科学計算研究室 Python ゼミ

～ 8. 数値積分 (Simpson 公式)～

2021-03-24 福田　浩

1　原理

1.1　考え方

関数 $f(x)$ の区間 a から b の定積分は，関数 $f(x)$ と x 軸で囲まれる面積となる．関数を細くスライスして，それぞれを $(x_i, 0), (x_i, y_i), (x_{i+1}, y_{y+1}), (x_{i+1}, 0)$ を頂点とする台形の面積 $\{f(x_i) + f(x_{i+1})\}(x_{i+1} - x_i)/2$ の総和で近似する．のが台形公式であった．この方式ではスライス幅が大きいと誤差が大きくなる．

台形公式では (x_i, y_i) と (x_{i+1}, y_{y+1}) の間を 1 次関数で近似していることと等価である．これを 2 次関数で近似するだけで，精度向上が見込める．

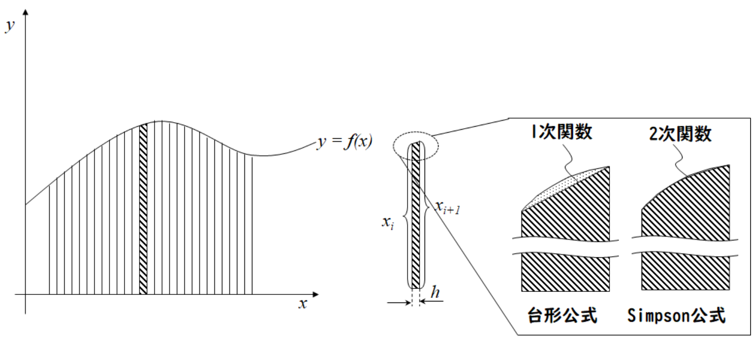

Figure 1: Simpson 公式のイメージ.

近似する 2 次関数を

$$y = \alpha x^2 + \beta x + \gamma \tag{1}$$

とすると，スライスの左端 (x_1, y_1)，中央 (x_2, y_2)，右端 (x_3, y_3) は式 1 を満たすので，

$$\begin{aligned} y_1 &= \alpha x_1^2 + \beta x_1 + \gamma & (2) \\ y_2 &= \alpha x_2^2 + \beta x_2 + \gamma & (3) \\ y_3 &= \alpha x_3^2 + \beta x_3 + \gamma & (4) \end{aligned}$$

が成り立つ．ここで，$x_2 - x_1 = h, x_3 - x_2 = h$ であることを考慮すると，

$$\begin{aligned} \alpha &= \frac{y_1 - 2y_2 + y_3}{2h^2} & (5) \\ \beta &= \frac{-(x_2 + x_3)y_1 + 2(x_3 + x_1)y_2 - (x_1 + x_2)y_3}{2h^2} & (6) \\ \gamma &= \frac{x_2 x_3 y_1 - 2x_3 x_1 y_2 + x_1 x_2 y_3}{2h^2} & (7) \end{aligned}$$

となる．式 1 を x_1 から x_3 まで積分して，式 5～7 を代入すると

$$\begin{aligned} \int_{x_1}^{x_3} (\alpha x^2 + \beta x + \gamma)\, dx &= \frac{\alpha(x_3^3 - x_1^3)}{3} + \frac{\beta(x_3^2 - x_1^2)}{2} + \gamma(x_3 - x_1) & (8) \\ &= \frac{h}{3}(y_1 + 4y_2 + y_3) & (9) \end{aligned}$$

と整理できる．

1.2　アルゴリズム

台形公式のプログラムとほぼ同じだが，16〜18 行目で，式 9 の計算を行う．

2　課題

n 次多項式の定積分に対し，Simpson 公式による数値積分プログラムを Python で実装せよ

- 積分区間 `[x1,x2]` を標準入力から入力する機能を備えること
- 多項式の次数 `d` と，台形の個数 `n` を標準入力から入力する機能を備えること
- 各係数 `a[]` を標準入力から入力する機能を備えること
- Simpson 公式による数値積分 (数値解) を求めるとともに，解析解，数値解，数値解の誤差 (%表示) を標準出力に出力する機能を備えること
- 以下の 2 つの定積分をスライスの数を 10 個，20 個，30 個の場合で求め，解析解，数値解，数値解の誤差を求めよ

$$\int_0^2 (-x^2 + 2x)\, dx$$

$$\int_1^2 (x^6 - 21x^5 + 175x^4 - 735x^3 + 1624x^2 - 1764x + 740)\, dx$$

本課題は，7. 数値積分 (台形公式) と内容がかなり近しい．協力して課題にあたり，プログラム等を共有しても良いし，それぞれ別々に取り組み，プログラムに個性を出しても良い

3 参考：C++ソースコード

```cpp
#include <iostream>
#include <math.h>
using namespace std;

double simp(int d, int n, double x1, double x2, double *a){

    double s, x, h;

    s = 0;
    x = x1;
    h = (x2-x1)/(2*n);

    for(int i=0;i<2*n-1;i=i+2){
        x = x1+h*i;
        for(int j=0;j<d+1;j++){
            s += a[j]*pow(x    ,j)*h/3.0;
            s += a[j]*pow(x+h  ,j)*h/3.0*4.0;
            s += a[j]*pow(x+h*2,j)*h/3.0;
        }
    }
    return s;
}

double exact(int d, double x1, double x2, double *a){

    double theory = 0;
    for(int i=0;i<d+1;i++) theory += a[i]/(i+1)*(pow(x2,i+1)-pow(x1,i+1));
    return theory;
}

int main(void){

    int d, n;
    double x1, x2, a[10], calc_result, theory_result;

    cin >> x1 >> x2;
    cin >> d >> n;
    for(int i=0;i<d+1;i++) cin >> a[i];

    calc_result = simp(d, n, x1, x2, a);
    theory_result = exact(d, x1, x2, a);

    cout << "approx.:" << "\t" << calc_result << endl;
    cout << "exact:"<< "\t\t" << theory_result << endl;
    cout << "error:"<< "\t\t" << theory_result - calc_result << "\t(";
    cout << (calc_result -  theory_result)/theory_result*100 << "%)" << endl;

    return 0;
}
```

4　更なる検討

Simpson 公式による数値解でも，解析解と比較して誤差が出る．誤差がマイナスの場合もあれば，プラスの場合もある．以下の 2 つの定積分を台形公式で求め，その誤差の符号を考察せよ．

$$\int_0^2 (-x^2 + 2x)\, dx$$

$$\int_0^2 (x^2 - 2x + 1)\, dx$$

ヒント：可視化 (グラフ化) してみること．