MATH50003
Numerical Analysis

IV.1 Polynomial Interpolation and Regression

Dr Sheehan Olver

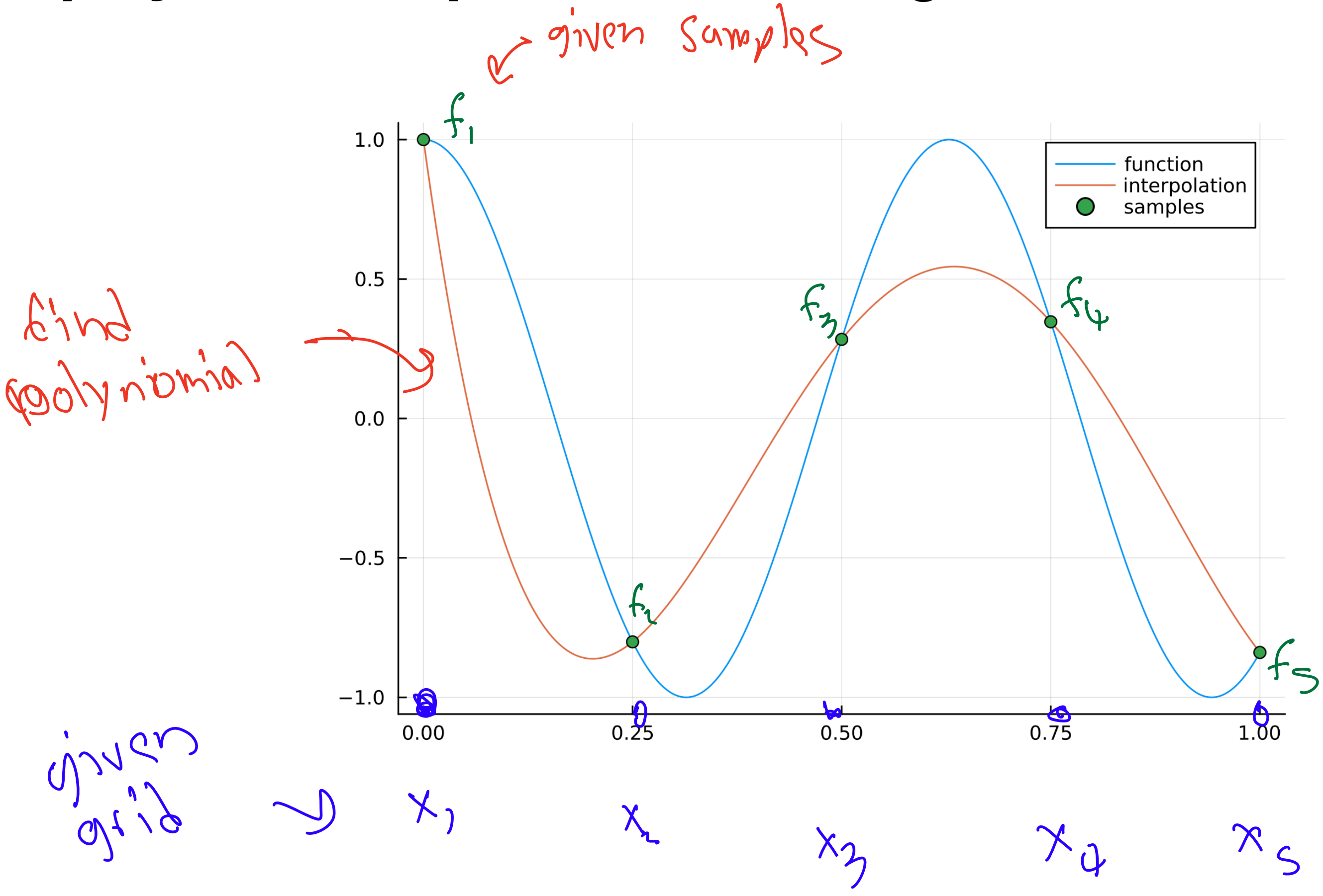
Part IV

Applications of Linear Algebra

1. Polynomial Interpolation and Regression for approximating data
2. Differential Equations via finite difference

IV.1.1 Polynomial interpolation

Find a polynomial equal to data at a grid



grid

Definition 25 (interpolatory polynomial). Given *distinct* points $\boldsymbol{x} = [x_1, \ldots, x_n]^\top \in \mathbb{C}^n$ and *data* $\boldsymbol{f} = [f_1, \ldots, f_n]^\top \in \mathbb{C}^n$, a degree $n-1$ *interpolatory polynomial* $p(x)$ satisfies

