数值积分 Newton-cotes 公式误差估计证明、cotes 系数的计算及 Newton-cotes 公式局限性的具体分析

韩之楠 曹绍祯
（武汉大学数学与统计学院，武汉市武昌区，430072）

摘要：

本文分成三部分，分别是 Newton-cotes 公式误差估计证明、cotes 系数的计算、Newton-cotes 公式局限性的具体分析。

第一部分证明：引入 $\omega_n(x)$ 的原函数 $\Omega_n(x)$，将误差估计转化为对插值误差的积分，再将 Lagrange 插值转化为对 Newton 插值误差的积分，最后通过研究差商的性质得到结论。

第二部分由 Newton-cotes 公式的构造导出一种与求解 Vandermonde 方程组有关的计算方法。

第三部分列出 Newton-cotes 公式的三条局限性。

关键词：数值积分、Newton-Cotes 公式，误差估计

The proof of error estimation of Newton Cotes formula for numerical integration, the calculation of Cotes coefficient and the concrete analysis of the limitation of Newton Cotes formula

Zhinan Han　　Shaozhen Cao
(School of mathematics and statistics, Wuhan University, Wuchang District, Wuhan City,430072)

Abstract:

This paper is divided into three parts, which are the proof of error estimation of Newton Cotes formula, the calculation of Cotes coefficient, and the specific analysis of the limitations of Newton Cotes formula.

In the first part, we prove that the error estimation is transformed into the integral of the interpolation error by introducing the original function, and then the Lagrange interpolation is transformed into the integral of the Newton interpolation error. Finally, we get the conclusion by studying the properties of the difference quotient.

In the second part, a calculation method related to solving Vandermonde equations is derived from the construction of Newton Cotes formula.

The third part lists three limitations of Newton Cotes formula.

Key words：Numerical integration，Newton-Cotes formula，error estimation

1、Newton-cotes 公式误差估计的证明

1.1 往证定理：若 $f(x)\in C^{n+2}[a,b]$，在该区间上取 n+1 个分点的等距划分

$a = x_0 < x_1 < ... < x_n = b$，取数值积分系数 Ak 为 Lagrange 基函数在（a，b）上的积分，记数值积分 $I_n(f) = \sum_{i=0}^{n} A_i f(x_i)$ ，记 $I(f)$ 为 $f(x)$ 在 (a,b) 上积分的精确值，则有如下误差估计：

$$I(f) - I_n(f) = \begin{cases} \dfrac{(b-a)^{n+2}}{n^{n+2}(n+1)!}\displaystyle\int_0^n t(t-1)...(t-n)dt f^{(n+1)}(\eta) & n\text{为奇数} \\ \dfrac{(b-a)^{n+3}}{n^{n+3}(n+2)!}\displaystyle\int_0^n t(t-1)...(t-n)dt f^{(n+2)}(\eta) & \text{n为偶数} \end{cases}$$

问题分析：由数值积分的格式和积分系数易得，Newton-cotes 公式计算积分本质上是对 f（x）的插值多项式进行积分，故其误差估计可以转化为如下形式（具体证明可见参考书目 2）：

$$e_n(f) = \int_a^b \omega_n(x) f[x_0,..,x_n,x]dx$$

接下来的处理需要以下引理：

1.1.1 引理：对于区间 $[a,b]$ 上的等距划分 $a = x_0 < x_1 ... < x_n = b$，定义

$$x_{n/2} = \frac{a+b}{2} = x_0 + \frac{n}{2}h \text{（无论 n 为奇偶）} \qquad \omega_n(x) = \prod_{i=o}^{n}(x - x_i)$$

则 $\omega_n(x)$ 具有以下性质：

（1） $\forall\xi \quad \omega_n(x_{n/2} + \xi) = (-1)^{n+1}\omega_n(x_{n/2} - \xi)$，即 $\omega_n(x)$ 关于区间中点有对称性。

