计算物理讲义

冯旭

1　数值计算的基础

2　线性方程组的直接解法

3　内插与函数的计算

前一章中我们讨论了数值线性方程的解。本章中我们将涉及函数的内插或外推问题。这类问题广泛地出现在各类的实验科学中。

内插面对的基本问题大致如下: 假定我们在一系列控制变量 (自变量)x 取值处 –例如我们在 $x = x_0, x_1, \cdots, x_n$ 这 $(n+1)$ 个两两不同的点处 –获得了相应的函数 $y = f(x)$ 的数值:$y_0, y_1, \cdots, y_n$，并且如果我们对函数 $y = f(x)$ 的宏观性状做某些合理的假定 (例如，它是什么类型的函数等等)，我们是否可以根据这些信息完全确定，或者近似地确定这个函数本身? 如果能够“近似地”确定这个函数，那么我们就不再需要去测量其他 x 处 (即 $x \neq x_i, i = 0, \cdots, n$ 处) 的函数值，而可以直接“近似地”计算它。这种方法是有意义的。因为一方面在我们没有测量同时又希望了解的那些点处的实验可能是十分耗时，甚至是不可能实现的; 另一方面，我们有时候需要在一定的定义域内获得相应物理量 Y 的一个简单的表达式，这个表达式可以用于进一步计算。

不失一般性，我们可以假设已经测量的这些自变量的点是按照由小到大排列的。它们被称为内插问题中的支撑点 (support points) 或者节点 (nodes)。即:$x_0 < x_1 < \cdots < x_n$ 而整数 n 则称为支撑点的数目。我们要求内插的函数 $f(x)$ 必须满足:

$$f(x_i) = y_i, \quad i = 0, 1, \cdots, n \tag{1}$$

显然，支撑点数目越大，我们对函数的了解就越精细。如果我们希望计算的其他点 $x \in [x_0, x_n]$，这个问题称为内插问题; 而如果 $x < x_0$ 或 $x > x_n$，这个问题称为外推问题。当然，笼统地说我们也可以统称其为内插问题。

外推和内差还是不太一样。因为这时候我们感兴趣的是函数在某个已知区间之外的点的函数值。如果我们不能够对函数的形式进行任何的限制的话，外推比内差更可能得到完全疯狂的结果。

注意内插问题与我们后面会讨论的拟合问题是不太一样的。内插问题中我们假定在支撑点 x_i 处测得的函数值 y_i 是严格的。在拟合问题中，我们一般是需要考虑 y_i 的误差的。由于在内插问题中，我们认为在支撑点处的值是严格的。因此，在待定函数之中的参数的数目一般也与支撑点的数目相同。所不同的是函数的形式。函数形式的选择则依赖于我们对问题的物理理解。一般来说，如果我们认为待定的函数在我们研究的区间应当是无限光滑解析的，我们会选择诸如多项式这类的函数; 反之，如果我们认为所研究的函数在该区间中可能有奇异性，我们则可以选择具有极点的分式函数，或者称为有理分式内插。在下面的几节中我们依次来讨论这些内插方法。最后我们会讨论使用样条函数进行分段内

差的情况，这种情形是函数在整个区间中并没有一个统一的形式。样条函数特别是三次样条函数在计算机图像学中有非常广泛的应用。

本章中我们还将讨论函数的计算问题。这包括两种情形。一种情形是函数具有明确的解析表达式。这个表达式可能是由级数或者一个积分表达式给出。我们需要的是从数值上准确地计算这个函数值。另外一类是首先利用某种近似来描写目标函数，然后再计算相应的函数。这一类与内插问题有类似之处。需要强调的是，函数的计算是我们后面几章讨论函数的积分、方程求根以及函数极值问题的基础。

3.1 多项式内插

本节中我们讨论多项式的内插。考虑区间 $[x_0, x_n]$ 上面的 n 次多项式，

$$P_n(x) = a_0 + a_1 x + \cdots + a_n x^n. \tag{2}$$

它具有 $(n+1)$ 个参数：$a_i,\ i = 0, 1, \cdots, n$。我们试图利用这个多项式来解决区间 $[x_0, x_n]$ 上面的内插问题。也就是说，我们希望有，$P_n(x_i) = y_i$, $i = 0, \cdots, n$。这个问题的一般解是著名的拉格朗日多项式。

我们考虑如下形式的 n 次多项式：

$$L_j(x) = \prod_{0 \le m \le n, m \ne j} \frac{x - x_m}{x_j - x_m}, \quad j = 0, 1, \cdots, n. \tag{3}$$

