ML Assignment 3

Ans 1.1) a) Weight for the training examples updates by this rule: $w_i = w_i + \underbrace{\Delta w}$

$$\underbrace{\Delta w_{ji}}_{\text{wt. from } i \text{ to } j \text{ [input from node } i \text{ to unit } j]} = -\eta \frac{\partial E_d}{\partial w_{ji}}$$

E_d: error for each training example d

$$E_d(\vec{w}) = \frac{1}{2} \sum_{k \in o} (t_k - o_k)^2 \qquad [k \in \text{output} \equiv k \in o]$$

t_k → target output, o_k → output computed by unit 'k'

To implement stochastic gradient descent, we compute $\frac{\partial E_d}{\partial w_{ji}}$

$$\frac{\partial E_d}{\partial w_{ji}} = \frac{\partial E_d}{\partial net_j} \cdot \frac{\partial net_j}{\partial w_{ji}}$$

$$= \underbrace{\frac{\partial E_d}{\partial net_j}}_{\text{computing this term firstly}} \cdot x_{ji} \qquad \left[\frac{\partial net_j}{\partial w_{ji}} = \frac{\partial \sum w_{ji} x_{ji}}{\partial w_{ji}} = x_{ji} \right]$$

$$\frac{\partial E_d}{\partial net_j} = \frac{\partial E_d}{\partial o_j} \frac{\partial o_j}{\partial net_j}$$

Case 1) training rule for output unit weights

$$\frac{\partial E_d}{\partial net_j} = \underbrace{\frac{\partial E_d}{\partial o_j}} \cdot \frac{\partial o_j}{\partial net_j} \quad - (1)$$

solving this term first

$$\frac{\partial E_d}{\partial o_j} = \frac{\partial}{\partial o_j} \frac{1}{2} \sum_{k \in o} (t_k - o_k)^2$$

The derivatives $\frac{\partial}{\partial o_j}(t_k - o_k)^2$ will be zero for all outputs units k except when $k = j$

We drop the summation & set $k = j$

$$\frac{\partial E_d}{\partial o_j} = \frac{\partial}{\partial o_j} \frac{1}{2} (t_j - o_j)^2$$

$$= \frac{1}{2} \cdot 2 (t_j - o_j)(-1) = -(t_j - o_j)$$

$$- (2)$$

Now lets consider the next term

$$\frac{\partial o_j}{\partial net_j} = \frac{\partial(\tanh(net_j))}{\partial net_j}$$

This is simply the derivative of tanh function

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

$$\frac{d}{dx}\tanh(x) = \frac{(e^{x}+e^{-x})(e^{x}+e^{-x}) - (e^{x}-e^{-x})(e^{x}-e^{-x})}{(e^{x}+e^{-x})^2}$$

$$= 1 - \left(\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}\right)^2 = 1 - \tanh^2(x)$$

$$\therefore \frac{\partial o_j}{\partial net_j} = 1 - (o_j)^2 \qquad - (3)$$

Putting (2), (3) in (1), we get

$$\frac{\partial E_d}{\partial net_j} = -(t_j - o_j)(1-(o_j)^2)\, x_{ji}$$

$$\Delta w_{ji} = -\eta \cdot [-(t_j - o_j)(1-(o_j)^2)\, x_{ji}]$$

$$= \eta \underbrace{(t_j - o_j)(1-(o_j)^2)}_{\delta_j} x_{ji}$$

$$\therefore \boxed{\Delta w_{ji} = \eta \delta_j x_{ji}}$$

where $\delta_j = (t_j - o_j)(1-(o_j)^2)$

Case 2) training rule for hidden unit weights

$$\frac{\partial E_d}{\partial net_j} = \sum_{k \in DS(j)} \frac{\partial E_d}{\partial net_k} \cdot \frac{\partial net_k}{\partial net_j} \quad [k \in DS(j) \equiv k \in \text{downstream}(j)]$$

$$= \sum_{k \in DS(j)} -\delta_k \cdot \frac{\partial net_k}{\partial net_j}$$

$$= \sum_{k \in DS(j)} -\delta_k \cdot \underbrace{\frac{\partial net_k}{\partial o_j}}_{w_{kj}} \cdot \underbrace{\frac{\partial o_j}{\partial net_j}}_{(1-(o_j)^2)}$$

