

Polynomial approximation

Spring 2021, A. Donov

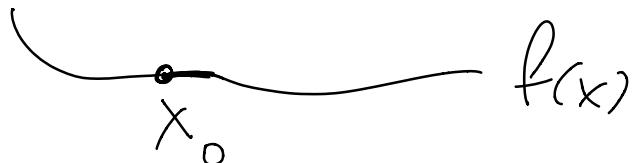
How to approximate nonlinear functions

$$f(x) = \exp(x), \sin(x), e^{\cos(x)}$$

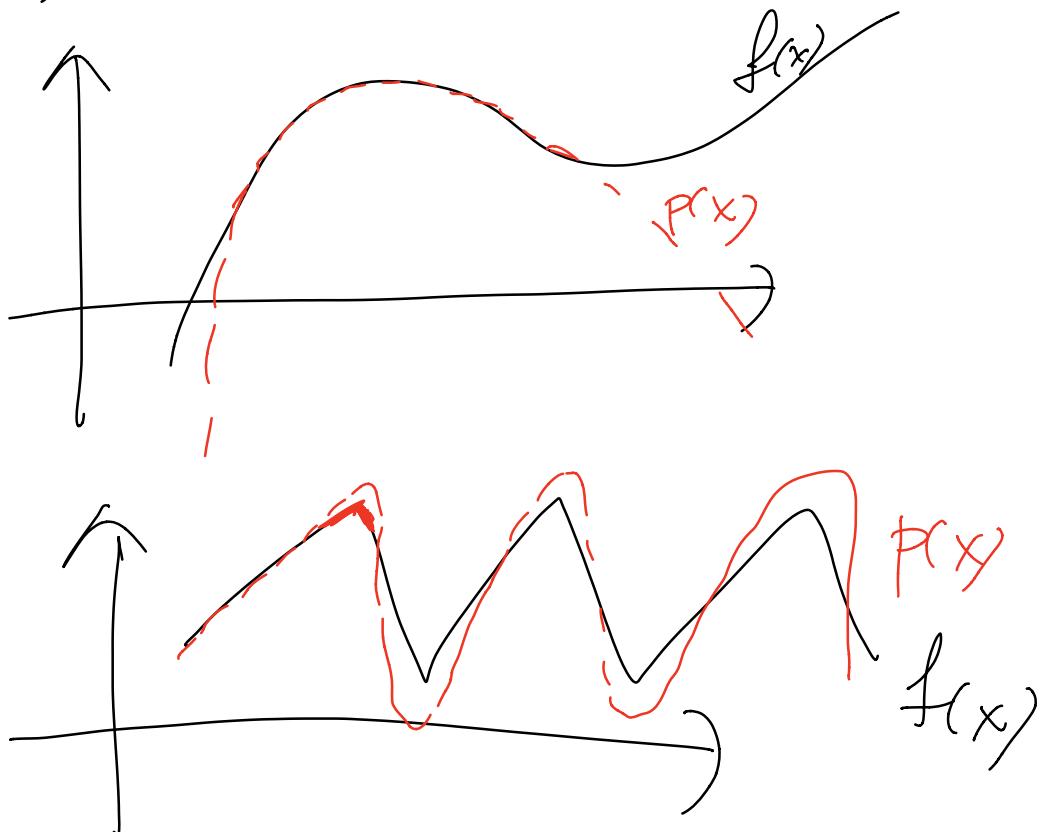
$f(x)$ can be locally approximated by a polynomial on $[a, b]$

