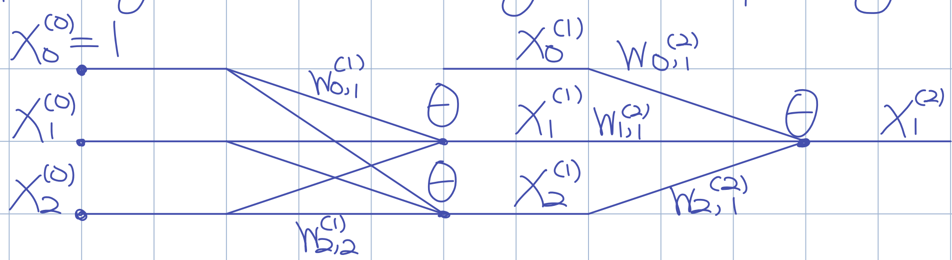
Example (2 Layer Neural Network)

Input Layer   Hidden Layer   Output Layer



$l=0$ $l=1$ $l=2$

Input Data Vector

$$\underline{x} = (x_1^{(0)}, x_2^{(0)}) \in \mathbb{R}^2$$

Input Into Node i (Layer l)

$$S_i^{(l)} = \sum_{j=0}^{d^{(l-1)}} x_j^{(l-1)} w_{j,i}^{(l)}$$

$d^{(l)}$ = number of nodes (except bias node) in layer l

$$x_i^{(l)} = \theta(S_i^{(l)})$$

Model Parameter

$$\underline{\Omega} = \{ w_{i,j}^{(l)} \}_{\substack{1 \le l \le L \\ 0 \le i \le d^{(l-1)},\ 1 \le j \le d^{(l)}}}$$

All Weights In Model

Given an input $(\underline{x}, y)$

Regression (squared error) loss

$$e(\Omega) = (x_1^{(2)} - y)^2$$

Training Set

$$D = \{(\underline{x_1}, y_1), (\underline{x_2}, y_2), \ldots, (\underline{x_N}, y_n)\}$$

$$\Omega^* = \underset{\Omega}{\operatorname{argmin}} \frac{1}{N} \sum_{n=1}^{N} e_n(\Omega)$$

SGD

Iteration #k, Let

$\Omega_k = \{w_{i,j}^{(l)}(k)\}$ denote the current choice of weights

$$w_{i,j}^{(l)}(k+1) = w_{i,j}^{(l)}(k) - \underset{\substack{\uparrow \\ \text{Learning Rate (Selected Experimentally)}}}{\varepsilon_k} \underbrace{\frac{\partial e_n(\Omega_k)}{\partial w_{i,j}^{(l)}(k)}}_{\text{How to evaluate?}}$$

Back Propagation Algorithm (Rumelhart Hinton Williams 1986)

Compute $\dfrac{\partial e(\Omega_k)}{\partial w_{i,j}^{(l)}(k)}$ in time that is linear in # of edges

Exact Calculation

$$\frac{\partial e(\Omega_k)}{\partial w_{0,1}^{(1)}} \approx \frac{e(\hat{\Omega}_k) - e(\Omega_k)}{\Delta}$$

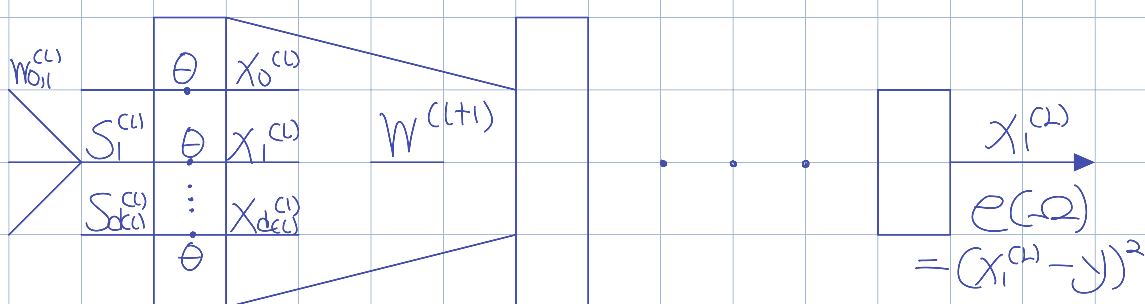
This is an approximation
$(\text{Number Of Edges})^2$
Computation Complexity