注意，因为分母上 $x_j - x_m$ 是已知的数，所以这个表达式是个典型的 n 次多项式。按照构造，这个多项式满足

$$L_j(x_i) = \delta_{ij}, \quad 0 \le i, j \le n. \tag{4}$$

于是如果我们令

$$P_n(x) \equiv \sum_{i=0}^{n} y_i L_i(x), \tag{5}$$

我们可以直接验证它满足 $P_n(x_i) = y_i,\ i = 0, 1, \cdots, n$。这个公式一般称为拉格朗日内插公式。

对于多项式内插而言，我们有如下的误差估计。假设我们需要写的函数 $f(x)$ 在区间 $[x_0, x_n]$ 上至少具有 $(n+1)$ 阶的导数，那么对于任意的 $x \in [x_0, x_n]$，我们一定可以找到一个 $\xi \in [x_0, x_n]$ 使得，

$$f(x) - P_n(x) = \frac{\omega(x) f^{(n+1)}(\xi)}{(n+1)!} \tag{6}$$

其中 $\omega(x) = (x - x_0) \cdots (x - x_n)$。也就是说，我们必定有

$$|f(x) - P_n(x)| \le \frac{|x_n - x_0|^n}{(n+1)!} \max_{x_0 \le \xi \le x_n} |f^{(n+1)}(\xi)| \tag{7}$$

这个误差估计可以由中值定理得到。定义

$$f(x) - P_n(x) = G(x)\omega(x) \tag{8}$$

把 $x = x^*$ 当成是 $[x_0, x_n]$ 区间中的某个我们感兴趣的点。定义

$$K(x) = f(x) - P_n(x) - G(x^*)\omega(x) \tag{9}$$

很明显，$K(x)$ 有 $x_0, x_1, \cdots, x_n$ 这 $n+1$ 个零点，并且 $x = x^*$ 也是 $K(x)$ 的零点。$K(x_i) = 0$ 和 $K(x_{i+1}) = 0$，必然意味着存在 $\xi_i \in [x_i, x_{i+1}]$，使得 $K'(\xi_i) = 0$，于是 $K'(x)$ 有 $n+1$ 个零点。以此类推，我们最后得到，在 $[x_0, x_n]$ 区间，必然存在 ξ，使得 $K^{(N+1)}(\xi) = 0$。其中 $K^{(n+1)}(x)$ 可以写成

$$K^{(n+1)}(x) = f^{(n+1)}(x) - ((n+1)!)G(x^*) \tag{10}$$

于是，我们得到，对于任意的 $G(x^*)$，$x^* \in [x_0, x_n]$，都存在 ξ，使得

$$f^{(n+1)}(\xi) - ((n+1)!)G(x^*) = 0 \tag{11}$$

或者说，对于任意 $x \in [x_0, x_n]$，都有

$$f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega(x) \tag{12}$$

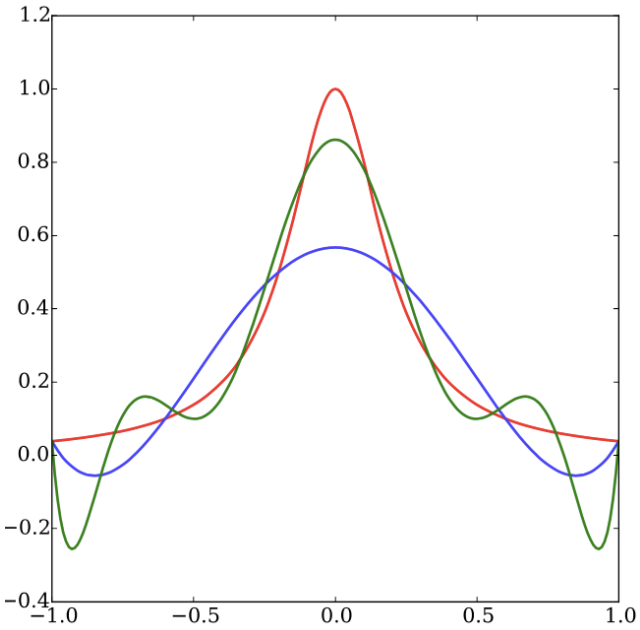
成立。如果导数 $|f^{(n+1)}(\xi)|$ 不是随着 n 增长得很快的话，那么基本上可以保证 $(n+1)! \gg |x_n - x_0|^n$。但很多情况下这个误差并不一定能够保证随着 n 的增加，误差就必定减小。

初看起来这个构造似乎已经完美解决了内插问题，至少是解决了利用多项式进行内插问题。但是实际上它是有一定的问题的。这个问题是当我们测量的支撑点的数目增加时，多项式的次数也随之增加。我们会发现，尽管在所有的支撑点处拉格朗日内插公式都满足 $P_n(x_i) = y_i$，但是在那些不是支撑点的地方，拉格朗日内插多项式可能与我们期望内插的函数 -这个函数的形式我们并不清楚 –可能相差很远。这个现象被称为 Runge 现象。

Runge 当年考虑的函数具体的形式为

$$f(x) = \frac{1}{1 + 25x^2} \tag{13}$$

这个函数一般称为 Runge 函数。



我们在图 3.1中画出了这个函数在 $x \in [-1, +1]$ 之间的图像 (红线)。同时显示的还有 n = 5 阶 (蓝线) 和 n = 9 阶 (绿线) 的拉格朗日多项式拟合。前者在 6 个支撑点处与 Runge 函数 (红线) 吻合; 后者

