数值分析实验一

赵浩宇,2016012390, 计科 60

2018 年 3 月 27 日

1 实验要求

1. 已知 $\ln 2 = 1 - \frac{1}{2} + \frac{1}{2} + \cdots + \frac{(-1)^{n-1}}{n} + \cdots$，令 $x_n = \sum_{i=1}^{n} \frac{(-1)^{k-1}}{k}$，试用单精度 float 计算 x_n，问 n 为何值时能满足精度要求？并分析理论上的 n 值与实际上的 n 值是否存在不同.

2. 对 $[-5, 5]$ 作等距划分 $x_i = -5 + ih, h = 10/n, 0 = 0, \ldots, n,$ 并对 Runge 给出的函数

$$f(x) = \frac{1}{1 + 16x^2}$$

做 Lagrange 插值和三次样条插值，分别取 $n = 10, 20$ 做 Lagrange 与三次样条插值，计算在 $x = 4.8$ 处的误差，并且画出图像。

2 算法描述

2.1 第一部分

第一部分的代码比较简单，直接每次是的 n 增加 1，并且判断是否满足精度要求即可。

2.2 第二部分

首先给出 Lagrange 插值的算法。

1. 首先使用一个 vector<double> 代表一个多项式,并且抽象出多项式乘法与加法 polyMul 与 polyAdd。
2. 之后抽象出计算 Lagrange 插值多项式的函数，该函数输入插值节点的值与函数值，返回一个 vector<double> 表示计算出来的插值多项式。计算时直接通过现有公式计算即可，打印的插值多项式的时候按照 Mathematica 语法打印，方便画图。并且在该函数中计算 $x = 4.8$ 的值。
3. 生成插值节点的值与函数值，并且传入上一步抽象出的计算 Lagrange 插值多项式的函数即可。
4. 使用 Mathematica 进行相应的画图。

之后给出三次样条插值的算法。

1. 首先抽象出矩阵与向量两个简单的数据结构，都是用 vector 实现.。

2. 之后抽象出追赶法解线性方程组的函数, 传入一个矩阵 A 与一个向量 b, 返回向量 x 满足 $Ax = b$。

3. 之后抽象出第一种边界条件的三次样条插值函数。首先根据公式计算出 $h, f[x_i, x_{i+1}], d, A$ 等基本变量，其中 A 是对角占优阵。之后调用之前抽象出的追赶法解方程，解出 $M, AM = d$。最后依据公式计算出相应的分段样条插值函数。输出格式与 Mathematica 格式相同，顺便计算误差。

4. 生成插值节点的值与函数值，并且计算两端的导数大小。

5. 使用 Mathematica 进行画图。

3 程序清单

程序清单见表 1。

名称	描述
code_1.cc	实验的第一部分的代码
code_2.cc	实验第二部分 Lagrange 插值的代码
code_3.cc	实验第二部分三次样条插值的代码
plot.nb	用于画图的 Mathematica Notebook 代码

表 1: 程序清单

4 运行结果

结果清单见表 2。

名称	描述
err.out	实验的第一部分的结果
lagrange.out	实验第二部分 Lagrange 插值的结果
spline.out	实验第二部分三次样条插值的结果
lag10.png	lagrange 插值多项式函数图像, n=10
lag20.png	lagrange 插值多项式函数图像, n=20
spl10.png	三次样条插值多项式函数图像, n=10
spl20.png	三次样条插值多项式函数图像, n=20

表 2: 运行结果清单

4.1 第一部分

若需要达到所需要的精度，需要 $n \geq 8975$。n 的值均在 err.out 文件中给出。理论上的 n 在 20000 左右，比实际的 n 要大，这是因为理论上的 n 是上界，计算是使用的是导数的最大值。但是实际上误差不一定是导数的最大值，所以实际上 n 的值比理论值小很多。

4.2 第二部分

首先给出四个函数在 $x = 4.8$ 处的误差。其中 L10,L20,S10,S20 分别代表 n=10,20 的 Lagrange 插值函数以及 n=10,20 的三次样条插值函数。见表 3。

函数	误差
L10	5.148610113257
L20	1080.742892271142
S10	0.000392438669
S20	0.000001102744