（2） i） 对于 $a < \xi + h \le x_{n/2} \quad \xi \ne x_j \ \ j = 0,1,...n$，有

$$|\omega_n(\xi + h)| < |\omega_n(\xi)|$$

ii） 对于 $x_{n/2} \le \xi < b \quad \xi \ne x_j \ \ j = 0,1,...n$， 有

$$|\omega_n(\xi)| < |\omega_n(\xi + h)|$$

此性质意味着越靠近区间的中点，$\omega_n(x)$ 的绝对值便越小。

证明：（1）显然，$\omega_n(x_{n/2} + \xi)$，$\omega_n(x_{n/2} - \xi)$ 都是关于 ξ 的 n+1 阶多项式，并且具有相同的 n+1 个零点

$$x_0 - x_{n/2}, x_1 - x_{n/2}, ...x_n - x_{n/2}$$

故二者只相差常数倍，比较最高次项的系数即得结论。

（2）两式相除

$$\left|\frac{\omega_n(\xi+h)}{\omega_n(\xi)}\right| = \left|\frac{(\xi+h-x_0)(\xi+h-x_1)\ldots(\xi+h-x_n)}{(\xi-x_0)(\xi-x_1)\ldots(\xi-x_n)}\right|$$
$$= \left|\frac{\xi+h-x_0}{\xi-x_n}\right| < 1$$

对于另外一侧的结论，只用使用（1）中的结论即可。

1.1.2 引理：若 $f(x) \in C^2[a,b]$， $x_j \in [a,b]$ ，定义

$$g_j(x) = \begin{cases} f[x, x_j] & x \neq x_j \\ f'(x_j) & x = x_j \end{cases} \qquad g_j'(x) = \begin{cases} \dfrac{df[x, x_j]}{dx} & x \neq x_j \\ \dfrac{1}{2} f''(x_j) & x = x_j \end{cases}$$

显然 $g(x)$ 连续，由泰勒公式将 f(x)在 x_j 处展开至 2 阶即可验证 $g'(x)$ 也是连续的。

1.1.3 引理：如果 $x, x_0, \ldots, x_n$ 是 n+2 个不同的点，则:

$$f[x_0, \ldots, x_n, x] = \sum_{j=0}^{n} \frac{f[x, x_j]}{\prod\limits_{k=0, k\neq j}^{n} (x_j - x_k)}$$

证明：由牛顿插值的误差估计和拉格朗日插值的格式，可将右边分子展开直接验证之。

1.1.4 引理：设 $a = x_0 < x_1 < \ldots < x_n = b$ 是区间 $[a,b]$.的一个划分，则在节点上适当补充定义后，以下两个函数都是连续的

$$f[x_0, \ldots x_n, x] \quad \frac{d}{dx} f[x_0, \ldots x_n, x]$$

证明：在引理 3 中记

$$c_{j=} \frac{1}{\prod\limits_{k=0, k\neq j}^{n} (x_j - x_k)}$$

则补充定义

$$f[x_0, \ldots x_n, x] = g(x) = \sum_{j=0}^{n} c_j g_j(x) \qquad \frac{d}{dx} f[x_0, \ldots x_n, x] = g(x) = \sum_{j=0}^{n} c_j g_j^{'}(x)$$

由引理 2 即得结论。

1.1.5 引理：差商的求导：$f[x_0, \ldots x_n, x_n] = \dfrac{d}{dx_n} f[x_0, \ldots, x_n]$，其中 $f[x_0, \ldots x_n, x_n]$ 由引理 4 所定义

$$
\begin{aligned}
f[x_0,\ldots x_n,x_n] &= \sum_{j=0}^{n-1} c_j g_j(x) + c_n g'_n(x_n) \\
&= \lim_{h\to 0} f[x_0,\ldots,x_n,x_n+h] \\
&= \lim_{h\to 0} \frac{f[x_0,\ldots,x_{n-1},x_n+h]-f[x_0,\ldots,x_n]}{h} \\
&= \frac{d}{dx_n} f[x_0,..,x_n]
\end{aligned}
$$

证毕。

作为引理 5 最直接的推论:

$$
\begin{aligned}
\frac{d}{dx} f[x_0,..,x_n,x] = f[x_0,\ldots x_n,x,x] \\
= \frac{f^{(n+2)}(\xi(x))}{(n+2)!}
\end{aligned}
$$

