MA2213: Numerical Analysis I

Lecture 10: Cubic spline interpolation

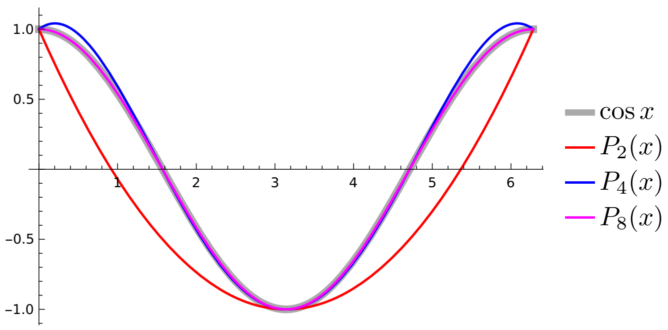
Zhenning Cai

October 17, 2019

In most cases, in Lagrange interpolation, if we place more nodes in an interval, the approximation is better. For example, we consider the Lagrange interpolation of the cosine function in $[0, 2\pi]$:

- $n = 2$, $x_k = k\pi$, $k = 0, 1, 2$:
$$P_2(x) = \frac{2}{\pi^2}s^2 - \frac{4}{\pi}s + 1.$$
- $n = 4$, $x_k = k\pi/2$, $k = 0, 1, 2, 3, 4$:
$$P_4(x) = -\frac{8}{3\pi^4}s^4 + \frac{32}{3\pi^3}s^3 - \frac{34}{3\pi^2}s^2 + \frac{4}{\pi}s + 1.$$
- $n = 8$, $x_k = k\pi/4$, $k = 0, 1, \cdots, 8$:
$$\begin{aligned} P_8(x) = {} & \frac{2048(12\sqrt{2} - 17)}{315\pi^8}s^8 - \frac{16384(12\sqrt{2} - 17)}{315\pi^7}s^7 + \frac{256(358\sqrt{2} - 507)}{45\pi^6}s^6 \\ & - \frac{512(306\sqrt{2} - 433)}{45\pi^5}s^5 + \frac{56(2672\sqrt{2} - 3777)}{45\pi^4}s^4 - \frac{32(2448\sqrt{2} - 3463)}{45\pi^3}s^3 \\ & + \frac{2(70592\sqrt{2} - 100633)}{315\pi^2}s^2 - \frac{4(1088\sqrt{2} - 1539)}{105\pi}s + 1. \end{aligned}$$

The following figure shows that from $P_2(x)$ to $P_8(x)$, the Lagrange interpolating polynomials provide better approximation to the cosine function:



However, for some functions, using more nodes may yield worse results. Such a phenomenon is called *Runge's phenomenon*.

1 Runge's phenomenon

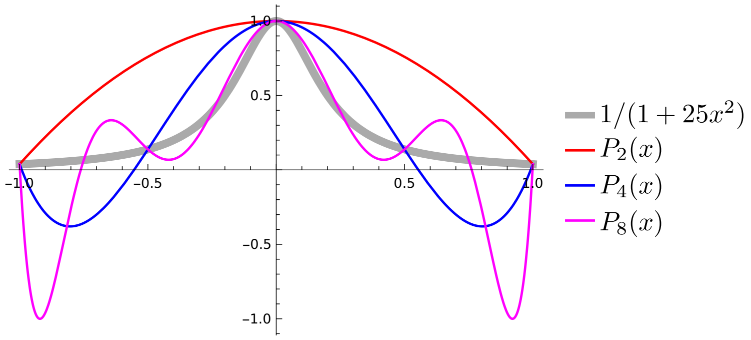
Define the Runge function

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

Simiarly, we consider its Lagrange interpolation on the nodes $x_k = -1+2k/n$, $k = 0, 1, \cdots, n$. When $n = 2, 4, 8$, the Lagrange interpolating polynomials are

$$\begin{aligned}
P_2(x) &= 1 - \frac{25}{26}s^2, \\
P_4(x) &= 1 - \frac{3225}{754}s^2 + \frac{1250}{377}s^4, \\
P_8(x) &= 1 - \frac{98366225}{7450274}s^2 + \frac{228601250}{3725137}s^4 - \frac{383000000}{3725137}s^6 + \frac{200000000}{3725137}s^8.
\end{aligned}$$

These functions are plotted in the following figure:



From which we see that more interpolation nodes cause larger deviation for x close to ± 1.

