MATH2089
Numerical Methods
Lecture 6

Interpolation and

Polynomial Approximation,

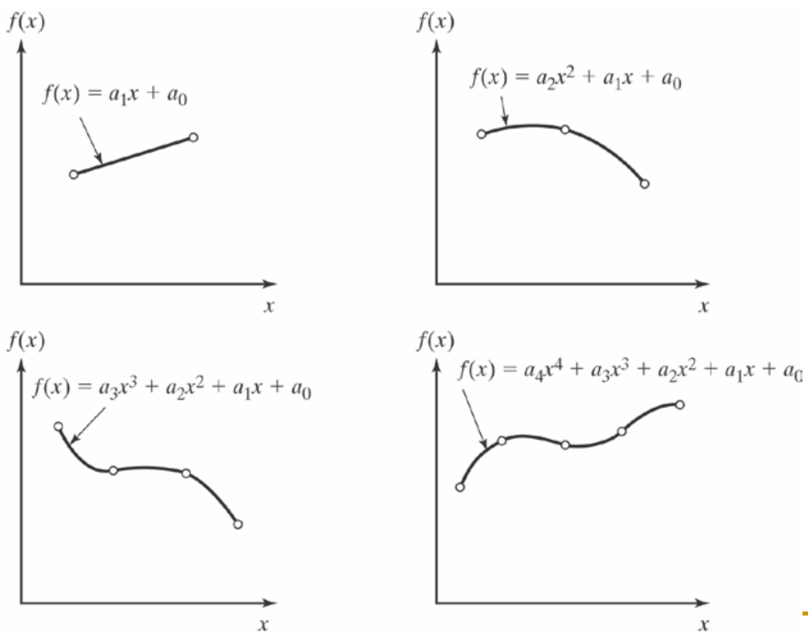
Curve Fitting

Interpolation and Curve Fitting

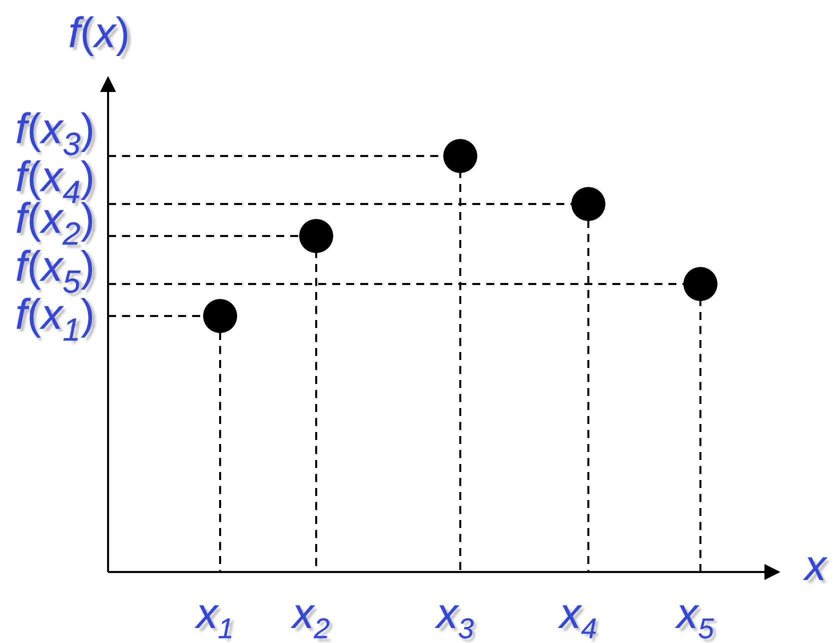
- For a set of data available at discrete points, we may require to fit a smooth and continuous function to this set of data
- Two approaches for curve fitting
  - Interpolation methods (*used when the data are known to be accurate or generated by evaluating a complicated function at a discrete set of points*)
  - Least Square Regression (*is useful when there are more data points than the number of unknown coefficients or when the data appear to have significant error or noise*)

Interpolation Methods

- Curve passes through every data point
- Four approaches
  - Interpolation using different degree-polynomials
  - Lagrange interpolating polynomial
  - Newton interpolating polynomial
  - Splines



Examples of interpolating different degree-polynomials



Interpolation of Different Degree-Polynomials

- Interpolation using a fourth degree-polynomials
- Fit a curve to a series of data and estimate density of helium gas of 17 K
- Five data points

Helium gas

T (K)	**ρ** (kg/m³)
4.22	16.9
7	7.53
10	5.02
20	2.44
30	1.62

$$\rho(T) = a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^4$$

$$\begin{aligned}
a_0 + a_1(4.22) + a_2(4.22)^2 + a_3(4.22)^3 + a_4(4.22)^4 &= 16.9 \\
a_0 + a_1(7) + a_2(7)^2 + a_3(7)^3 + a_4(7)^4 &= 7.53 \\
a_0 + a_1(10) + a_2(10)^2 + a_3(10)^3 + a_4(10)^4 &= 5.02 \\
a_0 + a_1(20) + a_2(20)^2 + a_3(20)^3 + a_4(20)^4 &= 2.44 \\
a_0 + a_1(30) + a_2(30)^2 + a_3(30)^3 + a_4(30)^4 &= 1.62
\end{aligned}$$

Example

- Set of equations in matrix form

$$\begin{bmatrix} 1 & 4.22 & 17.801 & 75.151 & 317.14 \\ 1 & 7 & 49 & 343 & 2401 \\ 1 & 10 & 100 & 1000 & 10^4 \\ 1 & 20 & 400 & 8000 & 1.6\times10^5 \\ 1 & 30 & 900 & 27000 & 8.1\times10^5 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{Bmatrix} = \begin{Bmatrix} 16.9 \\ 7.53 \\ 5.02 \\ 2.44 \\ 1.62 \end{Bmatrix}$$

- Using Gaussian elimination and pivoting

$$\begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{Bmatrix} = \begin{Bmatrix} 56.811 \\ -14.669 \\ 1.4821 \\ -0.062386 \\ 9.0796\times10^{-4} \end{Bmatrix}$$

Example (continue)

- Density equation

$$\rho(T) = 56.811 - 14.669T + 1.4821T^2 - 0.062386T^3 + (9.0796 \times 10^{-4})T^4$$

- Interpolated density at 17 K becomes

$$= 56.811 - 14.6699(17) + 1.4821(17)^2 - 0.062386(17)^3 + (9.0796 \times 10^{-4})(17)^4$$

$= 5.084$ kg/m^3

General Polynomial Fit

- Consider a polynomial of order *n* with *n +1* data points

$$f(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1} + a_n x^n$$

- If the polynomial passes through (x_i, y_i)

$$y_i = a_0 + a_1 x_i + a_2 x_i^2 + \ldots + a_{n-1} x_i^{n-1} + a_n x_i^n$$

General Polynomial Fit (continue)

- In matrix form

$$[A]\vec{a} = \vec{y} \qquad \begin{bmatrix} 1 & x_0 & x_0^2 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & x_n & x_n^2 & x_n^3 & \cdots & x_n^n \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{Bmatrix} = \begin{Bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{Bmatrix}$$

