Non-linear Least-Square Problem

No

Date

$$\min_x F(x) = \frac{1}{2}\|f(x)\|_2^2 \qquad f(x): \mathbb{R}^n \to \mathbb{R}$$

Objective: get x s.t. $F(x)$ has min

→ if easy → analytical form

↓ through

$\frac{dF}{dx} = 0$

↓

else hard. $\frac{dF}{dx}$ not easily solved.

algo

give initial value x_0

while true

  at $i = k$

    search Δx_k

    get $\|f(x_k + \Delta x_k)\|_2^2$

    if $\Delta x_k < b$

      return

  else

      continue

    $x_{k+1} = x_k + \Delta x_k$

How to get this?

@ x_k, $i = k$, get Δx_k

do Taylor Expansion:

Jacobian Hessian

$$F(x_k + \Delta x_k) \approx F(x_k) + J(x_k)^T \Delta x_k + \frac{1}{2}\Delta x_k^T H(x_k) \Delta x_k$$

1^{st} order method $F(x_k+\Delta x_k) \approx F(x_k) + J(x_k)^T \Delta x_k$	$\Delta x^* = -J(x_k)$ $\Delta x = -J(x_k) \cdot \lambda$ — length perameter steepest descent method
2^{nd} order method $F(x_k+\Delta x_k) \approx F(x_k) + J(x_k)^T \Delta x + \frac{1}{2}\Delta x_k^T H(x_k) \Delta x_k$	$\Delta x^* = \arg\min \underbrace{\left(F(x) + J(x)^T \Delta x + \frac{1}{2}\Delta x^T H \Delta x \right)}_{P(x)}$ $\frac{dP(x)}{d\Delta x} = J + H\Delta x = 0$ $H\Delta x = -J$ $\Delta x = -H^{-1}J$

$$x_{k+1} = x_k + \Delta x_k$$

1^{st} $\quad x_{k+1} = x_k - J(x_k) \cdot \lambda$

2^{nd} $\quad x_{k+1} = x_k - H^{-1}J$

Gauss-Newton Method

No.

Date

$$f(x) = \frac{1}{2}\sum_{j=1}^{m} r_j(x)^2 = \frac{1}{2}\sum_{j=1}^{m} \|r(x)\|_2^2$$

$$\begin{cases} r_j(x) = \phi(x; t_j) - y_j \\ j = 1, 2, 3 \dots m \\ r(x) = [r_1(x), r_2(x) \dots r_m(x)]^T \end{cases}$$

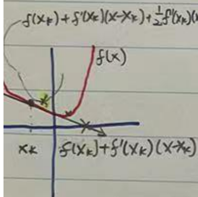
$$\nabla f(x_1, x_2) = \begin{pmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{pmatrix}$$

$$H(f(x_1, x_2)) = \begin{pmatrix} \frac{\partial f}{\partial x_1^2} & \frac{\partial f}{\partial x_1 \partial x_2} \\ \frac{\partial f}{\partial x_2 \partial x_1} & \frac{\partial f}{\partial x_2^2} \end{pmatrix}$$

X is the objective parameter (e.g. pose) $X \in R^n$

r is the residual result calculated from X (e.g. reprojection u, v) $m = 1 \& 2$

recall Newton's method



$$X_{k+1} = X_k - \underbrace{[\nabla^2 f(X_k)]^{-1} \nabla f(X_k)}_{\text{search direction might not be descent}}$$

for non-linear

to descent :

$$X_{k+1} = X_k - \boxed{H^{-1}} J_r(X_k)^T r(X_k)$$

$$-\nabla f(x^k)^T [\nabla^2 f(x^k)]^{-1} \nabla f(x^k) < 0$$

thus Gauss-Newton Method $f = \frac{1}{2}\sum_{j=1}^{m} r_j(x)^2$

w/ $\nabla f = J_r^T r \qquad \frac{\partial f}{\partial x_i} = \sum_{j=1}^{m} \frac{\partial r_j}{\partial x_i} r_j$

