

\* 미분: Derivative

$$\frac{df}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

$$(f(x)g(x))' = f'(x)g(x) + f(x)g'(x)$$

$$\left(\frac{f(x)}{g(x)}\right)' = \frac{f'(x)g(x) - f(x)g'(x)}{(g(x))^2}$$

$$(f(x) + g(x))' = f'(x) + g'(x)$$

$$(g(f(x)))' = g'(f(x)) f'(x)$$

\* Chain Rule

$$F(x) = f(g(h(u(v(x))))))$$

$$\frac{dF}{dx} = \frac{df}{dg} \frac{dg}{dh} \dots \frac{dv}{dx}$$

Ex)  $f(x) = \log(\sqrt{x})$

$$\frac{df}{dx} = \frac{d}{dg} \log(g) \cdot \frac{d}{dx} \sqrt{x}$$

$$= \frac{1}{\sqrt{x}} \cdot \frac{1}{2\sqrt{x}} = \frac{1}{2x}$$

\* Vector form

$$\frac{df(v)}{dx} = \frac{\partial f(v)}{\partial v} \cdot \frac{\partial v}{\partial x} = \nabla f(v) \cdot \frac{\partial v}{\partial x}$$

\* Partial Derivatives

$$\frac{\partial f}{\partial x_n} = \lim_{h \rightarrow 0} \frac{f(x_1, \dots, x_n+h) - f(x)}{h}$$

\* Jacobian

$$J(f(x)) = \nabla_x f(x) = \frac{df(x)}{dx}$$

↓

$$\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_m}{\partial x_1} & \dots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} = (\nabla f_i)^T$$

\* Gradient & Hessian

$$\begin{aligned} \nabla_x b^T x &= b \\ \nabla_x^2 b^T x &= 0 \end{aligned} \rightarrow \nabla_x A x = \nabla_x \begin{bmatrix} a_1^T x \\ \vdots \\ a_n^T x \end{bmatrix} = A^T$$

\* Quadratic Function

$$\begin{aligned} \nabla_x x^T A x &= 2A x \\ \nabla_x^2 x^T A x &= 2A \end{aligned} \rightarrow \begin{aligned} \nabla_x \|x\|_2^2 &= \nabla_x x^T x = 2x \\ \nabla_x (x^T A x) &= 2A^T x \\ \nabla_x (b^T A x) &= (b^T A)^T \end{aligned}$$

\* Iterative Optimization & Gradient Descent

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} J(\theta)$$

→ ~~수치적~~

$$D = \langle x^1, y^1 \rangle, \langle x^2, y^2 \rangle, \dots, \langle x^n, y^n \rangle$$

↳ Training

Dataset

\* Learning Mechanism

→ Stochastic

On-Time → 1-batch = 1-training Example

Batch → 1-batch = Training Set

Mini-Batch → Subset of Training Set