 特殊な場合: 無限領域での積分

積分範囲が無限:

$$\int_0^\infty e^{-x} dx = 1$$

$$\int_{-\infty}^\infty e^{-x^2} dx = \sqrt{\pi}$$

対処法:

- 積分範囲が有限になるように変数変換

進んだ話題: 洗練された変数変換法として、  
2重指数型積分公式というものがある。

# 数値積分：ここまでのまとめ

- ・中点則

$$\int_a^b f(x) dx \approx (b - a) f\left(\frac{a + b}{2}\right)$$

- ・台形則

$$\int_a^b f(x) dx \approx (b - a) \frac{f(a) + f(b)}{2}$$

- ・シンプソン則

$$\int_a^b f(x) dx \approx \frac{(b - a)}{6} \left[ f(a) + 4f\left(\frac{a + b}{2}\right) + f(b) \right]$$

全て記号的に右のように書ける:  $\int_a^b f(x) dx \approx \sum_k w_k f(x_k)$

$x_k$ : 積分点、 $w_k$ : 重み

# 高次の求積法

数値積分の公式を系統的に導出する方法

$$f(x) \approx p_N(x) = \sum_{k=0}^N f(x_k) L_{N,k}(x)$$

この公式は  $f(x)$  が有限次の多項式の場合、厳密。  
 $N+1$  個の点を通る関数  $f(x)$  は次数  $N$  の多項式で  
 近似することができる：

$$f(x) \approx p_N(x) = \sum_{k=0}^N f(x_k) L_{N,k}(x)$$

$$L_{N,k}(x) = \prod_{j \neq k} \frac{x - x_j}{x_k - x_j}$$

Lagrange補間多項式

このとき、積分は

$$\int_a^b f(x) dx \approx \int_a^b p_N(x) dx = \sum_{k=0}^N w_k f(x_k) \quad \text{重みは}$$

$$w_k = \int_a^b L_{N,k}(x) dx$$

# Lagrange 補間多項式

定理:  $n+1$  個の点  $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$  を通る  
次数  $n$  の多項式が必ず存在し、一意に定まる

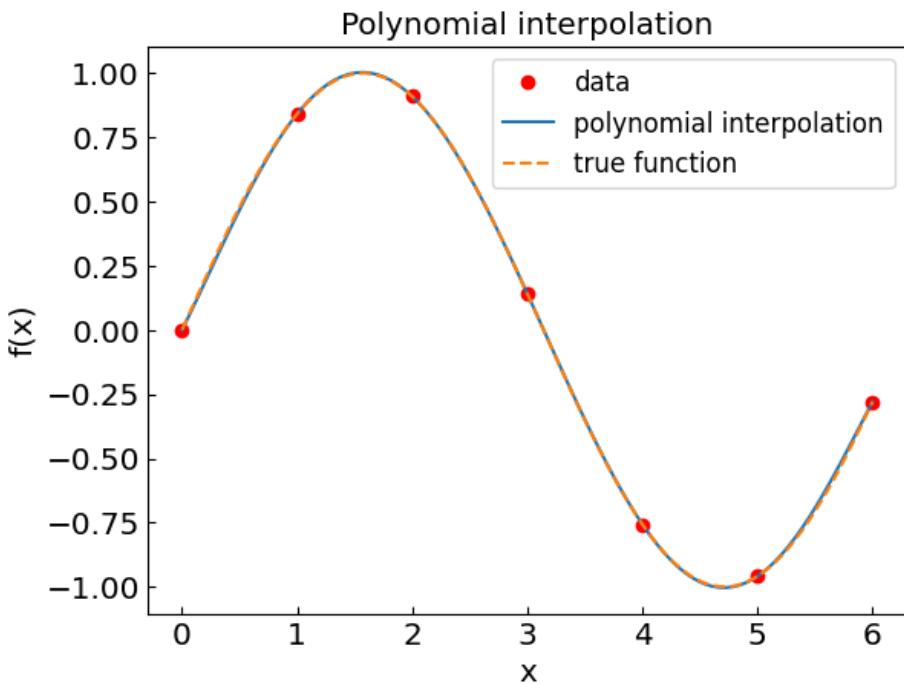
## Lagrange 補間多項式

$$L_j(x) = \prod_{k \neq j} \frac{x - x_k}{x_j - x_k}$$

$$x = x_k \text{ のとき } L_j(x_k) = \delta_{kj}$$

よって、

$$f(x) \approx p(x) = \sum_{j=0}^n y_j L_j(x)$$



# Newton-Cotes 求積法

被積分関数をLagrange補間多項式によって近似したもの

$$\int_a^b f(x) dx \approx \int_a^b p_N(x) dx = \sum_{k=0}^N w_k f(x_k)$$

積分点  $x_k$  は等間隔

- 閉じた Newton-Cotes (端点含む)

