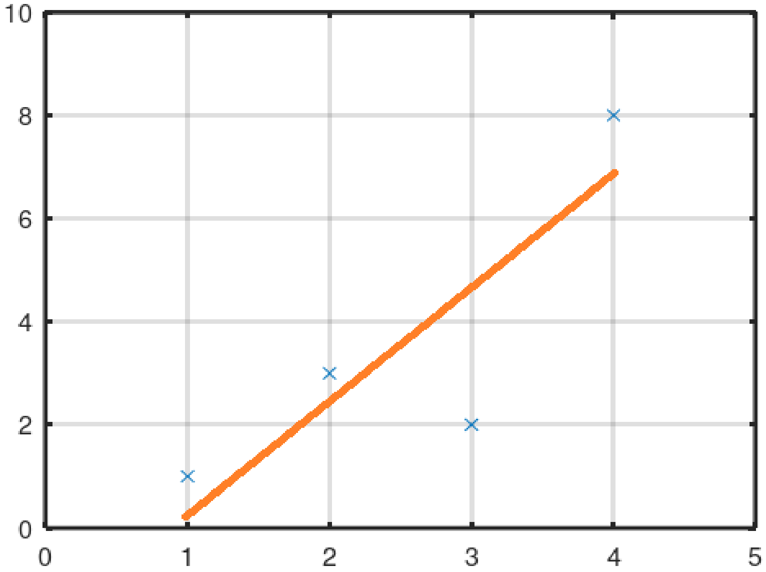
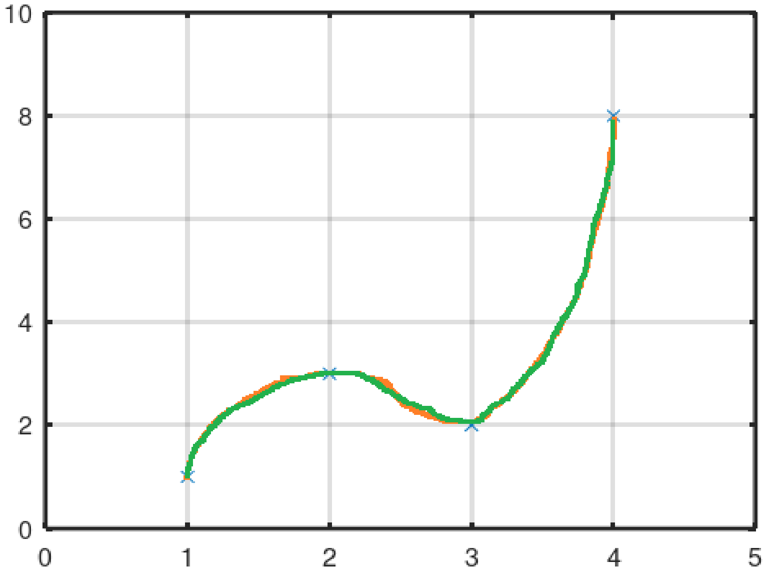
Numerical Modeling

Unit 2:Polynomial Approximations

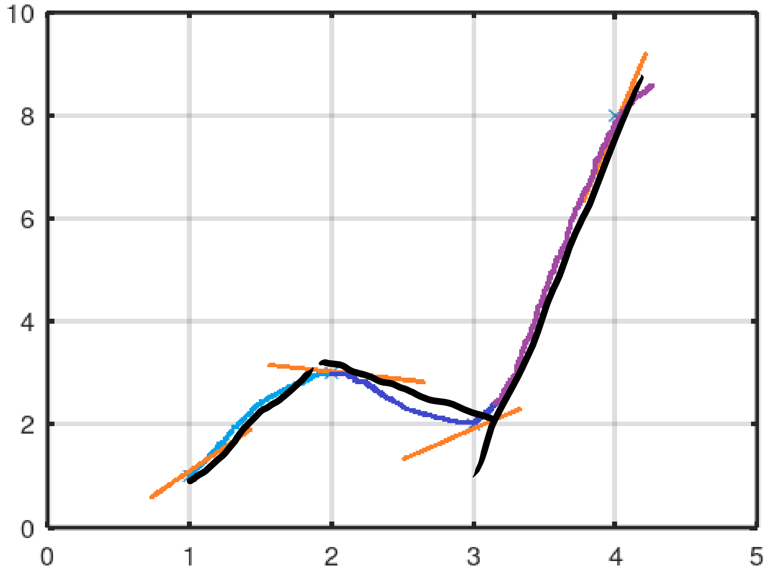
Polynomial Interpolation



Trend
- Least Squares



Interpolation with one polynomial
- Lagrange
- Newton



Interpolation using splines
- cubic

Trend - Least Squares

Trend / Linear approximation using the least squares method

Find best fit: $y = f(x) = ax + b$ for the set of points:

