Supplementary Notes 4

Tiago Salvador (tiago.saldanhasalvador@mail.mcgill.ca)

Abstract

We discuss the error in Lagrange interpolation, Chebyshev interpolation, Hermite interpolation.

Contents

1 Error in Lagrange interpolation

We discuss the error in Lagrange interpolation, the special case of interpolation on equally-spaced nodes and how to overcome the problems that arise.

Theorem 1.1. *Suppose $f \in C^{n+1}[a,b]$. Let L_n be the unique polynomial interpolation f at the nodes $\{x_j\}$ with $a = x_0 < x_1 < \ldots < x_n = b$. Then for each $x \in [a,b]$, there is $\xi \in [a,b]$ (depending on x) such that*

$$f(x) - L_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-x_0)(x-x_1)\ldots(x-x_n).$$

Moreover,

$$|f(x) - L_n(x)| \leq \frac{M_{n+1}}{(n+1)!} \max_{a\in[a,b]} |(x-x_0)\cdot(x-x_n)|\,,$$

where $M_{n+1} = \max_{a\in[a,b]} \left|f^{(n+1)}(x)\right|$. Moreover, if $h = \max_{i=0,\ldots,n} x_{i+1} - x_i$,

$$|f(x) - L_n(x)| \leq M_{n+1}\frac{h^{n+1}}{4(n+1)}.$$

Remark 1.2. *The above estimates do not imply that the error $\|f - L_n\|_\infty$ decreases to 0 when we interpolate with more and more points (increase n). Even though h may go to 0, the derivatives of f, i.e., $\left\|f^{(n)}\right\|_\infty$ can grow rapidly and therefore prevent convergence.*

Example 1.1 (Runge's example)**.** *Consider the function*

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

Runge found that if this function is interpolated at equidistant points x_i between -1 and 1 by a polynomial p of degree at most n, the resulting interpolation oscillates toward the end of the interval, i.e., close to -1 and 1 (see Figure 1). It can be proven that the interpolation error increases (without bound) when the degree of the polynomial is increased:

$$\lim_{n\to\infty} \max_{x\in[-1,1]} |f(x) - L_n(x)| = \infty.$$

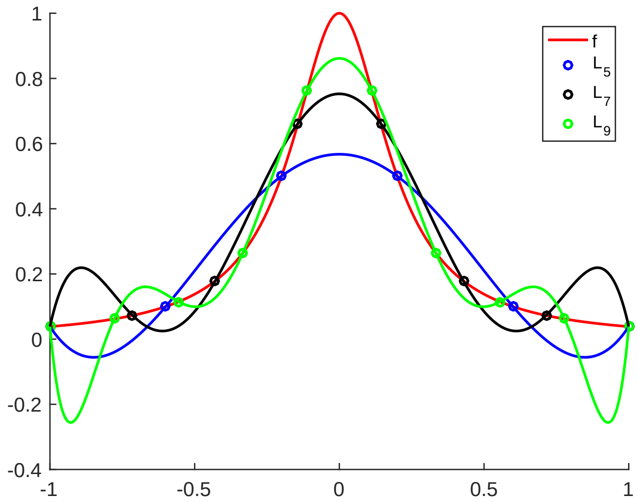

Figure 1: Plot of the Runge function and the Lagrange interpolation polynomials using equally spaced interpolating nodes for $n = 5, 7, 9$.

In fact, higher order polynomial interpolation on equally spaced nodes is notoriously bad, in particular close to the end points of the interval.

