Numerical Analysis
Semester 4

Contents

Chapter 1 **Interpolation** **Page 2**

1.1 Linear Interpolation 2

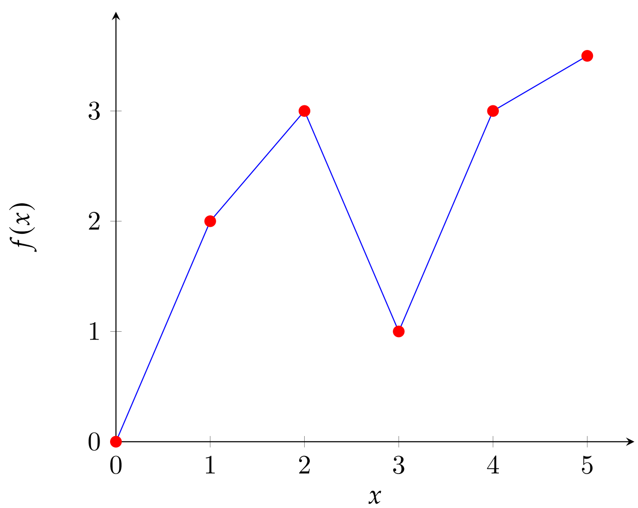
1.2 Polynomial Interpolation 2

Lagrange Polynomials — 2 • Newton Polynomial — 4 • Error due to polynomial interpolation — 6 • Hermite Interpolation — 6

Chapter 1

Interpolation

1.1 Linear Interpolation



x	$f(x)$
0	0
1	2
2	3
3	1
4	3
5	3.5

Linear interpolation is just drawing lines between the data points.

Definition 1.1.1: Linear Interpolation(lerp) equation

The equation of the lines between data points is

$$y = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}(x - x_i) + y_i.$$

Theorem 1.1.1 Error due to linear interpolation

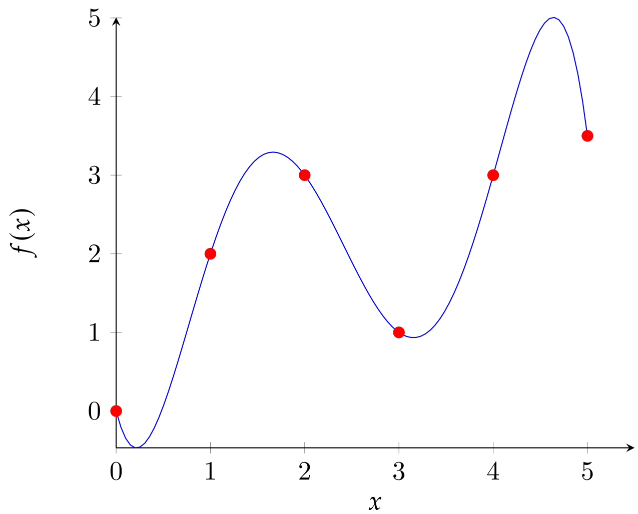
Let f be a continuous and differentiable on $[a, b]$. We define the error $z(x)$ to be

$$|z(x)| \leqslant \frac{(b-a)^2}{8} \sup_{a \leqslant x \leqslant b} |f''(x)|.$$

1.2 Polynomial Interpolation

1.2.1 Lagrange Polynomials

Really nice video here explaining Lagrange polynomials.



Theorem 1.2.1 Lagrange polynomial equation

Consider a set of n points $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$. The Lagrange polynomial for this set of data is

$$L(x) = \sum_{k=0}^{n} y_k \ell_k(x).$$

where

$$\ell_k(x) = \prod_{\substack{i=1\\ i\neq k}}^{n} \frac{x - x_i}{x_k - x_i}.$$

Case of equidistant points

If the set of x_i are equidistant from each other with a distance of $h = x_{i+1} - x_i$, then we can represent any point as $x_k = x_0 + kh$ where $k \in \mathbb{N}$ and any number $x = x_0 + sh$ where $s \in \mathbb{R}$. We can rewrite the formula as

$$Q(s) = \sum_{k=0}^{n} \ell_k(s) f(x_k).$$

where

$$\ell_k(s) = \prod_{\substack{j=0\\ k\neq k}}^{n} \frac{s - j}{k - j}.$$

by substitution

$$s = \frac{x - x_0}{h}.$$

Existence

Proof: $P(x)$ belongs to the vectorial space of polynomial of degree of, at most, n. Now, we must fins a basis for this vectorial space. Find the polynomial ℓ_k of degree $\leqslant n$ such that

$$\ell_k(x_i) = \delta_{ki} = \begin{cases} 1 & \text{if } i = k \\ 0 & \text{if } i \neq k \end{cases}.$$

