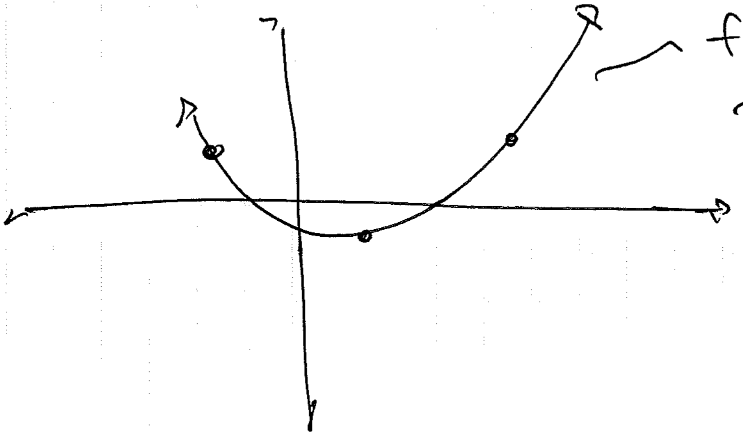
3.1 Data & Interpolating Functions

In general, given a data set: $\{(x_1,y_1),\dots,(x_n,y_n)\}$, we would like to find a function $f(x)$ such That $f(x_i)=y_i$ for all $1\le i\le n$.



$f(x)$ "interpolates" The data: $\{(x_1,y_1),(x_2,y_2)$ & $(x_3,y_3)\}$.

Definition The function $y=P(x)$ interpolates The data: $(x_1,y_1),\dots,(x_n,y_n)$ if $P(x_i)=y_i$ $\forall$ $1\le i\le n$.

∘) Note That P is required to be a function (i.e each input gets a single output). In general, because of its various "desirable" mathematical properties (i.e. low "complexity"), we generally aim to find $P(x)$ where $P(x)$ is a polynomial function.

∘) Polynomial functions are The most fundamental functions for digital computers.

∘) We can Think of function interpolation as The reverse/inverse of function evaluation. In addition, function interpolation (using polynomials) is an instance of data/compression.

Def. Given n data points the **Lagrange Interpolating Polynomial** of degree $\boxed{n-1}$ is defined:

$$\boxed{P_{n-1}(x) = y_1 L_1(x) + \cdots + y_n L_n(x)}$$

Note: no $(x-x_k)$ factor in numerator.

where:
$$\boxed{L_k(x) = \frac{(x-x_1)\cdots(x-x_{\underline{k-1}})(x-x_{\underline{k+1}})\cdots(x-x_n)}{(x_k-x_1)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}}$$

$\forall\ 1 \le k \le n$.

Ex. Given $\underline{3}$ data points: $(x_1,y_1), (x_2,y_2), (x_3,y_3)$ we have:

$$\cancel{L(x)} \Rightarrow P_{\underline{2}}(x) = y_1 \underbrace{\frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)}}_{L_1(x)} + y_2 \underbrace{\frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)}}_{L_2(x)} + y_3 \underbrace{\frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}}_{L_3(x)}$$

Claim: $P_2(x)$ interpolates the given data.

Check $P_2(x_1) = y_1 \cdot 1 + 0 + 0 = y_1$

$P_2(x_2) = 0 + y_2 \cdot 1 + 0 = y_2$ and $P_2(x_3) = y_3$. ✓

In general then, $P_{n-1}(x) = y_1 L_1(x) + \cdots + y_n L_n(x)$

$\ni$ $L_k(x_k) = 1$, else $L_k(x_j) = 0$ for $k \neq j$.

So the Lagrange Polynomial of degree n-1 interpolates n given data points, in general.

Q: How many interpolating polynomials of degree n-1 exist for a data set of n points (no repeated x_i's).

A: We seek to find coefficients:

$$P(x) = + \underline{a_{n-1}}x^{n-1} + \underline{a_{n-2}}x^{n-2} + \dots + \underline{a_1}x + \underline{a_0}$$ ~~Fact 2 inte~~

so that P(x) interpolates the data.

This problem gives rise to $\boxed{\text{n equations w/ n unknowns}}$.

which means that any such solution is unique. Thus for n points, an interpolating polynomial of degree n-1 is unique!

Hence, the Lagrange interpolating polynomial is the $\boxed{\text{unique}}$ n-1 degree polynomial that interpolates n data points.

