数值积分与数值微分上机实习报告

姓名:刘群 学号:2014211591 院系: 地学中心

Email: liu-q14@mails.tsinghua.edu.cn

(本文档由LaTeX编写)

1 问题的描述

试用不同数值积分方法计算$I(f) = \int_1^3 f(x)dx$ 的近似值, 其中$f(x) = \frac{1}{x^2}\sin\frac{2\pi}{x}$. 注：$I(f) = -0.238732414637843\cdots$.

1、把[1,3]分成4个子区间，用五点Gauss-Legendre 求积公式的复合求积公式计算。

2、用Romberg 求积算法计算积分，取$\varepsilon = 10^{-7}$, 并与第一种办法比较。

2 方法描述

在数值计算时我经常要计算某些积分的值, 一般来说, 对于比较简单的函数的积分来说, 我们可以通过Newton-Leibniz 公式得到原函数, 然后再进行计算. 但是现实中, 有些情况下原函数特别复杂, 有些时候根本没有可以解析的原函数. 另外, 很多时候我们并不需要知道原函数的表达式, 只是希望得到一个积分的结果. 在这种情况下, 我们就需要用到数值积分的方法来计算积分的值. 常见的数值积分方法有Newton-Cotes公式、复化求积公式、Gauss求积公式、Romberg求积等.

2.1 Gauss-Legendre公式

Gauss求积公式与Newton-Cotes 公式的区别就是, 求积公式中的节点$x_0, x_1, \cdots, x_n$ 选取的是[a,b]上以ρ为权的$n+1$次正交多项式的零点. 对于本题中, 由于权函数为1, 因此可以选择Legendre多项式的根(Legendre多项式是[-1,1]上以$\rho(x) \equiv 1$的正交多项式). 这样, 我们可以得到Gauss-Legendre求积的表达式如下:

$$\int_{-1}^{1} f(x)dx \approx \sum_{k=0}^{n} A_k f(x_k) \tag{1}$$

其中, Gauss点$x_0, x_1, \cdots, x_n$为$n+1$次Legendre正交多项式P_{n+1}的零点. 公式(1)中的系数A_k的表达式如下:

$$A_k = \frac{2}{(1-x_k^2)[P'_{n+1}(x_k)]^2}, k = 0, 1, \cdots, n. \tag{2}$$

或者

$$A_k = \frac{2}{n+1}\frac{1}{P_n(x_k)P'_{n+1}(x_k)}, k = 0, 1, \cdots, n. \tag{3}$$

其中(2)式和(3)式是等价的, 可以通过Legendre 多项式的性质得到.

Gauss-Legendre求积公式中的节点x_k和系数A_k可以实现计算得到, 如下表所示.

Table 1: Gauss-Legendre求积的节点和系数

n	x_k	A_k
1	0	2
2	±0.5773502692	1
3	±0.7745966692	0.5555555556
	0	0.8888888889
4	±0.8611363116	0.3478548451
	±0.3399810436	0.6521451549
5	±0.9061798459	0.2369268851
	±0.5384693101	0.4786286705
	0	0.5688888889
6	±0.9324695142	0.1713244924
	±0.6612093865	0.3607615730
	±0.2386191861	0.4679139346

在一般的区间[a,b]上进行求积, 如果采用Gauss-Legendre求积公式, 可以按照下式进行变量替换,

$$x = \frac{1}{2}(a+b) + \frac{1}{2}(b-a)t \tag{4}$$

从而可以使区间[a,b]变换到[-1,1], 并且有

$$\int_a^b f(x)dx = \frac{a+b}{2}\int_{-1}^1 f\left[\frac{1}{2}(a+b) + \frac{1}{2}(b-a)t\right]dt \tag{5}$$

对于本题中, 如果我们仅仅是采用Gauss-Legendre求积公式的话, 可以仅仅做变换

$$x = 2 + t \tag{6}$$

得到

$$\int_1^3 f(x)dx = 2\int_{-1}^1 f(2+t)dt = 2\int_{-1}^1 \frac{1}{t^2}\sin\frac{2\pi}{t}dt \tag{7}$$

但是, 这里我们要求采用复合五点Gauss-Legendre求积公式, 需要将区间[1,3]分为4个子区间, 因此, 我们下面就介绍一下如何构造复合五点Gauss-Legendre求积公式.

2.2 复合Gauss-Legendre求积公式

我们可以看到, 如果一味的提高求积公式的阶数, 有时候并不能获得很高的精度, 但是我们可以采用复合求积可以提高求积分的精度. 方法是将整个积分区间分成若干个子区间(通常是等分), 再在每个子区间上采用低阶的求积公式, 这样使整个区间的积分获得较高的精度.