则在 10 个支撑点处与 Runge 函数吻合。但是只要不在这些支撑点处，随着内插阶数的升高，拉格朗日多项式体现出越来越严重的震荡特性。不在这些支撑点处，拉格朗日多项式严重地偏离原函数。这就是所谓的 Runge 现象。事实上可以证明

$$\lim_{n\to\infty}\left(\max_{-1\le x\le+1}|f(x)-P_n(x)|\right)=+\infty \tag{14}$$

也就是说，随着内插阶数的升高，内插多项式与原函数的最大偏离的绝对值会发散。

在相当多的应用中，内插的目的恰恰是希望得到一定区间内没有测量过的点处的函数值。如果我们的内插函数在支撑点之外的地方可能严重偏离实际的函数，那这种偏差对这一类的应用是不可原谅的。因此我们希望克服这种弊端。具体到 Runge 函数本身，使用分式拟合就可以很好地解决这个问题。或者使用所谓的三次样条函数来进行内插。由于 Runge 函数本身就是一个分式的形式，如果使用正确的分式内插可以完全准确地确定这个函数；而如果我们使用三次样条函数来进行内插，我们也可以获得相当不错的结果。

3.2 有理分式内插

本节我们讨论利用有理分式进行内插。当函数在某个区间内的变化行为比较剧烈时，多项式内插会出现剧烈震荡的行为。这时候利用有理分式进行内插可能会更为合适一些。

考虑一个分式，

$$\Phi^{(m,n)}(x)=\frac{P_m(x)}{Q_n(x)} \tag{15}$$

其中分子和分母分别是 m 阶和 n 阶的多项式。我们下面将假定这两个多项式是互素的，也就是说并不存在一个公共的多项式因子。

下面我们直接构造这样的有理函数。从 $i=0,1,\cdots$ 开始，我们用下面的次序来递推:

i	x_i	y_i			
0	x_0	y_0			
1	x_1	y_1	$\phi(x_0,x_1)$		
2	x_2	y_2	$\phi(x_0,x_2)$	$\phi(x_0,x_1,x_2)$	
3	x_3	y_3	$\phi(x_0,x_3)$	$\phi(x_0,x_1,x_3)$	$\phi(x_0,x_1,x_2,x_3)$
$\cdots$			$\cdots$		

其中的各个函数 ϕ 可以按照下式进行递推地定义:

$$\begin{aligned}
\phi(x_i,x_j) &= \frac{x_i-x_j}{y_i-y_j}\\
\phi(x_i,x_j,x_k) &= \frac{x_j-x_k}{\phi(x_i,x_j)-\phi(x_i,x_k)}\\
&\cdots\\
\phi(x_i\cdots,x_l,x_m,x_n) &= \frac{x_m-x_n}{\phi(x_i,\cdots,x_l,x_m)-\phi(x_i,\cdots,x_l,x_n)}
\end{aligned} \tag{16}$$

显然，为了获得具体的数而不是发散的结果，我们应当尽可能选择函数单调的一个区间进行内插的构造。由于我们总是假设各个 x_i 是不相同的，因此如果函数是单调的，那么上述各个构造的差值比 ϕ 就不会发散。一旦获得了这些系数，我们可以构造一个有理分式:

$$\Phi^{(n,n)}(x)=\frac{P_n(x)}{Q_n(x)} \tag{17}$$

它是两个 n 次多项式的比。我们要求它能够经过 $(2n+1)$ 个点 (x_i, y_i)，$i = 0, 1, \cdots, 2n$。事实上这个结果可以写成一个连分数:

$$\Phi^{(n,n)}(x) = y_0 + \cfrac{x - x_0}{\phi(x_0, x_1) + \cfrac{x - x_1}{\phi(x_0, x_1, x_2) + \cfrac{x - x_2}{\ddots + \cfrac{x - x_{2n-1}}{\phi(x_0, \cdots, x_{2n})}}}}.$$

可以证明这个分式恰好通过给定的 $2n+1$ 个点，即满足 $\Phi^{(n,n)}(x_i) = y_i$，$i = 0, 1, \cdots, 2n$。

作为一个例子，我们考虑前面曾经提及的 Runge 函数:

$$f(x) = \frac{1}{1 + 25x^2} \tag{18}$$

我们选择其单调的区间，比如 $[0, 1]$ 来进行内插。假定知道下列三个点的数值: $x^2 = 0, 1/25, 1$，相应的函数值分别为:$1, 1/2, 1/26$。按照上面的构造，按照 x^2 的有理分式为,

$$f(x) = \Phi^{(1,1)}(x^2) = 1 + \frac{x^2}{\phi(0, 1/25) + \frac{x^2 - 1/25}{\phi(0, 1/25, 1)}} \tag{19}$$