并且等式中的函数都是连续的。

引入函数

$$\Omega_n(x)=\int_a^x \omega_n(s)ds, n=1,2,3\ldots.$$

1.1.6 引理: 当 n 为偶数时, $\Omega_n(x)$ 有如下性质:

(1) $\Omega_n(a)=\Omega_n(b)=0$ (2) $\Omega_n(x)>0 \quad a<x<b$

证明:

(1) 由 $\Omega_n(x)$ 的定义可以立刻得出 $\Omega_n(a)=0$, 再由引理可知, $\Omega_n(x)$ 是关于区间[a,b]的中点对称的, 当把区间中点看作原点时, $\Omega_n(x)$ 就是一个奇函数, 所以 $\Omega_n(b)=0$ 。

(2) 已知 $a,x_1,x_2,\ldots x_{n-1},b$ 是 $\omega_n(x)$ 的全部零点, 又有 $\omega_n(x)$ 为一个偶数阶的多项式, 则易得 $\omega_n(x)<0 \quad x<a$, 进一步地有 $\omega_n(x)>0 \quad a<x<x_1$, 因而

$$\Omega_n(x)>0 \quad a<x\le x_1$$

但是根据引理 1, $\omega_n(x)$ 在区间 $[x_1,x_2]$ 上对总体积分的贡献在大小上是小于其在 $[a,x_1]$ 上的积分的 , 则

$$\Omega_n(x)>0 \quad a<x<x_2$$

这个论断可以不加修改的推广到区间 $[a,x_{n/2}]$, 对于 $x>x_{n/2}$ 一侧的积分, 再由引理 1 即得结论。

然后，类似地可以说明：当 n 为奇数时，$\Omega_n(x)$有如下性质：

$$（1）\ \Omega_n(a)=0 \quad \Omega_n(b)=2\Omega_n(x_{n/2}) \qquad （2）\ \Omega_n(x)<0 \quad a<x\le b$$

1.2 定理的正式证明：

1.2.1 首先，我们证明 n 为偶数时的误差估计，此时，$e_n(f)$可以写成如下形式：

$$\begin{aligned}
e_n(f)&=\int_a^b \frac{d\Omega_n(x)}{dx} f[x_0,\ldots x_n,x]dx \\
&=\Omega_n(x)f[x_0,\ldots x_n,x]\Big|_a^b-\int_a^b \Omega_n(x)\frac{d}{dx}f[x_0,\ldots x_n,x]dx \\
&=-\int_a^b \Omega_n(x)\frac{d}{dx}f[x_0,\ldots x_n,x]dx
\end{aligned}$$

再由引理 5 的推论

$$e_n(f)=-\int_a^b \Omega_n(x)\frac{f^{(n+2)}(\xi(x))}{(n+2)!}dx$$

由引理 4

$$f^{(n+2)}(\xi(x))=\frac{d}{dx}f[x_{0,}\ldots x_n,x](n+2)!$$

是连续的，引理 6 保证$\Omega_n(x)$在区间$[a,b]$上不变号，故可以运用推广的积分第一中值定理得到

$$e_n(f)=-\frac{f^{(n+2)}(\eta)}{(n+2)!}\int_a^b \Omega_n(x)dx \quad a<\eta<b$$

再一次分部积分可得

$$\begin{aligned}
\int_a^b \Omega_n(x)dx&=x\Omega_n(x)\Big|_a^b-\int_a^b x\frac{d}{dx}\Omega_n(x) \\
&=-\int_a^b x\omega_n(x)dx \\
&=-\int_a^b x(x-x_0)(x-x_1)\ldots(x-x_n)dx \\
&\underline{\underline{x=x_0+th}}\int_0^n h^{n+3}t^2(t-1)(t-2)\ldots(t-n)dt
\end{aligned}$$

所以

$$e_n(f)=\frac{(b-a)^{n+3}}{n^{n+3}(n+2)!}\int_0^n t^2(t-1)(t-2)\ldots(t-n)dt$$

偶数情形证毕

1.2.2 现在证明 n 为奇数的情形，我们把误差的表达式拆分成两部分

$$e_n(f)=\int_a^{b-h}\omega_n(x)f[x_1,\ldots x_n,x]dx+\int_{b-h}^{b}\omega_n(x)f[x_1,\ldots x_n,x]dx$$