$$= \sum_{k \in DS(j)} -\delta_k \, w_{kj} \, (1-(o_j)^2)$$

$$\therefore \frac{\partial E_d}{\partial net_j} = -(1-(o_j)^2) \sum_{k \in DS(j)} \delta_k \, w_{kj}$$

$$\Delta w_{ji} = -\eta \frac{\partial E_d}{\partial w_{ji}} = -\eta \frac{\partial E_d}{\partial net_j} \cdot x_{ji}$$

$$= -\eta \left[-(1-(o_j)^2) \sum_{k \in DS(j)} \delta_k \, w_{kj} \right] x_{ji}$$

$$= \eta \underbrace{(1-(o_j)^2) \sum_{k \in DS(j)} \delta_k \, w_{kj}}_{\delta_j} \, (x_{ji})$$

$$\therefore \boxed{\Delta w_{ji} = \eta \, \delta_j \, x_{ji}} \quad \text{where } \delta_j = (1-(o_j)^2) \sum_{k \in DS(j)} \delta_k \, w_{kj}$$

Ans 1) b) ReLu

$$E_d(\bar{w}) = \frac{1}{2} \sum_{k \in O} (t_k - o_k)^2$$

$$\Delta w_{ji} = -\eta \frac{\partial E_d}{\partial w_{ji}}$$

$$\frac{\partial E_d}{\partial w_{ji}} = \frac{\partial E_d}{\partial net_j} x_{ji} \quad \text{(from before)}$$

Case 1) training rule for output unit weights

$$\frac{\partial E_d}{\partial net_j} = \cancel{\frac{\partial E_d}{\partial net}} \quad \underbrace{\frac{\partial E_d}{\partial o_j}} \cdot \frac{\partial o_j}{\partial net_j}$$

$$\frac{\partial E_d}{\partial o_j} = \frac{\partial}{\partial o_j} \frac{1}{2} \sum_{k \in O} (t_k - o_k)^2$$

$$\frac{\partial E_d}{\partial o_j} = -(t_j - o_j)$$

$\frac{\partial o_j}{\partial net_j}$ is derivate of ReLu function

$$ReLu(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases}$$

$$\frac{\partial}{\partial x}(ReLu(x)) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases}$$

$$\frac{\partial o_j}{\partial net_j} = \begin{cases} 1 & net_j > 0 \\ 0 & net_j \leq 0 \end{cases} \quad - (4)$$

$$\therefore \frac{\partial E_d}{\partial net_j} = \begin{cases} -(t_j - o_j) & net_j > 0 \\ 0 & net_j \leq 0 \end{cases}$$

$$\frac{\partial E_d}{\partial w_{ji}} = \begin{cases} -(t_j - o_j)\, x_{ji} & net_j > 0 \\ 0 & net_j \leq 0 \end{cases}$$

$$\Delta w_{ji} = -\eta \frac{\partial E_d}{\partial w_{ji}}$$

$$\boxed{\Delta w_{ji} = \begin{cases} \eta (t_j - o_j)\, x_{ji} & net_j > 0 \\ 0 & net_j \leq 0 \end{cases}} = \begin{cases} \eta \delta_k x_{ji} \\ 0 \end{cases}$$

where $\delta_j = (t_j - o_j)$

Case 2) training rule for hidden unit weights

$$\frac{\partial E_d}{\partial net_j} = \sum_{k \in DS(j)} \frac{\partial E_d}{\partial net_k} \cdot \frac{\partial net_k}{\partial net_j}$$

$$= \sum_{k \in DS(j)} -\delta_k \frac{\partial net_k}{\partial net_j}$$

$$= \sum_{k \in DS(j)} -\delta_k \underbrace{\frac{\partial net_k}{\partial o_j}}_{w_{kj}} \cdot \frac{\partial o_j}{\partial net_j}$$

$$\frac{\partial E_d}{\partial net_j} = \begin{cases} \sum_{k \in DS(j)} -\delta_k\, w_{kj} & net_j > 0 \\ 0 & net_j \leq 0 \end{cases} \quad [\text{from } (4)]$$