2 Chebyshev Interpolation

The idea behind Chebyshev interpolation is to find the "best" Lagrange approximation minimizing the $|f(x) - L(x)|$. As we saw in the previous section,

$$|f(x) - L_n(x)| \leq \frac{M_{n+1}}{(n+1)!} \max_{x\in[a,b]} |(x - x_0) \cdots (x - x_n)| \, .$$

We want to choose the "best" $\{x_k\}_{k=0}^n$, i.e, we want find the $\{x_k\}_{k=0}^n$ that minimize

$$\max_{x\in[a,b]} |(x - x_0) \cdots (x - x_n)| \, .$$

One can prove that for the interval $[-1, 1]$ the optimal interpolation points are the $(n + 1)$ roots of the $(n + 1)$-th Chebyshev polynomial, i.e.,

$$t_k = \cos\left(\frac{(2k+1)\pi}{2(n+1)}\right) \quad k = 0, \ldots, n.$$

We then get the optimal interpolation points for the interval $[a, b]$, up to a linear transformation to the interval $[a, b]$. They are given by applying the transformation

$$x = \frac{1}{2}(a + b) + \frac{1}{2}(b - a)t \in [a, b], \quad \text{for} \quad t \in [-1, 1]$$

that maps the interval $[-1, 1]$ to $[a, b]$. Hence the optimal interpolation nodes for the interval $[a, b]$ are

$$x_k = \frac{1}{2}(a + b) + \frac{1}{2}(b - a)\cos\left(\frac{(2k+1)\pi}{2(n+1)}\right) \quad k = 0, \ldots, n.$$

So Chebyshev interpolation is just Lagrange interpolation with the above nodes.

Remark 2.1. *In Chebyshev interpolation the points aren't equally distant. However, they are the projections of equally "spaced" angles. One of the disadvantages of the Chebyshev interpolation is that the interpolation nodes are determined a priori and we might not be able to obtain the function value there. One of the advantages is that the error always goes to* 0 *if the function is differentiable, as opposed to the Lagrange interpolation error that might diverge as* $n \to \infty$ *(see Figure 2).*

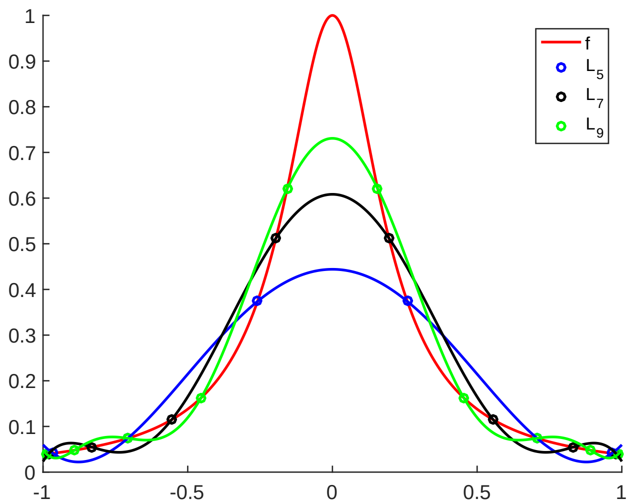

Figure 2: Plot of the Runge function and the 5^{th} and 9^{th} order interpolating polynomials using the Chebyshev interpolating nodes.

3 Hermite Interpolation

The idea of Hermite interpolation is to find a polynomial that not only matches the function value at the nodes, but also the first derivative.

Hermite interpolation problem: Given a function $f \in C^1[a, b]$ and $n+1$ distinct points satisfying $a = x_0 < x_1 < \ldots < x_n = b$, find an interpolation polynomial H such that

$$\begin{aligned} f(x_j) &= H(x_j) \quad j = 0, \ldots, n+1, \\ f'(x_j) &= H'(x_j) \quad j = 0, \ldots, n+1. \end{aligned}$$

We have $2n + 2$ conditions that need to be satisfied and therefore we look for a polynomial of degree $2n + 1$. The next theorem tells us that such a polynomial always exists and is unique.