并且注意到 $\omega_n(x)$ 在区间 $\left[b-h,b\right]$ 上是不变号的，而且由引理 5，类似偶数情形我们也可以运用推广的积分第一中值定理得到

$$e_n(f)=\int_a^{b-h}\omega_n(x)f[x_1,\ldots x_n,x]dx+\frac{f^{(n+1)}(\xi_1)}{(n+1)!}\int_{b-h}^{b}\omega_n(x)dx \quad a<\xi_1<b$$

为了处理第一项的积分，我们把 $\omega_n(x)$，$\Omega_{n-1}(x)$ 写作

$$\omega_n(x)=\omega_{n-1}(x)(x-x_n) \qquad \omega_n(x)=\omega_{n-1}(x)(x-x_n)$$

则由差商的递推性质可知

$$\int_a^{b-h}\omega_n(x)f[x_1,\ldots x_n,x]dx=\int_a^{b-h}\frac{d\Omega_{n-1}(x)}{dx}(f[x_0,\ldots x_{n-1},x]-f[x_0,\ldots x_n])dx$$

又 n 是奇数，则 n-1 是偶数，所以

$$\Omega_{n-1}(a)=\Omega_{n-1}(b-h)=0 \quad 且 \quad \int_a^{b-h}\frac{d\Omega_{n-1}(x)}{dx}dx=\Omega_{n-1}(x)\Big|_a^{b-h}=0$$

现在，含有常数 $f[x_0,\ldots x_n]$ 的一项为 0，由偶数情形的证明可知，另一项可以由分部积分和积分中值定理进行处理，最后得到

$$\int_a^{b-h}\omega_n(x)f[x_1,\ldots x_n,x]dx=-\frac{f^{(n+1)}(\xi_2)}{(n+1)!}\int_a^{b-h}\Omega_{n-1}(x)dx \quad a<\xi_2<b$$

现在误差 $e_n(f)$ 可以写为如下形式

$$e_n(f)=-[Af^{(n+1)}(\xi_1)+Bf^{(n+1)}(\xi_2)]$$

其中

$$A=-\frac{1}{(n+1)!}\int_{b-h}^{b}\omega_n(x)dx \qquad B=\frac{1}{(n+1)!}\int_a^{b-h}\Omega_{n-1}(x)dx$$

但是我们知道 b 是 $\omega_n(x)$ 的最大零点，且 $\omega_n(x)>0 \ \ x>b$ ，所以 $\omega_n(x)\le 0 \quad x\in[b-h,h]$，这意味这 A>0，由引理 5 可以保证 B>0，再由 $f(x)\in C^{n+2}[a,b]$ ，所以其 n+1 阶导数是连续的，所以由连续函数的介值定理可以存在一个 $\xi\in(\xi_1,\xi_2)$ 使得

$$e_n(f) = -(A+B) f^{(n+1)}(\xi)$$

又

$$\omega_n(x) = \omega_{n-1}(x)(x-b)$$

分部积分得到

$$\int_a^{b-h} \omega_n(x)dx = \Omega_{n-1}(x)(x-b)\Big|_a^{b-h} - \int_a^{b-h} \Omega_{n-1}(x)dx$$
$$= -\int_a^{b-h} \Omega_{n-1}(x)dx$$

所以

$$A+B = -\frac{1}{(n+1)!}\int_a^b \omega_n(x)dx$$

这样我们就得到了奇数情形下的误差估计

$$e_n(f) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\int_a^b \omega_n(x)dx = \frac{f^{(n+1)}(\xi)}{(n+1)!}\int_a^b (x-x_0)(x-x_1)\ldots(x-x_n)dx$$
$$\underline{\underline{x = x_0 + th}}\frac{1}{(n+1)!}\int_0^n h^{n+2} t(t-1)(t-2)\ldots(t-n)dt\, f^{(n+1)}(\xi)$$
$$= \frac{(b-a)^{n+2}}{n^{n+2}(n+1)!}\int_0^n t(t-1)(t-2)\ldots(t-n)dt\, f^{(n+1)}(\xi)$$

