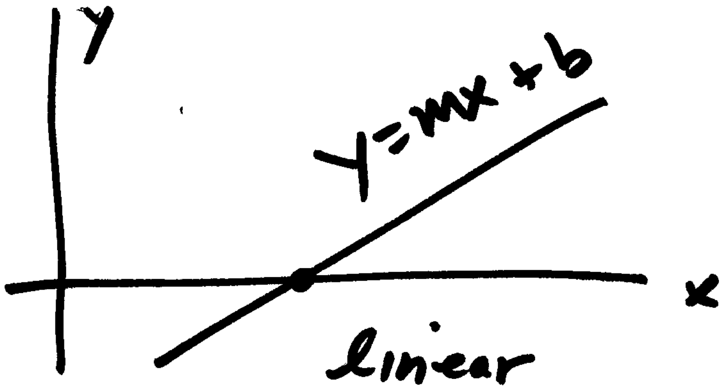
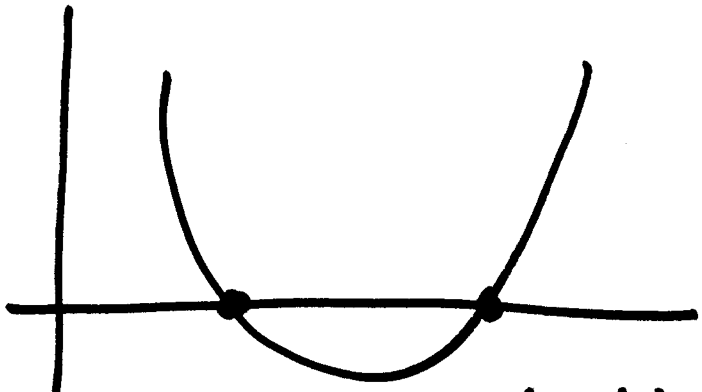
finding roots of equation