$$x_k = a + hk, \quad k = 0 \dots N, \quad h = (b - a)/N$$

N = 1: 台形則

N = 2: シンプソン則

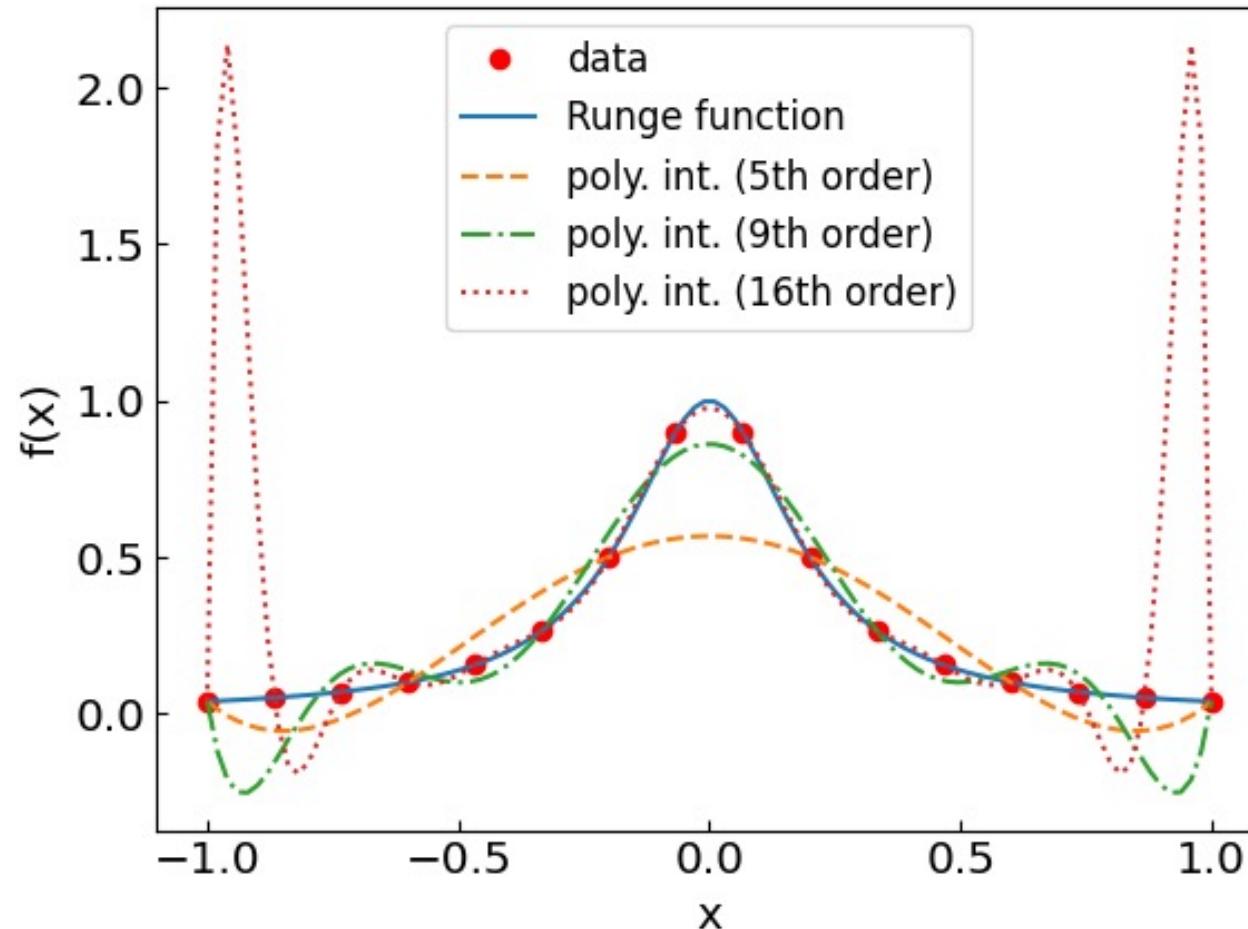
- 開いた Newton-Cotes (端点除く)

$$x_k = a + hk, \quad k = 1 \dots N + 1, \quad h = (b - a)/(N + 2)$$

N = 0: 中点則

# Runge 現象

Runge関数:  $f(x) = \frac{1}{1 + 25x^2}$



# Newton-Cotes 求積法と Runge 現象

$$I = \int_{-1}^1 \frac{dx}{1 + 25x^2} = 0.5493603\dots$$

N	I_N
0	2.0000000000000000
1	0.5294117647058825
2	-0.2988505747126436
3	0.2666666666666667
4	2.0404749055585549
5	0.9320668542657328
6	-2.0045340869981669
7	-0.1816307907657775