$\hat{w}_{0,1}^{(1)} = w_{0,1}^{(1)} + \Delta$

$\hat{\Omega}_k$ = set of weights in Ω_k where $w_{0,1}^{(1)}$ is changed by $\hat{w}_{0,1}^{(1)}$

Repeat this approximation for each edge

Complexity $\equiv O((\#\text{ of edges})^2)$ Too Slow



Layer l Layer $l+1$ Output Layer L

$$\frac{\partial e(\Omega)}{\partial w_{ij}^{(l)}}$$

$$e(\Omega) = e\left(S_1^{(l)}, S_2^{(l)}, \ldots, S_{d^{(l)}}^{(l)}, \underline{W^{(l+1)}}, \underline{W^{(l+2)}}, \ldots, \underline{W^{(L)}}\right)$$

$$\frac{\partial e(\Omega)}{\partial w_{i,j}^{(l)}} = \frac{\partial e(\Omega)}{\partial S_j^{(l)}} \times \frac{\partial S_j^{(l)}}{\partial w_{i,j}^{(l)}}$$

Hard To Compute

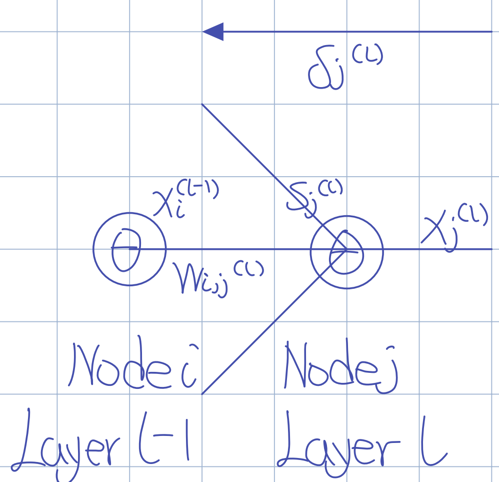
Only Effects $S_j^{(l)}$ — Easy To Compute

$$\frac{\partial e(\Omega)}{\partial S_j^{(l)}} = \delta_j^{(l)}$$

"Backward Message" in node j in layer l

$$\frac{\partial S_j^{(l)}}{\partial w_{i,j}^{(l)}} = \frac{\partial}{\partial w_{i,j}^{(l)}} \sum_{k=0}^{d^{(l-1)}} x_k^{(l-1)} w_{k,j}^{(l)} = x_i^{(l-1)}$$

$$S_j^{(l)} = \sum_{k=0}^{d^{(l-1)}} x_k^{(l-1)} w_{k,j}^{(l)}$$



$$\frac{\partial S_j^{(l)}}{\partial w_{i,j}^{(l)}} = x_i^{(l-1)} \, \delta_j^{(l)}$$

Forwarding Backwarding

How to compute backward messages

$$\frac{\partial e(\Omega)}{\partial s_j^{(l)}}$$

$1 \leq l \leq L$

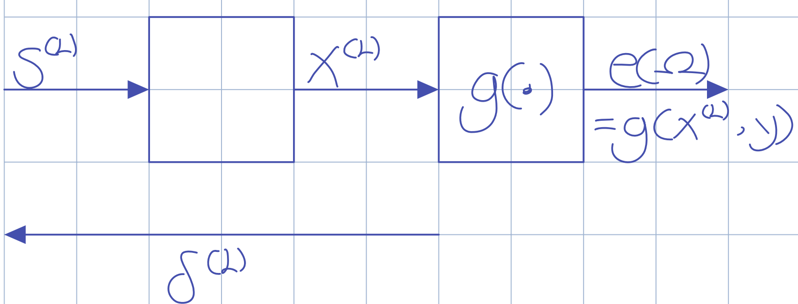
$1 \leq j \leq d^{(l)}$

Back Propogation Algorithm

Consider Output Layer $(l = L)$

Regression Setting

Loss Function $g(x^{(L)}, y) = (x^{(L)} - y)^2$



$$\delta^{(L)} = \frac{\partial e(\Omega)}{\partial s^{(L)}} = \frac{\partial e(\Omega)}{\partial x^{(L)}} \times \frac{\partial x^{(L)}}{\partial s^{(L)}}$$