奇数情形证毕。

2 cotes 系数的计算

现在，我们叙述计算 cotes 系数的方法，我们返回到 newton-cotes 公式的构造来获得一些 cotes 系数的信息。Newton-cotes 公式计算数值积分的思想是用 Lagrange 插值多项式替换原被积函数，所以当原函数为不超过 n 次的多项式时，这种替换是精确的，也就是说，当 f（x）是不超过 n 次的多项式时，Newton-cotes 公式计算积分的误差为 0，这启发我们通过取 $f(x)=x^k$ 来获得 cotes 系数的方程组。

由于 cotes 系数与实际计算积分的求和系数只相差一个区间长度，所以我们不妨假设积分区间[a,b]=[0,1]，此时 cotes 系数就是求和的系数。

取[0,1]区间的 n+1 个节点的等距划分

$$0 = x_0 < x_1 < \ldots < x_n = 1, \quad x_i = \frac{i}{n} \;\; i = 0,1,\ldots,n$$

由此前的分析我们得到

$$I_n(x^k) = c_1^{(n)} x_1^{\,k} + c_2^{(n)} x_2^{\,k} + \ldots + c_n^{(n)} x_n^{\,k} = \frac{1}{k+1} = I(x^k) \;\; k = 1,2,\ldots,n$$

写成方程组形式即为

$$\begin{bmatrix} 1 & 2 & \dots & n \\ 1^2 & 2^2 & \dots & n^2 \\ 1^3 & 2^3 & \dots & n^3 \\ \vdots & \vdots & \dots & \vdots \\ 1^n & 2^n & \dots & n^n \end{bmatrix} \begin{bmatrix} c_1^{(n)} \\ \vdots \\ \vdots \\ \vdots \\ c_n^{(n)} \end{bmatrix} = \begin{bmatrix} n/2 \\ n^2/3 \\ \vdots \\ \vdots \\ n^n/n+1 \end{bmatrix}$$

于是，cotes 系数即为一个 Vandermonde 线性方程组的解（可在 MATLAB 中求解）

对于 Vandermonde 方程组

$$\begin{bmatrix} 1 & 2 & \dots & n \\ 1^2 & 2^2 & \dots & n^2 \\ 1^3 & 2^3 & \dots & n^3 \\ \vdots & \vdots & \dots & \vdots \\ 1^n & 2^n & \dots & n^n \end{bmatrix} \begin{bmatrix} u_1^{(n)} \\ \vdots \\ \vdots \\ \vdots \\ u_n^{(n)} \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ \vdots \\ d_n \end{bmatrix}$$

解也可以 LSV 算法给出：

$$u_i = \frac{(-1)^{n-i} \sum_{j=1}^{n} (-1)^{j+1} \sigma_{j,i}^{(n)} d_{n-j+1}}{i!(n-i)!}, \quad i = 1, 2, .., n$$

所以 cotes 系数计算公式为

$$\mathrm{C}_i^{(n)} = \frac{(-1)^{n-i}}{i!n(n-i)!} \left(\sum_{j=1}^{n} (-1)^{j+1} \frac{n^{n-j+2}}{n-j+2} \sigma_{j,i}^{(n)} \right), \quad i = 1, 2, \dots, n$$

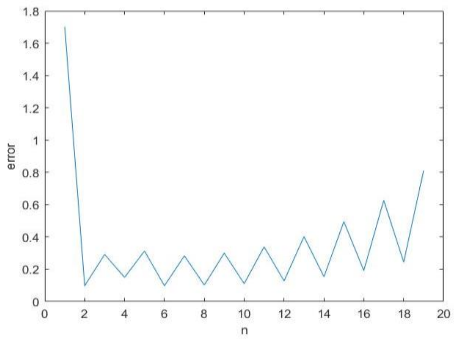
其中σ定义见参考文献 5，这说明 LSV 算法可以为我们提供一种十分准确的求 cotes 系数的方法。

