Newton-divided differences + barycentric interpolation
Wednesday, October 11, 2023 1:52 PM

We will 3 ways of evaluating the intrpolation polynomial:

- Lagrange
- Newton - Divided differences
- Barycentric interpolation ← rewrite of Lagrange to be more stable.

<u>Thm</u> Let $f \in C^{(n+1)}[a,b]$ with the interpolation nodes $\{x_j\}_{j=0}^{n} \in [a,b]$. Then the polynomial through the data is unique.

<u>Proof</u>: Proof by contradiction: suppose $p(x)$ & $q(x)$ are interpolating polynomials of degree n & $p(x) \neq q(x)$

By def: $p(x_j) = f(x_j)$ $\quad q(x_j) = f(x_j)$
for $j = 0, \cdots, n$

Define $w(x) = p(x) - q(x)$

What is the degree of w? $\leq n$

$w(x_j) = p(x_j) - q(x_j) = 0$ for $j = 0, \cdots, n$

How many roots does $w(x)$ have? $\underline{\underline{n+1}}$

This is impossible for a polynomial of deg > 1. $\Rightarrow$ $\deg(w) = 0$.

$\Rightarrow$ $w(x) = 0$
$\quad p(x) - q(x) = 0$
$\Rightarrow$ $p(x) = q(x)$ ← This is our contradiction.

$\therefore$ Interpolation is unique.//  = fireworks

How much does it cost to do Lagrange interpolation?

Given $\{(x_j, f(x_j)\}_{j=0}^{n}$

our Lagrange polynomial is

$$p(x) = \sum_{j=0}^{n} f(x_j)\, \ell_j(x)$$

BAD: - Cost $O(n^2)$ to evaluate a 1 location

Good = Stable.

Newton - Divided differences: Good: Cost $O(n)$

Bad: not very stable. :(

Idea: Write our polynomial as follows

$$p(x) = a_0 + a_1(x - x_0) + a_2(x-x_0)(x-x_1) + \cdots + a_n(x-x_0)\cdots(x-x_{n-1})$$

The coefficients are unknowns that we have to find.

let's try. plug in the nodes but in a "smart"

$$p(x_0) = a_0 = f(x_0)$$

$$p(x_1) = a_0 + a_1(x_1 - x_0) = f(x_1)$$

$$\Rightarrow a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} := f[x_0, x_1]$$

$$p(x_2) = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) = f(x_2)$$

Yuck!

$$a_2 = \frac{f(x_2) - f[x_0, x_1](x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} := f[x_0, x_1, x_2]$$

We can continue but instead we build a table which makes it easier to evaluate coefficients.

evaluate coefficients.

$$f[x_j, \dots, x_k] := \frac{f[x_{j+1}, \dots x_k] - f[x_j, \dots x_{k-1}]}{x_k - x_j}$$

We need an easier way of evaluating the coefficients.

Fortunately we can build them via a table

The Table

$$\begin{array}{llllll}
x_j & f(x_j) = \boxed{a_0} = \text{constant} & \boxed{a_1} \text{ linear} & & \boxed{a_2} = \text{quadratic} & \boxed{a_3} = \text{cubic} \\
x_0 & f(x_0) & f[x_0, x_1] := \dfrac{f(x_1) - f(x_0)}{x_1 - x_0} & & & \\
x_1 & f(x_1) & & f[x_0, x_1, x_2] := \dfrac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} & & \\
x_2 & f(x_2) & f[x_1, x_2] & & f[x_0, x_1, x_2, x_3] = \dfrac{f[x_1, x_2, x_3] - f[x_0, x_1, x_2]}{x_3 - x_0} & \\
x_3 & f(x_3) & f[x_2, x_3] & f[x_1, x_2, x_3] & &
\end{array}$$

$\underbrace{x_j \quad f(x_j)}_{\text{interpolation data}}$

Our last interpolation scheme is called Barycentric interpolation.

A rewrite of Lagrange so that it has linear cost to evaluate.

For $x \neq x_j$, the Lagrange polynomial is

$$l_j(x) = \frac{(x - x_0) \cdots (x - x_{j-1})(x - x_{j+1}) \cdots (x - x_n)}{(x_j - x_0) \cdots (x_j - x_{j-1})(x_j - x_{j+1}) \cdots (x_j - x_n)}$$

This is missing the $x - x_j$ in the numerator compared w/ the other polynomials $l_i(x)$

Lets multiply by 1

$$\ell_j(x) = \frac{(x-x_0)\cdots(x-x_{j-1})\cancel{(x-x_j)}(x-x_{j+1})\cdots(x-x_n)}{(x_j-x_0)\cdots(x_j-x_{j-1})\cancel{(x-x_j)}(x_j-x_{j+1})\cdots(x_j-x_n)}$$

$$= \frac{\ell(x)\, w_j}{(x-x_j)}$$

where $\ell(x) = \prod_{k=0}^{n} (x-x_k)$

& $$w_j = \frac{1}{(x_j-x_0)\cdots(x_j-x_{j-1})(x_j-x_{j+1})\cdots(x_j-x_n)}$$

$$= \boxed{\frac{1}{\ell'(x_j)}}$$ Remember this is here!

Now we can use this to rewrite our interpolating polynomial.

$$p(x) = \sum_{j=0}^{n} \ell_j(x) f(x_j) = \sum_{j=0}^{n} \frac{\ell(x)\, w_j\, f(x_j)}{x-x_j}$$

$$= \ell(x) \sum_{j=0}^{n} \frac{w_j\, f(x_j)}{x-x_j}$$

Cost: creating w_j for all j is $O(n^2)$

Cost of evaluating $p(x) = O(n)$