Course overview

Matrix calculus for machine learning + Deep learning.

Newton's thought → linearization

$$\delta y = f'(x)\,\delta x$$

$$dy = f'(x)\,dx$$

$$f(x)-f(x_0) = \cancel{diff}\; f'(x_0)\,dx = f'(x_0)(x_0-x)$$

$$\underline{df = f'(x)\,dx}$$ → Change of df w.r.t dx is linear in local domain.

ex $f(x)=x^2$, eval local domain of 3.

$$df = f(3+\Delta x)-f(3) = f'(3)(3+\Delta x-3) = 6\Delta x$$

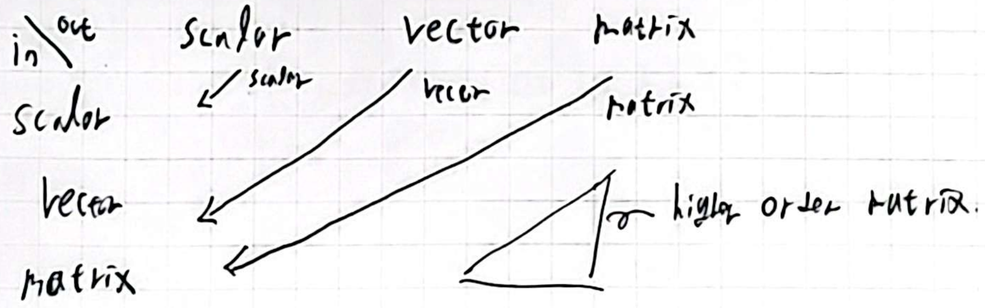
• Notation

$\begin{bmatrix}1\\2\end{bmatrix}\begin{bmatrix}2\\4\end{bmatrix}$

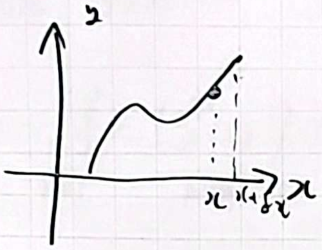
Element-wise mul $x .* y = x \odot y = \begin{bmatrix}3\\8\end{bmatrix}$

$\nabla f \to [\ ]_{m\times 1}$ col vectors.

$\frac{df}{dx} \to [\quad]_{1\times m}$ row vectors



• Derivative : linearization



higher order. $\longrightarrow$

$$f(x+\delta x) - f(x) = f'(x)\,\delta x + \underbrace{(\ )(\delta x)^2 \cdots}_{\delta x \text{ small negligible}}$$

$o(\delta(x))$

decays faster than δx.

$$\delta f = f(x+\delta x) - f(x) \approx f'(x)\,\delta x \cancel{+ o(\delta x)}$$

$\cancel{+o(\delta x)}$

• Differential notation:

$$\underbrace{df}_{\text{differential}} = f(x+dx) - f(x) = \underbrace{f'(x)}_{\text{derivative}}\, dx$$

Δoutput = (linear operator) Δx,,

부 $v \in V$ (vector space)

linear operator: $L[v]$ or Lv

↳ def $v, w \in V$, $c \in \mathbb{R}$,,

$$L[cv] = cL[v]$$

$$L[v+w] = L[v] + L[w]$$

$$L[v] \in V$$

$f(\vec{x}) \in \mathbb{R}^n$

$$df = f(\vec{x} + d\vec{x}) - f(\vec{x}) = \underbrace{f'(x)}_{\nabla f = \text{grad}}\, d\vec{x} = \text{scalar}$$

∇f^T

$f(\vec{x}) = x^T A x$

$$f(x+dx) - f(x) = (x+dx)^T A (x+dx) - x^T A x$$

$= \cancel{2dx^T A x + dx^T A dx}$ $dx^T A x + x^T A dx + \cancel{dx^T A dx}$

$= \cancel{2x^T A^T dx}$ $(x^T A + x^T A^T)\, dx$

$\cancel{f'(x) = \nabla f^T}$ $f' = \nabla f^T$

$\cancel{\nabla f = 2Ax}$ $\nabla f = (A + A^T)\, x$

vec function $\vec{f}(\underset{\in \mathbb{R}^n}{\vec{x}}) \in \mathbb{R}^m$

$\Rightarrow \underset{m\times 1}{d\vec{f}} = \underset{m\times n}{(f'(x))}\ \underset{n\times 1}{d\vec{x}}$ // end of lec 1,,