其中的几个系数,

$$\begin{aligned} \phi(0, 1/25) &= \frac{0 - 1/25}{1 - 1/2} = -\frac{2}{25} \\ \phi(0, 1) &= \frac{0 - 1}{1 - 1/26} = -\frac{26}{25} \\ \phi(0, 1/25, 1) &= \frac{1/25 - 1}{-2/25 + 26/25} = -1 \end{aligned} \tag{20}$$

将这些数值带入前一式我们发现，我们竟然可以完全重构 $f(x)$。

3.3 样条函数内插

样条这个词最早源于早期工程师绘图所用的薄木条，将它固定在一些给定的数据点上，就可以绘出一条连接各点的光滑曲线。我们可以想象一下，要连接 2 个点，用一个一次多项式就可以，但是我们无法保证在端点处曲线是光滑的，也就是一阶导数连续。如果我们用一个二次多项式去连接 2 个点，我们就能保证曲线在端点处是光滑的。这就是所谓的二次样条插值。

定义：给定区间 $[a, b]$ 的一个 n 段分割：$a = x_0 < x_1 \cdots x_{n-1} < x_n = b$, 二次样条函数 $S(x)$ 满足以下条件：

- $S(x)$ 在每个区间 $[x_{i-1}, x_i]$ 上是一个二次多项式；
- $S(x)$ 在所有节点处满足 x_i $(i = 1, 2, \cdots, n-1)$ 上具有一阶连续导数；
- $S(x)$ 在所有节点处满足 $S(x_i) = y_i$ $(i = 0, 1, \cdots, n)$。

在每个小区间 $[x_{i-1}, x_i]$ 上是一个二次多项式，有 3 个系数，因此要确定 $S(x)$ 就要确定 $3n$ 个待定参数，而由 $S(x_i - 0) = S(x_i + 0) = y_i$ $(i = 1, \cdots, n-1)$，得到 $2(n-1)$ 个方程；在两个端点处，$S(x_i) = y_i$，有 2 个方程。然后根据导数连续 $S'(x_i - 0) = S'(x_i + 0)$ $(i = 1, \cdots, n-1)$ 再得到 $(n-1)$ 个方程。一共是 $3n - 1$ 个方程。为了确定一个待定的样条插值函数，还需增加 1 个条件，这个条件通常是在区间 $[a, b]$ 的两端处给出，即边界条件，边界条件根据实际问题的需求来确定，其类型很多，常见的边界条件类型有：

- 给定初始端点或者终端的一阶导数值：$S'(x_0) = y'_0$ 或者 $S'(x_n) = y'_n$
- 给定初始端点或者终端的二阶导数值：$S''(x_0) = y''_0$ 或者 $S''(x_n) = y''_n$
- 若插值函数为周期函数时，此时 $y_0 = y_n$，可以给定周期性边界条件：$S'(x_0) = S'(x_n)$
- 非结点条件：假定 $[x_0, x_2]$ 区间用一个统一的二次多项式描述，或者 $[x_{n-2}, x_n]$ 之间用一个统一的二次多项式描述

对于最后一个条件，我们原本在 $[x_0, x_1]$ 和 $[x_1, x_2]$ 区间各有一个二次多项式，总共 6 个参数，方程是 $S(x_0) = y_0$，$S(x_1 - 0) = S(x_1 + 0) = y_1$，$S(x_2 - 0) = y_2$，$S'(x_1 - 0) = S'(x_1 + 0)$，总共 5 个方程。现在 $[x_0, x_2]$ 区间用一个统一的二次多项式描述，那么我们一共有 3 个参数。方程也刚好是 3 个：$S(x_0) = y_0$，$S(x_1) = y_1$，$S(x_2 - 0) = y_2$。原来多出一个参数的情况，现在变成参数数目和方程数目正好相等。

假设 $S'(x_j) = M_j$ $(j = 0, \cdots, n)$，在各个小区间内，$S'(x)$ 为一次多项式，我们可以写成

$$S'(x) = M_j \left(\frac{x_{j+1} - x}{x_{j+1} - x_j} \right) + M_{j+1} \left(\frac{x - x_j}{x_{j+1} - x_j} \right), \quad x \in [x_j, x_{j+1}] \tag{21}$$

我们可以定义 $h_j = x_{j+1} - x_j$。做一次积分以后，得到

$$S(x) = -\frac{M_j}{2h_j}(x - x_{j+1})^2 + \frac{M_{j+1}}{2h_j}(x - x_j)^2 + a_j \tag{22}$$

