$$F(x+\Delta x) = F(x) + \Delta x \cdot \frac{dF}{dx} + \frac{1}{2}(\Delta x)^2 \frac{d^2F}{dx^2} + \cdots$$

single variable version.

$$\boxed{H_{ij} = \frac{\partial^2 F}{\partial x_i \partial x_j}}$$

If $X = (x_1, \cdots, x_m)$.

Hessian

$$F(x+\Delta x) = F(x) + (\Delta x)^T \cdot \nabla F(x) + \frac{1}{2}(\Delta x)^T \boxed{H} (\Delta x).$$

$$\nabla F(x) = \begin{bmatrix} \frac{\partial F}{\partial x_1} \\ \vdots \\ \frac{\partial F}{\partial x_m} \end{bmatrix} \quad [\cdots\cdots] \quad [\quad].$$

$$H = \begin{bmatrix} \frac{\partial^2 F}{\partial x_1 \partial x_1} & \frac{\partial^2 F}{\partial x_1 \partial x_2} & \cdots & \\ \frac{\partial^2 F}{\partial x_2 \partial x_1} & & \ddots & \\ \vdots & & & \frac{\partial^2 F}{\partial x_m \partial x_m} \end{bmatrix}.$$

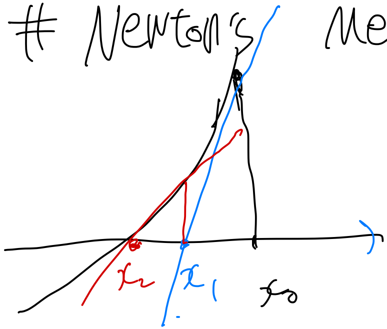
$$f(x,y) = 3x^2 + 2xy + 5y^2 \Rightarrow [x \quad y]\begin{bmatrix} 3 & 1 \\ 1 & 5 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}$$

$$f_x = 6x + 2y$$

$$f_{xx} = 6$$

$$H = \begin{bmatrix} f_{xx} & f_{xy} \\ f_{yx} & f_{yy} \end{bmatrix}$$

Newton's Method (solve $f(x)=0$)



$$\underbrace{f(x_k+\Delta x)}_{\to 0} = f(x_k) + f'(x_k)\cdot\Delta x.$$

$(\Delta x = x_{k+1} - x_k)$.

$$0 = f(x_k) + f'(x_k)(x_{k+1} - x_k).$$

$$\Rightarrow \boxed{x_{k+1} = x_k - f'(x_k)^{-1}\cdot f(x_k).}$$

$$\left(x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}\right)$$

$f = (f_1, f_2, f_3)$.

$$\Rightarrow \boxed{x_{k+1} = x_k - J(x_k)^{-1}\cdot f(x_k).}$$

$$\left(\text{Jacobian } J_{jk} = \frac{\partial f_j}{\partial x_k}.\right)$$

ex) $f(x) = x^2 - 9$.

$$x_{k+1} = x_k - J(x_k)^{-1} \cdot f(x_k).$$

$$J(x_k) = 2x_k \quad , \quad f(x_k) = x_k^2 - 9.$$

$$x_{k+1} = x_k - \frac{1}{2x_k}(x_k^2 - 9),$$

Convergence rate?

$$(x_{k+1} - 3) = x_k - \frac{1}{2x_k}(x_k^2 - 9) - 3.$$

$$= \frac{1}{2x_k}\{2x_k^2 - x_k^2 + 9 - 6x_k\}.$$

$$= \frac{1}{2x_k}(x_k - 3)^2.$$

$$\Rightarrow (x_{k+1} - 3) = \frac{1}{2x_k}(x_k - 3)^2.$$

→ very powerful method.

minimize $F(x)$. ($\approx$ solving $\nabla F = 0$).

Convergence rate.

I Steepest Descent.

$$x_{k+1} = x_k - s_k \cdot \nabla F.$$

$\leftarrow$ linear

II Newton's Method.

$\leftarrow$ quadratic.

cost?

