Scientific Computation

Xiao Yunxuan

November 2017

1 正交多项式

1.1 勒让德多项式

1.1.1 形式

勒让德多项式是带权$\rho(x) = 1$的正交函数族, 是$\{1, x, x^2, \cdots, x^n, \cdots\}$通过正交化得到的,自然的次数从$1 - n$

$$\begin{cases} P_0(x) = 1 \\ P_n(x) = \dfrac{1}{2^n n!} \dfrac{d^n}{dx^n}(x^2 - 1)^n \end{cases}$$

归一化后形式:

$$\widetilde{P_n}(x) = \frac{n!}{(2n)!} \frac{d^n}{dx^n}(x^2 - 1)^n = \frac{2^n (n!)^2}{(2n)!} P_n(x)$$

1.1.2 性质

性质1：正交性

$$\int_{-1}^{1} P_n(x) P_m(x) dx = \begin{cases} 0, & m \neq n; \\ \dfrac{2}{2n+1}, & m = n; \end{cases}$$

1.1.3 应用

1.勒让德多项式P_n零点作为n-rank Gauss-Legendre 求积公式插值点

2 数值积分

2.1 代数精度

若某种求积公式对次数$\leq m$的多项式准确成立

$$\int_a^b f(x)dx = \sum_{k=0}^{n} A_k f(x_k)$$

则其具有m次代数精度

直白表述：即能找到确定的插分点$\{x_k\}$及其对应的系数$\{A_k\}$,使得其线性组合能精确表示一组基$\{1, x, x^2, \cdots, x^m\}$，则其必对$P_m$精确成立。

化归为解线性方程组：

$$\begin{cases} \sum A_k = b - a \\ \sum A_k x_k = \frac{1}{2}(b^2 - a^2) \\ \qquad \vdots \\ \sum A_k x_k^m = \frac{1}{m+1}(b^{m+1} - a^{m+1}) \end{cases}$$

要刻画一个n次多项式，至少需要$n+1$个点，加上$n+1$个系数组成的$2n+2$个未知量的方程组

1.一般会告诉插分点,带进去解系数A_k

2.继续向下验证是否对x^{m+1}也成立，直到推到最大代数精度

PS: n阶求积公式——n+1插值点——最少n代数精度——对P_n精确成立

2.2 插值型求积公式

通过插值点$a \le x_0 < x_1 < \cdots < x_n \le b$ 得到插值函数$L_n(x)$，以插值函数的积分近似代替I。

(梯形公式与中矩形公式均是特殊的插值求积公式)

$$\begin{aligned}\int L_n(x)dx &= \int \sum f(x_k) l_k(x) dx \\ &= \sum f(x_k) \int l_k(x) dx \\ &= \sum f(x_k) A_k\end{aligned}$$

故系数即插值基函数的积分:

$$A_k = \int_a^b l_k(x) dx$$

求积公式的余相:

$$R[f] = \int [f(x) - L_n(x)] = \int R_n(x) = \int \frac{f^{(n+1)}(\xi)}{(n+1)!} \omega_{n+1}(x)$$

1.对于 $\forall f(x) \in P_n(x)$： $R[f] = 0$精确成立

2.对于 $\forall f(x) \notin P_n(x)$： 有泛化公式:

$$R[f] = \int_a^b f(x)dx - \sum_{k=0}^{n} A_k f(x_k) = \boldsymbol{K} f^{(n+1)}(\xi)$$

据此可求出$n+1$**阶函数求积余项公式** (WARNING：此时$f(x) = x^{n+1}$)

$$\boldsymbol{K} = \frac{1}{(n+1)!}[\int_a^b x^{n+1} dx - \sum_{k=0}^{n} A_k x_k^{n+1}]$$