- [*A*] is known as a Vandermonde matrix, prone to ill conditioning

Polynomial Fit - example

Suppose we want to fit the parabola $f(x) = a_0 + a_1 x + a_2 x^2$

that passes through the last three density values from previous example

$$x_0 = 10 \quad f(x_0) = 5.02$$

$$x_1 = 20 \quad f(x_1) = 2.44$$

$$x_2 = 30 \quad f(x_2) = 1.62$$

Helium gas

T (K)	**ρ** (kg/m^3)
4.22	16.9
7	7.53
10	5.02
20	2.44
30	1.62

$$\begin{bmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \end{Bmatrix}$$

Lagrange Interpolating Polynomial

- Lagrange polynomial of order 2

$$f(x) = y = b_0(x - x_1)(x - x_2) + b_1(x - x_0)(x - x_2) + b_2(x - x_0)(x - x_1)$$

- Substitution 3 data points

$$\begin{aligned} y_0 &= b_0(x_0 - x_1)(x_0 - x_2) & & b_0 = y_0 / (x_0 - x_1)(x_0 - x_2) \\ y_1 &= b_1(x_1 - x_0)(x_1 - x_2) & \Rightarrow & b_1 = y_1 / (x_1 - x_0)(x_1 - x_2) \\ y_2 &= b_2(x_2 - x_0)(x_2 - x_1) & & b_2 = y_2 / (x_2 - x_0)(x_2 - x_1) \end{aligned}$$

Example

- 1^{st} order Lagrange interpolating polynomial

$$f_1(x) = y_0 \frac{(x - x_1)}{(x_0 - x_1)} + y_1 \frac{(x - x_0)}{(x_1 - x_0)} = 5.02 \frac{(x - 20)}{(10 - 20)} + 2.44 \frac{(x - 10)}{(20 - 10)}$$

- 2 points

Helium gas

T (K)	**ρ** (kg/m³)
4.22	16.9
7	7.53
10	5.02
20	2.44
30	1.62

$$f_1(17) = -0.502(17 - 20) + 0.244(17 - 10) = 3.214 \text{ kg/m}^3$$

Example (continue)

- 2^{nd} order Lagrange interpolating polynomial

$$f(x) = y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}$$

- 3 points

Helium gas

T (K)	**ρ** (kg/m³**)**
4.22	16.9
7	7.53
10	5.02
20	2.44
30	1.62

$$f_2(x) = 7.53\frac{(x-10)(x-20)}{(7-10)(7-20)} + 5.02\frac{(x-7)(x-20)}{(10-7)(10-20)} + 2.44\frac{(x-7)(x-10)}{(20-7)(20-10)}$$

$$f_2(17) = 0.1931(17-10)(17-20) - 0.1673(17-7)(17-20) + 0.01877(17-7)(17-10)$$

= 2.279 kg/m³

Example (continue)

- 3rd order Lagrange interpolating polynomial
- 4 points

Helium gas

T (K)	ρ (kg/m³)
4.22	16.9
7	7.53
10	5.02
20	2.44
30	1.62

$$f_3(x) = 7.53\frac{(x-10)(x-20)(x-30)}{(7-10)(7-20)(7-30)} + 5.02\frac{(x-7)(x-20)(x-30)}{(10-7)(10-20)(10-30)}$$
$$+2.44\frac{(x-7)(x-10)(x-30)}{(20-7)(20-10)(20-30)} + 1.62\frac{(x-7)(x-10)(x-20)}{(30-7)(30-10)(30-20)}$$

$$f_3(x) = -0.008395(x-10)(x-20)(x-30) + 0.008367(x-7)(x-20)(x-30)$$
$$-0.001877(x-7)(x-10)(x-30) + 0.0003522(x-7)(x-10)(x-20)$$

$f_3(17) = 2.605$ kg/m³

Compact Form

- General form of *n*th degree

$$f(x) = \sum_{i=0}^{n} y_i L_i = \sum_{i=0}^{n} y_i \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{(x - x_j)}{(x_i - x_j)}$$

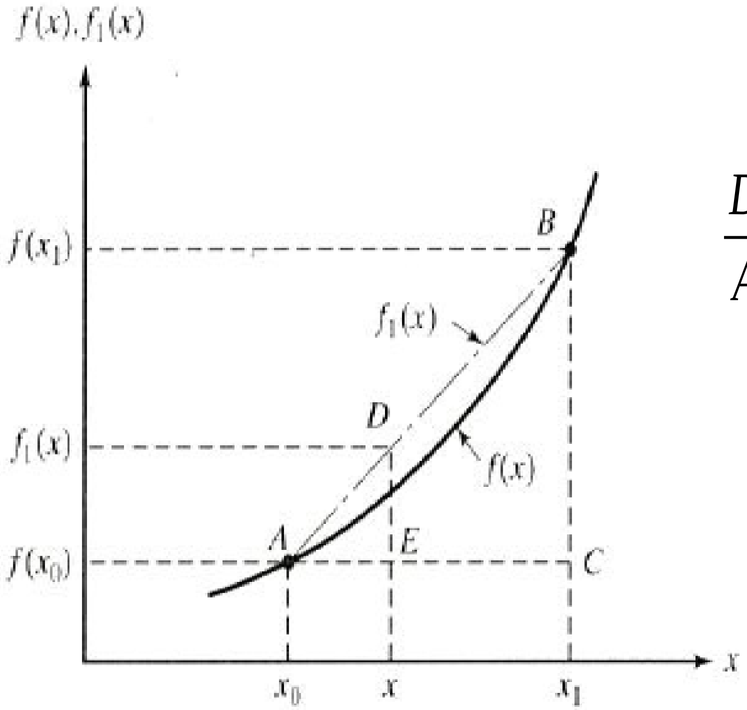
Newton Interpolating Polynomial

- Disadvantages of Lagrange interpolation polynomial
  - A large of number arithmetic operations
  - Cannot be used if data points are changed
  - Estimation of error in interpolation is not easy

- *n*th degree Newton interpolating polynomial

$$f_n(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + \cdots + c_n(x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

Newton Interpolating Polynomial (continue)



- 1^{st} order - liner interpolation

$$\frac{DE}{AE} = \frac{BC}{AC} \text{ or } \frac{f_1(x) - f(x_0)}{x - x_0} = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

