Mathematics for AI

Lecture 18

Jacobian Matrix, Chain Rule, Automatic Differentiation, Backpropagation

Remember: The Jacobian Matrix

K. N. Toosi
University of Technology

MA 18 (I)

$f: \mathbb{R}^n \rightarrow \mathbb{R}^m \qquad f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \end{bmatrix} \qquad f_i: \mathbb{R}^n \rightarrow \mathbb{R}$

$$J_f = \begin{bmatrix} \nabla f_1^T \\ \nabla f_2^T \\ \vdots \\ \nabla f_m^T \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & & \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \in \mathbb{R}^{m \times n}$$

$\cancel{f} \quad \cancel{f(x_0)}$ -

around x_0: $\underbrace{f(x)}_{m\times 1} \simeq \underbrace{f(x_0)}_{m\times 1} + \underbrace{J_f(x_0)}_{m\times n} \underbrace{(x - x_0)}_{n\times 1}$

$$D[u]f(x) = \frac{d}{d\alpha} f(x+\alpha u)\Big|_{\alpha=0} \in \mathbb{R}^m = \underbrace{J}_{\text{Jacobian}} u$$

$f(x) = Ax \qquad J = A$

$f(x) = Ax + b \qquad J = A$

Example: Derive Jacobian by directional derivative

$f: \mathbb{R}^n \to \mathbb{R}^n$ $\quad A \in \mathbb{R}^{n\times n}$ constant matrix

$$f(x) = \underbrace{x}_{n\times 1}\underbrace{x^T}_{1\times n}\underbrace{A}_{n\times n}\underbrace{x}_{n\times 1} \in \mathbb{R}^n$$

$$D[u]f(x) = \frac{d}{d\alpha} f(x+\alpha u)\Big|_{\alpha=0} = \frac{d}{d\alpha}(x+\alpha u)(x+\alpha u)^T A (x+\alpha u)\Big|_{\alpha=0}$$

$$= u\underbrace{x^T A x}_{1\times 1} + x\underbrace{u^T A x}_{1\times 1} + \boxed{x x^T A}\,u$$

$$\Big(\big(\underbrace{x^T A x}_{\text{scalar}}\big)\cdot I\Big)u + \underbrace{x x^T A^T}_{n\times n} u + \underbrace{x x^T A}_{n\times n} u$$

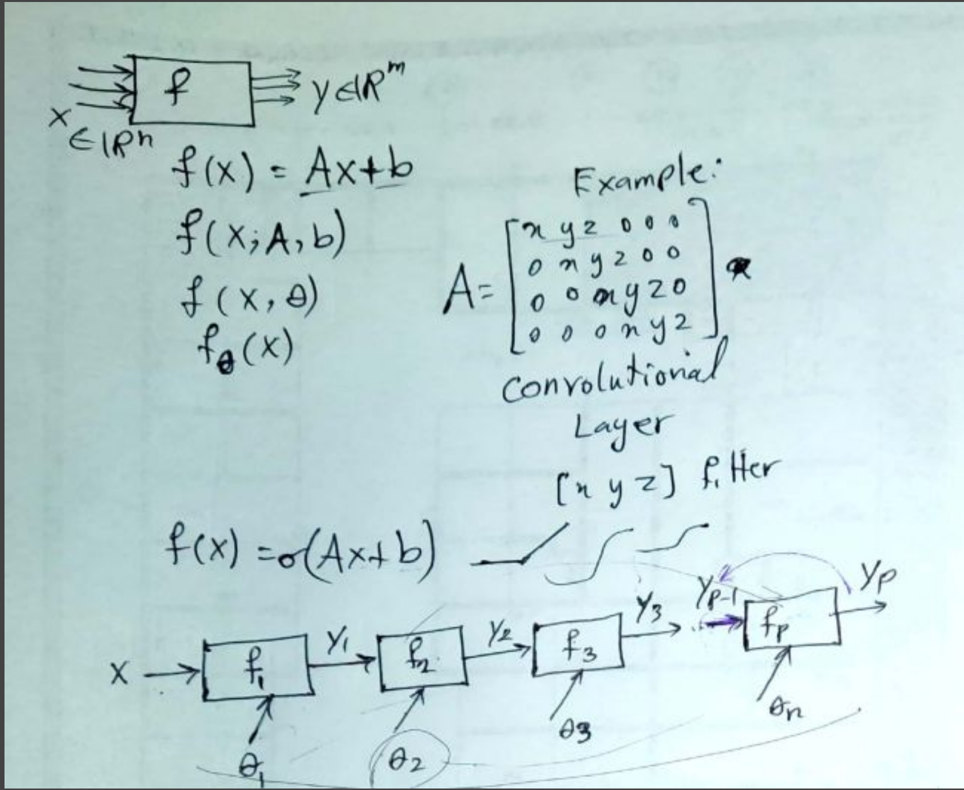
$$\boxed{\begin{aligned} &u + Au \\ &= (I+A)u \end{aligned}}$$