本题中, 我们要求解决的是五点Gauss-Legendre求积的复合求积公式, 并要求将区间[1,3]分为4 个子区间, 因此有

$$\int_1^3 f(x)dx = \int_1^{\frac{3}{2}} f(x)dx + \int_{\frac{3}{2}}^2 f(x)dx + \int_2^{\frac{5}{2}} f(x)dx + \int_{\frac{5}{2}}^3 f(x)dx \tag{8}$$

其中, $f(x) = \frac{1}{x^2}\sin\frac{2\pi}{x}$.

接下来, 在每个小区间上,我们分别按照公式(4)进行变量变换, 从而可以得到

Table 2: 变量转换的结果

$x \in [a,b]$	$t \in [-1,1]$	$x \to t$
$x \in [1, \frac{3}{2}]$	$t \in [-1,1]$	$x = \frac{1}{4}t + \frac{5}{4}$
$x \in [\frac{3}{2}, 2]$	$t \in [-1,1]$	$x = \frac{1}{4}t + \frac{7}{4}$
$x \in [2, \frac{5}{2}]$	$t \in [-1,1]$	$x = \frac{1}{4}t + \frac{9}{4}$
$x \in [\frac{5}{2}, 3]$	$t \in [-1,1]$	$x = \frac{1}{4}t + \frac{11}{4}$

从而有方程变为

$$\begin{aligned}\int_1^3 f(x)dx &= \frac{1}{4}\int_{-1}^1 f\left(\frac{t}{4}+\frac{5}{4}\right)dt + \frac{1}{4}\int_{-1}^1 f\left(\frac{t}{4}+\frac{7}{4}\right)dt \\ &+ \frac{1}{4}\int_{-1}^1 f\left(\frac{t}{4}+\frac{9}{4}\right)dt + \frac{1}{4}\int_{-1}^1 f\left(\frac{t}{4}+\frac{11}{4}\right)dt\end{aligned} \tag{9}$$

其中, $f(x)=\frac{1}{x^2}\sin\frac{2\pi}{x}$. 根据Table 1中的数据, 对于式(9)中右边的每一项,有

$$\begin{aligned}\frac{1}{4}\int_{-1}^{1} f\left(\frac{t}{4}+\frac{5}{4}\right)dt=\frac{1}{4}\Big[&0.2369268851\times\frac{16}{(5+0.9061798459)^2}\sin\frac{8\pi}{5+0.9061798459}\\&+0.2369268851\times\frac{16}{(5-0.9061798459)^2}\sin\frac{8\pi}{5-0.9061798459}\\&+0.4786286705\times\frac{16}{(5+0.5384693101)^2}\sin\frac{8\pi}{5+0.5384693101}\\&+0.4786286705\times\frac{16}{(5-0.5384693101)^2}\sin\frac{8\pi}{5-0.5384693101}\\&0.5688888889\times\frac{16}{(5+0)^2}\sin\frac{8\pi}{5+0}\Big]\end{aligned}\tag{10}$$

$$\begin{aligned}\frac{1}{4}\int_{-1}^{1} f\left(\frac{t}{4}+\frac{7}{4}\right)dt=\frac{1}{4}\Big[&0.2369268851\times\frac{16}{(7+0.9061798459)^2}\sin\frac{8\pi}{7+0.9061798459}\\&+0.2369268851\times\frac{16}{(7-0.9061798459)^2}\sin\frac{8\pi}{7-0.9061798459}\\&+0.4786286705\times\frac{16}{(7+0.5384693101)^2}\sin\frac{8\pi}{7+0.5384693101}\\&+0.4786286705\times\frac{16}{(7-0.5384693101)^2}\sin\frac{8\pi}{7-0.5384693101}\\&0.5688888889\times\frac{16}{(7+0)^2}\sin\frac{8\pi}{7+0}\Big]\end{aligned}\tag{11}$$

$$\begin{aligned}\frac{1}{4}\int_{-1}^{1} f\left(\frac{t}{4}+\frac{9}{4}\right)dt=\frac{1}{4}\Big[&0.2369268851\times\frac{16}{(9+0.9061798459)^2}\sin\frac{8\pi}{9+0.9061798459}\\&+0.2369268851\times\frac{16}{(9-0.9061798459)^2}\sin\frac{8\pi}{9-0.9061798459}\\&+0.4786286705\times\frac{16}{(9+0.5384693101)^2}\sin\frac{8\pi}{9+0.5384693101}\\&+0.4786286705\times\frac{16}{(9-0.5384693101)^2}\sin\frac{8\pi}{9-0.5384693101}\\&0.5688888889\times\frac{16}{(9+0)^2}\sin\frac{8\pi}{9+0}\Big]\end{aligned}\tag{12}$$