(More formally)

Thm: Fund. Theorem of Polynomial Interpolation

Let $(x_1, y_1), \dots, (x_n, y_n)$ be n points in the plane w/ distinct x_i. Then there exists one & only one polynomial P of degree n-1 (or less) that satisfies $P(x_i) = y_i \ \forall \ 1 \le i \le n$.

Pf: (Existence is proved by the explicit Lagrange interpolation formulas given above).

Uniqueness Proof: (By Contradiction) Suppose not, & say both P(x) & Q(x) have degree n-1 & both interpolate the data.

So, $P(x_1) = Q(x_1) = y_1$, $P(x_2) = Q(x_2) = y_2$, etc.

Define a new polynomial, $H(x) = P(x) - Q(x)$.

Then $\deg(H(x)) \leq n-1$, clearly (why?). Also: $H(x_1) = P(x_1) - Q(x_1) = 0$, & in general, $H(x_i) = P(x_i) - Q(x_i) = 0 \quad \forall \; 1 \leq i \leq n$.

This shows that H has n distinct zeros. But, according to the FTA (Fund. Theorem of Algebra), a $n-1$ degree can have at most $n-1$ distinct zeros, unless the function is identically zero. Consequently $H(x) \equiv 0$, which shows $P(x) \equiv Q(x)$, proving uniqueness,

Ex. Find the (Lagrange) interpolating polynomial of degree 3 or less for: $(0,2)$, $(1,1)$, $(2,0)$ & $(3,-1)$.

$$P(x) = 2\,\frac{(x-1)(x-2)(x-3)}{(0-1)(0-2)(0-3)} + 1\,\frac{(x-0)(x-2)(x-3)}{(1-0)(1-2)(1-3)} + 0\,\frac{(x-0)(x-1)(x-3)}{(2-0)(2-1)(2-3)} - 1 \cdot \frac{(x-0)(x-1)(x-2)}{(3-0)(3-1)(3-2)}$$

$$= -\frac{1}{3}(x^3 - 6x^2 + 11x - 6) + \frac{1}{2}(x^3 - 5x^2 + 6x) - \frac{1}{6}(x^3 - 3x^2 + 2x)$$

$$= \boxed{-x+2}$$

Our theorem says the interpolating polynomial of degree 3 or less is unique, so our result indicates the data set is $\boxed{\text{co-linear.}}$

Newton's Divided Difference

Lagrange interpolation, despite its seeming utility is rarely used in practice. Instead, as we shall see, we prefer to use a recursive method known as Newton's Divided Difference.

Def Denote by $f[x_1, \ldots, x_n]$ the coefficient of the x^{n-1} term in the unique polynomial that interpolates: $(x_1, f(x_1)), \ldots, (x_n, f(x_n))$.

Ex. (one more Lagrange)

Interpolate: $(0,1)$, $(2,2)$ & $(3,4)$

$$P_2(x) = 1 \cdot \frac{(x-2)(x-3)}{(0-2)(0-3)} + 2 \cdot \frac{(x-0)(x-3)}{(2-0)(2-3)} + 4\,\frac{(x-0)(x-2)}{(3-0)(3-2)}$$

$$= \boxed{\frac{1}{2}x^2 - \frac{1}{2}x + 1}$$

Using our new notation, then: $f[0\ 2\ 3] = \frac{1}{2}$.

We now give a remarkable alternative interpolating formula:

$$\begin{aligned} P(x) = {} & f[x_1] + f[x_1\ x_2](x-x_1) + f[x_1\ x_2\ x_3](x-x_1)(x-x_2) \\ & + f[x_1\ x_2\ x_3\ x_4](x-x_1)(x_4-x_2)(x-x_3) \\ & + \cdots + f[x_1\ x_2 \ldots x_n](x-x_1)\cdots(x-x_{n-1}) \end{aligned}$$

→ Newton's Divided Difference.

Moreover, the coefficients $f[x_1 \ldots x_k]$ can be calculated <u>recursively</u>

First, list the data points in a table:

$$\begin{array}{c|c} x_1 & f(x_1) \\ x_2 & f(x_2) \\ \vdots & \vdots \\ x_n & f(x_n) \end{array}$$

Now define the <u>divided differences</u>, which are real numbers:

$$f[x_k] = f(x_k)$$

$$f[x_k \; x_{k+1}] = \frac{f[x_{k+1}] - f[x_k]}{x_{k+1} - x_k}$$

$$f[x_k \; x_{k+1} \; x_{k+2}] = \frac{f[x_{k+1} \; x_{k+2}] - f[x_k \; x_{k+1}]}{x_{k+2} - x_k}$$

$$\vdots$$

$$f[x_k \; x_{k+1} \; x_{k+2} \; x_{k+3}] = \frac{f[x_{k+1} \; x_{k+2} \; x_{k+3}] - f[x_k \; x_{k+1} \; x_{k+2}]}{x_{k+3} - x_k}$$