$\left(x_{k+1} = x_k - J^{-1}(x_k) \cdot f(x_k) \right.$ now $f \to \nabla f$)

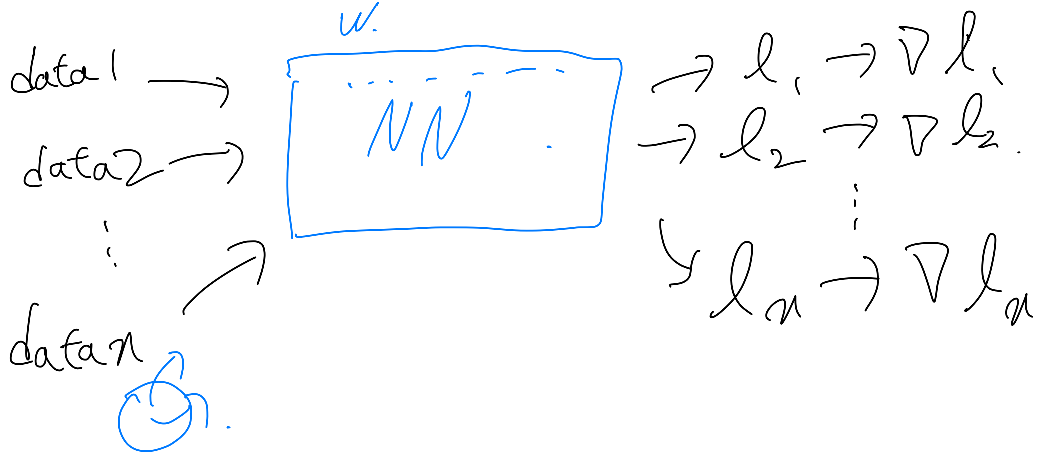
$\leftarrow$ cheap.

$J_{jk} = \frac{\partial f_j}{\partial x_k}$. $\nabla f_j = \frac{\partial f}{\partial x_j}$. $\to$ $J_{jk} = \frac{\partial^2 f}{\partial x_k \partial x_j}$.

$\leftarrow$ expensive (H).

$$x_{k+1} = x_k - H^{-1}(\nabla F).$$

Stochastic Gradient descent. (SGD).



$$\cancel{\nabla L = \frac{1}{n}\sum_{i=1}^{n} \nabla l_i}$$

$$\nabla L = \nabla l_k.$$

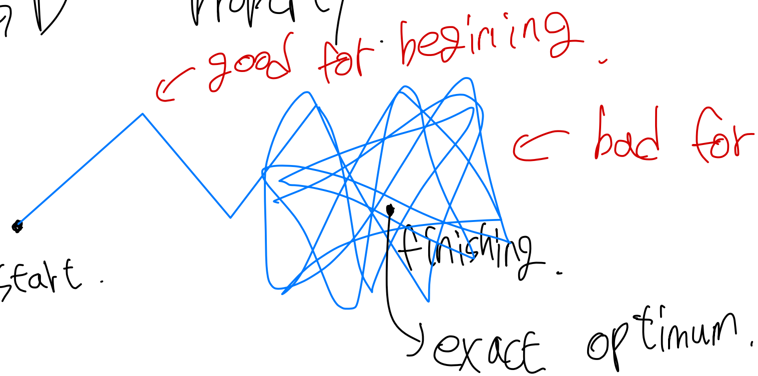
How pick data k? (for k=0, 1, ..., n).

Option 1: Randomly pick an index i with replacement.

Option 2: Pick index i without replacement.

⇒ option 2! (shuffle 후 차례대로).

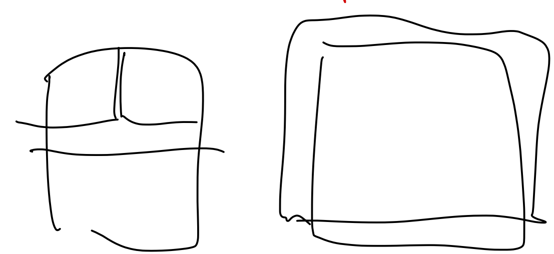
SGD Property



If I don't care about getting to the best optimum?

(exact optimal → over fitting).

⇒ "SGD is great"



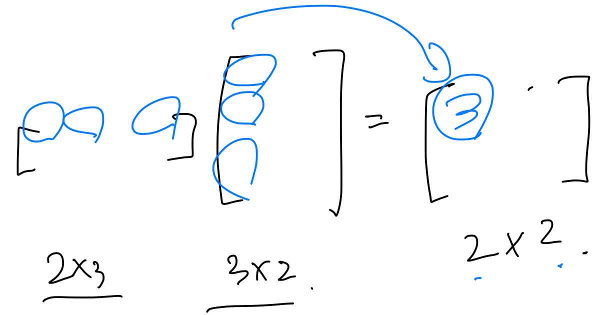
(mini-batch).

→ GPU 연산에 적합.

Back-Propagation.