$$p(x) \approx f(x) \text{ on } [a, b]$$

Recall Taylor series



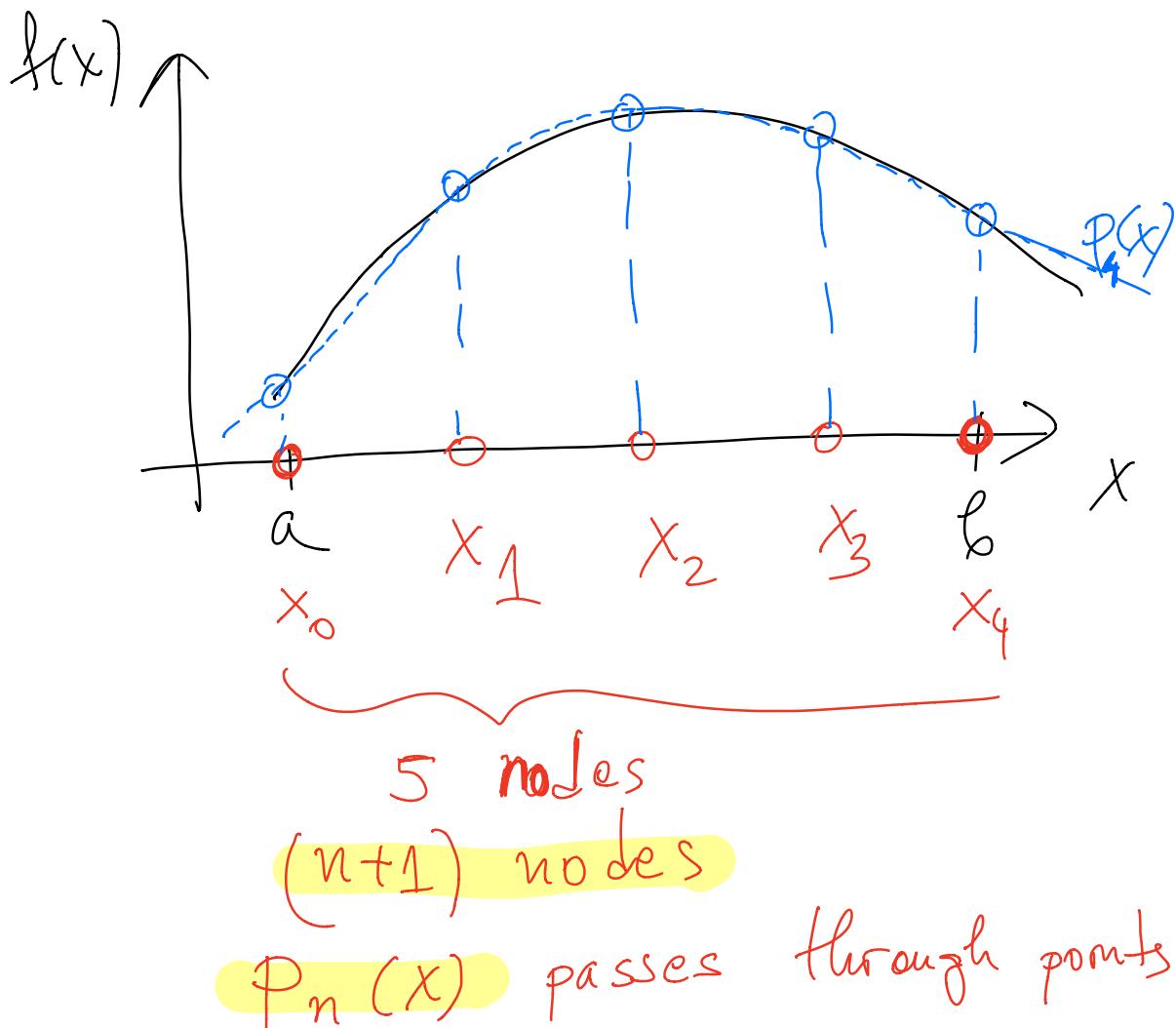
This will work "well" if
 $f(x)$ is smooth



Weierstrass approximation th

$\forall \varepsilon > 0$, $\exists p(x)$ s.t.
 $\max_{a \leq x \leq b} |f(x) - p(x)| < \varepsilon$

Interpolation



$(n+1)$ distinct points uniquely
define a polynomial of degree n

Let's find $P_n(x)$

$$P_n = \underbrace{a_n x^n}_{\text{monomials}} + \underbrace{a_{n-1} x^{n-1}}_{\dots} + \dots + a_1 x + a_0$$

$$\vec{a} = [a_0, a_1, \dots, a_n]$$

$$\left\{ \begin{array}{l} P_n(x_0) = f(x_0) = y_0 \\ P_n(x_1) = f(x_1) = y_1 \\ \dots \\ P_n(x_n) = f(x_n) = y_n \end{array} \right.$$