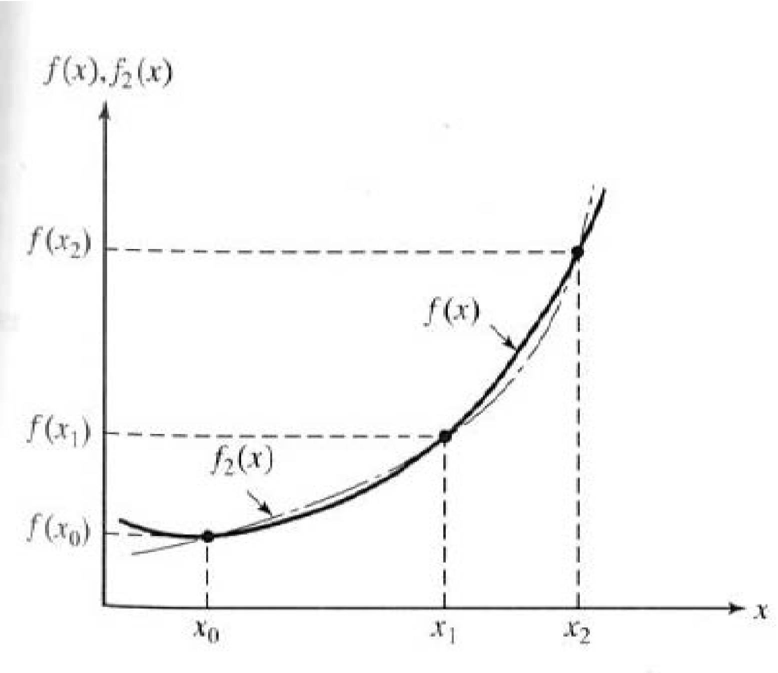
$$f_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0)$$

$$f_1(x) = c_0 + c_1(x - x_0)$$

where $c_0 = f(x_0)$

$$c_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = f[x_1, x_0]$$

Newton Interpolating Polynomial (continue)



- 2nd order quadratic interpolation

$$f_2(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1)$$

where

$$c_0 = f(x_0)$$

$$c_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = f[x_1, x_0]$$

$$c_2 = \frac{1}{(x_2 - x_0)}\left\{\frac{f(x_2) - f(x_1)}{x_2 - x_1} - \frac{f(x_1) - f(x_0)}{x_1 - x_0}\right\} = f[x_2, x_1, x_0]$$

Newton Interpolating Polynomial (continue)

- Substituting (x_0, y_0), we obtain $f_n(x_0) = c_0 = y_0$
- Substituting (x_1, y_1), and use $c_0 = y_0$, we obtain

$$f_n(x_1) = c_0 + c_1\left(x_1 - x_0\right) = y_1$$

$$c_1 = f[x_1, x_0] = \frac{y_1 - y_0}{x_1 - x_0}$$

First divided difference between x_0 and x_1

$$c_2 = f[x_2, x_1, x_0] = \frac{f[x_2, x_1] - f[x_1, x_0]}{x_2 - x_0} = \frac{1}{x_2 - x_0}\left(\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}\right)$$

$$c_3 = f[x_3, x_2, x_1, x_0] = \frac{f[x_3, x_2, x_1] - f[x_2, x_1, x_0]}{x_3 - x_0}$$

$$= \frac{1}{x_3 - x_0}\left(\frac{f[x_3, x_2] - f[x_2, x_1]}{x_3 - x_1} - \frac{f[x_2, x_1] - f[x_1, x_0]}{x_2 - x_0}\right)$$

$$\boxed{f_n(x) = c_0 + c_1\left(x - x_0\right) + c_2\left(x - x_0\right)\left(x - x_1\right) + \cdots + c_n\left(x - x_0\right)\left(x - x_1\right)\cdots\left(x - x_{n-1}\right)}$$

Newton Interpolating Polynomial (continue)

$$c_4 = f[x_4, x_3, x_2, x_1, x_0] = \frac{f[x_4, x_3, x_2, x_1] - f[x_3, x_2, x_1, x_0]}{x_4 - x_0}$$

$$\vdots$$

$$c_n = f[x_n, x_{n-1}, \ldots, x_1, x_0] = \frac{f[x_n, x_{n-1}, \ldots, x_2, x_1] - f[x_{n-1}, x_{n-2}, \ldots, x_1, x_0]}{x_n - x_0}$$

$$\boxed{f_n(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + \cdots + c_n(x - x_0)(x - x_1)\cdots(x - x_{n-1})}$$

Newton Interpolating Polynomial (continue)

- Coefficients $c_0, c_1, \ldots\ldots, c_n$ determined from divided differences from tabulated values

i	x	y	c_1 1st Diff.,	c_2 2nd Diff.,	c_3 3rd Diff.,	c_4 4th Diff.,
0	x_0	$y_0 \to$	$f[x_1, x_0] \to$	$f[x_2, x_1, x_0] \to$	$f[x_3, x_2, x_1, x_0] \to$	$f[x_4, x_3, x_2, x_1, x_0]$
1	x_1	$y_1 \to$	$f[x_2, x_1] \to$	$f[x_3, x_2, x_1] \to$	$f[x_4, x_3, x_2, x_1]$	
2	x_2	$y_2 \to$	$f[x_3, x_2] \to$	$f[x_4, x_3, x_2]$		
3	x_3	$y_3 \to$	$f[x_4, x_3]$			
4	x_4	y_4				

Example

- 1^{st} order

Helium gas

T (K)	**ρ** (kg/m³)
4.22	16.9
7	7.53
10	5.02
20	2.44
30	1.62

$$c_0 = y_0 = 5.02$$

$$c_1 = f[x_1, x_0] = \frac{y_1 - y_0}{x_1 - x_0} = \frac{2.44 - 5.02}{20 - 10} = -0.258$$

i	x	y	First
0	10	5.02	-0.2580
1	20	2.44	

$$f_1(x) = 5.02 - 0.258(x - 10)$$

$$f_1(17) = 5.02 - 0.258(17 - 10) = 3.214 \text{ kg/m}^3$$

Example (continue)

- 2^{nd} order

Helium gas

T (K)	ρ (kg/m^3)
4.22	16.9
7	7.53
10	5.02
20	2.44
30	1.62

i	x	y	First	Second
0	7	7.53	-0.8367	0.04451
1	10	5.02	-0.2580	
2	20	2.44		

$$f_2(x) = 7.53 - 0.8367(x-7) + 0.04451(x-7)(x-10)$$

$$f_2(17) = 7.53 - 0.8367(17-7) + 0.04451(17-7)(17-10) = 2.279 \text{ kg/m}^3$$

- Note: Curve fitting from Lagrange and Newton interpolating polynomials are the same as that of the polynomial fit

Newton-Gregory Formulas

- Applicable for uniformly spaced data
- Newton interpolating polynomial can be simplified

$$h = \left(x_n - x_0\right)/n \qquad x_i = x_0 + ih$$

$$c_0 = \frac{\Delta^0 f_0}{h^0} = f_0 \qquad c_1 = \left\{\frac{f_1 - f_0}{x_1 - x_0}\right\} = \frac{\Delta^1 f_0}{h^1} = \frac{\Delta f_0}{h}$$

$$c_2 = \frac{1}{\left(x_2 - x_0\right)}\left\{\frac{f_2 - f_1}{\left(x_2 - x_1\right)} - \frac{f_1 - f_0}{\left(x_1 - x_0\right)}\right\} = \frac{f_2 - 2f_1 + f_0}{2h^2} = \frac{\Delta^2 f_0}{2!h^2}$$

$$\vdots$$

$$c_n = \frac{\Delta^n f_0}{n!h^n}$$

Forward difference

Newton-Gregory Formulas (continue)

- *n*th order Newton-Gregory forward interpolating polynomial

$$
\begin{aligned}
f_n(x) = f(x_0) &+ \frac{\Delta f_0}{h}(x - x_0) + \frac{\Delta^2 f_0}{2!h^2}(x - x_0)(x - x_0 - h) \\
&+ \cdots + \frac{\Delta^n f_0}{n!h^n}(x - x_0)(x - x_0 - h)\cdots(x - x_0 - (n-1)h)
\end{aligned}
$$