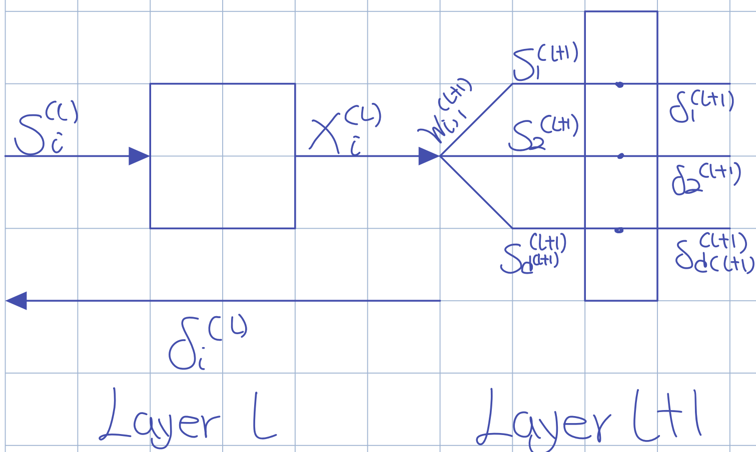
$x^{(L)} = \theta(s^{(L)})$

$$\frac{\partial x^{(L)}}{\partial s^{(L)}} = \theta'(s^{(L)})$$

$$\frac{\partial e(\Omega)}{\partial x^{(L)}} = 2(x^{(L)} - y)$$

$$\delta^{(L)} = 2(x^{(L)} - y)\,\theta'(s^{L})$$

Intermediate Layer



Compute

$$\delta_i^{(l)} = \frac{\partial e(\Omega)}{\partial s_i^{(l)}}$$

In term of

$\delta_1^{(l+1)}, \delta_2^{(l+1)}, \ldots, \delta_{d^{(l+1)}}^{(l+1)}$

$$\frac{\partial e(\Omega)}{\partial s_i^{(l)}} = \frac{\partial e(\Omega)}{\partial x_i^{(l)}} \times \frac{\partial x_i^{(l)}}{\partial s_i^{(l)}}$$

$$x_i^{(l)} = \theta(s_i^{(l)})$$

$$\frac{\partial x_i^{(l)}}{\partial s_i^{(l)}} = \theta'(s_i^{(l)})$$

$$e(\Omega) = e(\underbrace{s_1^{(l+1)}, s_2^{(l+1)}, \dots, s_{d^{(l+1)}}^{(l+1)}}_{\text{Depend On } x_i^{(l)}}, \underline{W^{(l+2)}}, \dots, \underline{W^{(L)}})$$

$$\frac{\partial e(\Omega)}{\partial x_i^{(l)}} = \sum_{j=1}^{d^{(l+1)}} \underbrace{\frac{\partial e(\Omega)}{\partial s_j^{(l+1)}}}_{\delta_j^{(l+1)}} \times \underbrace{\frac{\partial s_j^{(l+1)}}{\partial x_i^{(l)}}}_{w_{i,j}^{(l+1)}}$$

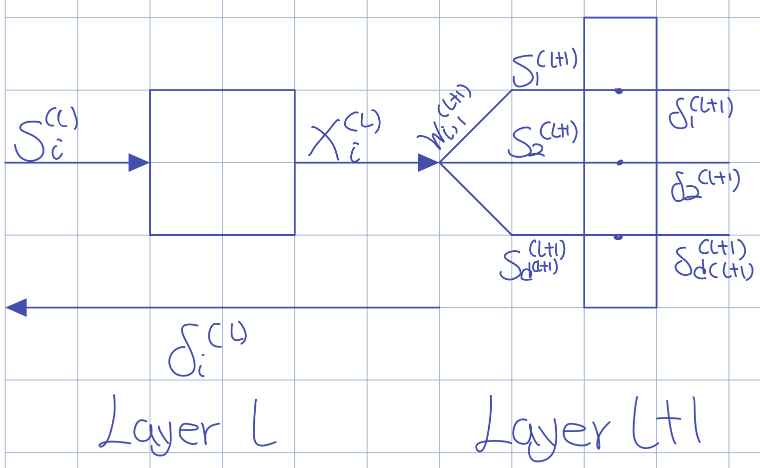
$$f(x(t), y(t), z)$$

$$\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial t}$$

$$\delta_j^{(l+1)} = \frac{\partial e(\Omega)}{\partial s_j^{(l+1)}}$$

$$\frac{\partial s_j^{(l+1)}}{\partial x_i^{(l)}} = \frac{\partial}{\partial x_i^{(l)}}\left(\sum_{k=0}^{d^{(l)}} x_k^{(l)} w_{k,j}^{(l+1)}\right)$$