$$
\begin{aligned}
\frac{1}{4}\int_{-1}^{1} f\left(\frac{t}{4}+\frac{11}{4}\right)dt = \frac{1}{4}\Bigg[&0.2369268851\times\frac{16}{(11+0.9061798459)^2}\sin\frac{8\pi}{11+0.9061798459}\\
&+0.2369268851\times\frac{16}{(11-0.9061798459)^2}\sin\frac{8\pi}{11-0.9061798459}\\
&+0.4786286705\times\frac{16}{(11+0.5384693101)^2}\sin\frac{8\pi}{11+0.5384693101}\\
&+0.4786286705\times\frac{16}{(11-0.5384693101)^2}\sin\frac{8\pi}{11-0.5384693101}\\
&0.5688888889\times\frac{16}{(11+0)^2}\sin\frac{8\pi}{11+0}\Bigg]
\end{aligned}
\tag{13}
$$

这样就可以得到用计算得到了用五点Gauss-Legendre积分公式的复合公式的计算公式.

2.3 Romberg积分

Romberg 求积方法是一种数值积分的加速算法, 它可以看成是Richardson 外推方法的一种特例, 下面将Richardson外推方法介绍如下: 假定$F(h)$, 当$h\to 0$时有$F(h)\to F^*$(F^*与h无关), 并有

$$
F^*-F(h)=\sum_{k=1}^{\infty}\alpha_k h^{p_k}, 0<p_1<p_2<\cdots \tag{14}
$$

其中p_k,α_k为与h无关的常数, $\alpha_k\neq 0, h\geqslant 1$,则由

$$
\begin{aligned}
&F_1(h)=F(h)\\
&F_{m+1}(h)=\frac{F_m(qh)-q^{p_m}F_m(h)}{1-q^{p_m}}, m=1,2,\cdots
\end{aligned}
\tag{15}
$$

确定的序列$\{F_m(h)\}$有

$$
F^*-F(h)=\sum_{k=1}^{\infty}\alpha_{m+k}^{(m+1)}h^{p_{m+k}}, \tag{16}
$$

式中$\alpha_{m+k}^{(m+1)}$为与h无关的非零常数, $0<q<1$.

特别地, 如果我们在Richardson外推算法中取$q=\frac{1}{2}, p_k=2k$, 则可以得到Romberg 求积方法, 即

$$
\begin{aligned}
(T_1 f)(h) &= \frac{h}{2}\sum_{m=1}^{n}[f(x_{m-1})+f(x_m)], \\
(T_{j+1} f)(h) &= \frac{4^j (T_j f)\left(\frac{h}{2}\right)-(T_j f)(h)}{4^j-1}, j=1,2,\cdots
\end{aligned}
\tag{17}
$$

在Romberg积分中, 我们可以引入记号

$$
T_j^k f=(T_j f)\left(\frac{h}{2^k}\right), j=1,2,\cdots
$$

此称为Romberg数列, 在未外推时(即T_1^k), k表示在复合梯形公式中区间[a, b]分成2^k等份, 此时有(17)式可以表示为

$$
T_{j+1}^k f=\frac{4^j T_j^{k+1} f-T_j^k f}{4^j-1} \tag{18}
$$

从而, Romberg求积方法的计算过程如下:

(1) 求梯形面积$T_1^0 f=\frac{h}{2}[f(a)+f(b)], h=b-a$.

(2) 将区间[a,b]分半, 求出两个小梯形面积之和, 记为$T_1^1 f$, 应用公式(18)有

$$
T_2^0 f=\frac{4T_1^1 f-T_1^0 f}{4-1} \quad \text{Simpson求积公式}
$$

置$l=1$, 转入(4).

(3) 对区间[a,b]作2^l等分,其符合梯形求积值记为$T_1^l f$, 构造数列

$$
T_{j+1}^{k-1} f=\frac{4^j T_j^k f-T_j^{k-1} f}{4^j-1}
$$

由此可得$T_{l+1}^0 f$.

(4) 若$|T_l^0 f-T_{l+1}^0 f|<varepsilon$(预先给定的误差控制), 则停止计算, 否则转(3).

当误差小于ε时, 我们就取T_{l+1}^0的值作为定积分的值.在本题中, 我们取$\varepsilon=10^{-7}$, 然后计算出最后的结果.