- *n*th order Newton-Gregory backward interpolating polynomial

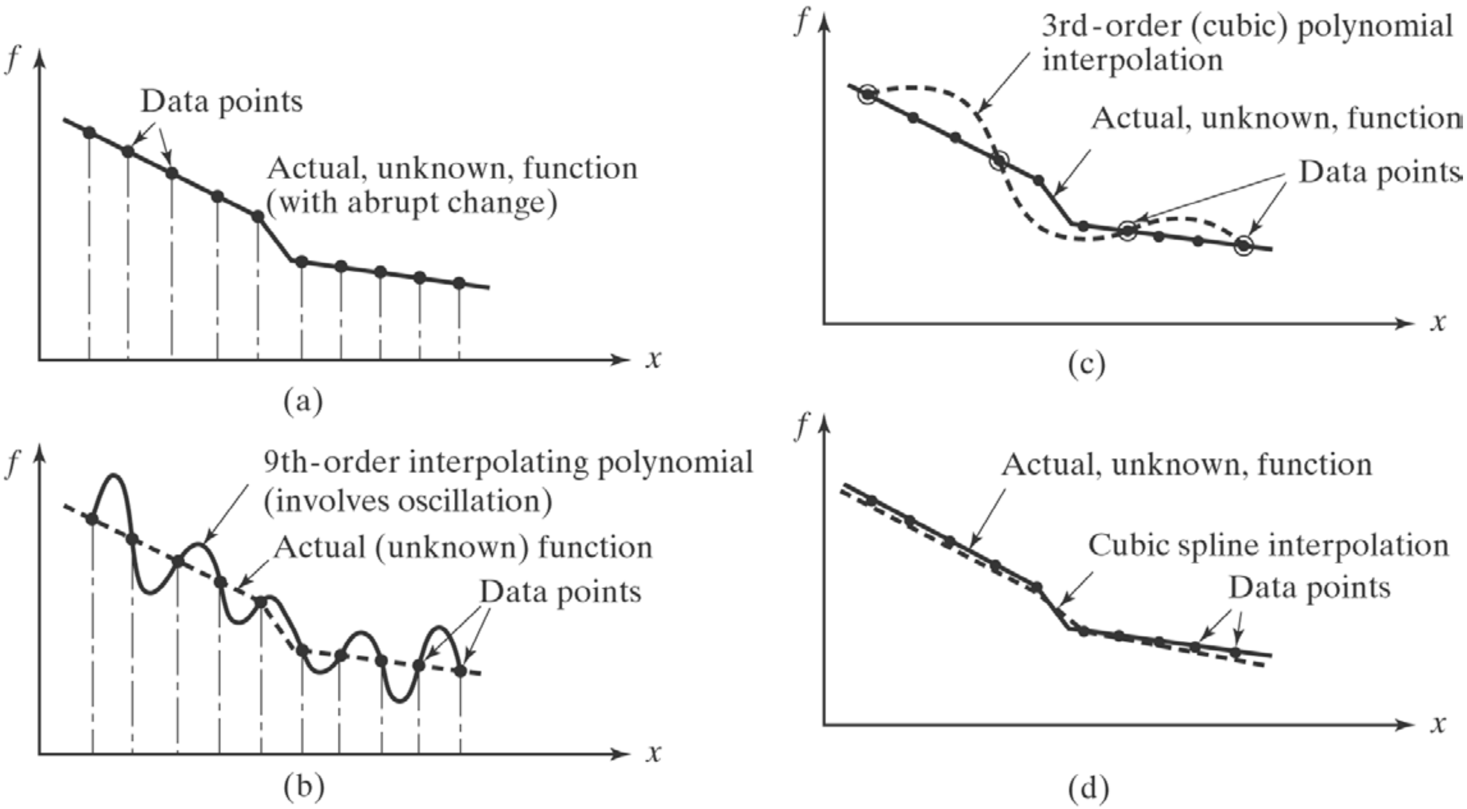
$$
\begin{aligned}
f_n(x) = f(x_n) &+ \frac{\nabla f_n}{h}(x - x_n) + \frac{\nabla^2 f_n}{2!h^2}(x - x_n)(x - x_n + h) \\
&+ \cdots + \frac{\nabla^n f_n}{n!h^n}(x - x_n)(x - x_n + h)\cdots(x - x_n + (n-1)h)
\end{aligned}
$$

- If *x* is close to x_0, choose forward difference
- If *x* is close to x_n, choose backward difference

Splines

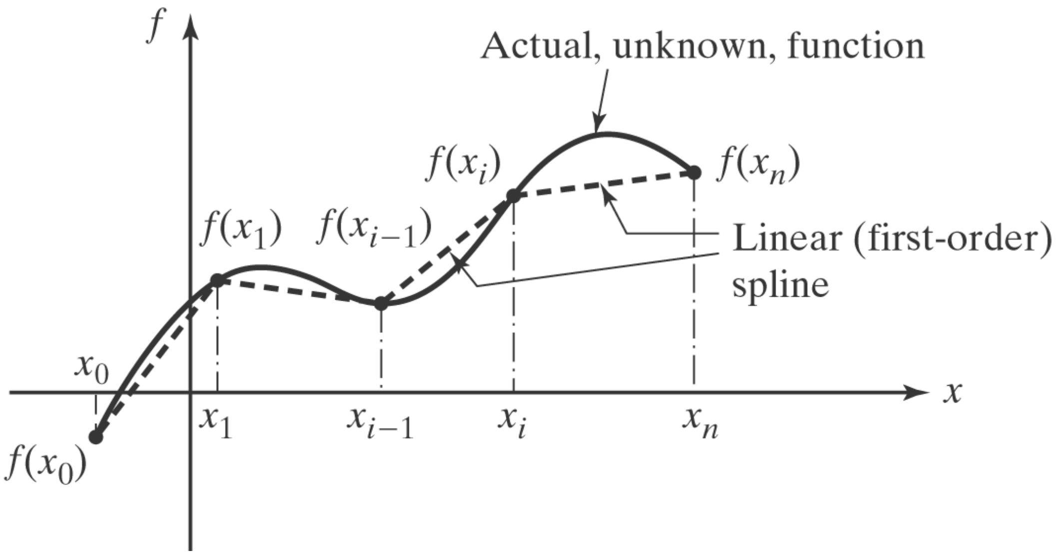
- For Lagrange and Newton interpolation polynomials, although the polynomial pass through all the data points, errors of a single polynomial tend to increase drastically as its order *n* becomes large
- Often, a high-order polynomial introduces unnecessary oscillations or wiggles when the function undergoes an abrupt change in the range of interpolation

Splines (continue)



Linear Spline

- Linear spline represents a straight line joining any two neighboring data points (knots), see figure below
- A linear polynomial in the *i*th interval between point (x_i, y_i) and (x_{i+1}, y_{i+1}) is $f_i(x) = a_i(x - x_i) + b_i$