Scanned by CamScanner

$$\Delta w_{ji} = -\eta \frac{\partial E_d}{\partial w_{ji}}$$

$$= -\eta \frac{\partial E_d}{\partial net_j} \cdot x_{ji}$$

$$= \begin{cases} +\eta \underbrace{\sum_{k \in DS(j)} +\delta_k w_{kj}}_{\delta_j} x_{ji} & net_j > 0 \\ 0 & net_j \leq 0 \end{cases}$$

$$\therefore \boxed{\Delta w_{ji} = \begin{cases} \eta \delta_j x_{ji} & net_j > 0 \\ 0 & net_j \leq 0 \end{cases} \quad \text{where } \delta_j = \sum_{k \in DS(j)} \delta_k w_{kj}}$$

Ans 1.2)

$$o = w_0 + w_1(x_1 + x_1^2) + \ldots + w_n(x_n + x_n^2)$$

x_i : inputs
w_i : weights
o : output

$$E_d(\bar{w}) = \frac{1}{2}\sum_{k \in o}(t_k - o_k)^2$$

$$\Delta w_{ji} = -\eta \frac{\partial E_d}{\partial w_{ji}}$$

$$\frac{\partial E_d}{\partial w_{ji}} = \underbrace{\frac{\partial E_d}{\partial net_j}} \cdot \underbrace{\frac{\partial net_j}{\partial w_{ji}}}_{(x_j + x_j^2)}$$

$$\frac{\partial E_d}{\partial net_j} = \underbrace{\frac{\partial E_d}{\partial o_j}} \cdot \underbrace{\frac{\partial o_j}{\partial net_j}}_{= f'(x) = 1}$$

[activation function is $f(x) = x$]

$$\frac{\partial E_d}{\partial o_j} = \frac{\partial}{\partial o_j} \frac{1}{2}(t_j - o_j)^2$$
$$= \frac{1}{2} \cdot 2(t_j - o_j)(-1) = -(t_j - o_j)$$

$$\Delta w_{ji} = -\eta\left[-(t_j - o_j)(x_j + x_j^2)\right]$$

$$\therefore \boxed{\Delta w_{ji} = \eta(t_j - o_j)(x_j + x_j^2)}$$

Ans 1.3) a) i: input   o: output

$$i_3 = x_1 w_{31} + x_2 w_{32}$$

$$i_4 = x_1 w_{41} + x_2 w_{42}$$

$$o_3 = h(x_1 w_{31} + x_2 w_{32})$$

$$o_4 = h(x_1 w_{41} + x_2 w_{42})$$

$$i_5 = w_{53} o_3 + w_{54} o_4$$

$$i_5 = w_{53}[h(x_1 w_{31} + x_2 w_{32})] + w_{54}[h(x_1 w_{41} + x_2 w_{42})]$$

$$o_5 = h(i_5)$$

$$o_5 = y_5 = h\left(w_{53} \cdot h(x_1 w_{31} + x_2 w_{32}) + w_{54} \cdot h(x_1 w_{41} + x_2 w_{42})\right)$$

b) $X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ $W^{(1)} = \begin{bmatrix} w_{3,1} & w_{3,2} \\ w_{4,1} & w_{4,2} \end{bmatrix}$ $W^{(2)} = [w_{5,3} \quad w_{5,4}]$

$$W^{(1)} \cdot X = \begin{bmatrix} w_{31} x_1 + w_{32} x_2 \\ w_{41} x_1 + w_{42} x_2 \end{bmatrix}$$

output for hidden layer $= \begin{bmatrix} h(w_{31} x_1 + w_{32} x_2) \\ h(w_{41} x_1 + w_{42} x_2) \end{bmatrix} = X_1$

$$W^{(2)} X_1 = [w_{53} \cdot h(w_{31} x_1 + w_{32} x_2) + w_{54} \cdot h(w_{41} x_1 + w_{42} x_2)]$$

output for node 5 $= y_5 = h(W^{(2)} X_1)$

$$= [h(w_{53} \cdot h(w_{31} x_1 + w_{32} x_2) + w_{54} \cdot h(w_{41} x_1 + w_{42} x_2))]$$