3 Newton-cotes 公式局限性分析

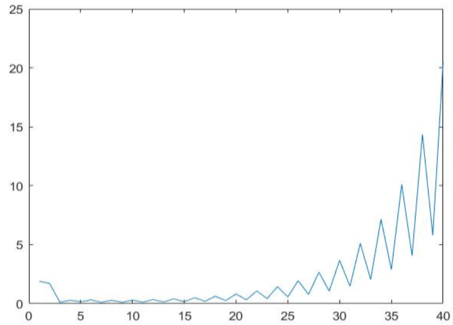
3.1 Runge 现象导致计算误差过大

Newton-cotes 积分方法作为插值型积分，必然会把函数插值中的一些不好的性质带到积分的计算当中，Runge 现象就是一个十分典型的例子，在对一些函数作插值的时候会出现插值多项式次数增大的同时误差也增大的现象，所以在用 Newton-cotes 公式计算这类函数的积分时也会出现较大的误差。接下来以函数$1/(1+x^2)$在$[-3,3]$上的积分为例展现这种现象：

在 MATLAB 中分别用 Newton-cotes 公式和内置 int 函数计算，计算二者之间的误差并绘制图像（程序代码见文章末尾）：

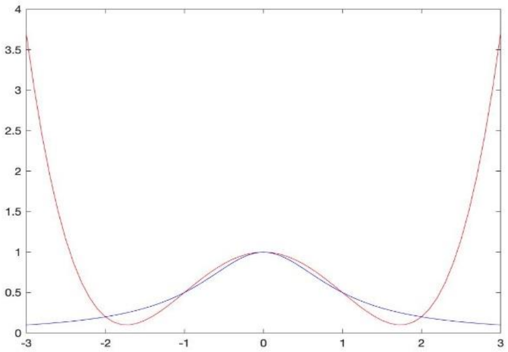


n≤20



n≤40

插值多项式（红）与原函数（蓝）的图像对比：

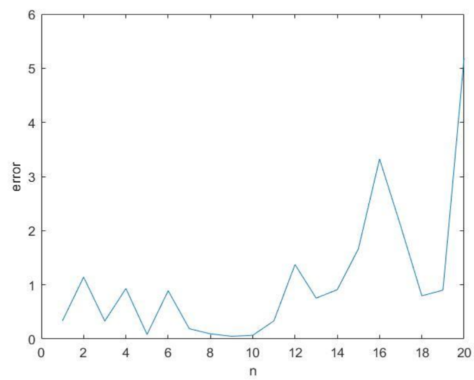


我们可以清楚的看到当 n 增大时，误差也是逐渐增大并且达到了 O(1)的量级（积分的精确值约为 2.498），并且 n 大于 25 之后，计算结果已经完全偏离，说明 Newton-cotes 公式此时已经无法使用。

从上述计算结果还可以看出，误差图像呈锯齿状，可以看出 n 为偶数时误差要比 n 为奇数时小很多，这是因为 n 为偶数时代数精度为 n+1，而 n 为奇数时代数精度为 n。这个结果也印证了在实际计算中基本不采用奇数形式的 Newton-cotes 公式。

3.2 对某些图像分布不均匀的函数计算积分时误差较大

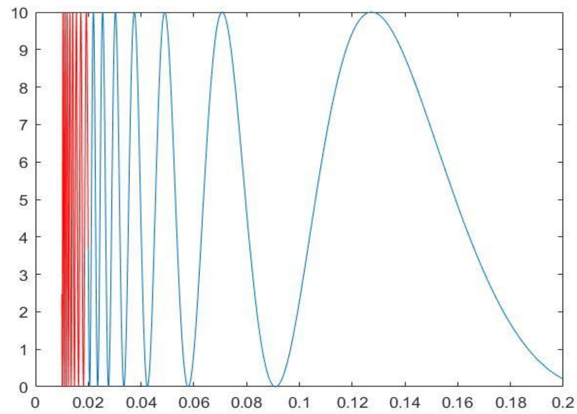
例如函数$5sin\left(\frac{1}{x}\right)+5$在区间$[0.01,0.5]$上的积分，在 MATLAB 中用 Newton-cotes 公式计算并与精确值对比，画出误差图像如下（程序代码见文章末尾）：



从误差图像我们可以看出，误差是$O(1)$的量级，数值表现较差。这是因为函数

$5sin\left(\frac{1}{x}\right)+5$在 0 附近震荡十分剧烈，但是这种震荡只维持了很短的距离，而且在距离 0 不是很近的地方函数图像是很平滑的，在用 Newton-cotes 公式计算时，取的是积分区间上的等距划分，这就导致虽然在 0 附近函数震荡剧烈对整体积分值影响很大，但是被采用的函数信息却完全不足，所以计算结果有较大偏差。