$$= w_{i,j}^{(l+1)}$$



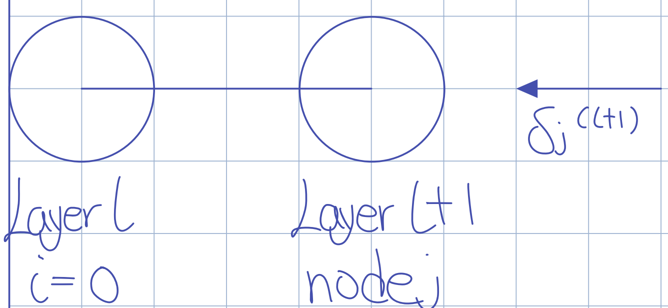
$$\frac{\partial e(\Omega)}{\partial x_i^{(l)}} = \sum_{j=1}^{d^{(l+1)}} \delta_j^{(l+1)} w_{i,j}^{(l+1)}$$

$$\delta_i^{(l)} = \frac{\partial e(\Omega)}{\partial s_i^{(l)}}$$

$$= \left(\sum_{j=1}^{d^{(l+1)}} \delta_j^{(l+1)} w_{i,j}^{(l+1)} \right) \theta'(s_i^{(l)})$$

Layer l, $i = 0$

Bias Node



$$\frac{\partial e(\Omega)}{\partial w_{0,j}^{(l)}} = x_0^{(l)} \delta_j^{(l+1)}$$

$$j = 1, 2, \ldots, \delta^{(l+1)}$$

Define

$$\underline{S^{(l)}} = \begin{bmatrix} s_1^{(l)} \\ \vdots \\ s_{d^{(l)}}^{(l)} \end{bmatrix}$$

Forward Message

$$\underline{\delta^{(l)}} = \begin{bmatrix} \delta_1^{(l)} \\ \vdots \\ \delta_{d^{(l+1)}}^{(l)} \end{bmatrix}$$

Backward Message At Layer l

Weight Matrix

$$\hat{W}^{(l+1)} = \begin{bmatrix} W_{1,1}^{(l+1)} & W_{1,2}^{(l+1)} & \cdots & W_{1,d^{(l+1)}}^{(l+1)} \\ \vdots & \vdots & \ddots & \vdots \\ W_{d^{(l)},1}^{(l+1)} & W_{d^{(l)},2}^{(l+1)} & & W_{d^{(l)},d^{(l+1)}}^{(l+1)} \end{bmatrix}$$

$$\delta^{(l)} = \left[\hat{W}^{(l+1)} \delta^{(l+1)}\right] \otimes \theta'(s^{(l)})$$

elementwise multiplication

Backward Propagation Algorithm

Input (x, y)

$\Omega = \{W^{(1)}, \ldots, W^{(L)}\}$

Output

$\delta^{(l)}$ for $l = 1, 2, \ldots, L$ and

$\frac{\partial e}{\partial w_{i,j}^{(l)}}$ $\forall i, j, l$

1. Run forward propagation to compute $\{s^{(l)}, x^{(l)}\}_{l=1,2,\ldots,L}$ and

$e(\Omega) = g(x^{(L)}; y)$

2. $\delta^{(L)} = \nabla_{x^{(L)}}\, g(x^{(L)}, y) \otimes \theta'(s^{(L)})$

3. for $l = L-1, \ldots, 1$ do

$$\underline{\delta^{(l)}} = \left[\hat{W}^{(l+1)} \delta^{(l+1)}\right] \otimes \theta'(s^{(l)})$$

$$\frac{\partial e}{\partial w_{i,j}^{(l)}} = x_i^{(l-1)} \delta_j^{(l)} \qquad w_{i,j}^{(l)} \leftarrow w_{i,j}^{(l)} - \varepsilon_k \frac{\partial e}{\partial w_{i,j}^{(l)}}$$

end for