2.3 Newton-Cotes Formula

单区间：等距型插值求积公式

2.3.1 Trapezoidal Formula

$$n = 1, h = b - a$$

$$I \approx (\frac{b-a}{2})[f(x_0) + f(x_1)]$$

$$R[f] = -\frac{(b-a)^3}{12} f^{(2)}(\xi) \qquad Accuracy = 1$$

2.3.2 Simpson Formula

$$n = 2, h = \frac{b-a}{2}$$

$$I \approx \frac{(b-a)}{6}[f(x_0) + 4f(x_1) + f(x_2)]$$

$$R[f] = -\frac{b-a}{180}(\frac{b-a}{4})^4 f^{(4)}(\xi) \qquad Accuracy = 3$$

2.3.3 Newton Formula

$$n = 3, h = \frac{b-a}{3}$$

$$I \approx \frac{3h}{8}[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)]$$

$$R[f] = -\frac{(b-a)}{80}(\frac{b-a}{3})^4 f^{(4)}(\xi) \qquad Accuracy = 3$$

2.3.4 Cotes Formula

$$n = 4, h = \frac{b-a}{4}$$

$$I \approx \frac{2h}{45}[7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4)]$$

$$R[f] = -\frac{2(b-a)}{945}(\frac{b-a}{4})^6 f^{(6)}(\xi) \qquad Accuracy = 5$$

规律：偶次阶求积公式具有n+1阶精度。

定理

证明偶次阶求积公式具有n+1阶精度:

我们只需证明n = Even Number时，Newton-Cotes 公式对$f = x^{n+1}$余项为0

引入变换$x = a + th$

$$
\begin{aligned}
R[f] &= \int_a^b \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x) \\
&= \int_a^b \omega_{n+1}(x) \\
&= h^{n+1}\int_0^n \prod_{j=0}^{n}(t-j)dht \quad (Alter: t = u + \frac{n}{2}) \\
&= h^{n+2}\int_{-\frac{n}{2}}^{\frac{n}{2}} \prod_{j=0}^{n}(u + \frac{n}{2} - j)du \quad (Moving\_towards\_left) \\
&= 0 \quad (Odd\_function)
\end{aligned}
$$

2.4 复合求积公式

多区间：小区间求和插值公式

注意：此处n(小区间等分个数)非彼处n(单区间等分个数)

此处h(小区间步长)非彼处h(单区间步长)

2.4.1 复合Trapezoidal公式

$$
\begin{aligned}
I = T_n + R_n &= \frac{h}{2}\sum_{k=0}^{n-1}[f(x_k)+f(x_{k+1})] + R_n \\
&= \frac{h}{2}[f(a) + 2\sum_{k=1}^{n-1} f(x_k) + f(b)] + R_n
\end{aligned}
$$

$$
\begin{aligned}
Where \quad R_n &= \sum_{k=0}^{n-1}\big[-\frac{h^3}{12}f^{(2)}(\eta_k)\big] = -\frac{h^3}{12}n * f^{(2)}(\eta) \\
&= -\frac{b-a}{12}h^2 f^{(2)}(\eta) \qquad \boldsymbol{Error = O(h^2)}
\end{aligned}
$$

2.4.2 复合Simpson公式

$$
\begin{aligned}
I = T_n + R_n &= \frac{h}{6}\sum_{k=0}^{n-1}[f(x_k)+4f(x_{k+\frac{1}{2}})+f(x_{k+1})] + R_n \\
&= \frac{h}{2}[f(a) + 4\sum_{k=1}^{n-1} f(x_{k+\frac{1}{2}}) + 2\sum_{k=1}^{n-1} f(x_k) + f(b)] + R_n
\end{aligned}
$$

$$
\begin{aligned}
Where \quad R_n &= \sum_{k=0}^{n-1}\big[-\frac{h}{180}\big(\frac{h}{2}\big)^4 f^{(4)}(\eta_k)\big] = -\frac{h}{180}\big(\frac{h}{2}\big)^4 n * f^{(4)}(\eta) \\
&= -\frac{b-a}{180}\big(\frac{h}{2}\big)^4 f^{(4)}(\eta) \qquad \boldsymbol{Error = O(h^4)}
\end{aligned}
$$