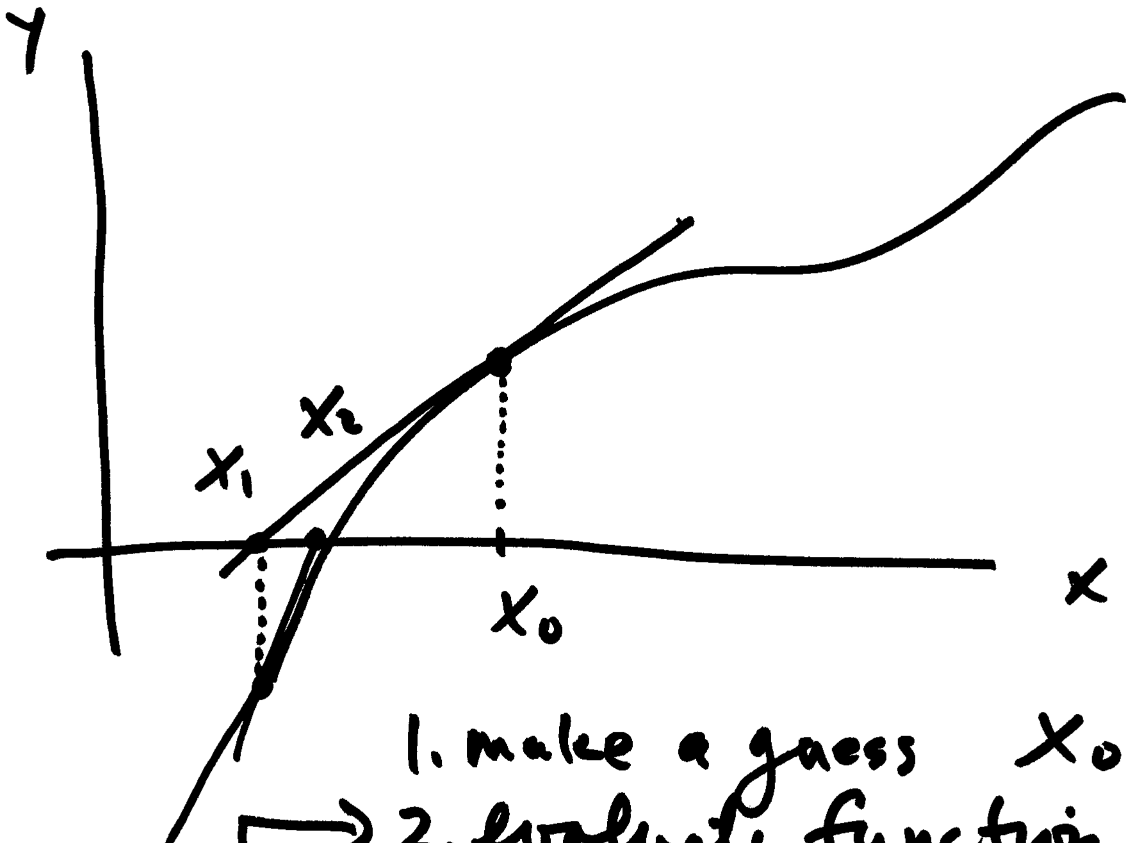


$$y = 0 = mx + b$$
$$mx = -b$$
$$x = -b/m$$



quadratic

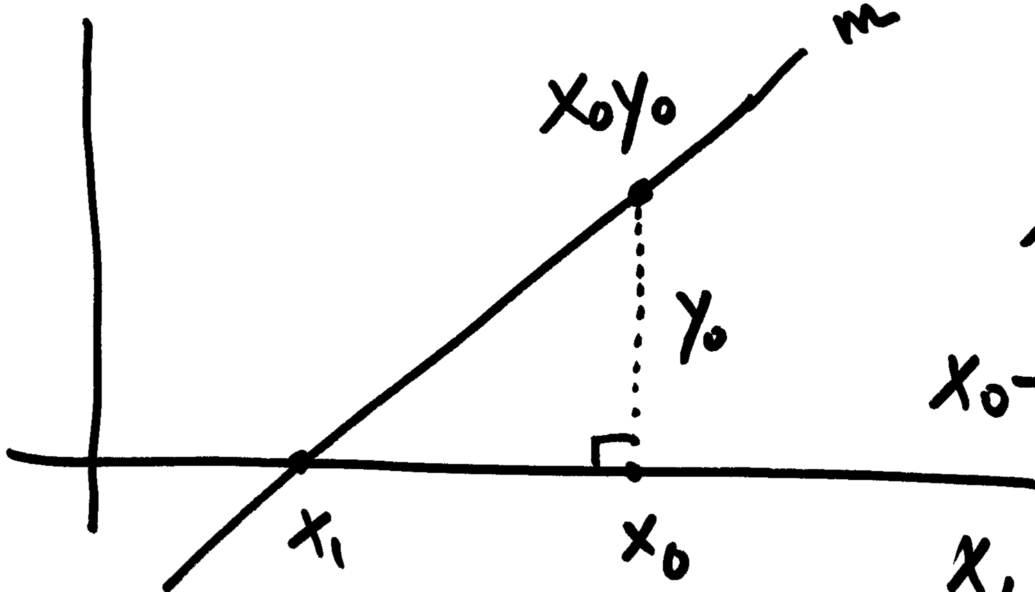
$$y = Ax^2 + Bx + c$$
$$0 = Ax^2 + Bx + C$$
$$x = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$



any nonlinear function

1. make a guess x_0
2. evaluate function + derivative
3. find root of that linear approx
   that becomes new approximation