根据 $S(x_j) = y_j$，$S(x_{j+1}) = y_{j+1}$，我们可以得到

$$-M_j h_j/2 + a_j = y_j, \quad M_{j+1} h_j/2 + a_j = y_{j+1} \tag{23}$$

下一步，就是得到递推关系式

$$M_{j+1} = \frac{2(y_{j+1} - y_j)}{h_j} - M_j \tag{24}$$

再加上边界条件。我们就能确定所有 M_j 的值以及 a_j 的值，于是 $S(x)$ 确定。

有二次样条函数，就有 k 次样条函数。当多项式的阶数比较小的时候，插值函数不够光滑；当多项式阶数很高的时候，比方说 n 阶多项式，事实上样条函数等同于多项式插值，这个时候就没法避免 Runge 现象的发生。所以简单的分析告诉我们，阶数并不是越大越好，也不是越小越好。事实上，当我们考虑某个区间 $[a, b]$ 上的 $m > 0$ 阶以下的所有阶导数连续且其 m 阶导数平方可积的函数构成的函数空间，将其记为 $\mathcal{K}^m[a, b]$。对于任意的函数 $f \in \mathcal{K}^2[a, b]$，我们可以定义一个函数的模

$$||f|| = \int_a^b dx \, |f''(x)|^2 \tag{25}$$

我们知道一个函数 $f(x)$ 的曲率 $R(x)=f''(x)(1+f'(x)^2)^{-3/2}$。如果在一个区间上 $f'(x)^2$ 比起 1 来说要小很多话，那么函数的曲率几乎就等于 $f''(x)$。因此，上面这个模从某种意义上是衡量了函数在一个区间上曲率模方的大小。可以证明，三次样条函数实际上是使得这个模最小的函数。换句话说，它是“最光滑的”函数。

对于三次样条插值函数，我们要求

- $S(x)$ 在每个区间 $[x_{i-1},x_i]$ 上是一个三次多项式；
- $S(x)$ 在所有节点处满足 x_i $(i=1,2,\cdots,n-1)$ 上具有一阶连续导数和二阶连续导数；
- $S(x)$ 在所有节点处满足 $S(x_i)=y_i$ $(i=0,1,\cdots,n)$。

对于三次样条插值函数，要确定 $S(x)$ 就要确定 $4n$ 个待定参数。由 $S(x_i-0)=S(x_i+0)=y_i$ $(i=1,\cdots,n-1)$，得到 $2(n-1)$ 个方程；在两个端点处，$S(x_i)=y_i$，有 2 个方程。然后根据一阶、二阶导数连续 $S'(x_i-0)=S'(x_i+0)$，$S''(x_i-0)=S''(x_i+0)$ $(i=1,\cdots,n-1)$ 再得到 $2(n-1)$ 个方程。一共是 $4n-2$ 个方程。所以需要增加 2 个边界条件来确定样条函数。常见的边界条件类型有：

- 同时给定初始端点和终端的一阶导数值：$S'(x_0)=y_0'$，$S'(x_n)=y_n'$
- 同时给定初始端点和终端的二阶导数值：$S''(x_0)=y_0''$，$S''(x_n)=y_n''$
- 若插值函数为周期函数时，此时 $y_0=y_n$，可以给定两个周期性边界条件：$S'(x_0)=S'(x_n)$，$S''(x_0)=S''(x_n)$
- 非结点条件：假定 $[x_0,x_2]$ 区间用一个统一的三次多项式描述，并且 $[x_{n-2},x_n]$ 之间也可以用一个统一的三次多项式描述

关于三次样条插值函数的构造，我们可以采取类似的方式。假设 $S''(x_j)=M_j$ $(j=0,\cdots,n)$，在各个小区间内，$S''(x)$ 为一次多项式，我们可以写成

$$S''(x)=M_j\left(\frac{x_{j+1}-x}{x_{j+1}-x_j}\right)+M_{j+1}\left(\frac{x-x_j}{x_{j+1}-x_j}\right),\quad x\in[x_j,x_{j+1}] \tag{26}$$

对这个式子做两次积分

$$S(x)=-\frac{M_j}{6h_j}(x-x_{j+1})^3+\frac{M_{j+1}}{6h_j}(x-x_j)^3+A_j(x-x_j)+B_j,\quad x\in[x_j,x_{j+1}] \tag{27}$$

根据 $S(x_j)=y_j$，$S(x_{j+1})=y_{j+1}$，我们可以得到

$$A_j=\frac{y_{j+1}-y_j}{h_j}-\frac{h_j}{6}(M_{j+1}-M_j),\quad B_j=y_j-M_j\frac{h_j^2}{6} \tag{28}$$

我们来回顾一下，在 $S''(x)$ 的构造过程中，我们已经利用了 $S''(x)$ 连续的条件；在确定 A_j，B_j 时，用掉了 $S(x_j)=y_j$ 的条件；我们还剩一个条件没有用，就是 $S'(x)$ 连续。$S'(x)$ 的表达式是

$$S'(x)=-\frac{M_j}{2h_j}(x-x_{j+1})^2+\frac{M_{j+1}}{2h_j}(x-x_j)^2+A_j,\quad x\in[x_j,x_{j+1}] \tag{29}$$

