计 算 方 法 实 验 一

Error, Interpolation, Curve Fitting, Integration

何军辉
hejh@scut.edu.cn

实验说明

- 实验内容：
  - 编写程序调试，用作业题验证算法
  - 基于Lagrange插值的秘密图像共享
- 考核方式：
  - 实验报告（90%）
  - 实验考勤（10%）
- 提交方式：
  - 文件命名：学号-XXX-LAB1
  - 以班级为单位提交电子版至下面邮箱：hejh@scut.edu.cn

1.1 有效数字丢失现象观察

例：下面两个理论上等价的函数，当x不断增大

$$f_{1(x)}=\sqrt{x}\left(\sqrt{x+1}-\sqrt{x}\right), f_2(x)=\frac{\sqrt{x}}{\sqrt{x+1}+\sqrt{x}}$$

```
At x=                1,  f1(x)=0.41421356237309510,  f2(x)=0.41421356237309509
At x=               10,  f1(x)=0.48808848170151475,  f2(x)=0.48808848170151548
At x=              100,  f1(x)=0.49875621120889946,  f2(x)=0.49875621120890273
At x=             1000,  f1(x)=0.49987506246102187,  f2(x)=0.49987506246096486
At x=            10000,  f1(x)=0.49998750062485442,  f2(x)=0.49998750062496089
At x=           100000,  f1(x)=0.49999875000592886,  f2(x)=0.49999875000624994
At x=          1000000,  f1(x)=0.49999987504634191,  f2(x)=0.49999987500006249
At x=         10000000,  f1(x)=0.49999998740115092,  f2(x)=0.49999998750000058
At x=        100000000,  f1(x)=0.50000000555883162,  f2(x)=0.49999999874999995
At x=       1000000000,  f1(x)=0.50000007799750634,  f2(x)=0.49999999987499999
At x=      10000000000,  f1(x)=0.49999944167211652,  f2(x)=0.49999999998750005
At x=     100000000000,  f1(x)=0.50004449631168080,  f2(x)=0.49999999999875000
At x=    1000000000000,  f1(x)=0.50003807246685030,  f2(x)=0.49999999999987499
At x=   10000000000000,  f1(x)=0.49919454697383597,  f2(x)=0.49999999999998751
At x=  100000000000000,  f1(x)=0.50291419029235840,  f2(x)=0.49999999999999872

sqrt(x+1) = 31622776.6016838100000, sqrt(x) = 31622776.601683792000
  diff=0.0000000186264514923095 7, sum=63245553.2033676060000000000000
```

1.2 n次插值的Lagrange形式

- Lagrange型n次插值算法
  ① 读入插值节点$x_i, y_i (i = 0,1,\cdots,n)$和点$xx$
  ② $yy = 0$
  ③ 对$i = 0,1,\cdots,n$做如下工作
    - $t = 1$
    - 对$k = 0,1,\cdots,i-1,i+1,\cdots,n$做
    - $t = t \times \frac{xx - x_k}{x_i - x_k}$
    - $yy = yy + t \times y_i$
  ④ 输出点xx相应的函数近似值yy

1.3 n次插值Newton形式

- Newton型n次插值算法

① 读入插值节点$x_i, y_i (i = 0,1,\cdots,n)$和点$xx$

② $\omega = 1, V_0 = y_0, yy = y_0$

③ 对$k = 1,2,\cdots,n$做如下工作

- $V_k = y_k$
- 对$i = 0,1,\cdots,k-1$做
- $V_k = \frac{V_i - V_k}{x_i - x_k}$
- $\omega = \omega \times (xx - x_{k-1})$
- $yy = yy + \omega \times V_k$

④ 输出插值点xx相应的函数近似值yy

1.4 三次样条插值

三次样条插值算法

① 读入插值节点$x_i, y_i (i = 0,1,\cdots,n)$和点$xx$

② 对$i = 0,1,\cdots,n-1$计算$h_i = x_{i+1} - x_i$

③ 计算α_i和$\beta_i (i = 0,1,\cdots,n)$

- 对第一种边界条件：

$$\alpha_0 = 0, \beta_0 = 2m_0;$$
$$\alpha_n = 1, \beta_n = 2m_n$$

- 对第二种边界条件：

$$\alpha_0 = 1, \beta_0 = \frac{3}{h_0}(y_1 - y_0);$$
$$\alpha_n = 0, \beta_n = \frac{3}{h_{n-1}}(y_n - y_{n-1})$$

三次样条插值算法

- 对$i=1,2,\cdots,n-1$做

$$\alpha_i=\frac{h_{i-1}}{h_{i-1}+h_i};\beta_i=3\left[\frac{1-\alpha_i}{h_{i-1}}(y_i-y_{i-1})+\frac{\alpha_i}{h_i}(y_{i+1}-y_i)\right]$$

