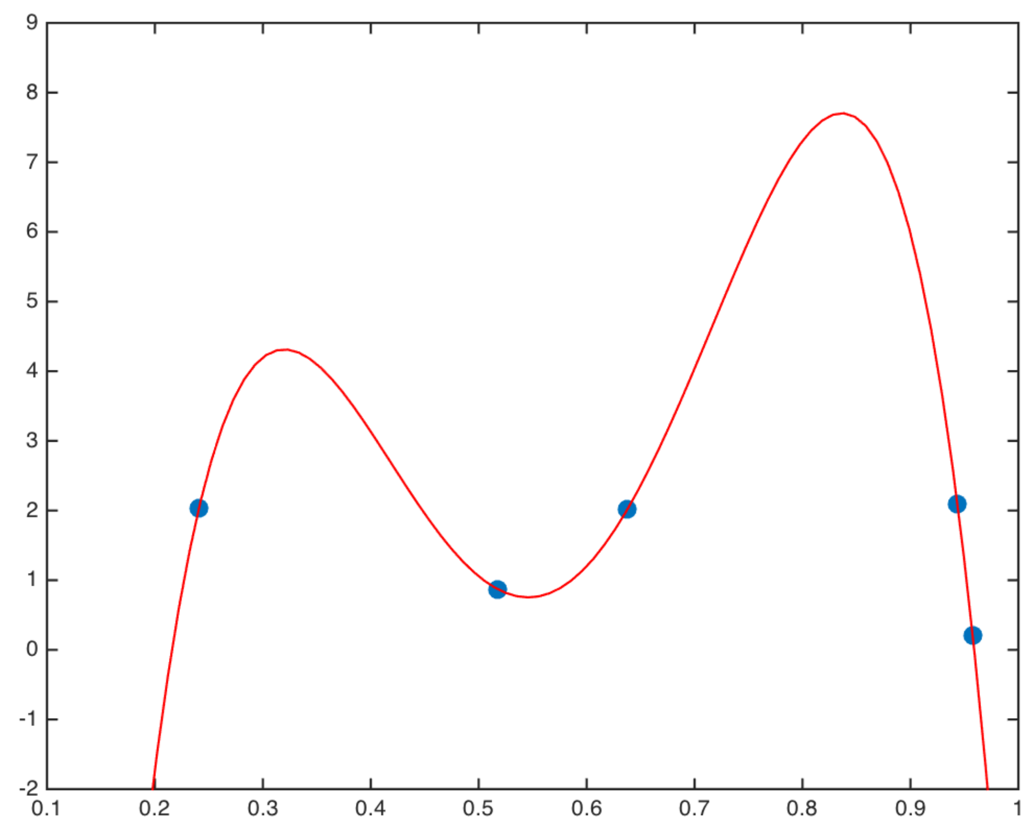
Polynomial interpolation

Vandermonde matrix systems

Polynomial interpolation

Suppose we want to find a curve that exactly fits a given set of set of data points. Perhaps we want to approximate a function with a polynomial, or we are looking to approximate our data in some precise sense.



Polynomial interpolation

Suppose we want to find an n^{th} degree polynomial interpolant

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0$$

Each of $n+1$ data points (x_i, y_i) must then satisfy the condition that

$$P_n(x_i) = a_n x_i^n + a_{n-1} x_i^{n-1} + \ldots + a_1 x_i + a_0 = y_i$$

Such a system is called a *Vandermonde* system and can be written succinctly as

$$V\mathbf{a} = \mathbf{y}, \quad V \in R^{(n+1)\times(n+1)}$$

where the solution $\mathbf{a}$ contains the coefficients a_0, a_1, $\ldots$, a_n.

Vandermonde matrix

The Vandermonde system looks very much like the system we constructed for linear least squares :

$$\underbrace{\begin{bmatrix} x_0^n & x_0^{n-1} & \dots & x_0 & 1 \\ x_1^n & x_1^{n-1} & \dots & x_1 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_n^n & x_n^{n-1} & \dots & x_n & 1 \end{bmatrix}}_{V} \underbrace{\begin{bmatrix} a_n \\ a_{n-1} \\ \vdots \\ a_0 \end{bmatrix}}_{\mathbf{a}} = \underbrace{\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}}_{\mathbf{y}}$$

If we can invert the matrix, we can solve the system and find the unique polynomial that interpolates our data.

Inverting the Vandermonde matrix

How do we know we can solve this system?

Consider the 2×2 system for interpolating a line :

$$\begin{bmatrix} x_0 & 1 \\ x_1 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \end{bmatrix}$$

This system will have a unique solution if

$$\det(V) = (x_0 - x_1) \neq 0$$

$$P_1(x) = \left(\frac{y_0 - y_1}{x_0 - x_1}\right) x + \frac{y_1 x_0 - x_1 y_0}{x_0 - x_1}$$

m b

Inverting the Vandermonde system

Consider the 3×3 system for interpolating a second degree polynomial :

$$\begin{bmatrix} x_0^2 & x_0 & 1 \\ x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \end{bmatrix} \begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \end{bmatrix}$$

Again, this system will have a unique solution if

$$\det(V) = (x_0 - x_1)(x_0 - x_2)(x_1 - x_2) \neq 0$$

or that the x_i's are distinct.

Inverting the Vandermonde system

In general, the Vandermonde system will be invertible if

$$\det(V) = \prod_{0 \le i < j \le n} (x_i - x_j) \neq 0$$

or if the x_i's are distinct.

This also demonstrates that the polynomial that interpolates n+1 distinct points is *unique*, and in theory at least, the coefficients can be written down as

$$\mathbf{a} = V^{-1}\mathbf{y}$$

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0$$

In Matlab

Construct the Vandermonde matrix system using **`vander`** and use the backslash to invert and solve for the coefficients.

Use **`polyfit`** to fit a polynomial of a given degree to your data. For the polynomial interpolation problem, this solves the Vandermonde system.

Caution : the Vandermonde system becomes increasingly ill-conditioned as the polynomial size grows, and so there are in fact better ways to compute the polynomial interpolant.

Uniform approximation

Weierstrass Approximation Theorem. Let f be continuous on the closed interval $[a, b]$. Given any $\epsilon > 0$, there exists a polynomial P such that

$$\|f - P\|_\infty \equiv \max_{x \in [a,b]} |f(x) - P(x)| < \epsilon$$

In other words, given a continuous function f(x) on an interval, we can find a polynomial that comes as "close as we want" to our function (in theory, at least).

Vandermonde Matrix system