System of $n+1$ linear
equations for \vec{a}

Interpolation

$$\sum_{k=0}^n a_k x_i^k = y_i$$

$i = 0, \dots, n+1$ nodes

$$\begin{matrix} V & \xrightarrow{\quad} & a \\ & \xleftarrow{\quad} & = \\ & & y \end{matrix}$$

Square linear system

(Fitting V was not square)

$$V = \begin{bmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \dots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{bmatrix} \begin{bmatrix} a_0 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_0 \\ \vdots \\ y_n \end{bmatrix}$$

Vandermonde matrix

$$V \xrightarrow{\quad} a = y$$

LU factorization

Is V invertible?

$$\det(V) = \prod_{j < k} (x_k - x_j) \neq 0$$

If nodes are distinct
then V is invertible

~~~  
Is  $P_n$  unique? Yes

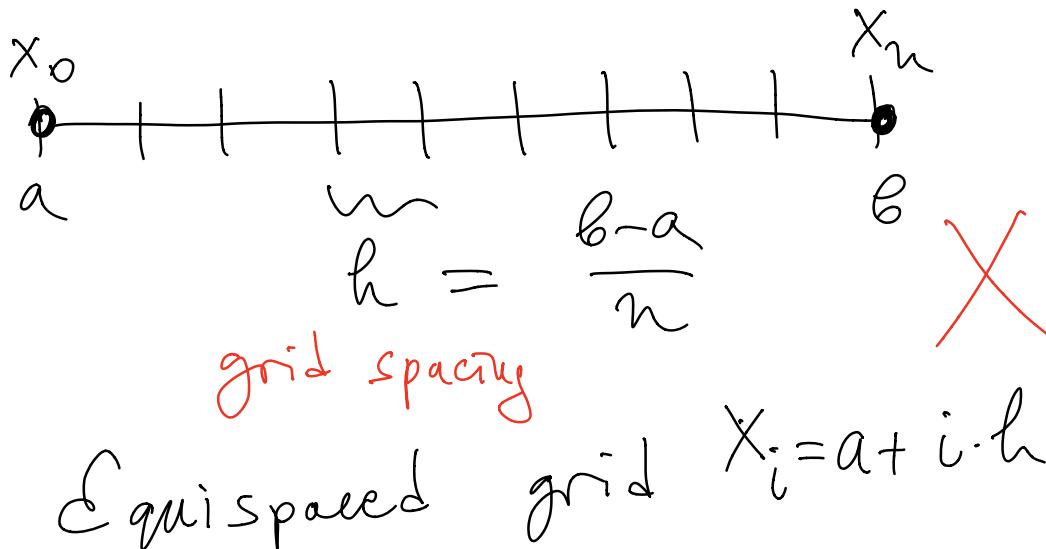
In Matlab you can do  
this using - polyfit

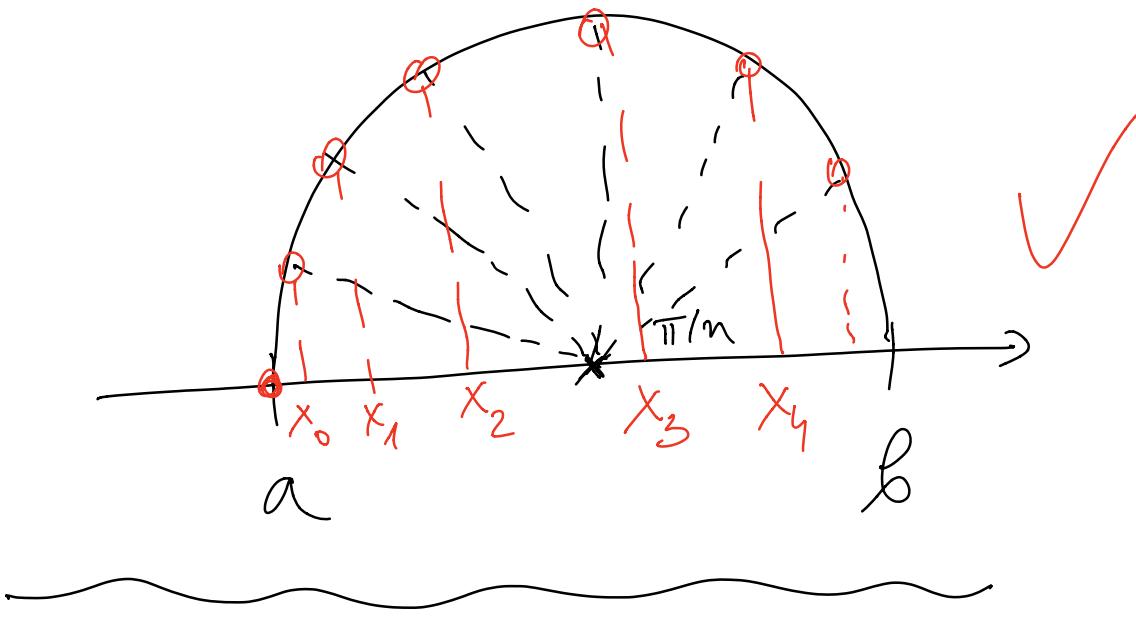
To evaluate  $p(x)$  - polyval

$$Va = y \quad X$$

① If  $n$  is large,  $O(n^3)$   
cost is large

② Is  $V$  well-conditioned?  
No - ill-conditioned matrix  
just like Hilbert





Abstract linear algebra

$P_n$  — linear space of polynomials of degree  $n$

$$\dim(P_n) = n+1$$

# of linearly independent polynomials that form a basis

$$P_n = \text{span} \left\{ b_0(x), b_1(x), \dots, b_n(x) \right\}$$

Span  $\left\{ x^0, x^1, \dots, x^n \right\}$

If we use monomials as basis, then we get Vandermonde matrix