$(x_0, y_0)\ (x_1, y_1) \dots (x_n, y_n)$

find a & b so that 'err (e)' is minimal

Derive a and b on OneNote

i.e. $e = \sum (y_i - f(x_i))^2 = \sum (y_i - a \cdot x_i - b)^2$

$$\begin{cases} \dfrac{\partial e}{\partial a} = \sum 2 \cdot (y_i - a x_i - b) \cdot (-x_i) = 0 \\ \dfrac{\partial e}{\partial b} = \sum 2 (y_i - a x_i - b) \cdot (-1) = 0 \end{cases}$$

$$\begin{cases} b = \bar{y} - a\bar{x} \\ a = \dfrac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - \sum x_i{}^2} \end{cases}$$

$$b = \bar{y} - a\bar{x}$$

$$a = \frac{\bar{y}\sum x_i - \sum x_i y_i}{\bar{x}\sum x_i - \sum x_i^2}$$

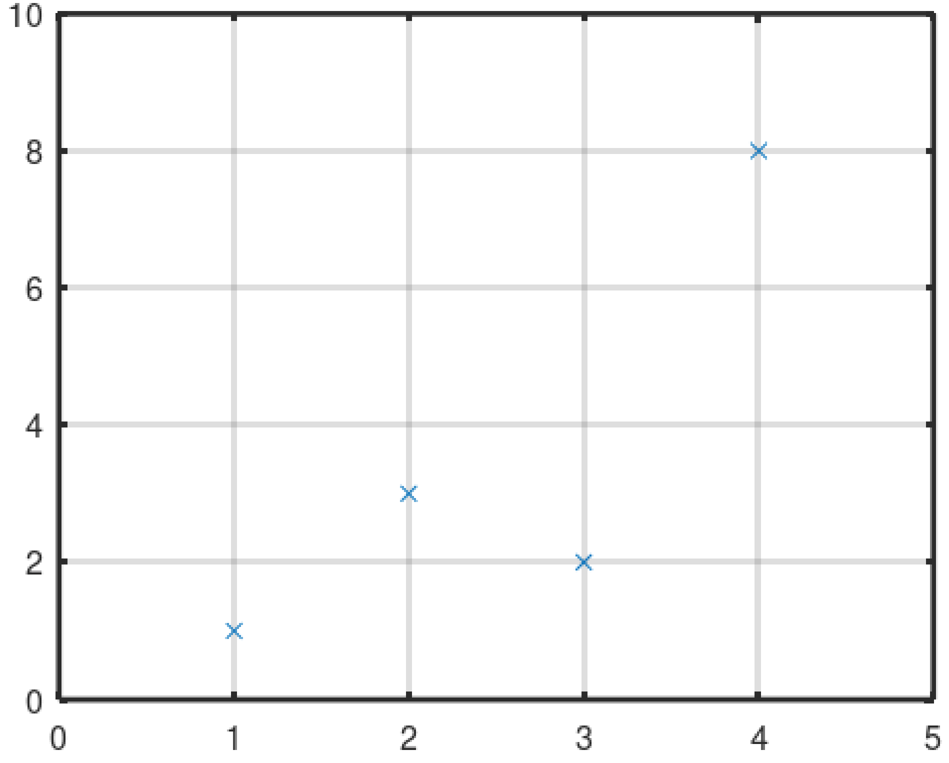
Code using matlab/octave

Polynomial Interpolation

Given the values	x0	x1	x2	...	xn		roots
	y0	y1	y2	...	yn		
	p0(x)	p2(x)	p3(x)		pn(x)		basis
	a0	a2	a3		an		coefficients
P(x)	a0*p0(x)	a1*p1(x)	a2*p2(x)		an*pn(x)		

$$P(x) = \sum_{i=0}^{n} a_i p_i(x)$$

```
x = [1, 2, 3, 4];
y = [1, 3, 2, 8];
```



Newton Interpolation

Newton Basis: $N_i(x) = (x - x_0) \ldots\ldots\ldots\ldots (x - x_{i-1}) = \prod_{j=0}^{i-1} (x - x_i)$