$$\alpha I = \begin{bmatrix} \alpha & & \\ & \alpha & \\ & & \alpha \end{bmatrix}$$

$$\Big(\underbrace{(x^T A x) I + x x^T A^T + x x^T A}_{J_f \,=\, \text{Jacobian at } x}\Big) u$$

Multi-layer neural nets

K. N. Toosi
University of Technology

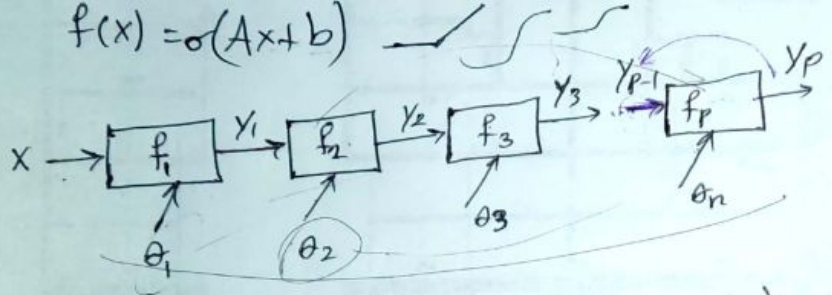


Multi-layer neural nets

K. N. Toosi
University of Technology

$[x\ y\ z]$ f_1 ...

$f(x) = \sigma(Ax+b)$



$$y_p = f_p(y_{p-1}, \theta_p) = f_p(f_{p-1}(y_{p-2}, \theta_{p-1}), \theta_p)$$

$$y_p = f_p(\ldots\ldots f_3(f_2(f_1(x, \theta_1), \theta_2), \theta_3) \ldots, \theta_p)$$

$$C(\theta_1, \theta_2 \ldots \theta_p) = \sum_{i=1}^{N} d(y_p^i, f_p(f_{p-1}(\ldots f_2(f_1(\underline{x}^i, \theta_1), \theta_2), \ldots))$$

$$\frac{\partial C}{\partial \theta_i} = \nabla_C^{\theta_i}$$

Derivative of Composition of functions

K. N. Toosi
University of Technology

$$f : \mathbb{R}^n \longrightarrow \mathbb{R}^m \qquad \frac{\partial}{\partial\theta} g(f(\theta)) = \frac{\partial}{\partial\theta}\, g \circ f(\theta) = \nabla_{g \circ f}(\theta)$$

$$g : \mathbb{R}^m \longrightarrow \mathbb{R}^p$$

$$\underbrace{g(\overbrace{f(\underbrace{\theta}_{\mathbb{R}^n})}^{\mathbb{R}^m})}_{\mathbb{R}^p}$$

$$n = m = p = 1 \Rightarrow \frac{\partial}{\partial\theta} = \frac{d}{d\theta}\, g(f(\theta)) = g'(f(\theta))\, f(\theta)$$