$f(x)=5sin\left(\frac{1}{x}\right)+5$的图像如下：

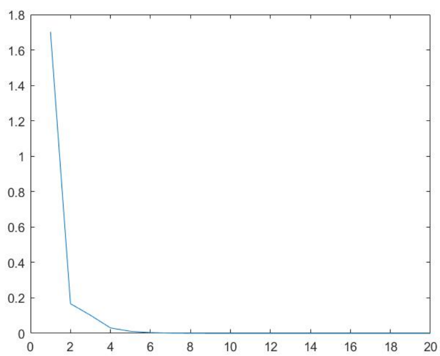


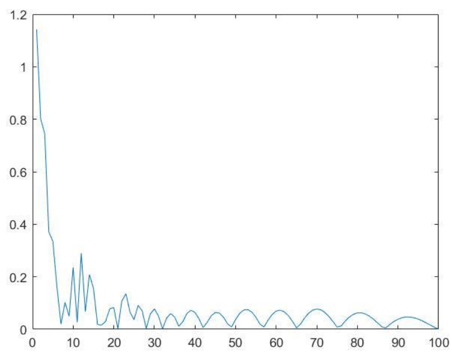
图中红色部分即为造成较大计算误差的部分，这部分函数图像十分密集，但是所包含区间的分点却很少，造成计算积分时贡献不足。

对于类似的这种函数，图像稀疏程度不均匀，通常在某些点或者某些区间非常稠密，Newton-cotes 公式的积分计算结果往往都有较大的误差。

解决办法：

解决以上两个问题的一个很好的办法就是使用复化的 Newton-cotes 公式，以复化的 Simpson 公式为例，在 MATLAB 中分别计算$1/(1+x^2)$在$[-3,3]$上和$5sin\left(\frac{1}{x}\right)+5$在$[0.01,0.5]$上的积分并做出误差图像（精确值分别约为 2.498091 和 2.607913）：





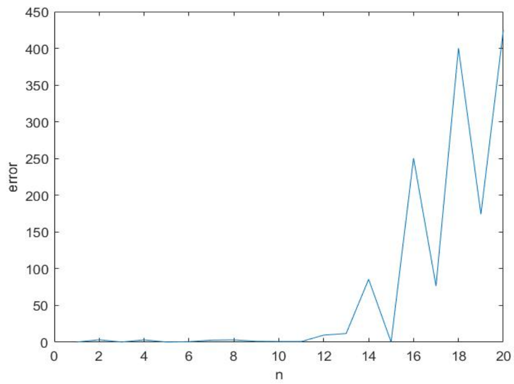
从图像可以看出$1/(1+x^2)$积分的计算得到了很好的改进，误差可以达到很高的精度要求，于是成功消除了 Runge 现象对数值积分的影响。

$5sin\left(\frac{1}{x}\right)+5$的积分虽然从图像上看结果还并不是十分精确，但是随着 n 的增大误差呈现递减的趋势，而且复化的 Simpson 公式可以对区间划分更加稠密，从而通过增大

n 来获得更精确的结果，但是经典的 Newton-cotes 却不可以任意增大 n，原因在 3 中给出。

3.3 cotes 系数在 n>7 时出现负数，造成计算不稳定

由 2 中计算结果看出，cotes 系数在 n > 7 时出现负数，且绝对值呈现发散的趋势，因此在作求和时被积函数值有一些很小的扰动便会导致较大的误差，以$\sin(10x)$在$[0,3\pi/2]$上的积分为例，在 MATLAB 中做出误差图像：



从图像可以看出 n > 10 之后的计算结果完全错误，说明 n 较大时 Newton-cotes 公式极不稳定，有时无法使用。

由第二部分 cotes 系数计算结果，n > 7 时 cotes 系数就出现了负数（虽然 n=9 时为正数，但是 9 为奇数故并不常用），这就解释了为什么一般只取 n≤8 时的 Newton-cotes 公式。再者，n 较大时 cotes 系数十分冗长，计算代价较大，且在计算机运算时可能会造成较大的舍入误差进而导致误差增大。

