Polynomial and Data Fitting

Polynomial interpolation.

$$P_n(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$$

Given two points (x_0, y_0) and (x_1, y_1), we can have

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

a_0 and a_1 are the coefficients to be determined and can be obtained by solving above linear system of equations.

Given a set of points $(x_0, y_0), (x_1, y_1), \cdots (x_n, y_n)$ with $x_0 < x_1 < \cdots < x_n$. A one-dimensional interpolation is to construct a function f such that $f(x_i) = y_i$, $i = 0, 1, 2, \cdots, n$, and f assumes reasonable values for $x \in [x_0, x_n]$. Similar to the above formulation for two points case, we have the following system of equations

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & & \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

When n becomes large, the coefficient matrix is surprisingly ill-conditioned. This makes the linear system quite difficult to solve.

In MATLAB, there are various tools for fitting data using polynomial.

Polynomial evaluation: $y = \text{polyval}(P, X)$, X can be a matrix.

Example $P(x) = x^2 - 3x + 1$, evaluate $p(x)$ at $x = 2, 3,$ and 5

```
p = [1 -3 1];
polyval (P, [2;3;5])
```

ans: ⇒

```
-1
 1
11
```

Data fitting: Given a set of data $\{(x_i, y_i)\}_{i=1}^{m}$, find a polynomial that fits the data

$p = \text{polyfit}(x, y, n)$ find the coefficients of a polynomial $P(x)$ of degree n that fits the data in a least squares sense. $(n \le m)$

Example $f(x) = \dfrac{1}{x + (1-x)^2}$ evaluated at 20 equally spaced points on the interval $[-2, 2]$

```
x = linspace (-2, 2, 20);
y = 1./(x + (1-x).^2);
P = polyfit ( x, y, 3);
plot (x, y, '*', x, polyval (P, x), '--')
```

any number less than or equal to 20

'spline' function can be used if the exact data interpolation is required.

```
x = linspace(-2, 2, 20);
y = 1./(x+(1-x).^2);
xx = linspace(-2, 2, 60)
yy = spline(x, y, xx);
plot(x, y, '*', xx, yy, '--')
```

Another 1-D data interpolation using MATLAB:

```
interp1(x, y, xi, method, 'extrap')
```

x, y : input data

xi : data ~~to~~ be evaluated.

method: 'nearest', 'linear', 'spline', 'pchip', 'cubic'.

Example.

```
x = 0:10;
y = sin(x);
xi = 0:0.25:10;
yi = interp1(x,y,xi, 'nearest')
plot(x,y,'*', xi,yi)
```

or others