Chain Rule Example - Jan 19, 2026

$$\sigma(x) = \frac{1}{1+e^{-x}} = \underbrace{(1+e^{-x})^{-1}}_{g(x)}$$

$$g(x) = 1 + (e^x)^{-1}$$
$$= 1 + h(x)^{-1}$$

$$\frac{d\sigma}{dx} = \frac{d\sigma}{dg} \cdot \frac{dg}{dh} \cdot \frac{dh}{dx}$$

$$\frac{dh}{dx} = e^x, \quad \frac{dg}{dh} = 0 + (-h(x)^{-2}), \quad \frac{d\sigma}{dg} = -g(x)^{-2}$$

subbing back in:

$$\frac{d\sigma}{dx} = -g(x)^{-2} \cdot -h(x) \cdot e^x$$

$$= -(1+e^{-x})^{-2} \cdot -(e^x)^{-1} \cdot e^x$$

$$= (1+e^{-x})^{-2} \cdot e^{-x} = \frac{e^{-x}}{(1+e^{-x})^2} //$$

→ commonly written as $\sigma(x)(1-\sigma(x))$

Linear Regression - Jan 21

$\frac{\partial MSE}{\partial \theta_0} = 0$ to find θ_0 that minimizes MSE

$$\frac{\partial}{\partial \theta_0}\left(\frac{1}{m}\sum_i \underbrace{(\theta_0 + \theta_1 x_i - y_i)^2}_{g}\right) \qquad \frac{\partial g}{\partial \theta_0} = 1 \;,\; \frac{\partial g}{\partial \theta_1} = x_i$$

$$= \frac{1}{m}\sum_i 2g^1 \cdot \frac{\partial g}{\partial \theta_0} = \frac{2}{m}\sum_i (\theta_0 + \theta_1 x_i - y_i)$$

$$0 = \frac{2}{m}\left(\sum_i \theta_0 + \sum_i (\theta_1 x_i - y_i)\right)$$

$$\hookrightarrow \theta_0 = \frac{\sum_i y_i}{m} - \theta_1 \frac{\sum x_i}{m} = \mu_y - \theta_1 \mu_x //$$

$$\frac{\partial}{\partial \theta_1} = \frac{1}{m}\sum_i 2g \cdot \frac{\partial g}{\partial \theta_1} = \frac{2}{m}\left(\sum_i (\theta_0 + \theta_1 x_i - y_i)\cdot x_i\right.$$

$$0 = \theta_0 \sum_i x_i + \theta_1 \sum_i x_i^2 - \sum x_i y_i$$

$$= (\mu_y - \theta_1 \mu_x)\sum_i x_i + \theta_1 \sum_i x_i^2 - \sum_i x_i y_i$$

$$= \mu_y \sum x_i - \sum x_i y_i - \theta_1(\mu_x \sum x_i - \sum x_i^2)$$

$$\therefore \theta_1 = \frac{\mu_y \sum x_i - \sum x_i y_i}{\mu_x \sum x_i - \sum x_i^2} //$$

Matrix Form

∇_θ : just like scalar, $\frac{d}{dx}(\vec{a}\vec{x}) = \vec{a}$, $\frac{d}{dx}(\vec{a}\vec{x}^T\vec{x}) = 2\vec{a}\vec{x}$, etc

First, expand

$$\frac{1}{m}(X\vec{\theta} - \vec{y})^T(X\vec{\theta} - \vec{y})$$

$(X\vec{\theta})^T = \vec{\theta}^T X^T$

$$\frac{1}{m}(\vec{\theta}^T X^T - \vec{y}^T)(X\vec{\theta} - \vec{y}) = \vec{\theta}^T X^T X\vec{\theta} - \underbrace{\vec{\theta}^T X^T \vec{y} - \vec{y}^T X\vec{\theta}}_{\text{can group, both end up as vectors}} + \vec{y}^T\vec{y}$$

$$= \vec{\theta}^T X^T X\vec{\theta} - 2\vec{\theta}^T X^T\vec{y} + \cancel{\vec{y}^T\vec{y}} \quad \nwarrow \frac{\partial}{\partial\theta} = 0$$

$\uparrow$ $\vec{\theta}^T\vec{\theta} = \sum_i \theta_i^2$, $\frac{\partial}{\partial\theta} = 2X^T X\vec{\theta}$