Example

- Fit the data with a linear spline
- Interval [0.0,1.0]

$$f_0(x_0) = a_0(x_0 - x_0) + b_0 = y_0$$
$$f_0(x_1) = a_0(x_1 - x_0) + b_0 = y_1$$

$$b_0 = 2$$
$$a_0(1-0) + b_0 = 4.4366, a_0 = 2.4366$$

- Interval [1.0,1.5]

$$f_1(x_1) = a_1(x_1 - x_1) + b_1 = y_1$$
$$f_1(x_2) = a_1(x_2 - x_1) + b_1 = y_2$$

$$b_2 = 6.7134$$
$$a_2(2.25-1.5) + b_2 = 13.913, a_2 = 9.5995$$

i	x	f(x)
0	0.0	2.0000
1	1.0	4.4366
2	1.5	6.7134
3	2.25	13.913

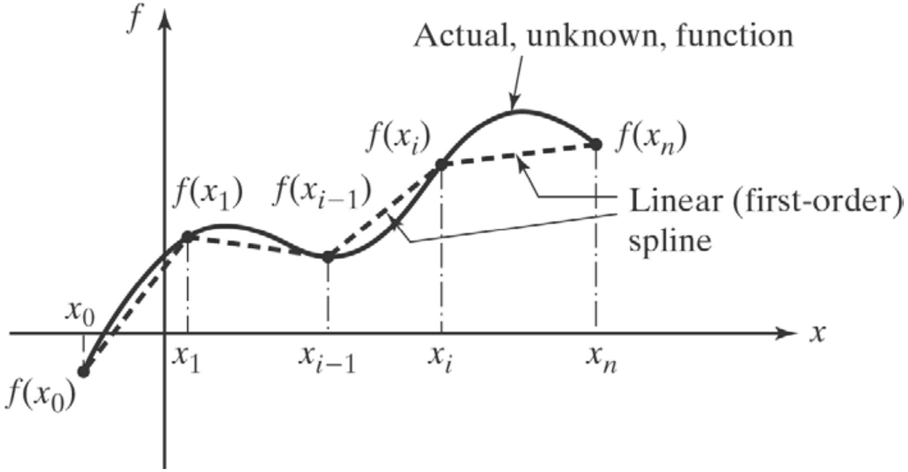
Example (continue)

- Interval [1.5,2.25]

$$f_2(x_2) = a_2(x_2 - x_2) + b_2 = y_2$$
$$f_2(x_3) = a_2(x_3 - x_2) + b_2 = y_3$$
$$b_2 = 6.7134$$
$$a_2(2.25 - 1.5) + b_2 = 13.913, a_2 = 9.5995$$

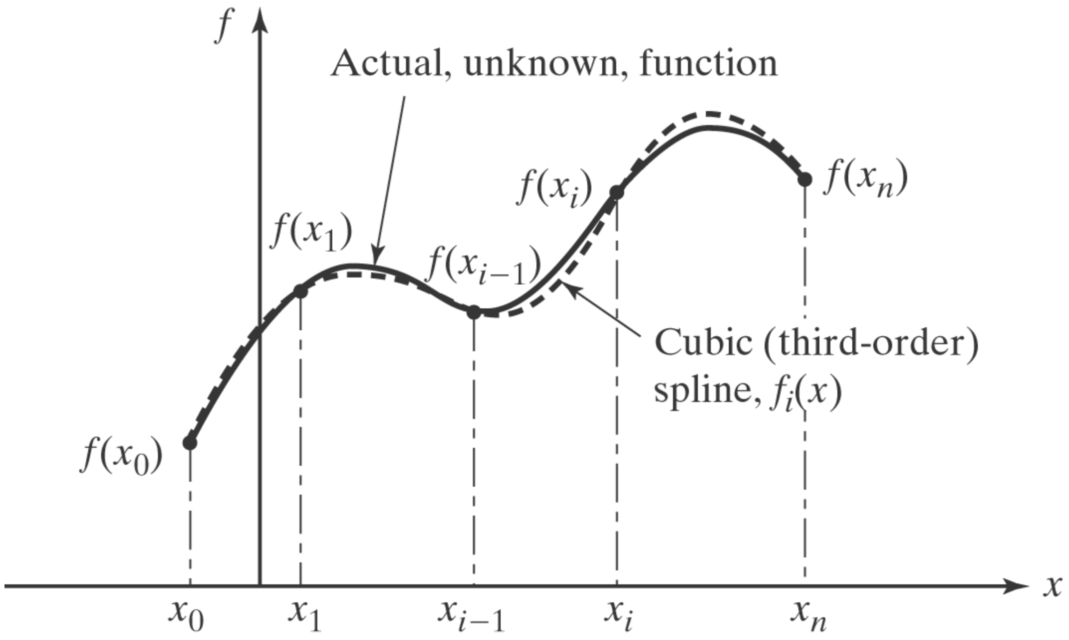
i	x	f(x)
0	0.0	2.0000
1	1.0	4.4366
2	1.5	6.7134
3	2.25	13.913

- Problem with the linear spline is that the slope is discontinuous at the points



Cubic Spline

- Cubic spline represents a cubic equation between any two neighboring data points (knots)
- A cubic polynomial in the *i*th interval between point (x_i, y_i) and (x_{i+1}, y_{i+1}) is $f_i(x) = a_i(x - x_i)^3 + b_i(x - x_i)^2 + c_i(x - x_i) + d_i$



Cubic Spline (continue)

- Satisfies the following conditions:

$$f_i(x_i) = y_i \quad \text{and} \quad f_n(x_n) = y_n$$

$$f_i(x_{i+1}) = f_{i+1}(x_{i+1}) \qquad \text{continuous}$$

$$f_i'(x_{i+1}) = f_{i+1}'(x_{i+1}) \qquad \text{slope}$$

$$f_i''(x_{i+1}) = f_{i+1}''(x_{i+1}) \qquad \text{curvature}$$

Cubic Spline (continue)

- Based on $f_i(x) = a_i(x - x_i)^3 + b_i(x - x_i)^2 + c_i(x - x_i) + d_i$

$$f_i(x_i) = a_i(x_i - x_i)^3 + b_i(x_i - x_i)^2 + c_i(x_i - x_i) + d_i$$

$$f_i(x_i) = d_i \qquad d_i = y_i$$

- Let $h_i = x_{i+1} - x_i$ be the width of *i*th interval

$$f_i(x_{i+1}) = a_i(x_{i+1} - x_i)^3 + b_i(x_{i+1} - x_i)^2 + c_i(x_{i+1} - x_i) + d_i$$

$$a_i h_i^3 + b_i h_i^2 + c_i h_i + y_i = y_{i+1}$$

- Slope and curvature, differentiate $f_i(x)$

$$f_i'(x) = 3a_i h_i^2 + 2b_i h_i + c_i$$

$$f_i''(x) = 6a_i h_i + 2b_i$$

Cubic Spline (continue)

- Let $S_i = f_i''(x_i)$ and $S_n = f_{n-1}''(x_n)$

$$S_i = 6a_i(x_i - x_i) + 2b_i$$

$$S_{i+1} = 6a_i(x_{i+1} - x_i) + 2b_i$$

$$b_i = \frac{S_i}{2} \qquad a_i = \frac{S_{i+1} - S_i}{6h_i}$$

- Solve for c_i , using a_i, b_i and d_i, *recalling* $d_i = y_i$

$$\left(\frac{S_{i+1} - S_i}{6h_i}\right)h_i^3 + \left(\frac{S_i}{2}\right)h_i^2 + c_i h_i + y_i = y_{i+1}$$

$$c_i = \frac{y_{i+1} - y_i}{h_i} - \frac{2h_i S_i + h_i S_{i+1}}{6}$$

Cubic Spline (continue)