$$\begin{matrix} \xrightarrow{\quad} & & \xleftarrow{\quad} & \end{matrix} \quad \begin{matrix} V \\ \xleftarrow{\quad} & \xrightarrow{\quad} \\ \text{monomials} & & \text{other basis} \end{matrix}$$

$$\begin{matrix} \xleftarrow{\quad} & \xrightarrow{\quad} & \end{matrix} \quad \begin{matrix} V' \\ a \\ = y \end{matrix}$$

$$V' = \text{I Identity}$$

$$\Rightarrow a = y$$

$$P_n = \text{Span} \left\{ L_0^{(x)}, L_1^{(x)}, \dots, L_n^{(x)} \right\}$$

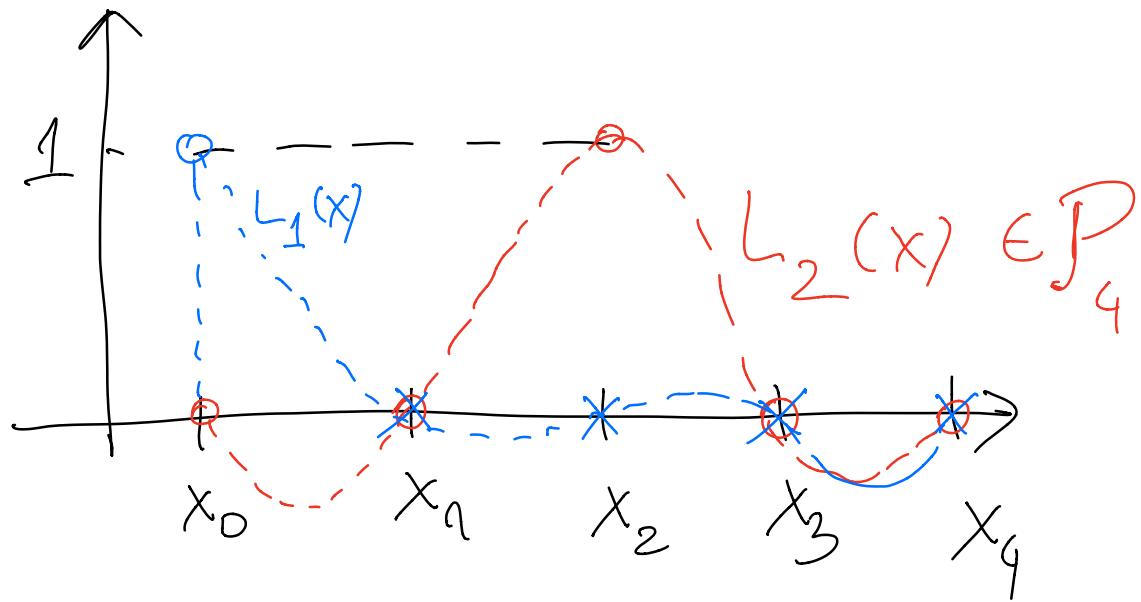
$$p(x) = \sum_{k=0}^n a_k L_k(x)$$

$$\begin{aligned} p(x_i) &= \sum_{k=0}^n a_k L_k(x_i) \\ &= \sum_{k=0}^n V_{ik} a_k \end{aligned}$$

best choice of  $L_k$

$$V_{ik} = \boxed{L_k(x_i) = \delta_{ik}}$$

$$\delta_{ik} = \begin{cases} 1 & \text{if } i=k \\ 0 & \text{otherwise} \end{cases} \quad \begin{matrix} \text{Kronecker} \\ \text{symbol} \end{matrix}$$



$$a = y$$

$\Rightarrow p(x) = \sum_{k=0}^n y_k \underline{\underline{L_k(x)}}$

Lagrange polynomials

$$L_k(x) = C(x - x_0)(x - x_1) \dots (x - x_n)$$

$$L_k(x) = C_k \prod_{\substack{k \neq j \\ j=0}}^n (x - x_j)$$

$$L_k(x_k) = 1$$

$$C_k \prod_{j \neq k} (x_k - x_j) = 1$$

$$C_k = \frac{1}{\prod_{j \neq k} (x_k - x_j)}$$

$$L_k(x) = \frac{\prod_{j \neq k} (x - x_k)}{\prod_{j \neq k} (x_k - x_j)}$$

Lagrange  
Interpolation  
formula

$$P(x) = \sum_{k=0}^n y_k \frac{\prod_{j \neq k} (x - x_j)}{\prod_{j \neq k} (x_k - x_j)}$$

Simple case : 2 points