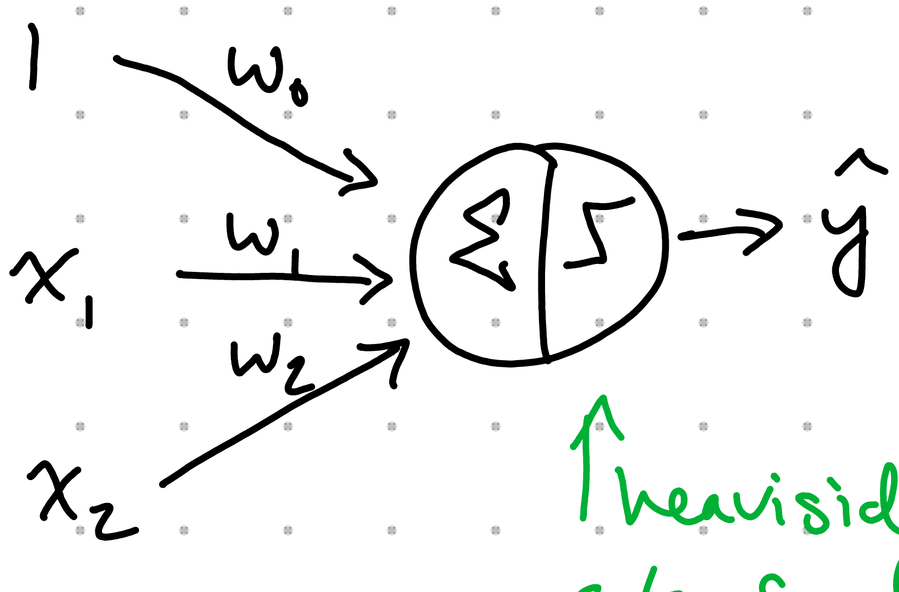
$$\nabla_\theta = \frac{1}{m}(2X^T X\vec{\theta} - 2X^T\vec{y}) = \frac{2X^T}{m}(X\vec{\theta} - \vec{y})$$

The gradient to descend

↳ can set $= 0$ to get:

$$\vec{\theta} = (X^T X)^{-1}(X^T\vec{y})$$ (closed form solution)

Perceptron: AND gate Example



x			y
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Start with $\vec{w} = \vec{0}$, then go one sample at a time

$$\hat{y}_1 = \vec{w}^T \vec{x}_1 = [0\ 0\ 0] \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \text{step}(0) = 1$$

mismatch! update $\vec{w}$:

$$\vec{w} = \vec{w} + \eta(0-1)\vec{x} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + (-1)\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}$$

sample 2:

$$\hat{y}_2 = [-1\ 0\ 0] \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \text{step}(-1) = 0 \checkmark$$

converges in 22 iterations

sample 3: same as $\hat{y}_2$

sample 4: $0 \neq 1$, update $\vec{w}$

$$\vec{w} = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} + (1-0)\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

↳ loop back to sample 1

Example: forward pass

$$\hat{y} = XW^{(1)}W^{(2)}$$

$$\begin{bmatrix} 2 & 3 \end{bmatrix} \begin{bmatrix} -0.78 & 0.13 \\ 0.85 & 0.23 \end{bmatrix} = \begin{bmatrix} 2\times -.78 + 3\times .85 & 2\times .13 + 3\times .23 \end{bmatrix}$$

$$= \begin{bmatrix} 0.99 & 0.95 \end{bmatrix}$$

$$\begin{bmatrix} 0.99 & 0.95 \end{bmatrix} \begin{bmatrix} 1.8 \\ 0.4 \end{bmatrix} = \begin{bmatrix} 0.99\times 1.8 + 0.95\times 0.4 \end{bmatrix} = 2.162$$

Note: I skipped writing this on the board and just showed the result in numpy

Entropy of Bernoulli distribution - Jan 28

$$H(x) = -E\left[\log P(x)\right] = -\sum_{x=0}^{1} P(x) \log\left(P(x)\right)$$

$$= -\left((1-p)\log(1-p) + p\log p\right)$$

for 50/50 coin toss: $p = 1-p = 0.5$. In bits:

$$H(x) = -2 \times .5 \log_2(.5) = -\log_2(.5) = 1$$

Variance in neural nets

$$Z = WX + b \quad , \quad X \sim P(0, \sigma^2) \, , \quad E[w] = E[b] = 0$$

X has n samples x k features (N x k matrix)

at a single neuron i: $z_i = \sum_{j=1}^{k} w_{ij} x_j$

$$E[z_i] = \mu_{z_i} = 0 \quad \text{for simplicity}$$

Variance:

$$\sigma_{z_i}^2 = E[z_i^2] - \cancel{E[z_i]^2}$$

$$= E\left[\left(\sum_k w_{ij} x_j\right)^2\right] = \sum_k E[w_{ij}^2] \, E[x_j^2]$$

$$k \times \sigma_w^2 \quad \sigma_x^2$$

Main takeaway: at each layer, variance scales by k features