$$p(x_j) = f_j$$

find p

given

Where

$$p(x) = \sum_{k=0}^{n-1} c_k \, x^k$$

find $\vec{c} = \begin{bmatrix} c_0 \\ \vdots \\ c_{n-1} \end{bmatrix}$

rectangular

Definition 26 (Vandermonde)**.** The $\vee$ *Vandermonde matrix* associated with $\boldsymbol{x} \in \mathbb{C}^m$ is the matrix

$$V_{\boldsymbol{x},n} := \begin{bmatrix} 1 & x_1 & \cdots & x_1^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & \cdots & x_m^{n-1} \end{bmatrix} \in \mathbb{C}^{m\times n}.$$

V is a linear map from $\begin{bmatrix} c_0 \\ | \\ c_{n-1} \end{bmatrix}$ to $\begin{bmatrix} p(x_1) \\ | \\ p(x_m) \end{bmatrix}$

ie

$$V\vec{c} = \begin{bmatrix} c_0 + x_1 c_1 + \cdots + x_1^{n-1} c_{n-1} \\ | \\ c_0 + x_m c_1 + \cdots + x_m^{n-1} c_{n-1} \end{bmatrix} = \begin{bmatrix} p(x_1) \\ | \\ p(x_m) \end{bmatrix}$$

$\Rightarrow$ if $m = n$ (square), assume V is invertible,

$$\vec{c} = V^{-1} \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix}$$

are coeffs of interpolating poly since

$$\begin{bmatrix} p(x_1) \\ \vdots \\ p(x_n) \end{bmatrix} = V \vec{c} = V V^{-1} \vec{f} = \vec{f}$$

Proposition 11 (interpolatory polynomial uniqueness). *Interpolatory polynomials are unique and therefore square Vandermonde matrices are invertible.*

Proof

uniqueness

Suppose $\underbrace{p \ \& \ \tilde{p}}_{\text{degree } n-1}$ s.t. $p(x_i) = \tilde{p}(x_i) = f_i$.

Consider

$$r(x) := \underbrace{p(x) - \tilde{p}(x)}_{\text{degree } n-1}.$$

Then

$$r(x_1) = r(x_2) = \text{———} = r(x_n) = 0$$

q_1

degree $n-1$ poly that's 0 at n points

$$\Rightarrow \quad r = 0 \Rightarrow p = \tilde{p}.$$

Fundamental Theorem of Algebra

invertible

If $V\vec{c} = 0$ then

$p(x) = \sum_{j=0}^{n-1} c_k x^k$ is 0 at $\{x_i\}$ since

$$\begin{bmatrix} p(x_1) \\ | \\ p(x_n) \end{bmatrix} = V\vec{c} = 0 \Rightarrow p = 0 \Rightarrow \vec{c} = 0.$$