repeat until Δx is small



$$\frac{y_0}{x_0 - x_1} = m$$

$$x_0 - x_1 = y_0/m$$

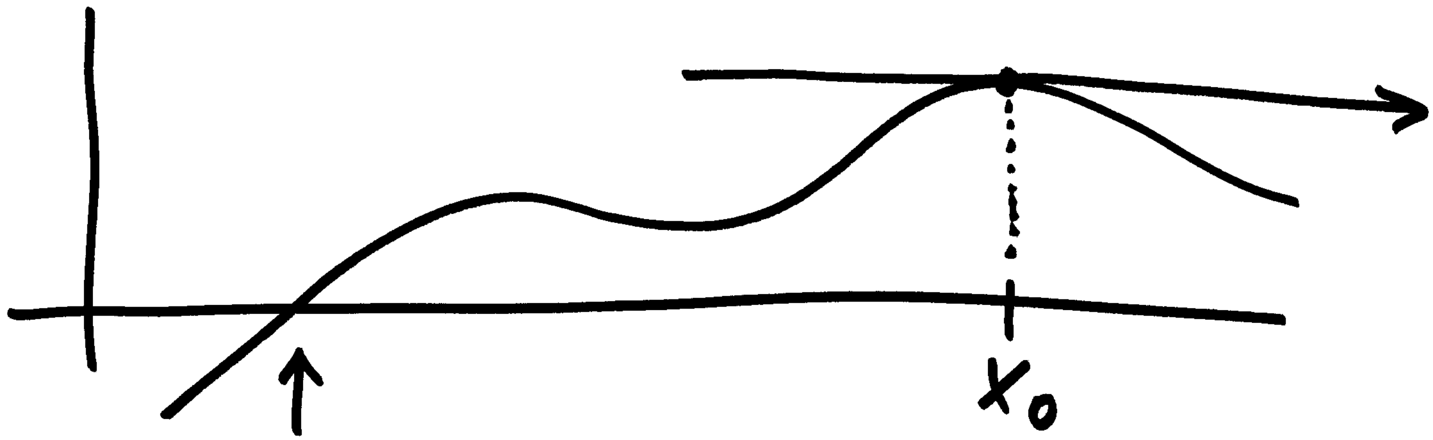
$$x_1 = x_0 - y_0/m$$

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

$$\boxed{x_{i+1} = x_i + \underbrace{\frac{-f(x_i)}{f'(x_i)}}_{\Delta x}}$$

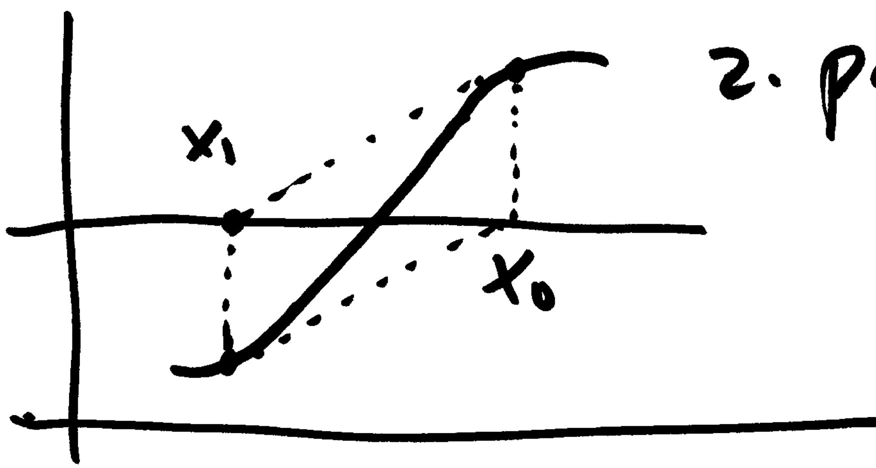
iteration formula for Newton Iteration

$$x_{i+1} = x_i + \Delta x$$



1. poor initial approximation, or badly behaved function

symptom : DIVERGE



2. pathological function

1D newton method : 1 NL eq. / 1 unknown

nD " " : n " / n "

$$\left.\begin{aligned} f_1(x_1, x_2, \ldots, x_n) &= 0 \\ f_2(x_1, x_2, \ldots, x_n) &= 0 \\ &\vdots \\ f_n(x_1, x_2, \ldots, x_n) &= 0 \end{aligned}\right\}$$

how to solve this ?

equivalent of "straight line approximation"