$$\nabla^2 f = J_r^T J_r + \sum_{i=1}^{m} r_i \nabla^2 r_i$$

$$= J_r^T J_r + \cancel{Q}$$

neglect this

$$\nabla^2 f \approx J_r^T J_r$$

$$J(x) = \begin{bmatrix} \frac{\partial r_j}{\partial x_i} \end{bmatrix} \begin{matrix} j = 1 \dots m \\ i = 1 \dots n \end{matrix} = \begin{bmatrix} \nabla r_1(x)^T \\ \nabla r_2(x)^T \\ \vdots \\ \nabla r_m(x)^T \end{bmatrix}$$

$$\therefore X_{k+1} = X_k - \underbrace{\boxed{[J_r(X_k)^T J_r(X_k)]^{-1} J_r(X_k)^T r(X_k)}}_{\Delta X}$$

In sum : Gauss-Newton Approximate H w/ $J^T J$

sup: 1^{st} order

$$\nabla f(x) = \nabla \frac{1}{2}\sum_{j=1}^{m} (r_j(x))^2 = \sum_{j=1}^{m} r_j(x) \nabla r_j(x) = J(x)^T r(x)$$

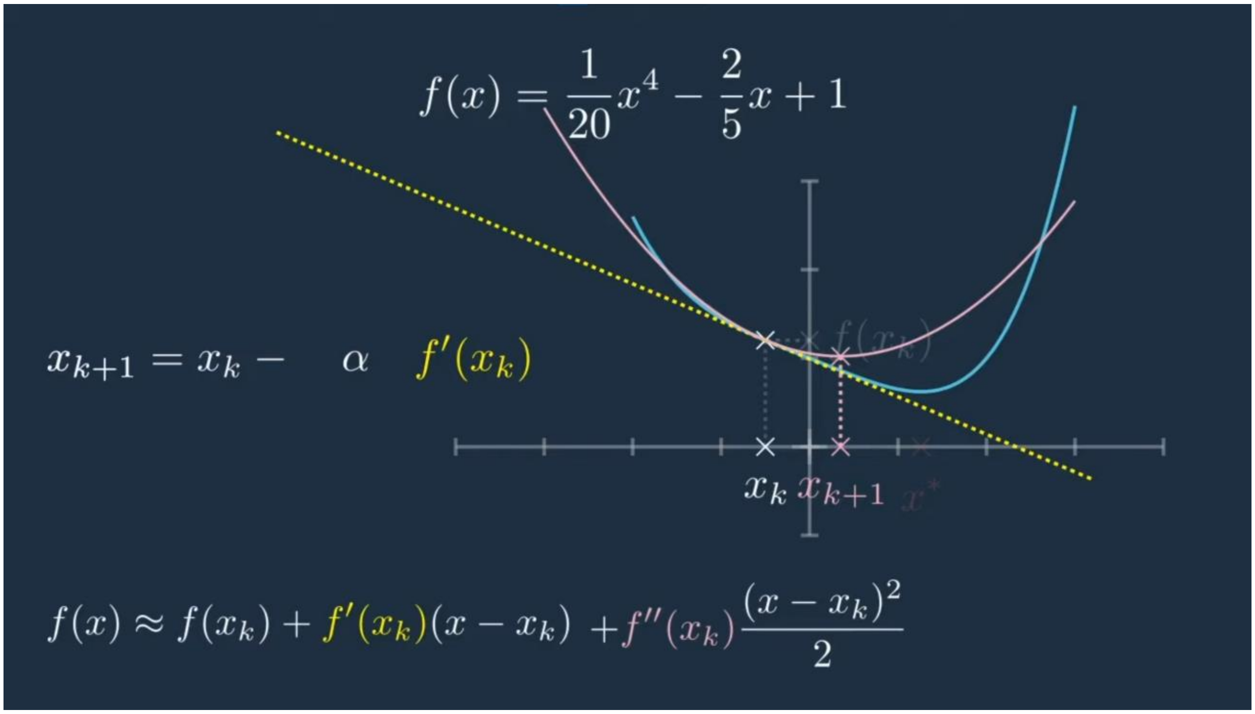
2^{nd} order

$$\nabla^2 f(x) = \nabla \sum_{j=1}^{m} r_j(x) \nabla r_j(x)$$

$$= \sum_{j=1}^{m} \nabla r_j(x)^T \nabla r_j(x) + \sum_{j=1}^{m} r_j(x) \nabla^2 r_j(x)$$