表 3: Caption

之后分别画出两个 Lagrange 插值以及两个三次样条差值的图像。

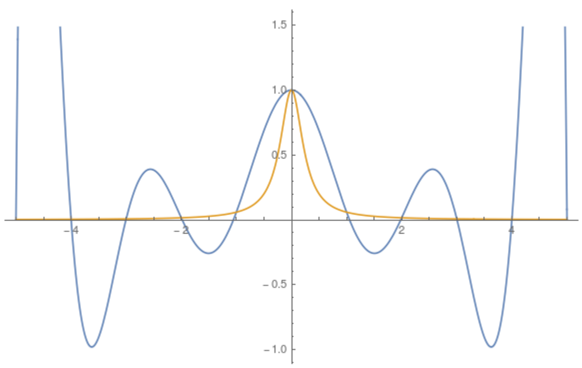

图 1: L10:n=10 的 Lagrange 插值

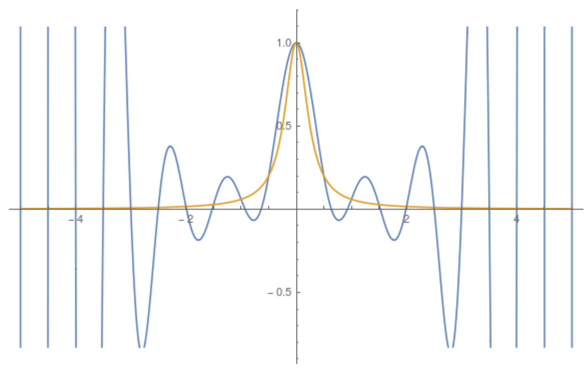

图 2: L20:n=20 的 Lagrange 插值

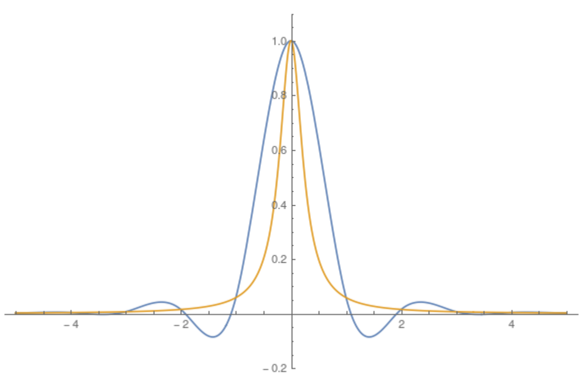

图 3: S10:n=10 的三次样条插值

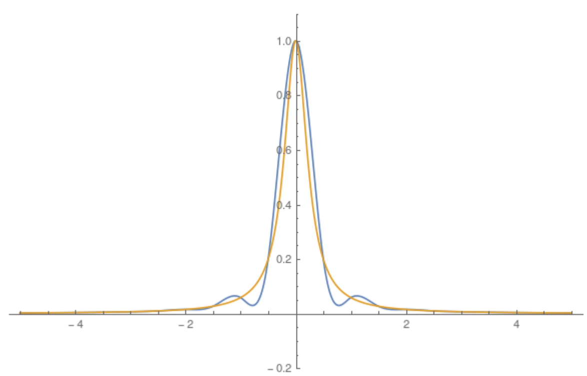

图 4: S20:n=20 的三次样条插值

Lagrange 插值的误差比三次样条插值要大，而且当 n 增加的同时，Lagrange 插值的误差增大，但是三次样条插值的误差在减小。为了更好的插值，可以选择采用更大的 n 进行三次样条插值。

关于 Lagrange 插值多项式的结果，以及在 $x = 4.8$ 处的误差均在 lagrange.out 文件中给出，关于三次样条插值多项式的结果，以及在 $x = 4.8$ 处的误差均在 spline.out 文件中给出。

5 体会与展望

通过该实验，更加深入的理解了误差分析。同时，通过代码实现，更加深入地理解了 Lagrange 插值与三次样条插值。在实现插值函数的同时，也锻炼了自己对程序模块进行抽象的能力。

通过函数插值的实验，知道了 Lagrange 插值与三次样条差值的优劣。Lagrange 插值更容易实现，但是在对特殊的函数进行差值的时候，会出现病态的性质。三次样条插值实现稍微麻烦一点，但是在函数的光滑性表先良好，且对特殊函数进行插值的时候会比 Lagrange 插值表现要好。