Then, $\ell_k(x) = \lambda(x - x_0)\dots(x - x_{k-1})(x - x_{k+1})\dots(x - x_n)$ where

$$\lambda = \frac{1}{(x_k - x_0)\dots(x_k - x_{k-1})(x_k - x_{k+1})\dots(x_k - x_n)}.$$

The $(n+1)$ polynomials $\ell_k(x)$ for a system of generators in the vectorial space of polynomials of degree at most n.

$$\lambda_0\ell_0(x) + \lambda_1\ell_1(x) + \cdots + \lambda_k\ell_k(x) + \cdots + \lambda_n\ell_n(x) = 0.$$

for $x = x_k$

$$\begin{aligned} \lambda_0\ell_0(x_k) + \lambda_1\ell_1(x_k) + \cdots + \lambda_k\ell_k(x_k) + \cdots + \lambda_n\ell_n(x_k) &= 0 \\ 0 + 0 + \cdots + \lambda_k 1 + \cdots + 0 = 0\lambda_k &= 0. \end{aligned}$$

$\therefore$ the set of ℓ_k for a basis in the vector space $\Rightarrow$ there has to exist a polynomial passing through the given set of points.

☺

Uniqueness

Proof: Let P and Q be 2 Lagrange polynomials of degrees $\leqslant n \Big/ P(x_i) = Q(x_i) = f(x_i) \quad \forall i = 0, 1, \ldots, n$.
Let

$$\left.\begin{aligned} &R = P - Q \text{ of degree } \leqslant n \\ &R = 0\ (n+1) \text{ times} \end{aligned}\right\} R \equiv 0 \Rightarrow P = Q\ \forall x.$$

☺

1.2.2 Newton Polynomial

Definition 1.2.1: Newton Polynomial equation

Consider a set of n points $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$. The Newton polynomial for this set of data is

$$p_n(x) = \underbrace{a_0}_{A_0} + \underbrace{a_1(x - x_0)}_{A_1} + \underbrace{a_2(x - x_0)(x - x_1)}_{A_2} + \cdots + \underbrace{a_n \prod_{i=0}^{n-1}(x - x_i)}_{A_n}.$$

where

$$a_i = f[x_0, x_1, \ldots, x_i].$$

Here $f[\ldots]$ is the divided difference of the inputted data.

Definition 1.2.2: Backwards formula

$$P_n(x) = f_n + A_1 + A_2 + \cdots + A_n.$$

where

$$A_i = f[x_n, x_{n-1}, \ldots, x_{n-i}] \prod_{j=n-i+1}^{n} (x - x_j).$$

The divided difference has 2 formulas, the recurrence formula

$$f[x_0, x_1, \ldots, x_{n+1}] = \frac{f[x_1, x_2, \ldots, x_{n+1}] - f[x_0, x_1, \ldots, x_n]}{x_{n+1} - x_0}.$$

and a general formula

$$f[x_0, x_1, \ldots, x_n] = \sum_{i=1}^{n} \frac{y_i}{\prod_{\substack{k=0 \\ k \neq i}}^{n}(x_i - x_k)}.$$

Now forget you ever saw those cause there is an easier method to finding the divided difference.

Divided Difference Table

$$
\begin{array}{cccccc}
x_0 & y_0 & & & & \\
 & & \frac{y_1-y_0}{x_1-x_0} = f[x_0,x_1] & & & \\
x_1 & y_1 & & \frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0} & & \\
 & & \frac{y_2-y_1}{x_2-x_1} = f[x_1,x_2] & & \dots & \\
x_2 & y_2 & & \frac{f[x_2,x_3]-f[x_1,x_2]}{x_3-x_1} & & \dots \\
 & & \frac{y_3-y_2}{x_3-x_2} = f[x_2,x_3] & & \dots & \\
x_3 & y_3 & & \frac{f[x_3,x_4]-f[x_2,x_3]}{x_4-x_2} & & \\
 & & \frac{y_4-y_3}{x_4-x_3} = f[x_3,x_4] & & & \\
x_4 & y_4 & & & &
\end{array}
$$