$$p(x) = y_0 \cdot \frac{x - x_1}{x_0 - x_1} \quad \left. \begin{matrix} j=1 \\ k=0 \end{matrix} \right\}$$

$$\underbrace{\phantom{...}}_{k=0}$$

$$+ y_1 \frac{x - x_0}{x_1 - x_0} \quad \left. \begin{matrix} j=0 \\ k=1 \end{matrix} \right\}$$

$$f \rightarrow \underbrace{\phantom{...}}_{k=1}$$

Are we done?

① How expensive is Lagrange interpolation

Evaluating  $p(x)$   $O(n^2)$  evaluation  
no cost to form  $p(x)$

② Can I evaluate  $p(x)$  without losing digits?

③ How good of an approximation is  $p(x)$  to  $f(x)$ ? meth

Answers:

$$\begin{aligned} \textcircled{1} \quad p_n &= a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n \\ &= a_0 + x \underbrace{(a_1 + a_2 x + \dots + a_n x^{n-1})}_{+ a_0 + x (a_1 + x (a_2 + a_3 x + \dots + a_{n-1} x^{n-2}))} \end{aligned}$$

$$= \dots \times (a_{n-3} + x(a_{n-2} + x(a_{n-1} + a_n x)))$$

$$\left. \begin{array}{l} b_{n-1} = a_{n-1} + a_n x \\ b_{n-2} = a_{n-2} + b_{n-1} x \\ \vdots \\ b_0 = a_0 + b_1 x = p(x) \end{array} \right\} 2n \text{ FLOPS}$$

$O(n)$  method -

Horner's scheme

Evaluating Lagrange polynomial  
 Interpolant costs  $O(n^2)$  FLOPS

## Barycentric formula

$$w_k = \prod_{j \neq k} \frac{1}{x_k - x_j}$$

weights  
 $k=0, \dots, n$

(pre-computation)

$$p_n(x) = \frac{\sum_{k=0}^n \frac{w_k}{x - x_k} y_k}{\sum_{k=0}^n \frac{w_k}{x - x_k}}$$

As long as  $x \neq x_j$   
this loses no digits