$\Rightarrow$ V is nonsingular / invertible ∎

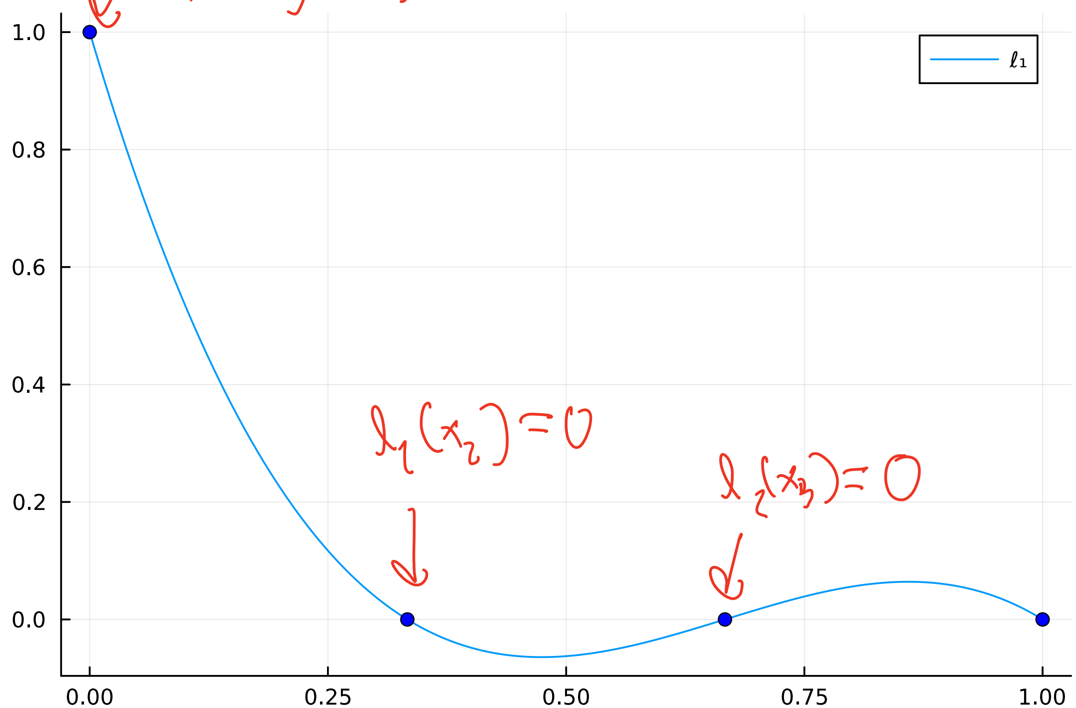
Definition 27 (Lagrange basis polynomial)**.** The *Lagrange basis polynomial* is defined as

$$\ell_k(x) := \prod_{j \neq k} \frac{x - x_j}{x_k - x_j} = \frac{(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_1)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}$$