Such a phenomenon can be explained from the remainder of the Lagrange interpolation:

$$|f(x) - P_n(x)| = \frac{|f^{(n+1)}(\xi)|}{(n+1)!}|(x-x_0)(x-x_1)\cdots(x-x_n)|,$$

where $\xi \in (x_0, x_n)$. Since $x_0, x_1, \cdots, x_n$ are equidistributed, we have

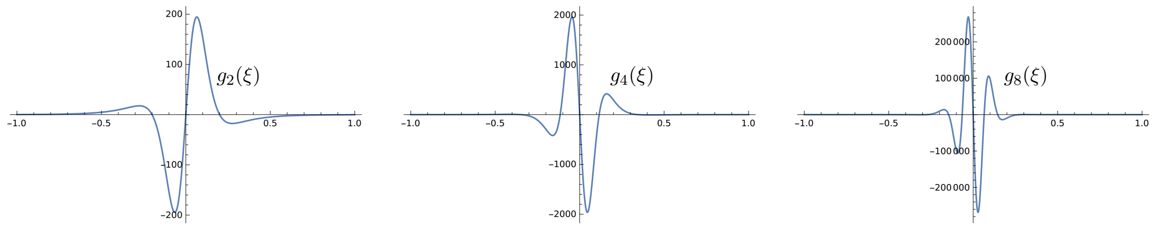
$$|(x-x_0)(x-x_1)\cdots(x-x_n)| \leqslant \left(\frac{x_n - x_0}{n}\right)^{n+1} n!.$$

Therefore we get the error estimation:

$$|f(x) - P_n(x)| \leqslant \frac{1}{n+1}\left(\frac{x_n - x_0}{n}\right)^{n+1} \max_{\xi\in[x_0,x_n]} f^{(n+1)}(\xi). \tag{1}$$

For the interpolation of the Runge function, the right-hand side of the above inequality increases as n increases, as can be seen from the graphs of the function

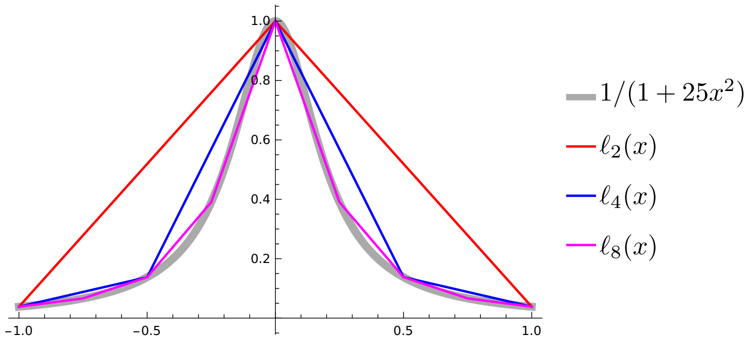
$$g_n(\xi) = \frac{1}{n+1}\left(\frac{2}{n}\right)^{n+1} f^{(n+1)}(\xi):$$



This shows that the error estimation (1) does not guarantee the decay of the error as n increases. In general, when $|f^{(n)}(x)|$ increases rapidly as n increases, the convergence of Lagrange interpolation is questionable.

2 Cubic spline interpolation

An alternative of Lagrange interpolation is the piecewise linear interpolation. Graphically, it is simply to connect all the points on the graph by line segments. For example, let $\ell_n(x)$ be the piecewise linear interpolation of the Runge function with n equidistributed nodes. Then the graphs of $\ell_n(x)$ for $n = 2, 4, 8$ are



Mathematically, such an interpolation can be expressed by

$$\ell_n(x) = f(x_{n-1})\frac{x - x_n}{x_{n-1} - x_n} + f(x_n)\frac{x - x_{n-1}}{x_n - x_{n-1}}, \qquad \text{for } x \in [x_{n-1}, x_n].$$

One can expect that when $n \to \infty$, the interpolating function converges to the original function $f(x)$. However, the functions $\ell_n(x)$ are non-smooth and the convergence is slow.