Generally, we write this calculations as a Tableau:

$$\begin{array}{c|ccccc} x_1 & f[x_1] & \searrow & & & \\ & & & f[x_1 \; x_2] & \searrow & \\ x_2 & f[x_2] & \nearrow & & & f[x_1 \; x_2 \; x_3] \\ & & \searrow & f[x_2, x_3] & \nearrow & \\ x_3 & f[x_3] & \nearrow & & & \end{array}$$

The interpolating polynomial is $\boxed{P(x) = \sum_{i=1}^{n} f[x_1 \ldots x_i](x - x_1) \cdots (x - x_{i-1})}$

Some <u>FACTS</u>: ① The unique interpolating polynomial is given by ↗ ; ② Divided Difference gives a <u>nested</u> i.e. efficiently expressed polynomial.

Ex. Interpolate w/ N's D.D.: (0,1), (2,2), (3,4)

$$\begin{array}{c|ccc} 0 & \boxed{1} & & \\ & & \boxed{\frac{1}{2}} & \\ 2 & 2 & & \boxed{\frac{1}{2}} \\ & & 2 & \\ 3 & 4 & & \end{array} \longrightarrow P(x) = 1 + \frac{1}{2}(x-0) + \frac{1}{2}(x-0)(x-2)$$

$$= \boxed{\frac{1}{2}x^2 - \frac{1}{2}x + 1} \checkmark$$

Computations:

$f[x_1] = 1$

$f[x_2] = 2$

$f[x_3] = 4$

$$f[x_1, x_2] = \frac{f[x_2] - f[x_1]}{x_2 - x_1} = \frac{2-1}{2-0} = \boxed{\frac{1}{2}}$$

$$f[x_2, x_3] = \frac{f[x_3] - f[x_2]}{x_3 - x_2} = \frac{4-2}{3-2} = \boxed{2}$$

$$f[x_1, x_2 x_3] = \frac{f[x_2 x_3] - f[x_1 x_2]}{x_3 - x_1} = \frac{2 - \frac{1}{2}}{3-0} = \boxed{\frac{1}{2}}$$

(*) A key advantage of using Newton's divided difference stems from the fact that if we add a point to our data set, only a small number of updated computations are required (because of recursion).

(*) On the other hand, if we computed $P_2(x)$ for 3 data points, and then add a single datum to our set (now $n=4$), computing $P_3(x)$ for this data would require us to start the process over from scratch (very inefficient).

$\boxed{\text{Ex.}}$ Add a fourth point: (1, 0) to the previous data set & compute $P_3(x)$, using Newton's Divided Diff.

$$\begin{array}{c|cccc}
0 & 1 & & & \\
 & & \frac{1}{2} & & \\
2 & 2 & & \frac{1}{2} & \\
 & & 2 & & -\frac{1}{2} \\
3 & 4 & & \boxed{0} & \\
 & & \boxed{2} & & \\
\boxed{1} & \boxed{0} & & &
\end{array}
\longrightarrow
\begin{aligned}
P_3(x) &= \underline{\underline{1}} + \underline{\underline{\tfrac{1}{2}}}(x-0) + \underline{\underline{\tfrac{1}{2}}}(x-0)(x-2) \\
&\quad \underline{+ \tfrac{1}{2}(x-0)(x-2)(x-3)} \\
&= \underline{\underline{P_2(x)}} + \left(-\tfrac{1}{2}\right)(x-0)(x-2)(x-3)
\end{aligned}$$

→ Newton's D.D. is often used for data that is <u>updated</u> in "real-Time".

<u>Q</u>: How many interpolating polynomials of degree 4 interpolate a data set consisting of 4 points?

<u>A</u>: An <u>Infinite-number</u>! <u>Why?</u>

$$\text{By N's D.D.} \rightarrow P_4(x) = \underbrace{P_3(x)}_{\substack{\text{uniquely determined} \\ \text{by 4 points}}} + \overset{\text{free variable!}}{c_1}(x-x_1)(x-x_2)(x-x_3)(x-x_4)$$

→ Polynomial interpolation is an example of <u>compression</u> in numerical analysis.

Calculator: What is $\sin(.123) = ?$

<u>Mechanism</u>: