Lagrange Interpolation

Tuesday, September 15, 2020 5:22 PM

One practical method for polynomial interpolation (and can be improved via Barycentric Lagrange interpolation)

Setup

⚠ we have $n+1$ nodes since starting at 0

Given $n+1$ distinct nodes $\{x_0, x_1, \dots, x_n\}$ with values $\{y_0, y_1, \dots, y_n\}$,

we want to find the unique n-degree polynomial that interpolates these values.

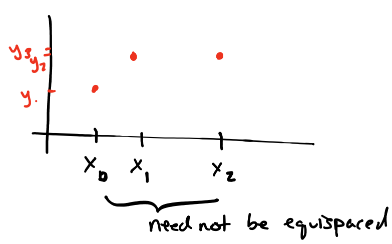
we proved previously

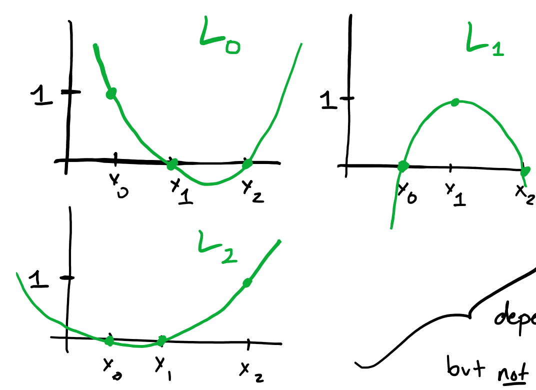
(Note: we could find an interpolating polynomial of degree $> n$, but then ① it's not unique, ② more likely to overfit)

One solution: Lagrange polynomials

it's not "the" Lagrange polynomials, since they will change based on the nodes

Idea: (for illustration, let $n=2$, so looking for a quadratic polynomial to interpolate 3 points)





depends on $\{x_i\}$ but not on $\{y_i\}$

$$L_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}$$

Quadratic

$L_0(x_0) = 1$

$L_0(x_1) = 0$

$L_0(x_2) = 0$

similarly for L_1, L_2

Now, putting it all together is easy:

$$p(x) = y_0 \cdot L_0(x) + y_1 \cdot L_1(x) + y_2 \cdot L_2(x)$$

so $p(x_0) = \underbrace{y_0 \cdot L_0(x_0)}_{=1} + \underbrace{y_1 \cdot L_1(x_0)}_{=0} + \underbrace{y_2 \cdot L_2(x_0)}_{=0} = y_0$ ✓

and similarly for x_1, x_2

General Form: (Sometimes we write $L_{n,k}$ to make it clear the dependence on n)

(*) $$L_{n,k}(x) = \frac{(x-x_0)(x-x_1)\cdot\ldots\cdot(x-x_{k-1})(x-x_{k+1})\cdot\ldots\cdot(x-x_n)}{(x_k-x_0)(x_k-x_1)\cdot\ldots\cdot(x_k-x_{k-1})(x_k-x_{k+1})\cdot\ldots\cdot(x_k-x_n)}$$

Lagrange Polynomial

$$= \prod_{\substack{i=0\\ i\neq k}}^{n} \frac{(x-x_i)}{(x_k-x_i)} \qquad \left(\text{So } L_{n,i}(x_k) = \begin{cases} 1 & k=i \\ 0 & k\neq i \end{cases} \right)$$

and the interpolating polynomial is

$$p(x) = \sum_{k=0}^{n} y_k \cdot L_{n,k}(x)$$

Lagrange form of the interpolating polynomial

Comments:

(+) The slowest part of the algorithm depends only on the nodes $\{x_i\}$ not the function values $\{y_i\}$. Sometimes we need to repeatedly interpolate different sets of $\{y_i\}$ but with the same nodes, so we can amortize the slow part.

(−) Evaluating $p(x)$ takes $O(n^2)$ flops

(−) Adding a new data point (x_{n+1}, y_{n+1}) requires you to start from scratch

ex: we keep adding points until the error in approximation is small

(−) Computation can be unstable

Fix is the

Barycentric version of the Lagrange formula (cf. Berrut and Trefethen, SIAM Review 2004)

Define $\ell(x) = (x-x_0)(x-x_2)(x-x_3)\ldots(x-x_n)$ (no missing terms)

$$\text{So } L_{n,k}(x) = \prod_{\substack{i=0\\ i\neq k}}^{n} \frac{(x-x_i)}{(x_k-x_i)} = \frac{\prod_{i\neq k}(x-x_i)}{\prod_{i\neq k}(x_k-x_i)} = \frac{\ell(x)/(x-x_k)}{\underbrace{\prod_{i\neq k}(x_k-x_i)}_{=w_k^{-1}}}$$

BARYCENTRIC

$$\boxed{\text{So } p(x) = \sum_{k=0}^{n} y_k L_{n,k}(x) = \ell(x)\cdot\sum_{k=0}^{n}\frac{w_k}{(x-x_k)}\, y_k} = \ell(x)\cdot\frac{w_k}{(x-x_k)}.$$

Precompute w_k in $O(n^2)$ flops* (each w_k takes $O(n)$ flops, and do this for $O(n)$ values of k)

*See code on github

Amortize: Spread out the cost over a long period of time (like buying a season pass... it makes sense if you use it a lot)

but now we can evaluate $p(x)$ at many different x, each one costing only $O(n)$. So we've amortized the $O(n^2)$ cost

$\underbrace{\text{costing only } O(n)}$: evaluate $\underset{O(n)}{\ell(x)}$ once, then $O(n)$ multiplies and sums

Furthermore, we have fast updates:

to add (x_{n+1}, y_{n+1}), update old w_k $(k=0,\dots,n)$ by $\div$ by $(x_k - x_{n+1})$

and compute new w_{n+1}

which is $O(n)$. Nice.

Note: an equivalent way to write the Barycentric Formula is ($p(x)$ is continuous - use L'Hôpital to see)

BARYCENTRIC FORMULA

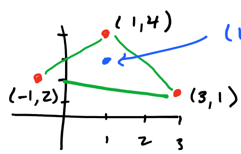
$$p(x) = \begin{cases} \dfrac{\sum_{k=0}^{n} \dfrac{w_k}{x - x_k} \cdot y_k}{\sum_{k=0}^{n} \dfrac{w_k}{x - x_k}} & \text{if } x \neq x_k \text{ for } k \in \{0,1,\dots,n\} \\ y_k & \text{if } x = x_k \end{cases}$$

↖ you might be worried that if x is close to some node x_k we get subtractive cancellation, but it turns out in this case it's OK since it happens in numerator and denominator

(Aside: what does Barycenter mean?)

The Barycenter of two bodies (planets, stars...) is their center of mass, so it's a weighted sum

Barycentric coordinates are coordinates with respect to the vertices of a simplex, ex.



(1,3) in Cartesian coordinates

or $(1,3) = 0 \cdot (-1,2) + 8/11\,(1,4) + 1/11\,(3,1)$

so $(0, 8/11, 1/11)$ in Barycentric coord.

Special Cases*

*Chebyshev nodes also simplify, see Berrut + Trefethen

If nodes are $\boxed{\text{equispaced}}$, $\longrightarrow$ $x_{k+1} - x_k = h$

then formulas simplify and are faster,

i.e., $\underbrace{(x_k - x_0)}_{(k)\cdot h} \cdot \underbrace{(x_k - x_1)}_{(k-1)\cdot h} \cdots \underbrace{(x_k - x_{k-1})}_{h} \cdot \underbrace{(x_k - x_{k+1})}_{-h} \cdots \underbrace{(x_k - x_n)}_{-(n-k)\cdot h}$

$$= (-1)^{n-k} \cdot h^n \cdot k! \cdot (n-k)!$$

$$= (-1)^{n-k}\, h^n \left(\binom{n}{k} \Big/ n! \right)^{-1}$$

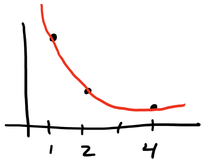
} Please do not memorize! Not that important

Example

Take $x_0 = 1$, $y_0 = 1$ (eg, $y = 1/x$)

$x_1 = 2$, $y_1 = 1/2$

$x_2 = 4$, $y_2 = 1/4$. Find quadratic interpolant p and evaluate $p(3)$



Solution: ① One way, compute $L_0(x) = (x-x_1)(x-x_2)\cdot\left(\overbrace{\frac{1}{(x_0-x_1)(x_0-x_2)}}^{w_0}\right)$

$$= (x-2)(x-4)\left(\underbrace{\frac{1}{(1-2)(1-4)}}\right) \qquad = \frac{1}{(-1)(-3)} = \frac{1}{3} = w_0$$

$$L_1(x) = (x-x_0)(x-x_2)\cdot\left(\underbrace{\frac{1}{(x_1-x_0)(x_1-x_2)}}_{w_1 = \frac{1}{(2-1)(2-4)} = \frac{-1}{2}}\right)$$

$$L_2(x) = (x-x_0)(x-x_1)\left(\underbrace{\frac{1}{(x_2-x_0)(x_2-x_1)}}_{w_2 = \frac{1}{(4-1)(4-2)} = \frac{1}{6}}\right)$$

then

$$p(x) = 1\cdot L_0(x) + \tfrac{1}{2}\cdot L_1(x) + \tfrac{1}{4} L_2(x)$$

and plug $x=3$ in (I won't do this now since I'll do it the 2nd way)

