Numerical Optimization 2015 Homework 1 104062544 劉康軍

1.

Step1: Taylor expansion series at x_0:

$$f(x) \approx f(x_k) + f'(x_k)(x - x_k) + \frac{f''(x_k)}{2!}(x - x_k)^2$$

Step2: Find the optimal point:

$$\Rightarrow Taylor\ expansion\ at\ x_0 = \frac{f^{'} \cdot (x_k)}{2} X^2 + (f'(x_k) - f''(x_k)x_k)x + \cdots$$

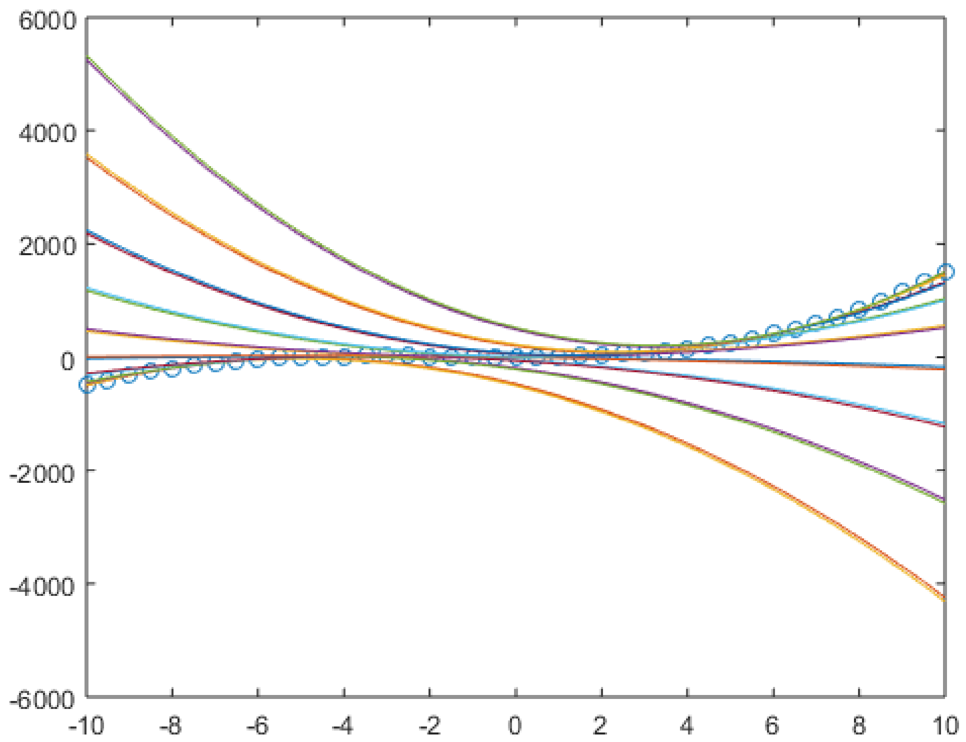
$$\Rightarrow x = \frac{x_k f''(x_k) - f'(x_k)}{f''(x_k)} = x_k - \frac{f'(x_k)}{f''(x_k)}$$

Step3: $\Rightarrow x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)} \Rightarrow x_{k+1} - x_k = -\frac{f'(x_k)}{f''(x_k)} \blacksquare \alpha$

2.

(1) When α is small enough and the target function is smooth, the approximate function is almost the same to Taylor expansion. We can see it from the figure below.

The target function: $\chi^3 + 5x^2 + x + 5$, $\alpha = 2$ (為了看出些微不同，故 alpha



取高了點)

$So\ we\ can$ know that in this situation, the interpolation version method will fail as Newton method fail. From Wiki pedia, a good fail example is "$\frac{3}{4}x^{\frac{4}{3}}, x > 0$".

Then, the iteration of its derivative will be $x_{k+1} = -2x_k$ which diverges as k goes to infinity.

(我提供了一份 my_interpoate.m 可以跑這個例子，當初始為 1，最終結果會是 NaN)

(2) First, when α is small by(1), the numerical property like convergence order is almost the same to Newton's method's. Second, this method doesn't need a cloed form of the derivative of the function which is better than Newton's method. Third, the cost of time of this method may be more than Newton's method. Since when x is not high dimension variable , the iteration of Newton's method doesn't need to calculate the inverse of matrices. However this method needs. But when it needs more discussion when x comes to high dimensions.

(3) $\begin{bmatrix} x_1^3 & x_1^2 & x_1 & 1 \\ 3x_1^2 & 2x_1 & 1 & 0 \\ x_2^3 & x_2^2 & x_2 & 1 \\ 3x_2^2 & 2x_2 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} f(x_1) \\ f'(x_1) \\ f(x_2) \\ f'(x_2) \end{bmatrix}$

3.

(1) $\nabla f(x,y) = (x, 9y),\ \ H = \begin{pmatrix} 1 & 0 \\ 0 & 9 \end{pmatrix}$

(2) Tolerance is 0.0005

Steepest descent method: (0.0340, -0.0038) with 25 iterations

Newton's method: (0,0) with 2 iteration

(3) Solid line: Newton's method

Dashed line: Steepest descend