3 方案设计

我们首先编写了一个主函数quadrature.m, 在里面定义了区间[a,b], 函数表达式等信息, 然后分别调用相应的函数进行积分求解. 其中, 调用CompoundGLfive.m 函数计算五点Gauss-Legendre积分公式的复合公式的计算结果, 调用Romberg.m来计算Romberg积分的值. 最后计算了二者结果各自的绝对误差和相对误差.

针对第一个问题, 即五点Gauss-Legendre积分公式的复合公式, 我们编写了一个函数来计算. (文件名为CompoundGLfive.m) 该函数主要就是计算五点Gauss-Legendre 积分复合公式在区间[a, b] 上的值, 其中它有四个参数, 分别是a(区间端点最小值)、b(区间端点最大值)、n(复合求积中区间分的段数)、f(要求积的函数的表达式, 以字符串的形式). 这样我们就可以返回求积计算的结果了. 在程序中, 我们主要用符号函数表示了要求积分的函数, 并且通过定义t, 完成了变量替换, 构成了新的积分的函数, 然后代入五点Gauss-Legendre求积公式的节点值和系数值, 这样就求出了一个小区间上的积分的值. 最后在外层循环中, 将所有的区间段上的积分加和, 就得到了最终的计算结果. 调用格式为CompoundGLfive(1, 3, 4, '1/x^2*sin(2*pi/x)').

针对Romberg积分, 我们首先编写了函数CompoundTrapezoid.m来计算复合梯形公式的积分, 其中该函数有4个参数, 分别是f,a,b,n. 其中f是要求积分的函数(以字符串的形式表示),a、b分别为区间[a,b]的端点, n 为复合求积中区间[a,b]要分的份数. 在函数中, 同样, 我们采用了Matlab内置的函数将字符串表示的函数转化为句柄函数.该函数的返回值即为复合积分的结果. 接下来, 我们按照前面介绍的Romberg求积算法的流程编写了函数Romberg.m来计算Romberg积分, 其中, 在该函数中, 我们调用了CompoundTrapezoid.m函数来计算中间用到的复合梯形求积的值. Romberg.m函数共有4个参数, 分别为a, b, epsilon, f. 其中, a和b为区间[a,b]的端点值, epsilon为控制误差, f 为要积分的函数, 以字符串的形式表达. 在函数中, 我们通过一个while循环对算法的整个流程进行控制, 当误差小于epsilon时程序停止. 同时, 需要特别指出的是, 在程序中, 我特别引入了一个矩阵T, 将Romberg积分中的每一个中间结果都记录了下来.

4 计算结果及其分析

计算的结果如下(真值为$I(f) = -0.238732414637843\cdots$):

(1) 当我们采用五点Gauss-Legendre公式的复合公式进行计算, 并将区间[1, 3]分为四个小段后, 计算的结果为$I(f) \approx -0.238732340666179$, 并且可以得出该方法的绝对误差为$7.3971663683281 \times 10^{-8}$, 相对误差为$3.09851780268281 \times 10^{-7}$.

(2) 当我们采用Romberg积分时, 我们得到的积分结果为$I(f) \approx -0.238732414621623$, 并且每一步计算的结果如Table 3 所示, 并且可以得出该方法的绝对误差为$1.62196922559588 \times 10^{-11}$,

相对误差为$6.79408880464097 \times 10^{-11}$.

Table 3: Romberg积分的结果

k	T_1^k	T_2^k	T_3^k	T_4^k	T_5^k	T_6^k	T_7^k	T_8^k
0	0.096225045							
1	0.048112522	0.032075015						
2	-0.121371008	-0.177865519	-0.191861554					
3	-0.206400287	-0.23474338	-0.238535237	-0.239276089				
4	-0.230583448	-0.238644501	-0.238904576	-0.238910438	-0.238909005			
5	-0.23669501	-0.238732198	-0.238738044	-0.238735401	-0.238734714	-0.238734544		
6	-0.238223125	-0.238732497	-0.238732516	-0.238732429	-0.238732417	-0.238732415	-0.238732414	
7	-0.238605097	-0.238732421	-0.238732416	-0.238732415	-0.238732415	-0.238732415	-0.238732415	-0.238732415

两种方法所得的数值结果对比如Table 4 所示. 从中可以看出, 采用Romberg积分计算的误差较小.