- Invoke the condition where the slopes are the same that join (x_i, y_i)

$$f'_{i-1}(x_i) = f'_i(x_i)$$

$$3a_{i-1}(x_i - x_{i-1})^2 + 2b_{i-1}(x_i - x_{i-1}) + c_{i-1} = 3a_i(x_i - x_i)^2 + 2b_i(x_i - x_i) + c_i$$

$$a_{i-1}h_{i-1}^2 + b_{i-1}h_{i-1} + c_{i-1} = c_i$$

- Substituting a_i, b_i, c_i and d_i and simplifying

$$h_{i-1}S_{i-1} + 2(h_{i-1} + h_i)S_i + h_i S_{i+1} = 6\left(\frac{y_{i+1} - y_i}{h_i} - \frac{y_i - y_{i-1}}{h_{i-1}}\right)$$

Cubic Spline (continue)

- For natural spline, $S_0 = 0$ and $S_n = 0$
- We can write S_i in matrix form

$$\begin{bmatrix} 2(h_0+h_1) & h_1 & & & \\ h_1 & 2(h_1+h_2) & h_2 & & \\ & h_2 & 2(h_2+h_3) & h_3 & \\ & & & \ddots & \\ & & & h_{n-2} & 2(h_{n-2}+h_{n-1}) \end{bmatrix} \begin{Bmatrix} S_1 \\ S_2 \\ S_3 \\ \vdots \\ S_{n-1} \end{Bmatrix} = 6 \begin{Bmatrix} f[x_2,x_1]-f[x_1,x_0] \\ f[x_3,x_2]-f[x_2,x_1] \\ f[x_4,x_3]-f[x_3,x_2] \\ \vdots \\ f[x_n,x_{n-1}]-f[x_{n-1},x_{n-2}] \end{Bmatrix}$$

- After S_i values are obtained

$$a_i = \frac{S_{i+1}-S_i}{6h_i} \qquad b_i = \frac{S_i}{2} \qquad c_i = \frac{y_{i+1}-y_i}{h_i} - \frac{2h_iS_i + h_iS_{i+1}}{6} \qquad d_i = y_i$$

Example

- Fit the data with a cubic spline
- For 4 points, $S_0 = 0 \quad S_3 = 0$

i	x	h	y
0	0.0	1.0	2.0000
1	1.0	0.5	4.4366
2	1.5	0.75	6.7134
3	2.25		13.913

$$\begin{bmatrix} 2(h_0+h_1) & h_1 \\ h_1 & 2(h_1+h_2) \end{bmatrix} \begin{Bmatrix} S_1 \\ S_2 \end{Bmatrix} = 6 \begin{Bmatrix} f[x_2,x_1]-f[x_1,x_0] \\ f[x_3,x_2]-f[x_2,x_1] \end{Bmatrix}$$

$$\begin{bmatrix} 2(1+0.5) & 0.5 \\ 0.5 & 2(0.5+0.75) \end{bmatrix} \begin{Bmatrix} S_1 \\ S_2 \end{Bmatrix} = 6 \begin{Bmatrix} \dfrac{6.7134-4.4366}{0.5} - \dfrac{4.4366-2}{1.0} \\ \dfrac{13.913-6.7134}{0.75} - \dfrac{6.7134-4.4366}{0.5} \end{Bmatrix}$$

$$\begin{bmatrix} 3 & 0.5 \\ 0.5 & 2.5 \end{bmatrix} \begin{Bmatrix} S_1 \\ S_2 \end{Bmatrix} = \begin{Bmatrix} 12.702 \\ 30.2754 \end{Bmatrix} \qquad \begin{Bmatrix} S_1 \\ S_2 \end{Bmatrix} = \begin{Bmatrix} 2.292 \\ 11.6518 \end{Bmatrix}$$

$$a_0 = \frac{S_1 - S_0}{6h_0} = \frac{2.292-0}{6(1)} = 0.382 \qquad b_0 = \frac{S_0}{2} = 0 \qquad d_0 = y_0 = 2$$

$$c_0 = \frac{y_1 - y_0}{h_0} - \frac{2h_0S_0 + h_0S_1}{6} = \frac{4.4366-2}{1.0} - \frac{2(1)(0)+(1)(2.292)}{6} = 2.0546$$

Example (continue)

i	x	h	y
0	0.0	1.0	2.0000
1	1.0	0.5	4.4366
2	1.5	0.75	6.7134
3	2.25		13.913

- Interval [0.0,1.0]

$$f_0(x) = 0.382(x-0)^3 + 2.0546(x-0) + 2$$

- Interval [1.0,1.5]

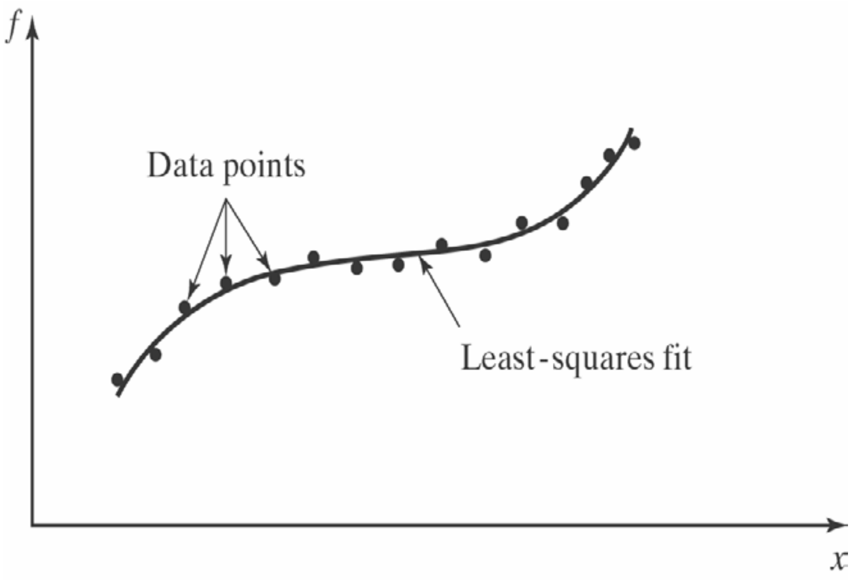
$$f_1(x) = 3.1199(x-1)^3 + 1.146(x-1)^2 + 3.2005(x-1) + 4.4366$$