$$= g'\big|_{f(\theta)}\, f\big|_{\theta}$$

Chain Rule

Derivative of Composition of functions

$$f : \mathbb{R}^n \to \mathbb{R}^m \implies J_f \in \mathbb{R}^{m\times n}$$

$$g : \mathbb{R}^m \to \mathbb{R}^p \implies J_g \in \mathbb{R}^{p\times m}$$

$$J_{g\circ f} = ?$$

Intuitive ~~(Proof?)~~ Argument

$$D[u](g\circ f)(x) = \frac{d}{d\alpha}$$

$$D[u](g\circ f)(x) = \frac{d}{d\alpha}\, g(f(x+\alpha u))\Big|_{\alpha=0}$$

$$\simeq \frac{d}{d\alpha}\, g(f(x) + J_f(\alpha u))$$

$$= \frac{d}{d\alpha}\, g(f(x) + \alpha J_f u)\Big|_{\alpha=0}$$

$$= D[J_f u]\, g\Big|_{f(x)}$$

$$= J_g\Big|_{f(x)} J_f\Big|_{x}\, u$$

$$= \underbrace{J_g(f(x))\, J_f(x)}_{J_{f\circ g}(x)}\, u$$

Derivative of Composition of functions

K. N. Toosi
University of Technology

$$C(\theta) = f_n(f_{n-1}(\cdots f_2(f_1(\theta))))$$

$$J_C^{\theta} = J_{f_n}\Big|\, J_{f_{n-1}}\Big| \cdots J_{f_2}\Big|_{f_1(\theta)} J_{f_1}\Big|_{\theta}$$ chain Rule

$$J = \frac{\partial f_n}{\partial f_{n-1}} \cdots \frac{\partial f_2}{\partial f_1} \frac{\partial f_1}{\partial \theta}$$

Numeric Differentiation

K. N. Toosi
University of Technology

$\partial f_{n-1} \quad \partial f_1$

$C(f\partial)$ Numeric ~~Computation~~ Differentiation

$$h(x) = h\left(\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}\right) \qquad \frac{\partial h}{\partial x_1} \simeq \frac{h\left(\begin{bmatrix} x_1+\varepsilon \\ x_2 \\ \vdots \\ x_n \end{bmatrix}\right) - h\left(\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}\right)}{\varepsilon}$$

$$h(x) = f_n(f_{n-1}(\cdots f_2(f_1(x))\cdots)) \qquad \nabla = \begin{bmatrix} \frac{\partial h}{\partial x_1} \\ \frac{\partial h}{\partial x_2} \\ \vdots \end{bmatrix}$$

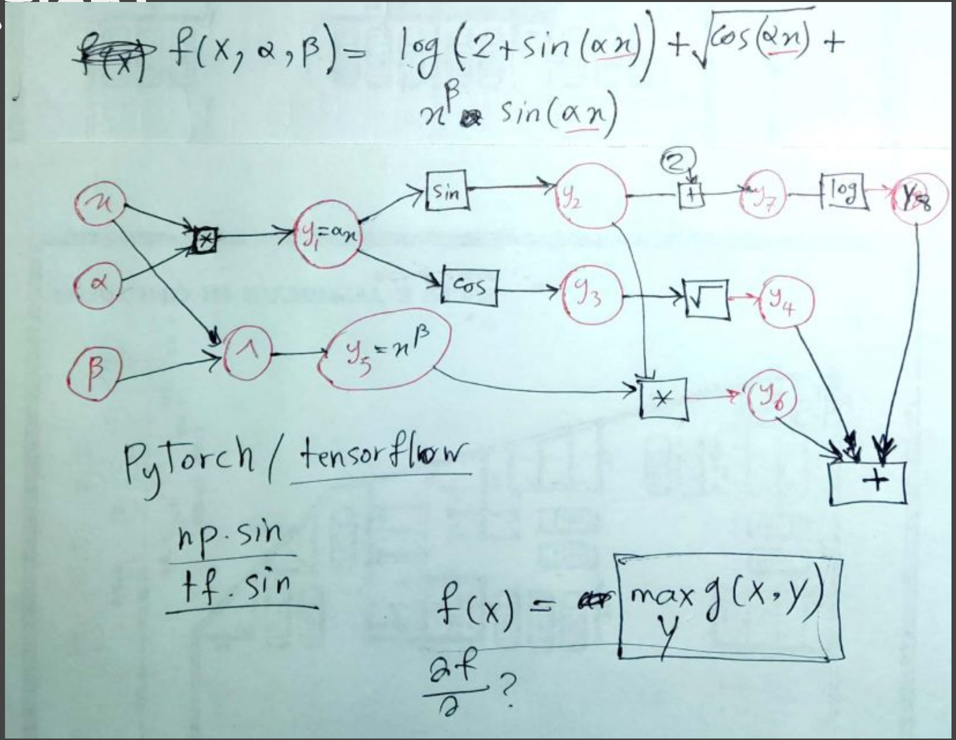
Symbolic Differentiation

Symbolic Differentiation

$f(X)$ $\quad X * X$

$\rightarrow 2X$ $\quad \Longrightarrow$ too Slow

Algorithmic Differentiation (Automatic Differentiation)



https://en.wikipedia.org/wiki/Automatic_differentiation