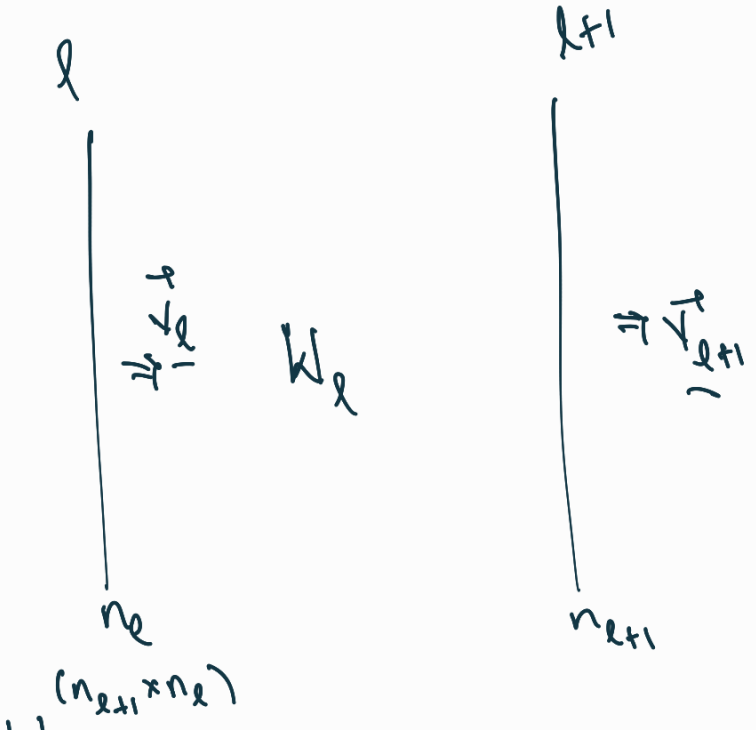
Neural networks (tensor formulation)



W_L

$W_L \cdot \vec{v}_l + b_{l+1} = \vec{v}_{l+1}$

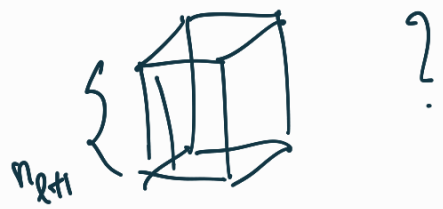
$$W_L \leftarrow W_L - \lambda \frac{\partial L(y_L, \vec{y})}{\partial W_L}$$

$(n_{l+1} \times n_l)$ $\qquad$ $(n_{l+1} \times n_l)$

$$\frac{\partial L}{\partial W_L} = \frac{\partial L}{\partial \vec{v}_{l+1}} \frac{\partial \vec{v}_{l+1}}{\partial W_L}$$

$n_{l+1} \times n_l$ $\qquad$ n_{l+1} (vector) $\qquad$ n_l (vector) $\rightarrow$ 3D tensor

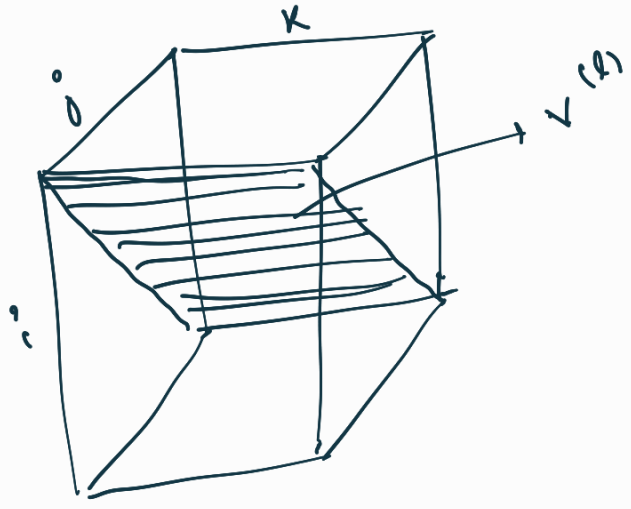
$n_{l+1} \times n_{l+1} \times n_l$



$\Delta w = dot(v, T, dim=0)$

$$v_i^{(l+1)} = b_i^{(l+1)} + \sum_{t=1}^{n_l} W_{it}^{(l)} \cdot v_t^{(l)}$$

$$\frac{\partial v_i^{(l+1)}}{\partial W_{jk}^{(l)}} = \begin{cases} 0 & \text{if } j \neq i \\ v_k^{(l)} & \text{if } j = i \end{cases}$$



$$\frac{L(v_L, \vec{y})}{\partial w_L}$$

$$f(g(x,y), h(x,y))$$

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial g}\frac{\partial g}{\partial x} + \frac{\partial f}{\partial h}\frac{\partial h}{\partial x}$$

$v_L \rightarrow$

$$\frac{\partial L(v_L)}{\partial w_{ij}^L} = \frac{\partial L}{\partial v_L}\frac{\partial v_L}{\partial w_{ij}}$$

initialize random weights →

for i in range(L):

$v_i^0 = W_{i-1} \cdot v_{i-1}$

dict $\{z_i : v_i\}$

backprop :