$$= J(x)^T J(x) + \sum_{j=1}^{m} r_j(x) \nabla^2 r_j(x)$$

$$= J(x)^T J(x) + Q(x)$$


$f(x) = \frac{1}{20}x^4 - \frac{2}{5}x + 1$
$x_{k+1} = x_k - \quad \alpha \quad f'(x_k)$
$f(x_k)$
x_k
x_{k+1}
x^*
$f(x) \approx f(x_k) + f'(x_k)(x - x_k) + f''(x_k)\frac{(x - x_k)^2}{2}$

2 Gauss-Newton method

The Gauss-Newton method is a simplification or approximation of the Newton method that applies to functions f of the form (1). Differentiating (1) with respect to x_j gives

$$\frac{\partial f}{\partial x_j} = \sum_{i=1}^{m} \frac{\partial r_i}{\partial x_j} r_i,$$

$f \to r_1$
$\to r_2$
$\vdots$
$\to r_n$

and so the gradient of f is

$$\nabla f = J_r^T \mathbf{r},$$

where $\mathbf{r} = [r_1, \ldots, r_m]^T$ and $J_r \in \mathbb{R}^{m,n}$ is the Jacobian of $\mathbf{r}$,

$$J_r = \left[\frac{\partial r_i}{\partial x_j}\right]_{i=1,\ldots,m,j=1,\ldots,n}.$$

$$\nabla f = \begin{bmatrix} \nabla r_1(x)^T \\ \nabla r_2(x)^T \\ \vdots \\ \nabla r_m(x)^T \end{bmatrix} = \left[\frac{\partial r_j}{\partial x_i}\right] \quad \begin{matrix} j=1\ldots m \\ i=1\ldots n \end{matrix}$$

$$= J_r^T r(x)$$

Differentiating again, with respect to x_k, gives

$$\frac{\partial^2 f}{\partial x_j \partial x_k} = \sum_{i=1}^{m} \left(\frac{\partial r_i}{\partial x_j} \frac{\partial r_i}{\partial x_k} + r_i \frac{\partial^2 r_i}{\partial x_j \partial x_k} \right),$$

and so the Hessian of f is

$$\nabla^2 f = J_r^T J_r + Q,$$

where

$$Q = \sum_{i=1}^{m} r_i \nabla^2 r_i.$$

The Gauss-Newton method is the result of neglecting the term Q, i.e., making the approximation

$$\nabla^2 f \approx J_r^T J_r. \tag{3}$$

Thus the Gauss-Newton iteration is

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - (J_r(\mathbf{x}^{(k)})^T J_r(\mathbf{x}^{(k)}))^{-1} J_r(\mathbf{x}^{(k)})^T \mathbf{r}(\mathbf{x}^{(k)}).$$

In general the Gauss-Newton method will not converge quadratically but if the elements of Q are small as we approach a minimum, we can expect fast convergence. This will be the case if either the r_i or their second order partial derivatives

$$\frac{\partial^2 r_i}{\partial x_j \partial x_k}$$

$$X_{k+1} = X_k + \Delta X_k$$

$$\Delta X = -\left[J(X_k)^T J(X_k)\right]^{-1} J(X_k)^T r(X_k)$$

$$\left[J(X_k)^T J(X_k)\right] \Delta X = J(X_k)^T \cdot \underbrace{r(X_k)}_{e.}$$

$$A \quad x \quad = \quad b$$

$:=$