Taylor Series Approximation

Taylor Series

1D $$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \cdots$$

(truncate as shown)

$$f(x) \approx f(a) + f'(a)(x-a)$$

"straight line approximation"

nD $$f(x_1, x_2, \ldots, x_n) \approx f(x_1^0, x_2^0, \ldots, x_n^0) + \frac{\partial f}{\partial x_1}(x_1 - x_1^0) + \frac{\partial f}{\partial x_2}(x_2 - x_2^0) + \cdots + \frac{\partial f}{\partial x_n}(x_n - x_n^0) =$$

$$f^0 + \frac{\partial f}{\partial x_1}\Delta x_1 + \frac{\partial f}{\partial x_2}\Delta x_2 + \cdots \frac{\partial f}{\partial x_n}\Delta x_n$$

$$f(x_1, x_2, \dots, x_n) \approx f^0 + \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \cdots & \frac{\partial f}{\partial x_n} \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \vdots \\ \Delta x_n \end{bmatrix}$$

extend to n eq. n unknowns

$$f_1(x_1 \cdots x_n) \approx f_1^0 + \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \end{bmatrix} \begin{pmatrix} \Delta x_1 \\ \vdots \\ \Delta x_n \end{pmatrix}$$

$$f_2(x_1 \cdots x_n) = f_2^0 + \begin{bmatrix} \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \end{bmatrix} \begin{pmatrix} \Delta x_1 \\ \vdots \\ \Delta x_n \end{pmatrix}$$

$$\vdots$$

$$f_n(x_1 \cdots x_n) = f_n^0 + \begin{pmatrix} \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix} \begin{pmatrix} \Delta x_1 \\ \vdots \\ \Delta x_n \end{pmatrix}$$

$$\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix} \approx \begin{bmatrix} f_1^0 \\ f_2^0 \\ \vdots \\ f_n^0 \end{bmatrix} + \begin{bmatrix} \partial f_1/\partial x_1 & \partial f_1/\partial x_2 & \cdots & \partial f_1/\partial x_n \\ \partial f_2/\partial x_1 & \partial f_2/\partial x_2 & \cdots & \partial f_2/\partial x_n \\ \vdots & & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \vdots \\ \Delta x_n \end{bmatrix}$$

$$\underset{(n,1)}{F} \approx \underset{(n,1)}{F^0} + \underset{(n,n)}{\partial F/\partial X} \cdot \underset{(n,1)}{\Delta X}$$

$J =$ jacobian

if we have written original equations $f(x \cdots x) = 0$

$$F \approx F^0 + J\Delta x = 0$$

solving: $J \cdot \Delta x = -F^0$

$$\Delta x = J^{-1}(-F^0)$$

rewrite in iteration formula

$$x_{i+1} = x_i + \Delta x$$

$$\boxed{x_{i+1} = x_i + J^{-1}(-F^0)}$$

nD version of iteration formula

$n,1 \qquad n,1 \qquad n,n \cdot n,1$

$$\boxed{x_{i+1} = x_i + \frac{-f(x_i)}{f'(x_i)}}$$

1D iteration formula

1D & nD formulae look very similar

how do LS techniques fit into Newton process ?

1. finding initial approximations of unknowns
2. linearize <u>equations</u> via Taylor Series (truncated)
3. classical Newton : solve uniquely $n \times n$ for ΔX

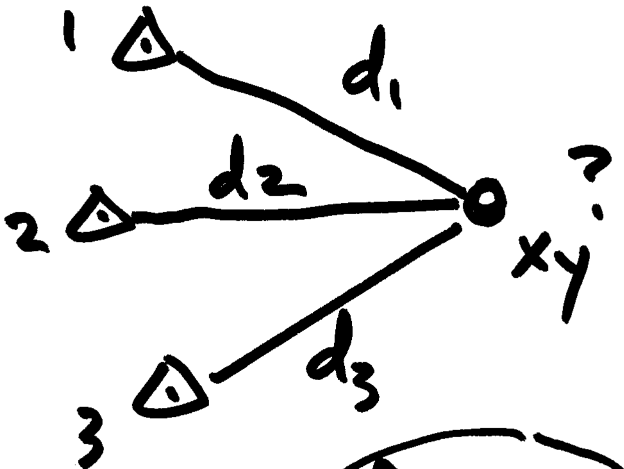
   LS newton : solve overdetermined (redundancy) for $\underline{\underline{\Delta X}}$
4. $X_{i+1} = X_i + \Delta X$

flow chart for ~~[illegible]~~ NL - LS

- analyze problem n, n_o, r
- choose I/O or O/O
- linearize (J)
- solve LS problem for Δ's
- refine approx

repeat until Δ's are "small"