④ 计算a_i和$b_i(i=0,1,\cdots,n)$

$$a_0=-\frac{\alpha_0}{2};b_0=\frac{\beta_0}{2}$$

- 对$i=1,2,\cdots,n$做

$$a_i=-\frac{\alpha_i}{2+(1-\alpha_i)a_{i-1}};\ b_i=\frac{\beta_i-(1-\alpha_i)b_{i-1}}{2+(1-\alpha_i)a_{i-1}}$$

□ 三次样条插值算法

⑤ 计算$m_i(i = 0,1,\cdots,n)$

$$m_n = b_n$$

$$m_i = a_i m_{i+1} + b_i \ (i = n-1,\cdots,1,0)$$

⑥ 判别点xx所在区间$[x_i, x_{i+1}](i = 0,1,\cdots,n-1)$，然后求出$s(x)$的值并输出

$$yy = s(xx)$$

$$= \left(1 + 2\frac{xx - x_i}{x_{i+1} - x_i}\right)\left(\frac{xx - x_{i+1}}{x_i - x_{i+1}}\right)^2 y_i$$

$$+ \left(1 + 2\frac{xx - x_{i+1}}{x_i - x_{i+1}}\right)\left(\frac{xx - x_i}{x_{i+1} - x_i}\right)^2 y_{i+1} + (xx - x_i)\left(\frac{xx - x_{i+1}}{x_i - x_{i+1}}\right)^2 m_i$$

$$+ (xx - x_{i+1})\left(\frac{xx - x_i}{x_{i+1} - x_i}\right)^2 m_{i+1}$$

1.5 单变量数据拟合（最小二乘法）

单变量线性拟合法算法

① 读入数据x_i和$y_i (i = 1,2,\cdots,n)$

② 计算

$$s_x = \sum_{i=1}^{n} x_i\,,\, s_y = \sum_{i=1}^{n} y_i\,,\, s_{xx} = \sum_{i=1}^{n} x_i^2\,,\, s_{xy} = \sum_{i=1}^{n} x_i y_i$$

③ 解正规方程组

$$\begin{cases} na + s_x b = s_y \\ s_x a + s_{xx} b = s_{xy} \end{cases}$$

④ 输出a和b

$$a = \frac{s_{xx}s_y - s_x s_{xy}}{n s_{xx} - s_x^2}, b = \frac{n s_{xy} - s_x s_y}{n s_{xx} - s_x^2}$$

1.6 自动选取步长梯形法

① 输入a,b和ϵ

② 计算$h=\frac{b-a}{2}$，$T1=\big(f(a)+f(b)\big)*h, n=1$

③ 计算$T0=T1$，$S=0$（$T0$表示前次积分近似值，$T1$表示后次积分近似值）

④ 对$k=1,2,\cdots,n$计算

$$S=S+f(a+(2*k-1)*h/n)$$

⑤ $T1=\frac{T0}{2}+S*\frac{h}{n}$

⑥ 若$|T1-T0|<3\epsilon$，则输出$T1$的值，结束计算，否则$n=2n$，返回③

1.7 Romberg求积法

① 输入a, b和ϵ

② 计算$T_0^{(0)}$：

$$T(0,0) = \frac{b-a}{2}[f(a) + f(b)]$$

③ $k = 1$(其中k用来记录把积分区间$[a, b]$2等分的次数)

④ 按复化梯形公式计算$T_0^{(k)}$：

$$T(0,k) = \frac{1}{2}\left[T(0,k-1) + \frac{b-a}{2^{k-1}}\sum_{i=1}^{2^{k-1}} f\left(a + (2i-1)\times\frac{b-a}{2^k}\right)\right]$$

1.7 Romberg求积法

⑤ 计算第$k+1$行元素$T_m^{(k-m)}$：

$$T(m,k-m)=\frac{4^m T(m-1,k-m+1)-T(m-1,k-m)}{4^m-1}$$

其中$m=1,2,\cdots,k$

⑥ 精度控制：

- 对指定的精度ϵ，若$|T(k,0)-T(k-1,0)|<\epsilon$，即$\left|T_k^{(0)}-T_{k-1}^{(0)}\right|<\epsilon$，则终止计算，并取$T(k,0)$即$T_k^{(0)}$作为满足精度要求的积分近似值；否则$k=k+1$，转回④继续计算.

Secret image sharing

- 参考文献 C.-C. Thien and J.-C. Lin, “Secret image sharing,” Computers & Graphics, vol. 26, no. 5, pp. 765–770, 2002.
- 实现基于Lagrange插值的秘密图像共享
  - 输入一幅灰度图像，n，k， k<=n
  - 输出 n 幅子图
  - 任意选择 k 幅子图重构原图像
  - 思考不截断的实现方法，即保持灰度图像像素值范围0-255


Thank You!