∘ To compute $\nabla f = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_m}\right)$.

$\Rightarrow$ AD (automatic differentiation) "Reverse mode".



순서가 중요?

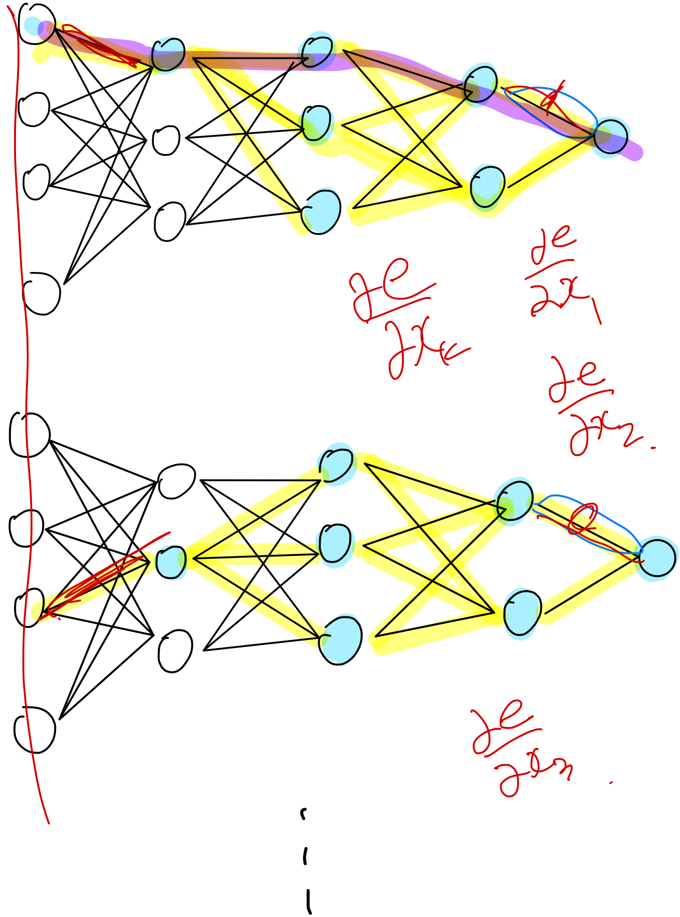
$\underset{m\times n}{A}\ \underset{n\times p}{B}\ \underset{p\times q}{C} \longrightarrow \underbrace{\boxed{(AB)}}_{m\times p}\underset{p\times q}{C} = \underset{m\times n}{A}\underbrace{(BC)}_{n\times q}$.

$mnp + mpq \;\Big|\; mnq + npq$.

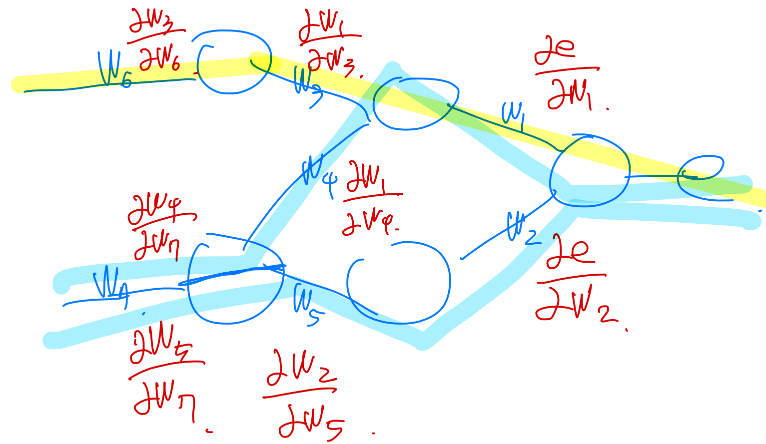
If $C = [\ ]$? $(q=1)$.

$mnp + mp \gg mn + np$.

In forward - mode



In backward - mode.



$$\frac{\partial e}{\partial w_6} = \frac{\partial e}{\partial w_1} \cdot \frac{\partial w_1}{\partial w_3} \cdot \frac{\partial w_3}{\partial w_6}.$$

$$\frac{\partial e}{\partial w_7} = \frac{\partial e}{\partial w_1} \cdot \frac{\partial w_1}{\partial w_4} \cdot \frac{\partial w_4}{\partial w_7} + \frac{\partial e}{\partial w_2} \cdot \frac{\partial w_2}{\partial w_5} \cdot \frac{\partial w_5}{\partial w_7}.$$

"ten million faster" 1000만배.