To improve the quality of interpolation, we consider in this lecture the interpolation using piecewise cubic polynomials. One commonly used method is called *cubic spline interpolation*, defined as follows:

Definition 1. Given a function f defined on $[a, b]$ and a set of nodes $a = x_0 < x_1 < \cdots < x_n = b$, the *spline* $S(x)$ is the *cubic spline interpolation* of $f(x)$ if $S(x)$ satisfies

- $S(x_k) = f(x_k)$ for all $k = 0, 1, \cdots, n$;
- $S(x)$ is a cubic polynomial on each subinterval $[x_{k-1}, x_k]$, $k = 1, 2, \cdots, n$;
- $S(x)$ is second-order differentiable;
- One of the following sets of boundary conditions is satisfied:

  1. $S''(x_0) = S''(x_n) = 0$ (natural boundary conditions);
  2. $S'(x_0) = f'(x_0)$, $S'(x_n) = f'(x_n)$ (clamped boundary conditions). □

Now we assume that $f(x_0), f(x_1), \cdots, f(x_n)$ are given, and we start to find $S(x)$. Assume that

$$S(x) = S_k(x), \qquad \text{for } x \in [x_{k-1}, x_k], \qquad k = 1, \cdots, n.$$

By the definition of the cubic spline, each $S_k(x)$ is a cubic polynomial. Since $S(x)$ is second-order differentiable, we have

$$S_k''(x_k) = S_{k+1}''(x_k) = S''(x_k), \qquad \text{for all } k = 1, \cdots, n-1.$$

Let $M_k = S''(x_k)$, $k = 0, 1, \cdots, n$. By the above equality, we get the following conditions for $S_k(x)$:

$$S_k''(x_{k-1}) = M_{k-1}, \qquad S_k''(x_k) = M_k. \tag{2}$$

Since $S_k(x)$ is a cubic polynomial, its second-order derivative $S_k''(x)$ is a linear polynomial. The unique linear polynomial satisfying (2) is

$$S_k''(x) = M_{k-1}\frac{x - x_k}{x_{k-1} - x_k} + M_k \frac{x - x_{k-1}}{x_k - x_{k-1}}.$$

Integrating this equation twice, we know that there exist two constants A_k and B_k such that

$$S_k(x) = M_{k-1}\frac{(x - x_k)^3}{6(x_{k-1} - x_k)} + M_k \frac{(x - x_{k-1})^3}{6(x_k - x_{k-1})} + A_k x + B_k. \tag{3}$$

To determine A_k and B_k, we insert the above formula to the equations $S_k(x_{k-1}) = f(x_{k-1})$ and $S_k(x_k) = f(x_k)$ to get

$$\begin{aligned}
&\frac{1}{6}M_{k-1}(x_{k-1} - x_k)^2 + A_k x_{k-1} + B_k = f(x_{k-1}),\\
&\frac{1}{6}M_k(x_k - x_{k-1})^2 + A_k x_k + B_k = f(x_k).
\end{aligned}$$

Then A_k and B_k can be solved as

$$\begin{aligned}
A_k &= \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} - \frac{1}{6}(M_k - M_{k-1})(x_k - x_{k-1}),\\
B_k &= \frac{x_k f(x_{k-1}) - x_{k-1} f(x_k)}{x_k - x_{k-1}} - \frac{1}{6}(M_k x_{k-1} - M_{k-1} x_k)(x_k - x_{k-1}).
\end{aligned}$$

By now, it remains only to find the values of $M_0, M_1, \cdots, M_n$. To determine these quantities, we need the first-order differentiability of $S(x)$:

$$S_k'(x_k) = S_{k+1}'(x_k), \qquad k = 1, \cdots, n-1. \tag{4}$$

Using (3), one can get the derivative of S_k:

$$S_k'(x) = M_{k-1}\frac{(x - x_k)^2}{2(x_{k-1} - x_k)} + M_k \frac{(x - x_{k-1})^2}{2(x_k - x_{k-1})} + A_k.$$

Therefore

$$S_k'(x_k) = M_k \frac{x_k - x_{k-1}}{2} + A_k = f[x_{k-1}, x_k] + \frac{2M_k + M_{k-1}}{6}(x_k - x_{k-1}). \tag{5}$$

Similarly,

$$S_{k+1}'(x_k) = f[x_k, x_{k+1}] - \frac{2M_k + M_{k+1}}{6}(x_{k+1} - x_k). \tag{6}$$

According to (4), we equate (5) and (6) to get

$$\begin{aligned}
\frac{x_{k+1} - x_k}{x_{k+1} - x_{k-1}} M_{k+1} + 2M_k + \frac{x_k - x_{k-1}}{x_{k+1} - x_{k-1}} M_{k-1} &= \frac{6\big(f[x_k, x_{k+1}] - f[x_{k-1}, x_k]\big)}{x_{k+1} - x_{k-1}}\\
&= 6f[x_{k-1}, x_k, x_{k+1}].
\end{aligned} \tag{7}$$

Thus we get $n-1$ equations for $M_0, M_1, \cdots, M_n$, and we need two more equations from the boundary conditions.