Theorem 3.1 (existence and uniqueness)**.** *For any set of* $n+1$ *distinct nodes* $x_0 < x_1 < \ldots < x_n$*, there exists a unique polynomial* H_{2n+1} *of degree at most* $2n+1$ *that solves the Hermite interpolation problem.*

Theorem 3.2 (interpolation error)**.** *Let* H_{2n+1} *be the Hermite interpolation polynomial and let* $f \in C^{2n+2}[a, b]$*. Then for some* $\xi \in [a, b]$ *(depending on* x*) we have*

$$f(x) - H_{2n+1}(x) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!}(x - x_0)^2 \cdots (x - x_n)^2.$$

Despite the use of information derivative, Runge's phenomenon still occurs in Hermite interpolation and it can be even worse (see Figure 3).

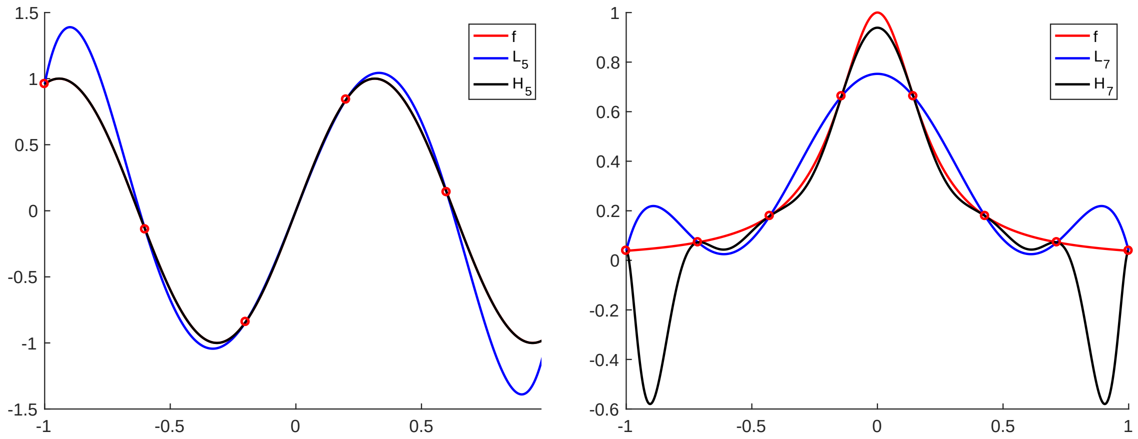

Figure 3: Comparison between Hermite and Lagrange interpolation on equally space nodes: $f(x) = \sin(x)$ (left); Runge function (right).

Exercise 3.1. *Given the function* $f(x) = \frac{\sin(\pi x)}{\pi}$ *compute the Hermite interpolation polynomial with nodes* $x_0 = 0$ *and* $x_1 = 1$.

Solution: Let H denote the Hermite interpolation polynomial. Then H is a polynomial of degree at most 3 that satisfies

$$\begin{cases} H(0) = f(0) \\ H(1) = f(1) \\ H'(0) = f'(0) \\ H'(1) = f'(1) \end{cases} \tag{1}$$

Since H is a polynomial of degree at most 3, we can write it as

$$H(x) = a + bx + cx^2 + dx^3,$$

where $a, b, c, d \in \mathbb{R}$ are unknowns. Since $H'(x) = b + 2cx + 3dx^2$, we can rewrite and solve the (linear) system (1):

$$\begin{cases} a = 0 \\ a + b + c + d = 0 \\ b = 1 \\ b + 2c + 3d = -1 \end{cases} \iff \begin{cases} a = 0 \\ b = 1 \\ c + d = -1 \\ 2c + 3d = -2 \end{cases} \iff \begin{cases} a = 0 \\ b = 1 \\ c = -1 \\ d = 0 \end{cases}$$

Hence $H(x) = x - x^2$.

References

[1] R. L. Burden and J. D. Faires. *Numerical Analysis.* 9^{th} edition. Brookes/Cole, 2004.

[2] A. Quarteroni, R. Sacco and F. Saleri. *Numerical Mathematics.* 2^{nd} edition. Springer, 2006.

[3] Tutorial notes written by Jan Feys