Runge現象により積分が激しく振動し、めちゃくちゃに

# Gauss-Legendre 求積法

$n$ 点の求積法:

$$\int_a^b f(x) dx \approx \sum_k w_k f(x_k)$$

は、 $f(x)$ が $n-1$ 次以下の多項式のとき厳密。  
任意の重なりのない積分点 $x_k$ について真。

積分点の位置 $x_k$ も自由に選ぶことができる。  
 →  $n$ 個の追加の自由度。

その場合、 $f(x)$ が $2n-1$ 次以下の多項式のときに  
**厳密**な求積法が存在。  
 → Gauss-Legendre 求積法

# Gauss-Legendre 求積法

被積分関数をLegendre級数によって近似したもの

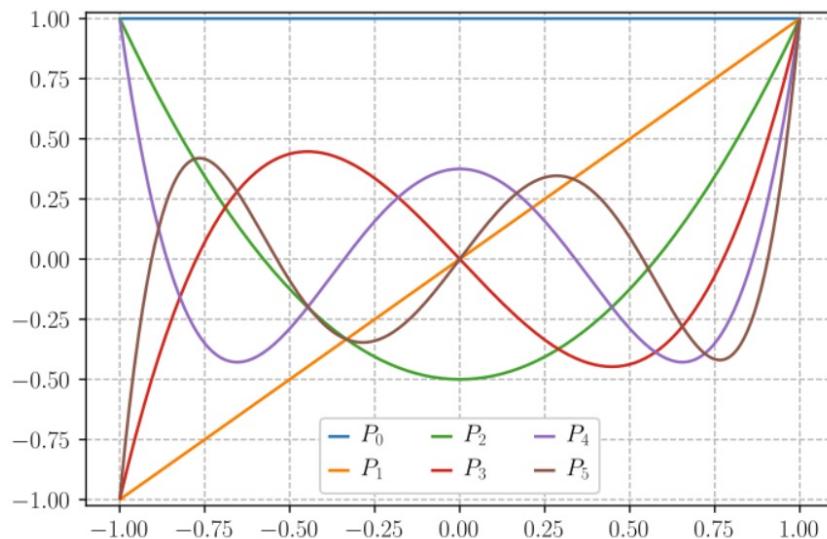
$$\int_{-1}^1 f(x)dx \approx \sum_{k=1}^n w_k f(x_k)$$

$x_k$ : Legendre多項式  $P_n(x)$  の零点

$w_k$ : 重み

$$w_k = \int_{-1}^1 L_{n-1,k}(x)dx = \frac{2}{(1-x_k^2)[P'_n(x_k)]^2} .$$

レポート問題



# 求積法の比較

- 中点・台形・シンプソン則
  - 高精度を要求しない簡易計算に適している
  - 被積分関数が滑らかでなくても良い、ノイズを含む・特異点を持つ場合、等間隔の点での計算に適している
- 適応求積法・Romberg法
  - 誤差の制御
  - 比較的滑らかな関数の等間隔の点での計算に適している
- Gauss-Legendre求積法
  - 関数が比較的滑らかなら理論的に最も高精度
  - 同じ種類の積分を何度も計算する場合に有効
  - 不等間隔の点での計算が必要
  - 滑らかでない関数の誤差制御が難しい場合がある
    - が、Kronrodによる拡張を使えば誤差が制御可能
  - 不連続な被積分関数には不向き

# 求積法の比較

方法

誤差の挙動

計算点の増え方  
(1次元で  $n$  分割した場合)

格子法(台形則など)  $\propto n^{-p}$  ( $p$  は次数)  $n^d$  (次元の呪い)

モンテカルロ法

 $\propto N^{-1/2}$  $N$  (次元依存しない)

次回 12/9



# 実習タイム

- 次の定積分を中点則とシンプソン則によって計算せよ。ただし、分割数を  $N=4, 16, 64$  の場合にとって計算せよ。

$$I = \int_0^2 x^7 dx = 32$$