- Interval [1.5,2.25]

$$f_2(x) = -2.5893(x-1.5)^3 + 5.8259(x-1.5)^2 + 6.6866(x-1.5) + 6.7134$$

Least Square Regression

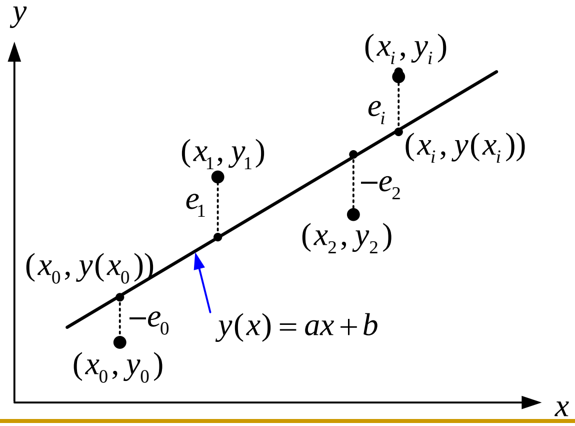
- Approach is useful when there are more data points than the number of unknown coefficients
- Four approaches
  - Linear Regression
  - Nonlinear Regression
  - Polynomial Regression
  - Multiple Regression

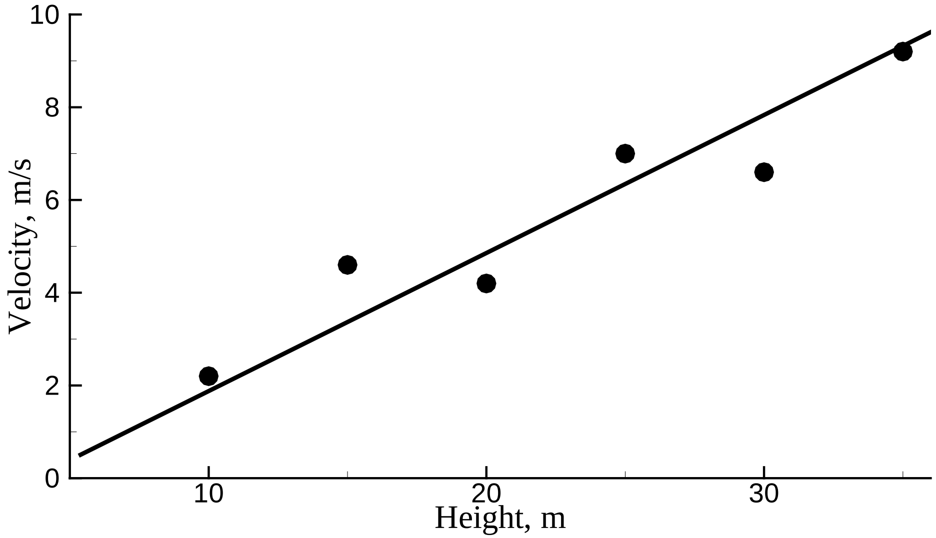


Linear Regression

- Linear regression is fitting the best straight line (linear equation) to the data

Height x (m)	Velocity y (m/s)
10	2.2
15	4.6
20	4.2
25	7
30	6.6
35	9.2





Linear Regression (continue)

- An approximate function $y(x) = ax + b$
- Error or deviation of the points (x_i, y_i) from the function $e_i = y_i - y(x_i) = y_i - (ax_i + b)$
- Minimize the magnitude of errors, least-square principle – sum of squares of errors

$$S = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left(y_i - ax_i - b\right)^2$$

- Determine the best values of *a* and *b* by

$$\frac{\partial S}{\partial b} = 0 = \sum_{i=1}^{n} 2\left(y_i - ax_i - b\right)(-1) \qquad \frac{\partial S}{\partial a} = 0 = \sum_{i=1}^{n} 2\left(y_i - ax_i - b\right)(-x_i)$$

$$a\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} b = \sum_{i=1}^{n} y_i \qquad a\sum_{i=1}^{n} x_i^2 + b\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} x_i y_i$$

Example

- In matrix form

$$\begin{bmatrix} \sum x_i & n \\ \sum x_i^2 & \sum x_i \end{bmatrix} \begin{Bmatrix} a \\ b \end{Bmatrix} = \begin{Bmatrix} \sum y_i \\ \sum x_i y_i \end{Bmatrix}$$

$$\begin{bmatrix} 135 & 6 \\ 3475 & 135 \end{bmatrix} \begin{Bmatrix} a \\ b \end{Bmatrix} = \begin{Bmatrix} 33.8 \\ 870 \end{Bmatrix}$$

$$\begin{Bmatrix} a \\ b \end{Bmatrix} = \begin{Bmatrix} 0.250286 \\ 0.00190476 \end{Bmatrix}$$

$$y = 0.250286x + 0.00190476$$

	Height x (m)	Velocity y (m/s)	x^2	xy
	10	2.2	100	22
	15	4.6	225	69
	20	4.2	400	84
	25	7	625	175
	30	6.6	900	198
	35	9.2	1225	322
Sum	**135**	**33.8**	**3475**	**870**

Accuracy of Linear Regression

- Correlation coefficient

$$r = \left(\frac{S_0 - S}{S_0} \right)^{1/2} \qquad S_0 = \sum_{i=1}^{n} (y_i - \bar{y})^2 \qquad \bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i \qquad S = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - a x_i - b)^2$$

- A perfect straight line is obtained if r = 1 or $S = 0$
- A worst straight line if r = 0 or $S = S_0$

	Height	Velocity			So	S
	x (m)	y (m/s)	x^2	xy	$(y-y_m)^2$	$(y-ax-b)^2$
	10	2.2	100	22	11.7878	0.0928816
	15	4.6	225	69	1.06778	0.712007
	20	4.2	400	84	2.05444	0.652258
	25	7	625	175	1.86778	0.549000
	30	6.6	900	198	0.934444	0.828982
	35	9.2	1225	322	12.7211	0.191919
Sum	**135**	**33.8**	**3475**	**870**	**30.4333**	**3.02705**
b =	0.00190476				r =	0.948965
a =	0.250286				r^2 =	0.900535

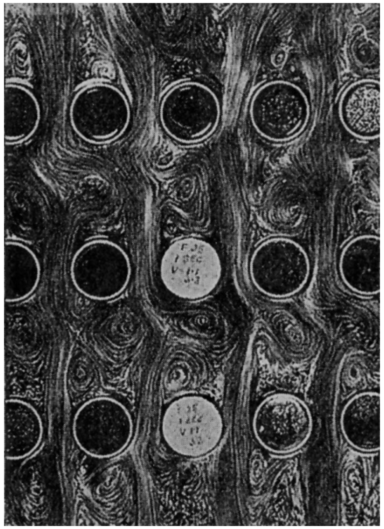
Still need to inspect the data plot even though r is close to 1

Example: heat transfer

The heat transfer coefficient (*h*) in a forced convection heat transfer in a cross flow past a cylinder is found to vary with the velocity of the fluid (*V*) flowing past the cylinder as follows:

V_i (m/s)	2	4	6	8
h_i (W/m^{2}K)	6.0	10.0	13.0	15.0

Fit a linear equation between *h* and *V* using the method of least squares.