② Second way, Barycentric formula. Uses $w_0 = 1/3$, $w_1 = -1/2$, $w_2 = 1/6$ as before

$$p(x) = \frac{\sum_{k=0}^{n} \frac{w_k}{x-x_k}\cdot y_k}{\sum_{k=0}^{n} \frac{w_k}{x-x_k}} = \frac{\frac{1}{3}\cdot\frac{1}{(3-1)}\cdot 1 + \frac{-1}{2}\frac{1}{(3-2)}\frac{1}{2} + \frac{1}{6}\cdot\frac{1}{(3-4)}\frac{1}{4}}{\frac{1}{3}\cdot\frac{1}{(3-1)} + \frac{-1}{2}\frac{1}{(3-2)} + \frac{1}{6}\cdot\frac{1}{(3-4)}}$$

$(x=3)$

$$= \frac{1/6\cdot 1 - 1/2\cdot 1/2 - 1/6\cdot 1/4}{1/6 - 1/2 - 1/6} = \frac{\frac{1}{6} - \frac{1}{4} - \frac{1}{24}}{-\frac{1}{2}}$$

$$= \frac{\frac{4-6-1}{24}}{-\frac{1}{2}} = \frac{+3}{24}\cdot 2 = \boxed{\frac{1}{4}}$$

Error

The $\{y_i\}$ can be arbitrary, but sometimes they are generated by a true "underlying" function f. In this case, how close is our interpolant p to f? Should p be a better approximation as we add nodes? (Yes!) Answered by following theorem:

Thm 3.3 (Burden and Faires)

Let $\{x_0, x_1, \dots, x_n\}$ be distinct points in $[a,b]$ and $f \in C^{n+1}([a,b])$.

Then $\forall x \in [a,b]$, $\exists\ \xi$ (unknown, depends on x) between $x_0, x_1, \dots x_n$

(hence $\xi \in (a,b)$) such that if p is the n^{th} degree polynomial interpolant w/ nodes $\{x_i\}_{i=0}^n$ and values $\{y_i = f(x_i)\}_{i=1}^n$, then

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

(i.e., error at $x = x_k$ is 0, and small if x is close to some node)
Similar to Taylor Series except $(x-x_0)\cdot\ldots\cdot(x-x_n)$ instead of $(x-x_0)^{n+1}$.

Proof Assume $x \neq x_k$ for any k, else it's trivial

Fix x, define $g(t) = f(t) - p(t) - (f(x) - p(x))\cdot\prod_{i=0}^{n} \frac{(t-x_i)}{(x-x_i)}$

then note $f \in C^{n+1}([a,b]) \Rightarrow g \in C^{n+1}([a,b])$

and $g(x_k) = 0$ for $k = 0, 1, \ldots, n$

$g(x) = 0$ also.

like MVT

So g has $(n+2)$ distinct zeros in $[a,b]$, so by generalized Rolle's Thm

$\exists\, \xi \in (a,b)$ s.t. $g^{(n+1)}(\xi) = 0$

Compute $g^{(n+1)}(\xi) = f^{(n+1)}(\xi) + p^{(n+1)}(\xi) - \overbrace{(f(x)-p(x))}^{\text{constants}} \frac{d^{n+1}}{dt^{n+1}} \prod \frac{(t-x_i)}{(x-x_i)}\Big|_{\xi}$

degree n polynomial so $\frac{d^{n+1}}{dt^{n+1}}(\ ) = 0$

degree $n+1$

so only leading term remains

$$= f^{(n+1)}(\xi) - (f(x) - p(x))\cdot\frac{(n+1)!}{\prod_{i=0}(x-x_i)} \qquad \square$$

Quiz

Let $x_0 = 2$, $x_1 = 6$

$y_0 = \frac{1}{2}$, $y_1 = \frac{1}{6}$. Compute $p(4)$ where p is the linear (degree $n=1$) interpolating polynomial.

Note: your answer should be an integer or fraction (simplify fractions), not a decimal

9