So that

$\ell_k(x_j) = \delta_{kj}$

$\ell_1(x_1) = 1$

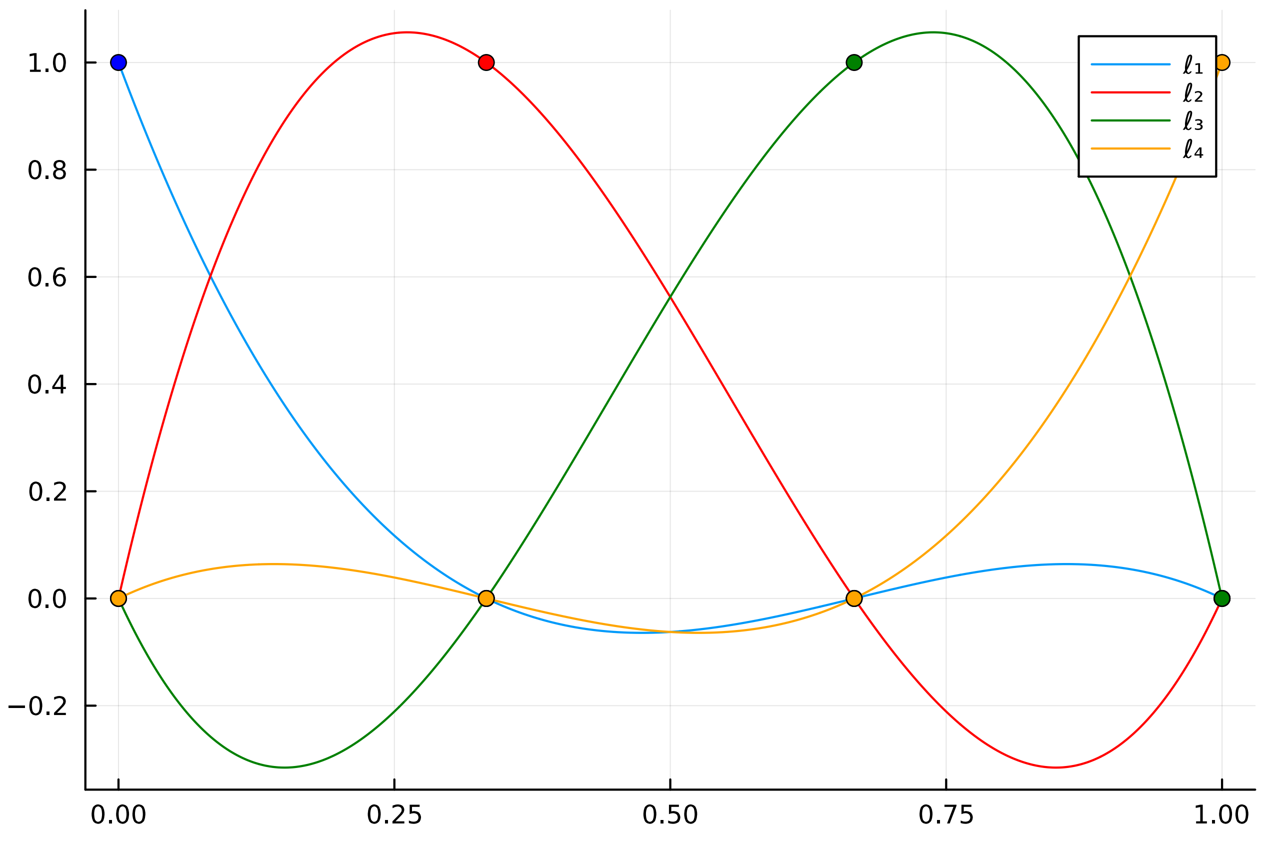


$\ell_1(x_2) = 0$

$\ell_2(x_3) = 0$

x_1 x_2 x_3 x_4

$$\ell_1(x) = \frac{(x-x_2) \cdots (x-x_n)}{(x_1-x_2)\cdots(x_1-x_n)}$$



1.0
0.8
0.6
0.4
0.2
0.0
−0.2
0.00
0.25
0.50
0.75
1.00
ℓ_1
ℓ_2
ℓ_3
ℓ_4

Theorem 9 (Lagrange interpolation)**.** *The unique interpolation polynomial is:*

$$p(x) = \underbrace{f_1\ell_1(x) + \cdots + f_n\ell_n(x)}_{\text{degree } n-1}$$

Since

$$p(x_j) = \sum_{k=1}^{n} f_k \underbrace{\ell_k(x_j)}_{\delta_{kj}} = f_j.$$

Example 18 (interpolating an exponential)**.**

Interpolate e^x at $[\underbrace{0}_{x_1}, \underbrace{1}_{x_2}, \underbrace{2}_{x_3}]$

by a quadratic. Here $\begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} 1 \\ e \\ e^2 \end{bmatrix}$

On a computer: invert Vandermonde.

By hand: use Lagrange:

$$\ell_1(x) = \frac{(x-1)(x-2)}{(-1)(-2)}$$

$$\ell_2(x) = \frac{x(x-2)}{1\cdot(-1)}$$

$$\ell_3(x) = \frac{x(x-1)}{2\cdot 1}$$

SEW Familiarisation: 7 March
Mock computer exam: 6 March
- Will be released on Github
- Do on own machine
- Stick to 1 hour time limit
- Email GTA for marking by end of day

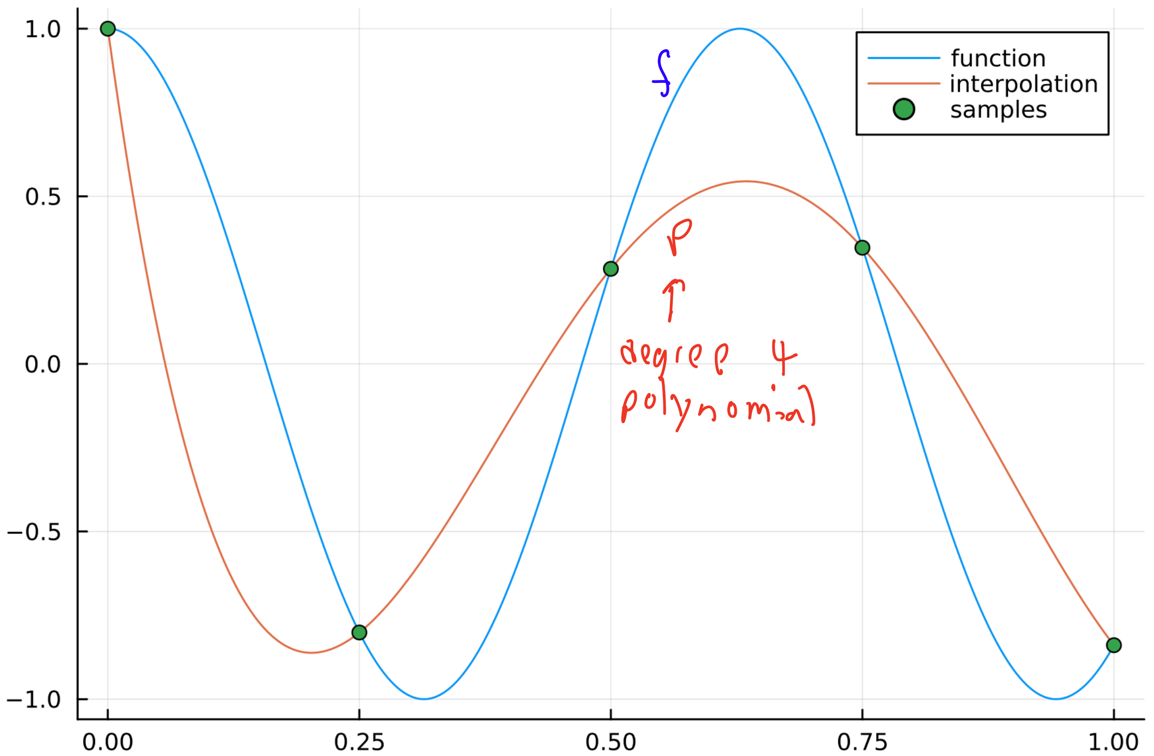
Computer exam: 14 March
- in SEW on college machines