After we have constructed the table we can find the divided difference we want by looking at the top diagonal

$$
\begin{array}{cccccc}
x_0 & y_0 & & & & \\
 & & f[x_0,x_1] & & & \\
x_1 & y_1 & & f[x_0,x_1,x_2] & & \\
 & & \dots & & f[x_0,x_1,x_2,x_3] & \\
x_2 & y_2 & & \dots & & f[x_0,x_1,x_2,x_3,x_4] \\
 & & \dots & & \dots & \\
x_3 & y_3 & & \dots & & \\
 & & \dots & & & \\
x_4 & y_4 & & & &
\end{array}
$$

Case of equidistant points

Bla bla bla the formula becomes

$$P(t) = a_0 + a_1(t-0) + a_2(t-0)(t-1) + \cdots + a_n \prod_{i=0}^{n-1}(t-i).$$

where in this case

$$a_k = \frac{\nabla^k[y](x_k)}{k!}.$$

and

$$x = x_0 + th.$$

where $\nabla^k[y]$ is the discrete difference.

$$\nabla[y](x_i) = y(x_i+h) - y(x_i).$$

and the backwards formula is

$$P(t) = f_n + A_1 + A_2 + \cdots + A_n.$$

where

$$A_i = \frac{\bar{\nabla}^i f_n}{i!} \prod_{j=n-i+1}^{n} (t-j).$$

Definition 1.2.3: Discrete Difference

Forward discrete difference:

$$\begin{aligned}
\nabla[y](x_i) &= y(x_i+h) - y(x_i)\\
\nabla^2[y](x_i) &= \nabla[y](x_i+h) - \nabla[y](x_i)\\
&= y(x_i+2h) - 2y(x_i+h) + y(x_i)\\
\nabla^k[y](x_i) &= \nabla\left(\nabla^{k-1}[y](x_i)\right)
\end{aligned}$$

Backwards discrete difference:

$$\begin{aligned}
\bar{\nabla}[y](x_i) &= y(x_i) - y(x_i-h)\\
\bar{\nabla}^k[y](x_i) &= \bar{\nabla}\left(\bar{\nabla}^{k-1}[y](x_i)\right)
\end{aligned}$$

$$\begin{array}{cccccc}
x_0 & y_0 & & & & \\
 & & \nabla[y](x_i) & & & \\
x_1 & y_1 & & \nabla^2[y](x_i) & & \\
 & & \dots & & \nabla^3[y](x_i) & \\
x_2 & y_2 & & \dots & & \nabla^4[y](x_i)\\
 & & \dots & & \dots & \\
x_3 & y_3 & & \dots & & \\
 & & \dots & & & \\
x_4 & y_4 & & & &
\end{array}$$

1.2.3 Error due to polynomial interpolation

Let $f(x)$ be of class $C^{n+1} \quad \forall x \in [a,b]$ and let the polynomial $P(x)$ interpolate it.

The error function is bounded by

$$|\text{Error}| = |f(x) - P(x)| \leqslant \frac{\left|\prod_{i=0}^{n}(x-x_i)\right|}{(n+1)!} \sup_{x\in[a,b]} |f^{(n+1)}(x)|.$$

1.2.4 Hermite Interpolation

Definition 1.2.4: Hermite interpolation formula

Consider $(n+1)$ sets of point (x_i, y_i, y_i') representing $f(x)$ ($y_i = f(x_i)$ and $y_i' = f'(x_i)$), the hermite polynomial $P(x)$ interpolates $f(x)$ such that $P'(x) = f'(x)$.

$$P(x) = \sum_{i=0}^{n} h_i(x) y_i + \sum_{i=0}^{n} k_i(x) y_i'.$$

where

$$\begin{aligned}
h_i(x) &= \left(1 - 2(x-x_i)\ell_i'(x_i)\right)\ell_i^2(x)\\
k_i(x) &= (x-x_i)\ell_i^2(x)\\
\ell_i(x) &= \prod_{\substack{j=0\\ j\neq i}}^{n} \frac{x-x_j}{x_i-x_j}
\end{aligned}$$

Theorem 1.2.2 Error due to Hermite interpolation

$$|\mathrm{Error}| = |f(x) - P(x)| \leqslant \frac{\left|\prod_{i=0}^{n}(x - x_i)^2\right|}{(2n+2)!} \sup_{x\in[a,b]} |f^{(2n+2)}(x)|.$$