(we imbed a linear LS solution inside of the newton iteration loop)



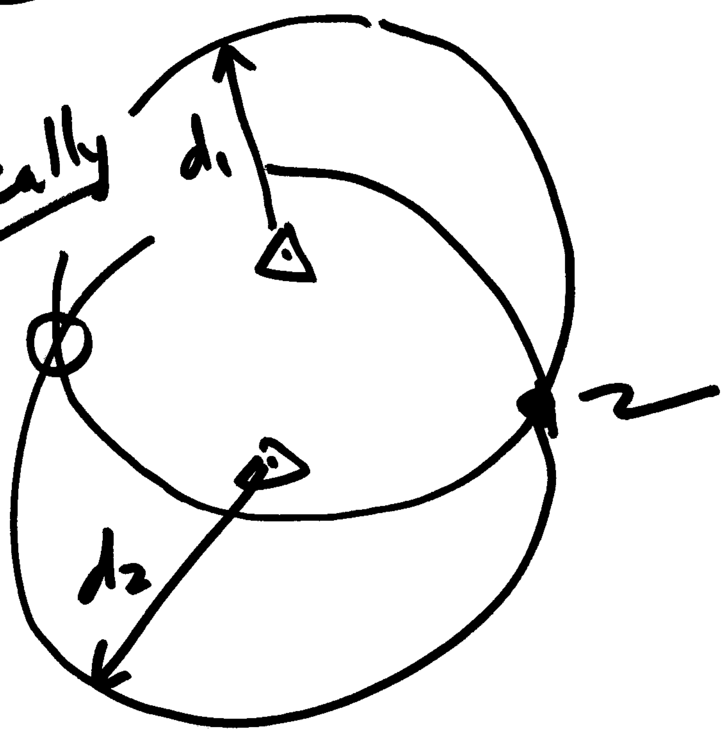
$x_1 y_1$, $x_2 y_2$, $x_3 y_3$

$n = 3$ indirect obs

$n_0 = 2$ $u = n_0 = 2$

$r = 1$ x, y

graphically



Example

$$\hat{l}_i = \left[(x - x_i)^2 + (y - y_i)^2\right]^{1/2}$$

$$F_i = 0 : \quad \hat{l}_i - \left[(x - x_i)^2 + (y - y_i)^2\right]^{1/2}$$

$$F_1 = \hat{d}_1 - \left[(x - x_1)^2 + (y - y_1)^2\right]^{1/2} = 0$$

$$F_2 = \hat{d}_2 - \left[(x - x_2)^2 + (y - y_2)^2\right)^{1/2} = 0$$

$$F_3 = \hat{d}_3 - \left[(x - x_3)^2 + (y - y_3)^2\right]^{1/2} = 0$$

as before we need B, f, $W = I_3$

$$\underset{(3,2)}{B} = \frac{\partial F}{\partial x} \qquad J \Delta x = \underline{\underline{F}}^0$$

$$B = \begin{bmatrix} \partial f_1/\partial x & \partial f_1/\partial y \\ \partial f_2/\partial x & \partial f_2/\partial y \\ \partial f_3/\partial x & \partial f_3/\partial y \end{bmatrix} \quad f = \begin{bmatrix} -F_1^\circ \\ -F_2^\circ \\ -F_3^\circ \end{bmatrix}, \quad W = I_3$$

$$\Delta = \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \left(B^T W B\right)^{-1} B^T W f$$

refine $\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \rightarrow$ new $\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$

will derive the linearized condition equations next time.