$$\Rightarrow \quad p(x) = \underbrace{1}_{e^0} \underbrace{\frac{(x-1)(x-2)}{2}}_{\ell_1(x)} + e^1 \underbrace{\frac{x(x-2)}{-1}}_{\ell^3(x)}$$

$+ \quad \underbrace{e^2}_{e^{x_3}} \quad \underbrace{\frac{x(x-1)}{2}}_{\ell_3(x)}$

IV.1.2 Interpolatory quadrature rules

Interpolate by polynomials and integrate exactly



If $f(x) \approx p(x)$ then

$$\int_a^b f(x)w(x)dx \approx \int_a^b p(x)w(x)dx$$

Definition 28 (interpolatory quadrature rule)**.** Given a set of points $\boldsymbol{x} = [x_1, \ldots, x_n]^\top$ the interpolatory quadrature rule is:

$$\Sigma_n^{w,\boldsymbol{x}}[f] := \sum_{j=1}^{n} w_j f(x_j)$$

where

$$w_j := \int_a^b \underbrace{\ell_j(x)}_{\text{Lagrange Basis}} \underbrace{w(x)}_{\text{weight}} \mathrm{d}x.$$

Since

$$p(x) = \sum_{j=1}^{n} f(x_j)\, \ell_j(x) \implies$$

$$\int_a^b p(x)\, w(x)\, dx = \sum_{j=1}^{n} f(x_j) \underbrace{\int_a^b \ell_j(x)\, w(x)\, dx}_{w_j}$$

Proposition 12 (interpolatory quadrature is exact for polynomials)**.** *Interpolatory quadrature is exact for all degree $n-1$ polynomials p:*

$$\int_a^b p(x)w(x)\mathrm{d}x = \Sigma_n^{w,\boldsymbol{x}}[p]$$

<u>Proof</u>

interpolation is unique $\Rightarrow$

$$p(x) = \sum_{j=1}^{n} p(x_j)\,\ell_j(x)$$

$\blacksquare$

Example 19 (3-point interpolatory quadrature). $x_1, x_2, x_3 = 0, 1/4, 1$ on $[0,1]$

with weight $w(x) = 1$

Given $\{x_i\}$ & $w(x)$, find w_i:

$$w_1 = \int_0^1 \underbrace{l_1(x)}_{\frac{(x-1/4)(x-1)}{(-1/4)(-1)}} \cancel{w(x)}\, dx = -1/6$$

$$w_2 = \int_0^1 \underbrace{l_2(x)}_{\frac{x(x-1)}{1/4(-3/4)}} dx = 8/9$$

$$w_3 = \int_0^1 \underbrace{\ell_3(x)}_{\frac{x(x-1/4)}{1\cdot(3/4)}}\, dx = 5/18$$

$$\Rightarrow \quad \Sigma_n^{w,\vec{x}}[f] = -\frac{f(0)}{6} + \frac{8}{9} f(1/4) + \frac{5}{18} f(1)$$