Polynomial form: $P(x) = \sum_{i=0}^{n} f(x_i) N_i(x)$

Expansion for n degree polynomial

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \ldots + a_n(x - x_0)(x - x_1)\ldots(x - x_{n-1})$$

Derive coefficients: *OneNote*

Lagrange Interpolation

y$= f(x) = ax + b$

$(x_0, y_0)\ (x_1, y_1)$

$f(x) = mx + b$

$$\begin{cases} y_0 = ax_0 + b \\ y_1 = ax_1 + b \end{cases}$$

$$y_0 - y_1 = a(x_0 - x_1)$$

$$a = \frac{y_0 - y_1}{x_0 - x_1}$$

$$y_0 = \frac{y_0 - y_1}{x_0 - x_1} x_0 + b \Rightarrow b = y_0 - \frac{y_0 - y_1}{x_0 - x_1} x_0$$

$$\frac{y_0(x_0 - x_1)}{x_0 - x_1} - \frac{x_0(y_0 - y_1)}{x_0 - x_1} = \frac{\cancel{y_0 x_0} - y_0 x_1 - \cancel{x_0 y_0} + x_0 y_1}{x_0 - x_1} = \frac{x_0 y_1 - x_1 y_0}{x_0 - x_1} = b$$

Lagrange Interpolation

$$y = \frac{y_0 - y_1}{x_0 - x_1} x$$

$$y = f(x) = \underbrace{\frac{x y_0}{x_0 - x_1} - \frac{x \boxed{y_1}}{x_0 - x_1}}_{} + \overbrace{\frac{x_0 \boxed{y_1}}{x_0 - x_1} - \frac{x_1 y_0}{x_0 - x_1}}^{b} =$$

$$y = \underbrace{\frac{x - x_1}{x_0 - x_1}}_{L_0} y_0 + \underbrace{\frac{x - x_0}{x_1 - x_0}}_{L_1} \boxed{y_1}$$

$$y_k = f(x_k)$$

$$L_0(x) = \frac{x - x_1}{x_0 - x_1}$$

$$L_0(x_0) = 1 \quad L_0(x_1) = 0$$

$$L_1(x) = \frac{x - x_0}{x_1 - x_0}$$

$$L_1(x_0) = 0 \quad L_1(x_1) = 1$$

$$L_{k\,=\,0}^{\;n}(x_i) = 0 \ \ if\ i \neq k$$

$$L_{k\,=\,0}^{\;n}(x_i) = 1 \ \ if\ i = k$$

$$L_k(x) = \frac{(x - x_0) \ldots\ldots\ldots\ldots (x - x_{k-1})(x - x_{k+1}) \ldots\ldots (x - x_n)}{(x_k - x_0) \ldots\ldots (x_k - x_{k-1})(x_k - x_{k+1}) \ldots\ldots (x_k - x_n)} = \prod_{\substack{i=0 \\ i \neq k}}^{n} \frac{(x - x_i)}{(x_k - x_i)}$$

$$P(x) = \sum_{k=0}^{n} \underbrace{f(x_k)}_{y_k} L_k(x)$$

Cubic Splines

Given a function f defined on [a,b] and a set of nodes a=$x_0 < x_1 < \cdots < x_n = b$, a **cubic spline interpolant** S for f is a function that satisfies the following conditions:

- S(x) is a cubic polynomial, denoted $S_j(x)$ on the subinterval $[x_j, x_{j+1}]$ for each j=0,1,….,n-1;
- $S(x_j) = f(x_j) for\ each\ j = 0,1, \ldots., n;$
- $S_{j+1}(x_{j+1}) = S_j(x_{j+1})\ for\ each\ j = 0,1, \ldots., n-2;$
- $S_{j+1}^1(x_{j+1}) = S_j^1(x_{j+1})\ for\ each\ j = 0,1, \ldots., n-2;$
- $S_{j+1}^2(x_{j+1}) = S_j^2(x_{j+1})\ for\ each\ j = 0,1, \ldots., n-2;$
- One of the following set of boundary conditions is satisfied:
  - $S^n(x_0) = S^n(x_n) = 0\ (\boldsymbol{free\ or\ natural\ baoundary})$
  - $S^1(x_0) = f^1(x_0)\ and\ S^1(x_n) = f^1(x_n)\ (\boldsymbol{clamped\ boundary})$