先考虑节点 x_j 处的左导数，它通过 $[x_{j-1},x_j]$ 区间上的函数表达式来计算，即

$$S'(x_j-0)=\frac{M_j}{2}h_{j-1}+A_{j-1}=\frac{h_{j-1}}{6}M_{j-1}+\frac{h_{j-1}}{3}M_j+\frac{y_j-y_{j-1}}{h_{j-1}} \tag{30}$$

同理，利用 $[x_j, x_{j+1}]$ 区间的函数表达式计算节点 x_j 的右导数

$$S'(x_j+0) = -\frac{M_j}{2}h_j + A_j = -\frac{h_j}{6}M_{j+1} - \frac{h_j}{3}M_j + \frac{y_{j+1}-y_j}{h_j} \tag{31}$$

要求一阶导数连续，我们得到

$$\mu_j M_{j-1} + 2M_j + \lambda_j M_{j+1} = d_j, \quad j = 1, \cdots, n-1 \tag{32}$$

其中，$\mu_j = \frac{h_{j-1}}{h_{j-1}+h_j}$，$\lambda_j = \frac{h_j}{h_{j-1}+h_j}$，$d_j = 6\left[\frac{y_{j-1}}{h_{j-1}(h_{j-1}+h_j)} + \frac{y_{j+1}}{h_j(h_{j-1}+h_j)} - \frac{y_j}{h_{j-1}h_j}\right]$。这个形式很像三对角方程。如果我们通过两个边界条件，得到

$$2M_0 + \lambda_0 M_1 = d_0, \quad \mu_n M_{n-1} + 2M_n = d_n \tag{33}$$

那么要确定 M_i 的值，只需要解出三对角方程

$$\begin{pmatrix} 2 & \lambda_0 & & & \\ \mu_1 & 2 & \lambda_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-1} & 2 & \lambda_{n-1} \\ & & & \mu_n & 2 \end{pmatrix} \begin{pmatrix} M_0 \\ M_1 \\ \cdots \\ M_{n-1} \\ M_n \end{pmatrix} = \begin{pmatrix} d_0 \\ d_1 \\ \cdots \\ d_{n-1} \\ d_n \end{pmatrix} \tag{34}$$

3.4 函数的近似与计算

本节我们讨论函数的近似与计算问题。我们将分为几类的函数的计算。一类是函数具有明确的表达式的计算问题。另一类是用某种函数近似后进行的数值计算。

3.4.1 级数表达的函数的计算

很多函数是以级数的形式表达的 (比如 Bessel 函数、Legendre 函数等等)。很多函数实际上也是这样去计算的。需要指出的是，尽管某些函数的级数展开在数学上是收敛的，但是直接按照级数去计算往往并不是最合适的方法。典型的例子如

$$\sin x = \sum_{k=0}^{\infty} (-)^k \frac{x^{2k+1}}{(2k+1)!} \tag{35}$$

我们知道这个级数对于任意的 x 都是收敛的，但是如果 $|x| \geq 1$，那么直接这么计算需要求和的项数是比较多的。事实上，一种更为有效的做法是首先将 x 化到第一或第四象限: $x \in [-\pi/2, \pi/2]$，然后我们可以利用公式:$\sin x = 3\sin(x/3) - 4\sin^3(x/3)$ 将其化为计算较小的角度 $(x/3)$ 的正弦问题。当然，这个过程可以进一步迭代直到 (x/3) 足够地小以至于一两项级数就足以满足精度要求为止。当然更为现代一些的算法是利用我们后面提及的 Chebyshev 近似来进行计算。事实上，多数的超越函数都可以利用 Chebyshev 近似的方法进行计算。(超越函数 (Transcendental Functions)，指的是变量之间的关系不能用有限次加、减、乘、除、乘方、开方运算表示的函数。)

3.4.2 函数的 Chebyshev 近似及其计算

首先我们来讨论一下连续函数的最佳平方逼近。假设对函数 $f(x)$, $x \in [a,b]$ 进行函数逼近,

$$S(x) = \sum_{n=0}^{N-1} c_n T_n(x) \tag{36}$$

其中 c_n $(n = 0, 1 \cdots, N-1)$ 是代定参数。我们用来逼近的函数 $T_n(x)$ 应该是形式简单的函数 (比方说 $T_n(x) = x^n$, 就是多项式逼近 ; 当然这里简单, 并不是指的狭义上的数学形式简单, 更多是指可以通过递推关系的构造, 在计算机算法上实现起来简单)。连续函数的最佳平方逼近问题就是求 $S(x)$, 使得 $||S(x) - f(x)||_2$ 达到最小值。也就是说, 我们要将

$$F = ||S(x) - f(x)||_2^2 = \int_a^b dx \left[\sum_{n=0}^{N-1} c_n T_n(x) - f(x)\right]^2 \tag{37}$$