For natural boundary conditions, we know that $M_0 = M_n = 0$. Therefore the equations (7) can be written as the following linear system:

$$\begin{pmatrix} 2 & \lambda_1 & & & \\ \mu_2 & 2 & \lambda_2 & & \\ & \mu_3 & 2 & \ddots & \\ & & \ddots & \ddots & \lambda_{n-2} \\ & & & \mu_{n-1} & 2 \end{pmatrix} \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_{n-2} \\ M_{n-1} \end{pmatrix} = \begin{pmatrix} 6f[x_0, x_1, x_2] \\ 6f[x_1, x_2, x_3] \\ \vdots \\ 6f[x_{n-3}, x_{n-2}, x_{n-1}] \\ 6f[x_{n-2}, x_{n-1}, x_n] \end{pmatrix},$$

where

$$\mu_k = \frac{x_k - x_{k-1}}{x_{k+1} - x_{k-1}}, \quad \lambda_k = \frac{x_{k+1} - x_k}{x_{k+1} - x_{k-1}}, \qquad k = 1, 2, \cdots, n-1. \tag{8}$$

For clamped boundary conditions, we suppose that $f'(x_0)$ and $f'(x_n)$ are given and

$$S_1'(x_0) = f'(x_0), \qquad S_n'(x_n) = f'(x_n).$$

By (6) with $k = 0$ and (5) with $k = n$, the above two equations become

$$\begin{aligned} f[x_0, x_1] - \frac{2M_0 + M_1}{6}(x_1 - x_0) &= f'(x_0), \\ f[x_{n-1}, x_n] + \frac{2M_n + M_{n-1}}{6}(x_n - x_{n-1}) &= f'(x_n). \end{aligned}$$

Define

$$\begin{aligned} f[x_0, x_0] &= f'(x_0), & f[x_n, x_n] &= f'(x_n), \\ f[x_0, x_0, x_1] &= \frac{f[x_0, x_1] - f[x_0, x_0]}{x_1 - x_0}, & f[x_{n-1}, x_n, x_n] &= \frac{f[x_n, x_n] - f[x_{n-1}, x_n]}{x_n - x_{n-1}}. \end{aligned}$$

We can rewrite the boundary conditions as

$$2M_0 + M_1 = 6f[x_0, x_0, x_1], \qquad M_{n-1} + 2M_n = 6f[x_{n-1}, x_n, x_n].$$

Thus the linear system for $M_0, M_1, \cdots, M_n$ is

$$\begin{pmatrix} 2 & \lambda_0 & & & & \\ \mu_1 & 2 & \lambda_1 & & & \\ & \mu_2 & 2 & \ddots & & \\ & & \ddots & \ddots & \lambda_{n-2} & \\ & & & \mu_{n-1} & 2 & \lambda_{n-1} \\ & & & & \mu_n & 2 \end{pmatrix} \begin{pmatrix} M_0 \\ M_1 \\ \vdots \\ M_{n-2} \\ M_{n-1} \\ M_n \end{pmatrix} = \begin{pmatrix} 6f[x_0, x_0, x_1] \\ 6f[x_0, x_1, x_2] \\ \vdots \\ 6f[x_{n-3}, x_{n-2}, x_{n-1}] \\ 6f[x_{n-2}, x_{n-1}, x_n] \\ 6f[x_{n-1}, x_n, x_n] \end{pmatrix}, \tag{9}$$

where $\lambda_0 = \mu_n = 1$, and other μ_k and λ_k are defined in (8).

Example 1. Consider the cubic spline interpolation for the Runge function with $n = 4$ and $x_k = -1 + k/2$ for $k = 0, 1, 2, 3, 4$. We first compute the divided differences:

x	0th divided differences	1st divided differences	2nd divided differences
-1	$f[-1] = 1/26$		
		$f[-1,-1/2] = 75/377$	
$-1/2$	$f[-1/2] = 4/29$		$f[-1,-1/2,0] = 575/377$
		$f[-1/2,0] = 50/29$	
0	$f[0] = 1$		$f[-1/2,0,1/2] = -100/29$
		$f[0,1/2] = -50/29$	
$1/2$	$f[1/2] = 4/29$		$f[0,1/2,1] = 575/377$
		$f[1/2,1] = -75/377$	
1	$f[1] = 1/26$		