Nonlinear Regression

- Fit with first degree polynomial

$$y = ax^b \qquad y = ae^{bx}$$

- By taking logarithms

$$\ln y = \ln a + b \ln x \qquad \ln y = \ln a + bx$$

- In matrix form

$$\begin{bmatrix} \sum \ln x_i & n \\ \sum (\ln x_i)^2 & \sum \ln x_i \end{bmatrix} \begin{Bmatrix} b \\ \ln a \end{Bmatrix} = \begin{Bmatrix} \sum \ln y_i \\ \sum \ln x_i \ln y_i \end{Bmatrix} \qquad \begin{bmatrix} \sum x_i & n \\ \sum x_i^2 & \sum x_i \end{bmatrix} \begin{Bmatrix} b \\ \ln a \end{Bmatrix} = \begin{Bmatrix} \sum \ln y_i \\ \sum x_i \ln y_i \end{Bmatrix}$$

Example

- Fit $y = ax^b$ to the following data

x	1	2	3	4	5
y	0.5	1.7	3.4	5.7	8.4

- Linearization by $\ln y = \ln a + b \ln x$

$$\begin{bmatrix} \sum \ln x_i & n \\ \sum (\ln x_i)^2 & \sum \ln x_i \end{bmatrix} \begin{Bmatrix} b \\ \ln a \end{Bmatrix} = \begin{Bmatrix} \sum \ln y_i \\ \sum \ln x_i \ln y_i \end{Bmatrix}$$

$$\begin{bmatrix} 4.7875 & 5 \\ 6.1995 & 4.7875 \end{bmatrix} \begin{Bmatrix} b \\ \ln a \end{Bmatrix} = \begin{Bmatrix} 4.93 \\ 7.5503 \end{Bmatrix}$$

$$\begin{Bmatrix} b \\ \ln a \end{Bmatrix} = \begin{Bmatrix} 1.7517 \\ -0.69128 \end{Bmatrix}$$

$$\ln y = -0.69128 + 1.7517 \ln x$$

$$a = \exp(-0.69128) = 0.50093$$

$$y = 0.50093 x^{1.7517}$$

x	y	ln x	ln y	(ln x)^2	(ln x)(ln y)
1	0.5	0	-0.69315	0	0
2	1.7	0.69315	0.53063	0.48045	0.36780
3	3.4	1.0986	1.2238	1.2069	1.3445
4	5.7	1.3863	1.7405	1.9218	2.4128
5	8.4	1.6094	2.1282	2.5903	3.4253
	Sum =	**4.7875**	**4.9300**	**6.1995**	**7.5503**
	b =	**1.7517**			
	ln a =	-0.69128	a =	**0.50093**	

Polynomial Regression

- Fit with second order degree polynomial or quadratic

$$y = a_0 + a_1 x + a_2 x^2 + e$$

$$e_i = y_i - a_0 - a_1 x_i - a_2 x_i^2$$

$$S = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left(y_i - a_0 - a_1 x_i - a_2 x_i^2 \right)^2$$

- Differentiating

$$\frac{\partial S}{\partial a_0} = 0 = -2 \sum_{i=1}^{n} \left(y_i - a_0 - a_1 x_i - a_2 x_i^2 \right) \qquad \frac{\partial S}{\partial a_1} = 0 = -2 \sum_{i=1}^{n} x_i \left(y_i - a_0 - a_1 x_i - a_2 x_i^2 \right)$$

$$\frac{\partial S}{\partial a_2} = 0 = -2 \sum_{i=1}^{n} x_i^2 \left(y_i - a_0 - a_1 x_i - a_2 x_i^2 \right)$$

Polynomial Regression (continue)

- In matrix form

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i^3 \\ \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i^3 & \sum_{i=1}^{n} x_i^4 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \\ \sum_{i=1}^{n} x_i^2 y_i \end{Bmatrix}$$

Example

- Fit a second order polynomial to the following data

x	0	1	2	3	4	5
y	2.1	7.7	13.6	27.2	40.9	61.1

- Solution

$$\begin{bmatrix} 6 & 15 & 55 \\ 15 & 55 & 225 \\ 22 & 225 & 979 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} 152.6 \\ 585.6 \\ 2488.8 \end{Bmatrix}$$

x	y	x^2	x^3	x^4	xy	x^2y
0	2.1	0	0	0	0	0
1	7.7	1	1	1	7.7	7.7
2	13.6	4	8	16	27.2	54.4
3	27.2	9	27	81	81.6	244.8
4	40.9	16	64	256	163.6	654.4
5	61.1	25	125	625	305.5	1527.5
15	**152.6**	**55**	**225**	**979**	**585.6**	**2488.8**

$$\begin{Bmatrix} a_0 \\ a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} 2.47857 \\ 2.35929 \\ 1.86071 \end{Bmatrix}$$

Gaussian elimination

$$y = 2.47857 + 2.35929x + 1.8607x^2$$

Example – electric motor

The vertical displacement of a large electric motor mounted on isolators due to the forced vibration caused by the rotating unbalance in the rotor is shown in the following table:

Speed of motor, v_i (rpm)	100	200	300	400	500
Displacement, d_i (mm)	0.10	0.35	0.70	0.40	0.35

You are required to develop a suitable polynomial relationship between *v* and *d* and fit a curve through data points.

Multiple Regression

- Extension of linear regression with two or more independent variables

$$y = a_0 + a_1 x_1 + a_2 x_2 + e$$

- For this 2-D case, the regression "line" becomes a "plane"

$$S = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left(y_i - a_0 - a_1 x_{1i} - a_2 x_{2i} \right)^2$$

$$\frac{\partial S}{\partial a_0} = 0 = -2 \sum_{i=1}^{n} \left(y_i - a_0 - a_1 x_{1i} - a_2 x_{2i} \right) \qquad \frac{\partial S}{\partial a_1} = 0 = -2 \sum_{i=1}^{n} x_{1i} \left(y_i - a_0 - a_1 x_{1i} - a_2 x_{2i} \right)$$

$$\frac{\partial S}{\partial a_2} = 0 = -2 \sum_{i=1}^{n} x_{2i} \left(y_i - a_0 - a_1 x_{1i} - a_2 x_{2i} \right)$$

Multiple Regression (continue)

- In matrix form

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{2i} \\ \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{1i}^2 & \sum_{i=1}^{n} x_{1i}x_{2i} \\ \sum_{i=1}^{n} x_{2i} & \sum_{i=1}^{n} x_{1i}x_{2i} & \sum_{i=1}^{n} x_{2i}^2 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_{1i}y_i \\ \sum_{i=1}^{n} x_{2i}y_i \end{Bmatrix}$$

Example

- Use linear multiple regression to the following data

x1	0	2	2.5	1	4	7
x2	0	1	2	3	6	2
y	5	10	9	0	3	27

- Solution

$$\begin{bmatrix} 6 & 16.5 & 14 \\ 16.5 & 76.25 & 48 \\ 14 & 48 & 54 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} 54 \\ 243.5 \\ 100 \end{Bmatrix}$$

$$\begin{Bmatrix} a_0 \\ a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} 5 \\ 4 \\ -3 \end{Bmatrix}$$

Gaussian elimination

x1	x2	y	x_1^2	x_2^2	x_1x_2	x_1y	x_2y
0	0	5	0	0	0	0	0
2	1	10	4	1	2	20	10
2.5	2	9	6.25	4	5	22.5	18
1	3	0	1	9	3	0	0
4	6	3	16	36	24	12	18
7	2	27	49	4	14	189	54
16.5	**14**	**54**	**76.25**	**54**	**48**	**243.5**	**100**