最小化。

根据 $p = 2$-范数和内积的关系, 我们可以把 F 写成

$$F = ||S - f||_2^2 = \left\langle \sum_{n=0}^{N-1} c_n T_n(x) - f, \sum_{n=0}^{N-1} c_n T_n(x) - f \right\rangle \tag{38}$$

由多元函数取极值的条件知道, 系数 c_n 应满足方程

$$\frac{\partial F}{\partial c_n} = 0, \quad n = 0, \cdots, N-1 \tag{39}$$

经过推导以后, 可以得到等价的方程

$$\sum_{n=0}^{N-1} c_n \langle T_n, T_m \rangle = \langle f, T_m \rangle, \quad m = 0, 1, \cdots, N-1 \tag{40}$$

我们只需要解这个方程组即可。特别地, 如果对于我们构造的函数有正交关系 $\langle T_n, T_m \rangle = A_n \delta_{mn}$, 那么这个方程组的解是很简单的

$$c_n = \langle f, T_n \rangle / A_n \tag{41}$$

有时候我们会引入加权正交的概念, 设函数 $f(x)$,$g(x)$ 是变量取值范围为 $[a,b]$ 的函数, $\rho(x)$ 为权函数, 如果

$$\langle f(x), g(x) \rangle = \int_a^b \rho(x) f(x) g(x) dx = 0 \tag{42}$$

则称 $f(x)$ 与 $g(x)$ 带权正交。若函数族 $\{T_0(x), \cdots, T_n(x), \cdots\}$ 满足

$$\langle T_n(x), T_m(x) \rangle = \int_a^b dx\, \rho(x) T_n(x) T_m(x) = A_n \delta_{mn} \tag{43}$$

则称函数族 $\{T_n(x)\}$ 是 $[a,b]$ 区间上的带权正交函数族。这里说明一下, 就好像矩阵的范数的定义不是唯一的一样, 连续函数的最佳平方逼近也不是唯一的, 我们可以在里面加入一个权重因子, 一般来讲, 只要这个因子满足正定和非奇异性就好。

对于变量取值范围为 $x \in [a,b]$ 的函数, 我们不妨做变量替换 $x \to x' = \frac{2(x-a)}{b-a} - 1$, 使得 $x' \in [-1,1]$。所以下面我们对于任意的 $x \in [-1,1]$, 引入 n 阶的第一类 Chebyshev 多项式 $T_n(x)$, 它可以明确地写为,

$$T_n(x) = \cos(n \arccos x). \tag{44}$$

对于最低阶的几个多项式，我们很容易写出它的明确表达式:

$$T_0(x)=1,\quad T_1(x)=x,\quad T_2(x)=2x^2-1,\quad T_3(x)=4x^3-3x$$
$$T_{n+1}(x)=2xT_n(x)-T_{n-1}(x) \tag{45}$$

Chebyshev 多项式满足一系列重要的性质。其中一个比较重要的性质是它满足加权正交关系

$$\int_{-1}^{1}dx\,\frac{T_i(x)T_j(x)}{\sqrt{1-x^2}}=\begin{cases}0 & i\neq j\\ \pi/2 & i=j\neq 0\\ \pi & i=j=0\end{cases} \tag{46}$$

另一个比较重要的性质是其零点以及极值点的位置。$T_n(x)$ 在 $[-1,1]$ 之中恰好有 n 个零点，它们的位置由下式给出:

$$x_{n,k}=\cos\left(\frac{\pi(k+1/2)}{n}\right),\quad k=0,1,\cdots,n-1 \tag{47}$$

而它的极值点共有 $(n+1)$ 个，其位置为,

$$\hat{x}_{n,k}=\cos\left(\frac{\pi k}{n}\right),\quad k=0,1,\cdots,n \tag{48}$$

在这些极值点处，Chebyshev 多项式恰好等于 $+1$(极大值) 或 -1(极小值)。也就是说，Chebyshev 多项式的绝对值总是在 1 以下的。正是这个特性使得它成为近似任意函数的有力工具，因为它可以很好地控制误差。

下面我们要用 Chebyshev 多项式来对函数 $f(x)$ 进行近似展开。我们首先来构造 Chebyshev 多项式，它的系数为 c_m, $m=0,1,\cdots,N-1$

$$c_m=\frac{1}{A_m}\int_{-1}^{1}dx\,\frac{f(x)T_m(x)}{\sqrt{1-x^2}},\quad \int_{-1}^{1}dx\,\frac{T_n(x)T_m(x)}{\sqrt{1-x^2}}=A_m\delta_{mn} \tag{49}$$

我们对这个积分进行近似，得到

$$c_m\approx c_{N,m}=\frac{2-\delta_{0m}}{N}\sum_{k=0}^{N-1}T_m(x_{N,k})f(x_{N,k}), \tag{50}$$