Sanity Check: is it exact for $1, x, x^2$? Yes!

$$\Sigma_n^{w,\vec{x}}[1] = -\frac{1}{6} + \frac{8}{9} + \frac{5}{18} = 1 = \int_0^1 1\, dx$$

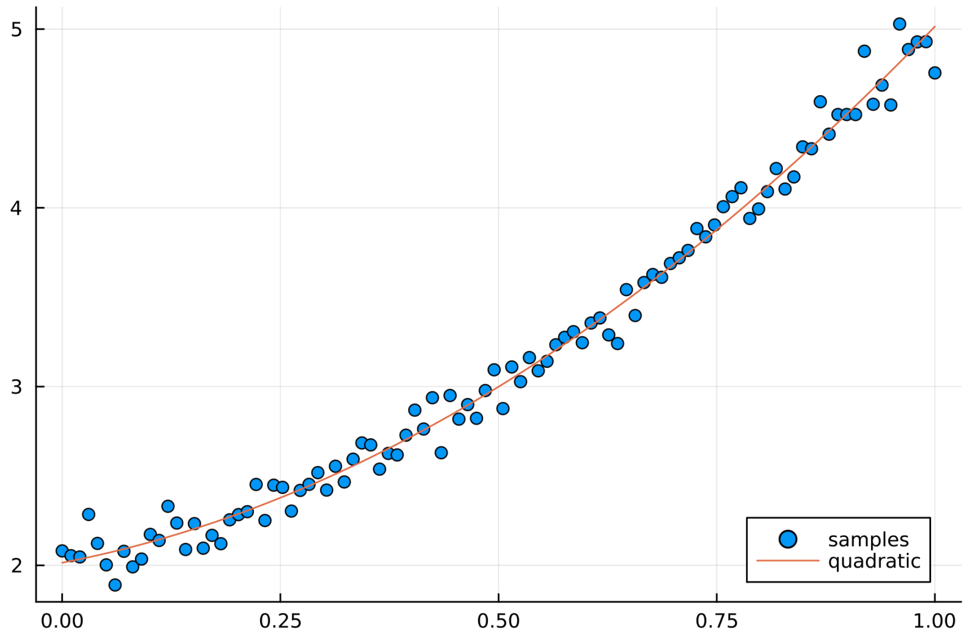
$$\Sigma_n[x] = \frac{8}{9} \times \frac{1}{4} + \frac{5}{18} = \frac{1}{2} = \int_0^1 x\, dx$$

$$\Sigma_n[x^2] = \int_0^1 x^2\, dx$$

$$\Sigma_n[x^3] = \frac{8}{9} \times \frac{1}{4^3} + \frac{5}{18} = \frac{7}{24} \neq \frac{1}{4} = \int_0^1 x^3\, dx$$

IV.1.3 Polynomial regression

How to fit a polynomial to lots of data?



Find

$p(x) = c_0 + c_1 x + c_2 x^2$

s.t.

$p(x_j) \approx f_j$

Find a (degree $n-1$) polynomial p such that:

s.t. $n < m$

$$\begin{bmatrix} p(x_1) \\ \vdots \\ p(x_m) \end{bmatrix} \overset{\text{least squares}}{\approx} \underbrace{\begin{bmatrix} f_1 \\ \vdots \\ f_m \end{bmatrix}}_{\boldsymbol{f}}.$$

Find p that minimises

$$\left\| \underbrace{\begin{bmatrix} p(x_1) \\ \vdots \\ p(x_m) \end{bmatrix}}_{\Rightarrow V\vec{c}} - \underbrace{\begin{bmatrix} f_1 \\ \vdots \\ f_m \end{bmatrix}}_{\vec{f}} \right\| = \sqrt{\sum_{j=1}^{m} \left(p(x_j) - f_j \right)^2}$$

Vandermonde matrix

where $p(x) = \sum_{k=0}^{n-1} c_k x^k = \begin{bmatrix} 1 & x & \cdots & x^{n-1} \end{bmatrix} \begin{bmatrix} c_0 \\ \vdots \\ c_{n-1} \end{bmatrix}$

ie minimise

$$\| V \vec{c} - \vec{f} \|$$

by computing $V = Q R_0$