Scanned by CamScanner

Ans) 1.3). c) $h_s(x) = \dfrac{1}{1+e^{-x}} = s$ (let) — ①

$h_t(x) = \dfrac{e^{x} - e^{-x}}{e^{x} + e^{-x}} = t$ (let) — ②

$$s = \frac{1}{1+\frac{1}{e^{x}}} = \frac{e^{x}}{e^{x}+1} \Rightarrow e^{x} = e^{x}\cdot s + s$$

$$e^{x}(1-s) = s$$

$$\therefore e^{x} = \frac{s}{1-s} \quad — ③$$

Putting ③ in ②, we get

$$t = \frac{\frac{s}{1-s} - \left(\frac{1-s}{s}\right)}{\frac{s}{1-s} + \left(\frac{1-s}{s}\right)} = \frac{\frac{s^2-(1-s)^2}{\cancel{(1-s)(s)}}}{\frac{s^2+(1-s)^2}{\cancel{(1-s)(s)}}}$$

$$= \frac{\cancel{s^2} - [1+\cancel{s^2}-2s]}{s^2+1+s^2-2s} = \frac{2s-1}{2s^2-2s+1}$$

$$\therefore h_t(x) = \frac{2h_s(x) - 1}{2(h_s(x))^2 - 2h_s(x) + 1}$$

Cont...

PTO

$$h_s(x) = \frac{e^x}{e^x+1}$$

$$h_s(2x) = \frac{e^{2x}}{e^{2x}+1}$$ [by noticing power of 2 in $h_t(x)$]

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

By hit & trial we note that '$2(h_s(2x))-1$' equals

$$\frac{2e^{2x}}{e^{2x}+1} - 1 = \frac{2e^{2x} - e^{2x} - 1}{e^{2x}+1} = \frac{e^{2x}-1}{e^{2x}+1}$$

which equals $h_t(x)$

$$\therefore h_t(x) = 2 \cdot h_s(2x) - 1$$

Hence output generated is same, with the parameters differing only by linear transformations & constants.

Ans 1.4) $E(\vec{w}) = \frac{1}{2} \sum_{d \in D} \sum_{k \in O} (t_{kd} - o_{kd})^2 + \gamma \sum_{i,j} w_{ji}^2$

let's assume the activation function to be sigmoid.

$\Delta w_{ji} = -\eta \frac{\partial E_d}{\partial w_{ji}}$

Δw_{ji} : wt. from i to j
η : learning rate
E_d : error for each training example D

Case 1: for output unit weights

we already derived Δw_{ji} for $E(\vec{w})$ without the term '$\gamma \sum_{i,j} w_{ji}^2$' in Q1.1) $\therefore$ We would only derive for this term now & use the previous result.

$$= \frac{\partial}{\partial w_{ji}} \gamma \sum_{i,j} w_{ji}^2$$

$$= 2\gamma w_{ji}$$

$$\therefore \Delta w_{ji} = -\eta \left[-(t_j - o_j) o_j (1 - o_j) x_{ji} + 2\gamma w_{ji} \right]$$

$$= \eta \underbrace{(t_j - o_j) o_j (1 - o_j)}_{\delta_j} x_{ji} - 2\gamma w_{ji}$$

$$\boxed{\Delta w_{ji} = \eta \delta_j x_{ji} - 2\gamma w_{ji}}$$

$$w_i^{new} = w_i^{old} + \Delta w_{ji}$$

$$= w_{ji} + \eta \delta_j x_{ji} - 2\gamma w_{ji}$$

$$\therefore \boxed{w_{ji} = \eta \delta_j x_{ji} - (2\gamma - 1) w_{ji}}$$

where $\delta_j = (t_j - o_j) o_j (1 - o_j)$

Scanned by CamScanner

Case 2: for hidden unit weights

$$\Delta w_{ji} = -\eta\left[-o_j(1-o_j)\sum_{k \in DS} \delta_k w_{kj} \cdot x_{ji} + 2\gamma w_{ji}\right]$$

$$= \eta \underbrace{o_j(1-o_j)\sum_{k \in DS} \delta_k w_{kj}}_{\delta_j} \cdot x_{ji} - 2\gamma x_{ji}$$

$$\boxed{\Delta w_{ji} = \eta \delta_j x_{ji} - 2\gamma x_{ji}}$$

$$w_i^{new} = w_i^{old} + \Delta w_{ji}$$

$$= w_{ji} + \eta \delta_j x_{ji} - 2\gamma x_{ji}$$

$$\boxed{w_{ji} = \eta \delta_j x_{ji} - (2\gamma - 1) w_{ji}}$$

where $\delta_j = o_j(1-o_j)\sum_{k \in DS} \delta_k w_{kj}$

In both the cases we get the same equation differing only by the value of the 'δ_j' term.

Hence it proves that this update rule can be implemented by multiplying each weight by some constant before performing the standard gradient descent update.

Scanned by CamScanner