Since $f'(-1) = 25/338$ and $f'(1) = -25/338$, we get

$$f[-1,-1,-1/2] = \frac{75/377 - 25/338}{1/2} = \frac{1225}{4901}, \quad f[1/2,1,1] = \frac{-25/338 - (-75/377)}{1/2} = \frac{1225}{4901}.$$

Therefore for clamped boundary conditions, the equations (9) are

$$\begin{pmatrix} 2 & 1 & & & \\ 1/2 & 2 & 1/2 & & \\ & 1/2 & 2 & 1/2 & \\ & & 1/2 & 2 & 1/2 \\ & & & 1 & 2 \end{pmatrix} \begin{pmatrix} M_0 \\ M_1 \\ M_2 \\ M_3 \\ M_4 \end{pmatrix} = \begin{pmatrix} 7350/4901 \\ 3450/377 \\ -600/29 \\ 3450/377 \\ 7350/4901 \end{pmatrix}.$$

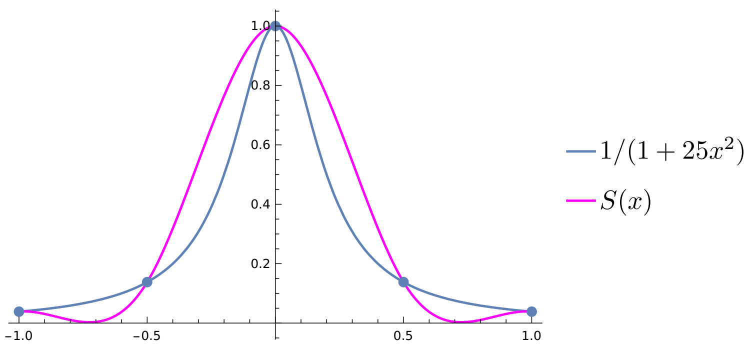
The solution is

$$M_0 = M_4 = -\frac{38225}{9802}, \qquad M_1 = M_3 = \frac{45575}{4901}, \qquad M_2 = -\frac{146975}{9802}.$$

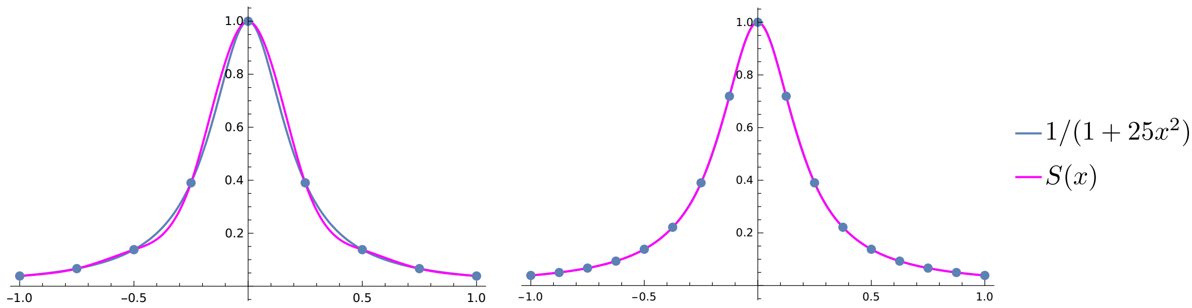
Inserting the above solution into (3), we obtain

$$\begin{aligned} S_0(x) &= \frac{50229}{19604} + \frac{91875}{9802}x + \frac{220525}{19604}x^2 + \frac{43125}{9802}x^3, \\ S_1(x) &= 1 - \frac{146975}{19604}x^2 - \frac{79375}{9802}x^3, \\ S_2(x) &= 1 - \frac{146975}{19604}x^2 + \frac{79375}{9802}x^3, \\ S_3(x) &= \frac{50229}{19604} - \frac{91875}{9802}x + \frac{220525}{19604}x^2 - \frac{43125}{9802}x^3. \end{aligned}$$

The graph of the function is plotted in the following figure:



When we increase the number of nodes, the approximation can be improved:



$S(x)$ for $n = 8$ $S(x)$ for $n = 16$

This shows when $n \to +\infty$, we can expect that $S(x)$ converges to $f(x)$. □

Exercise 1. For $n = 4$ and $x_k = -1 + k/2$, find the cubic spline interpolation of the Runge function using natural boundary conditions. □