科学计算引论作业(七)

谢悦晋　U202210333

Nov 13rd, 2023

5.1 试分别利用中矩公式、梯形公式及 Simpson 公式计算定积分

$$I=\int_0^{\frac{1}{2}}\exp(3x)\cos 2x\mathrm{d}x,$$

并比较其计算精度。

解：

中矩形公式：

$$\int_0^{\frac{1}{2}}\exp(3x)\cos 2x\mathrm{d}x\approx(\frac{1}{2}-0)f(\frac{1}{4})=0.9289211490504564$$

梯形公式：

$$\int_0^{\frac{1}{2}}\exp(3x)\cos 2x\mathrm{d}x\approx(\frac{1}{2}-0)\frac{f(\frac{1}{2})+f(0)}{2}=0.8553667347219239$$

Simpson公式：

$$\int_0^{\frac{1}{2}}\exp(3x)\cos 2x\mathrm{d}x\approx\frac{\frac{1}{2}-0}{6}(f(0)+4f(\frac{1}{4})+f(\frac{1}{2}))=0.9044030109409457$$

比较准确的解为$I=0.9082171883002617,\varepsilon=1.0083236338112719e-14$，可以看出Simpson公式精度最好

5.2 试确定下面求积公式

$$\int_{-1}^{1}f(x)\mathrm{d}x\approx a[f(x_0)+f(x_1)+f(x_2)],$$

使其具有 3 次代数精度，并由该公式计算定积分：

$$\int_{-1}^{1}\frac{x\sin x}{\sqrt{1+x^2}}\mathrm{d}x$$

解：

由代数精度的定义可以得到以下方程：

$$\begin{cases} a(1+1+1)=2 \\ a(x_0+x_1+x_2)=0 \\ a(x_0^2+x_1^2+x_2^2)=\dfrac{2}{3} \\ a(x_0^3+x_1^3+x_2^3)=0 \\ a(x_0^4+x_1^4+x_2^4)\neq 0 \end{cases} \Rightarrow \begin{cases} a=\dfrac{2}{3} \\ x_0=-\dfrac{\sqrt{2}}{2} \\ x_1=0 \\ x_2=\dfrac{\sqrt{2}}{2} \end{cases}$$

计算结果如下：

```
def f(x):
    return x * np.sin(x)/np.sqrt(1+x**2)

integrate.quad(f, -1, 1),2/3 * (f(-np.sqrt(2)/2) + f(0) + f(np.sqrt(2)/2))
```

Executed at 2023.11.08 22:20:33 in 4ms

```
((0.4833545313354184, 9.03548815670577e-09), 0.5000907488711981)
```

5.5 试构造两点 Gauss 型求积公式

$$\int_{-1}^{1} f(x)\mathrm{d}x \approx A_0 f(x_0)+A_1 f(x_1),$$

并由此计算积分:

$$\int_0^1 \sqrt{1+2x}\mathrm{d}x$$

解：

由代数精度的定义可以得到以下方程：

$$\begin{cases} A_0+A_1=2 \\ A_0x_0+A_1x_1=0 \\ A_0x_0^2+A_1x_1^2=\dfrac{2}{3} \\ A_0x_0^3+A_1x_1^3=0 \end{cases} \Rightarrow \begin{cases} A_0=1 \\ A_1=1 \\ x_0=-\dfrac{1}{\sqrt{3}} \\ x_1=\dfrac{1}{\sqrt{3}} \end{cases}$$

对于原积分不能直接使用上述的得到的Gauss积分公式，我们进行一个积分变换：

$$\int_0^1 \sqrt{1+2x}\mathrm{d}x \xlongequal{t=2x-1} \frac{1}{2}\int_{-1}^{1} \sqrt{2+t}\mathrm{d}t$$

计算结果如下：

```
def f(x):
    return np.sqrt(1+2 * x)

integrate.quad(f, 0, 1), 0.5 * (np.sqrt(2+1/np.sqrt(3)) + np.sqrt(2-1/np.sqrt(3)))
Executed at 2023.11.08 22:52:17 in 1ms
```

```
((1.3987174742355442, 1.5528883448418452e-14), 1.3990808081581056)
```

5.7 (实验题) 分别用变步长梯形求积公式和 Romberg 算法计算椭圆积分

$$\int_0^{\pi} \frac{\sqrt{2}}{(1+\sin^2 x)\sqrt{2-\sin^2 x}}\mathrm{d}x$$

要求其逼近值 $T_k, T_k^{(0)}$ 的计算精度分别满足 $|T_k - T_{k-1}| < 10^{-12}$ 和$|T_k^{(0)} - T_{k-1}^{(0)}| < 10^{-12}$.

解：

结果如下：

```
import numpy as np
from scipy import integrate

def trapezoid(f,a,b,N):
    h   = (b-a)/N
    xi  = np.linspace(a,b,N+1)
    fi  = f(xi)
    s   = 0.0
    for i in range(1,N):
        s = s + fi[i]
    s = (h/2)*(fi[0] + fi[N]) + h*s
    return s
# romberg method
def romberg(f,a,b,eps,nmax):
    Q = np.zeros((nmax,nmax),float)
    converged = 0
    for i in range(0,nmax):
        N= 2**i
        Q[i,0] = trapezoid(f,a,b,N)
        for k in range(0,i):
            n = k + 2
            Q[i,k+1] = 1.0/(4**(n-1)-1)*(4**(n-1)*Q[i,k] - Q[i-1,k])
        if (i > 0):
            if (abs(Q[i,k+1] - Q[i,k]) < eps):
                converged = 1
                break
    print(Q[i,k+1],N,abs(Q[i,k+1] - Q[i,k]))
    return Q[i,k+1],N,converged


f = lambda x: np.sqrt(2) / ((1 + np.sin(x) ** 2) * np.sqrt(2-np.sin(x) ** 2))
romberg(f, 0, np.pi, 1e-12, 12)

result = integrate.romberg(f, 0, np.pi, tol=1e-12, divmax=12, show=True)
Executed at 2023.11.13 08:48:17 in 4ms
```

```
2.546254733498455 128 3.339550858072471e-13
Romberg integration of <function vectorize1.<locals>.vfunc at 0x0000027A239B2480> from [0, 3.141592653589793]

 Steps  StepSize   Results
     1  3.141593  3.141593
     2  1.570796  2.681517  2.528159
     4  0.785398  2.549958  2.506105  2.504635
     8  0.392699  2.546258  2.545024  2.547619  2.548301
    16  0.196350  2.546255  2.546254  2.546336  2.546315  2.546308
    32  0.098175  2.546255  2.546255  2.546255  2.546254  2.546253  2.546253
    64  0.049087  2.546255  2.546255  2.546255  2.546255  2.546255  2.546255  2.546255
   128  0.024544  2.546255  2.546255  2.546255  2.546255  2.546255  2.546255  2.546255  2.546255

The final result is 2.546254733498458 after 129 function evaluations.
```