Table 4: 两种求积方法结果对比

求积方法	真值	计算所得结果	绝对误差	相对误差
五点Gauss-Legendre复合求积	$-0.238732414637843\cdots$	-0.238732340666179	$7.3971663683281 \times 10^{-8}$	$3.09851780268281 \times 10^{-7}$
Romberg积分	$-0.238732414637843\cdots$	-0.238732414621623	$1.62196922559588 \times 10^{-11}$	$6.79408880464097 \times 10^{-11}$

5 结论

无论是五点Gauss-Legendre复合积分公式, 还是Romberg积分, 二者都能在很高的精度下求得函数的积分, 但是从计算的结果我们也可以看出, 采用Romberg积分计算的相对误差更小, 更接近真值, 且收敛速度很快. 因此, 在实际中采用Romberg 积分的效果较好.

6 附录: Matlab程序

(1) **quadrature.m**

```
% 数值积分和数值微分上机实验
% Qun Liu 2014-12-24
clear;
clc;
% 定义要计算的函数f
f='1/x^2*sin(2*pi/x)';
% 定义区间[a,b]
a = 1;
b = 3;
```

```
% 求五点Gauss–复合求积Legendre分成个子区间,4
n = 4;
fGL = CompoundGLfive(a, b, n, f);
% 积分Romberg
% 定义误差
epsilon = 1e−7;
[fR, T] = Romberg(a, b, epsilon, f);
% 定义真值y
y=−0.238732414637843;
% 计算绝对误差和相对误差
fGL_abs = abs(fGL−y);
fGL_rel = fGL_abs /abs(y);
fR_abs = abs(fR−y);
fR_rel = fR_abs /abs(y);
disp(['五点Gauss−复合求积Legendre分成(' num2str(n) '段)的值为:' num2str(fGL,15) ', 绝对误差
    为' num2str(fGL_abs,15) ', 相对误差为' num2str(fGL_rel,15)])
disp(['积分的值为Romberg:' num2str(fR,15) ', 绝对误差为' num2str(fR_abs,15) ', 相对误差
    为' num2str(fR_rel,15)])
```

(2) **CompoundGLfive.m**

调用格式为: sum=CompoundGLfive(1, 3, 4, '1/x^2*sin(2*pi/x)');

```
function sum = CompoundGLfive(a, b, n, f)
% Gauss−Legendre Quadrature
% 采用五点Gauss−求积公式Legendre
% 其中, a, 为区间b[a, b的端点值], 为复合求积中要分的段数n, 为要求积分的函数f
% Qun Liu 2014−12−24
h = (b−a)/n;
aa = a : h : b−h;
bb = a+h : h : b;
% 积分的累积结果
sum = 0;
syms t
for i = 1 : n
    x2t = 0.5*( aa(i)+bb(i) ) + 0.5*( bb(i)−aa(i) )*t;
    f = subs(f, 'x', x2t);
    f = eval(['@(t)',vectorize(f)]);
    sum = sum + (bb(i)−aa(i))/2*0.5*(bb(i)−aa(i)) * ( f(0.9061798459)
        *0.2369268851...
        + f(−0.9061798459)*0.2369268851 + f(0.5384693101)*0.4786286705...
        + f(−0.5384693101)*0.4786286705+ f(0)*0.5688888889 );
end
```

(3) **CompoundTrapezoid.m**

```
function r=CompoundTrapezoid(f,a,b,n)
% Computing Compound Trapezoid Integration);
% f is a string containing x, denote a function
% Example: 'x/(4+x^2)'
```

```
% a,b are the start and end point of the interval
% n is the number of small segments of interval
% Qun Liu 2014-12-20

f=eval(['@(x)',vectorize(f)]);
h = (b-a)/n;
r = h/2*(f(a)+2*sum(f(a+h:h:b-h))+f(b));
```

(4) **Romberg.m**

```
function [f, T] = Romberg(a, b, epsilon, f)
% 计算函数的积分fRomberg
% a, b 为区间端点值,为控制误差epsilon, 为关于的函数fx, 字符串形式
% f 为计算的积分的结果, T 为计算过程中所有的中间结果
% Qun Liu 2014-12-24

L=1; % 区间[a,b要分割的份数]
T(1,1)=CompoundTrapezoid(f,a,b,L);
T(2,1)=CompoundTrapezoid(f,a,b,L+1);
T(1,2) = (4*T(2,1) - T(1,1)) / (4-1);
while(abs(T(1,L)-T(1,L+1)) > epsilon)
    L = L+1;
    T(L+1,1) = CompoundTrapezoid(f,a,b,2^L);
    for j=1:L
        k= L-j+1;
        T(k, j+1) =  (4^j*T(k+1, j) - T(k, j)) / (4^j-1);
    end
end
f = T(1, L+1);
for j=2:L+1
    T(j:L+1, j)=T(1:L+2-j, j);
    T(1:j-1,j)=0;
end
```