2.5 梯形算法外推化

将区间[a,b]n等分，再将n个区间二等分，得到$\{x_k, x_{k+\frac{1}{2}}\}$
使用复合梯形公式得到：

$$I = \frac{h}{4}[f(x_k) + f(x_k + \frac{1}{2})]$$

2.6 Richardson 加速

梯形公式余项为$O(h^2)$阶，可展开成如下形式

$$T(h) = I + a_1h^2 + a_2h^4 + \cdots + a_kh^{2k} + \cdots$$

令$h <= h/2$:

$$T(\frac{h}{2}) = I + a_1\frac{h^2}{4} + a_2\frac{h^4}{16} + \cdots + a_k(\frac{h}{2})^{2k} + \cdots$$

$4 * (2) - (1)$得：

$$S(h) = \frac{4T(\frac{h}{2}) - T(h)}{3} = I + c_1h^4 + c_2h^6 + \cdots$$

误差降低到$O(h^4)$

由梯形公式$T(h)\&T(h/2)$导出的$S(h)$恰是辛普森公式！！

不断继续下去便形成了Romberg算法。

注意：若使用Simpson公式外推，误差为$O(h^4)$阶，相应导出公式变为：

$$C(h) = \frac{16S(\frac{h}{2}) - S(h)}{15}$$

2.7 Romberg Algorithm

1.k等分区间梯形公式算$T_0^{(k)}$

2.依次使用$T_{m-1}^{(k+1)}T_{m-1}^{(k)}$计算一横排$T_m^{(k)}$

3.比较相邻对角元素$T_k^{(k)}T_{k-1}^{(k-1)}$与ϵ

4.停止or继续二分

思考：Richardson加速为什么更优呢？直接用n阶Newton-Cotes公式不行吗？

由于我们不知道k等分后计算出的结果是否可以达到误差范围，这样会造成我们不断k+1等分，重复计算大量值，且公式繁杂，每个积分公式系数都不同

而使用龙贝格方法优点为：

1.利用了之前运算结果，每次只需算k等分梯形公式值，利用前面结果，通过相同公式反复迭代，方便简洁稳定性好。

2.将先验误差转化为后验误差，根据结果调整计算过程，节约计算量。

3.加速算法，每次步长指数倍缩小，收敛更快。

2.8 Gauss 求积公式

我们知道，插值型求积公式最少精度为n，偶阶时精度为n+1，然而适当选取节点可以将精度提高到2n+1，这些点叫Gauss点

定义式：研究带权积分何时精确成立

$$\int_a^b f(x)\rho(x)dx \approx \sum_{k=0}^{n} A_k f(x_k)$$

观察余项：

$$R[f] = \frac{1}{(n+1)!}\int_a^b f^{(n+1)}(\xi)\omega_{n+1}(x)\rho(x)dx$$

只需要证明f为2n+1次多项式时，选Gauss点做插值点使得$R[q]=0$成立(q为n次多项式)即可

左推右：f的n+1阶导为n次，乘上ω_{n+1}为2n+1次，积分为0，OK

右推左：

$$f = p(x)\omega_{n+1}(x) + q(x)$$

因$\mathrm{R}=0$故

$$\int_a^b f(x)\rho(x)dx = \int_a^b q(x)\rho(x)dx$$

q(x)为n次多项式，精确成立

$$\int_a^b q(x)\rho(x)dx = \sum_{k=0}^{n} A_k q(x_k)$$

取高斯点时，$f(x_k) = q(x_k)$故

$$\int_a^b f(x)\rho(x)dx = \int_a^b q(x)\rho(x)dx = \sum_{k=0}^{n} A_k f(x_k)$$

总结：所以选择正交多项式的零点做插值点，这样ω_{n+1}就是归一化的正交多项式，就可以对任意n次多项式带权积分为0，保证对2n+1阶的f积分精确成立

其中系数A_k与f无关：

$$A_k = \int_a^b l_k(x)\rho(x)dx$$