这里 $x_{N,k}=\cos\left(\frac{\pi(k+1/2)}{N}\right)$ 是 T_N 的零点。得到 $c_{N,m}$ 后，可以构造

$$S(x)=\sum_{m=0}^{N-1}c_{N,m}T_m(x) \tag{51}$$

这个表达式也给出了函数 $f(x)$ 的一个非常好的近似。它和最佳平方逼近的差别在于 $c_{N,m}\approx c_m$。但它有一个很好的特性在于，在零点 $x_{N,k}$ 处，函数值严格等于 $f(x_{N,k})$。

$$S(x_{N,k})=\sum_{m=0}^{N-1}c_{N,m}T_m(x_{N,k})=\sum_{m=0}^{N-1}\frac{2-\delta_{0m}}{N}\sum_{k'=0}^{N-1}T_m(x_{N,k'})f(x_{N,k'})T_m(x_{N,k}) \tag{52}$$

可以证明

$$\frac{T_0(x_{N,k'})T_0(x_{N,k})}{2}+\sum_{m=1}^{N-1}T_m(x_{N,k'})T_m(x_{N,k})=\frac{N}{2}\delta_{kk'} \tag{53}$$

马上可以得到

$$S(x_{N,k}) = f(x_{N,k}) \tag{54}$$

由于 Chebyshev 具有模永远小于 1 的特性，因此它可以很好地控制误差。它非常接近于相应的最佳平方逼近，而且其计算也方便很多。由于截断到 N 阶的近似式的下一阶是 $c_N T_N(x)$ 而 $|T_N(x)| \leq 1$，因此误差主要由 $|c_N|$ 的大小来控制。一般来说随 N 的增加 c_N 衰减得非常快 (如果函数足够光滑的话，它们随着 N 的增加是指数衰减的)，所以我们往往仅仅需要几阶就够了。这正是 Chebyshev 强大的地方。我们运用 Chebyshev 最多的时候并不是利用很大的 N 的时候，而是往往利用不那么大的时候。

我们在定义 Chebyshev 多项式时，有用到

$$T_n(x) = \cos(n \arccos x). \tag{55}$$

那么定义域为 $[-1, 1]$。但如果我们采用递推关系的定义方式

$$\begin{aligned} T_0(x) &= 1, \quad T_1(x) = x, \\ T_{n+1}(x) &= 2xT_n(x) - T_{n-1}(x) \end{aligned} \tag{56}$$

那么，在任何实数域都可以得到 Chebyshev 多项式，或者说我们可以把 Chebyshev 多项式解析延拓到整个实空间。只是在 $[-1, 1]$ 区间，$T_n(x)$ 值的绝对值小于 1，而当 $|x| > 1$ 时，随着 n 变大，$T_n(x)$ 呈 $2^n x^n$ 阶的上升趋势。利用这个性质，我们可以构造 Chebyshev 过滤器 (Chebyshev filter)。我们首先定义一个函数

$$q(x; \alpha, \beta) = \frac{2x^2 - (\alpha^2 + \beta^2)}{\alpha^2 - \beta^2} \tag{57}$$

其中 $\alpha > \beta$。显然，这个函数在 $x = \beta$ 和 $x = \alpha$ 之间从 -1 变到 $+1$。然后我们把 $q(x; \alpha, \beta)$ 作为变量放到 Chebyshev 多项式中，比方说

$$T_n(q(x; \alpha, \beta)), \quad n = 10 \tag{58}$$

因为这个多项式在 $x > \alpha$ 和 $x < \beta$ 的值要远大于 $x \in [\beta, \alpha]$ 区间的值，等效于这个多项式把 $[\beta, \alpha]$ 的信息过滤掉了。Chebyshev 过滤器的一个应用是求一个庞大矩阵的若干个小本征值和本征矢量。我们之前曾经讨论过，如果能得到一个矩阵 A 的小于 λ_0 的本征值和本征矢量，那么利用它们我们可以把线性方程组求解的条件数从 $\mathrm{cond} = \lambda_{\max}/\lambda_{\min}$ 减小于 $\lambda_{\max}/\lambda_0$。但要求矩阵 A 的小本征值和本征矢量并不是那么容易，反而最大本征值和本征矢量是比较好求的。比方说我们可以构造矩阵多项式 $p(A)$，这个多项式作用在任意矢量 $v = \sum_i a_i v_i$ 上面，得到

$$p(A)v = \sum_i a_i p(\lambda_i) v_i \tag{59}$$

当多项式阶数比较高的时候，与大本征值对应的矢量被放大，与小本征值对应的矢量被压低。所以我们相对容易得到大本征值和与之相关的本征矢量。要求小本征值系统，我们就可以采用 Chebyshev 过滤器，对于

$$T_n(q(A; \alpha, \beta)) \tag{60}$$

来讲，所有在 $[\beta, \alpha]$ 区间的本征矢量都会被迅速压低，那么我们只需要设定 $\alpha > \lambda_{\max}$，$\beta \approx \lambda_0$，我们就能得到小于 λ_0 的本征矢量构成的子空间的信息。