参考文献

1、Analysis of Numerical Methods　Issacson and H.B. keller
2、数值计算与方法 郑慧娆 武汉大学出版社
3、数值分析　张平文 李铁军 北京大学出版社
4、A unified approach Newton-cotes quadrature formulae　Moawwad El-Mikkawy
5、An algorithm for solving Vandermonde systems　M.E.A. El-Mikkawy

代码：

```
%matlab内置函数求cotes系数
clear;
syms t
m=zeros(10,11);
for n=1:20
    for k=1:n+1
        l=1;
        for j=1:n+1
            if(j~=k)
                l=l*(t-j+1)/(k-j);
            end
        end
        s(n,k)=int(l,0,n)/n;
```

```
    end
end
m=s

%Vandermonde求解cotes系数
clear;
format long
for n=1:20
      x=zeros(1,n);
      A=zeros(n,n);
      b=zeros(1,n);
      c=zeros(1,n+1);
     for i=1:n
       x(i)=i;
     end
      %定义vandermonde矩阵
        for i=1:n
           for j=1:n
             A(i,j)=(x(j))^i;
           end
        end
      %右端向量
     for i=1:n
        b(i)=(n^i)/(i+1);
     end
     s=A\(b');
     c(1)=s(n);
  for i=2:n+1
    c(i)=s(i-1);
  end
  c1=c
  c2=rats(c)
end

clear
syms t x;

error=zeros(1,40);
f(x)=input('请输入函数f (x) =');
a=input('请输入区间下限: ');
b=input('请输入区间上限: ');
yy2=int(f,a,b); %精确值
for n=1:20
h=(b-a)/n;
```

```
for i=1:n+1    %将区间n等分后，计算各节点的横坐标
    xdata(i)=a+(i-1)*h;
end
ydata=subs(f,'x',xdata);

for k=1:n+1     %计算cotes系数
    l=1;
    for j=1:n+1
        if(j~=k)
            l=l*(t-j+1)/(k-j);
        end
    end
    s(k)=int(l,0,n)/n;
end
yy1=sum(s.*ydata);
If=(b-a)*yy1;   %用Newton-cotes公式计算求得结果
error(n)=abs(yy2-If);   %计算误差
error(n);
end
error
i=1:20;
plot(i,error(i))
```

2.复化 Simpson 公式求积

```
clear
format long;
syms  x;
f(x)=input('请输入函数f(x)=');
a=input('请输入区间下限:  ');
b=input('请输入区间上限:   ');
yy2=int(f,a,b);
error=zeros(1,100);
P=zeros(1,100);
for n=1:100
h=(b-a)/(2*n);

for i=1:2*n+1    %将区n等分后，计算各节点的横坐标
    xdata(i)=a+(i-1)*h;
end
ydata=subs(f,'x',xdata);
s=zeros(1,2*n+1);
s(1)=1;
```

```
s(2*n+1)=1;
for i=2:2*n
    if mod(i,2) == 0
        s(i)=4 ;  %复化simpson
    else
        s(i)=2;
    end
end
yy1=sum(s.*ydata);
P(n)=(h/3)*yy1;
error(n)=abs(P(n)-yy2);
end
P
error;
i=1:100;
plot(i,error(i))
```

3.Runge插值多项式图像绘制

```
a=-3;
b=3;
n=6;
h=(b-a)/n;
x=-3:0.1:3
for i=1:n-1
    x0(i)=a+(b-a)/n*i;
end
for i=1:n-1
    y0(i)=1/(1+x0(i)^2);
end
n=length(x0);
m=length(x);
for i=1:m
    z=x(i);
    s=0.0;
    for k=1:n
        p=1.0;
        for j=1:n
            if j~=k
                p=p*(z-x0(j))/(x0(k)-x0(j));
            end
        end
        s=p*y0(k)+s;
    end
```

```
    y(i)=s;
end
plot(x,y,'r')
hold on
x0=-3:0.1:3;
k=length(x0);
for i=1:k
    y0(i)=1/(1+x0(i)^2);
end
plot(x0,y0,'b')
hold off
```