Q1

$d = 0,45v\sqrt{h}$

$y = w_0 v + w_1 h$

$w = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}$

$x = \begin{bmatrix} v \\ h \end{bmatrix}$

$y_i = w^T x_i + b$

$L(y,d) = (y-d)^2$

$X = \begin{bmatrix} x_1^T \\ \vdots \\ x_I^T \end{bmatrix} = \begin{bmatrix} v_1 & h_1 \\ \cdots & \\ v_I & h_I \end{bmatrix}$

$d = \begin{bmatrix} d_1 \\ \vdots \\ d_I \end{bmatrix}$

I. why is $\hat{R}(w) = \frac{1}{I}\|Xw - d\|^2$?

* $R(w) = E(l(x,w)) = \frac{1}{I}\sum_i l(x_i,w) = \frac{1}{I}\sum_i L(f(x_i|w), y_i)$

$L(y,d) = (y-d)^2$

↳ $L(y,d) = (w_0 v + w_1 h - d)^2$

$R(w) = \frac{1}{I}\sum_i (w_0 v_i + w_1 h_i - d_i)^2 = \frac{1}{I}\sum_i (w^T x_i - d_i)^2$

$= \frac{1}{I}\left((w^T x_1 - d_1)^2 + (w^T x_2 - d_2)^2 + \cdots + (w^T x_I - d_I)^2\right)$

$= \frac{1}{I}\left(\begin{bmatrix} w^T x_1 - d_1 & \cdots & w^T x_I - d_I \end{bmatrix} \underbrace{\begin{bmatrix} w^T x_1 - d_1 \\ \vdots \\ w^T x_I - d_I \end{bmatrix}}_{H}\right)$

$= \frac{1}{I}\|H\|^2 = \frac{1}{I}\left\| \begin{bmatrix} w^T x_1 - d_1 \\ \vdots \\ w^T x_I - d_I \end{bmatrix} \right\|^2$ (a)

next page

$$\underbrace{\begin{bmatrix} w^T x_1 - d_1 \\ \vdots \\ w^T x_I - d_I \end{bmatrix}}_{H} = \begin{bmatrix} w^T x_1 \\ \vdots \\ w^T x_I \end{bmatrix} - \begin{bmatrix} d_1 \\ \vdots \\ d_I \end{bmatrix} = \begin{bmatrix} x_1^T w \\ \vdots \\ x_I^T w \end{bmatrix} - \underbrace{\begin{bmatrix} d_1 \\ \vdots \\ d_I \end{bmatrix}}_{d}$$

$$= \begin{bmatrix} x_1^T \\ \vdots \\ x_I^T \end{bmatrix} w - d = X w - d \quad (b)$$

(a), (b) $\Rightarrow$ $R(w) = \frac{1}{I} \| X w - d \|^2$ (c)

II. gradient of risk:

* gradient $f = \begin{bmatrix} f_1(w) \\ \vdots \\ f_n(w) \end{bmatrix} = \nabla f(w)$

$$\nabla R(w) \equiv \begin{bmatrix} R_1'(w) \\ \vdots \\ R_I(w) \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial w_0} R(w) \\ \vdots \\ \frac{\partial}{\partial w_1} R(w) \end{bmatrix}$$

$$R(w_0) = \frac{1}{I}\left((x_1^T w_0 - d_1)^2 + \cdots + (x_I^T w_0 - d_I)^2 \right)$$

$$R(w_0) = \frac{1}{I} \sum_i (x_i^T w_0 - d_i)^2$$

$$\frac{\partial}{\partial w_n}\left(\frac{1}{I} \sum_i (x_i^T w_0 - d_i)^2 \right) = \frac{1}{I} \sum_i \frac{\partial}{\partial w_n} (x_i^T w_0 - d_i)^2$$

$$\frac{\partial}{\partial w_n} (x_i^T w_0 - d_i)^2 = \frac{\partial}{\partial w_n} (w_0 v_i + w_1 h_i - d_i)^2$$

$$\frac{\partial}{\partial w_0} (w_0 v_i + w_1 h_i - d_i)^2 = 2(w_0 v_i + w_1 h_i - d_i)(v_i)$$

$$\frac{\partial}{\partial w_1} (w_0 v_i + w_1 h_i - d_i)^2 = 2(w_0 v_i + w_1 h_i - d_i)(h_i)$$

$$\rightsquigarrow \nabla R(w) = \begin{bmatrix} \frac{1}{I} \sum_i 2 v_i (w_0 v_i + w_1 h_i - d_i) \\ \frac{1}{I} \sum_i 2 h_i (w_0 v_i + w_1 h_i - d_i) \end{bmatrix}$$ ✓

$$w^{(t)} \leftarrow w^{(t-1)} - \eta \begin{bmatrix} \frac{1}{I} \sum_i 2 v_i (w_0^{(t-1)} v_i + w_1^{(t-1)} h_i - d_i) \\ \frac{1}{I} \sum_i 2 h_i (w_0^{(t-1)} v_i + w_1^{(t-1)} h_i - d_i) \end{bmatrix}$$ → update rule ✓

III. find minimizer

$$\nabla R(w) = \begin{bmatrix} \frac{1}{I}\sum_i 2v_i(w_0 v_i + w_1 h_i - d_i) \\ \frac{1}{I}\sum_i 2h_i(w_0 v_i + w_1 h_i - d_i) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\frac{1}{I}\sum_i 2v_i(w_0 v_i + w_1 h_i - d_i) = 0 \qquad \sum_i v_i(w_0 v_i + w_1 h_i - d_i) = 0$$

$$\frac{1}{I}\sum_i 2h_i(w_0 v_i + w_1 h_i - d_i) = 0 \longrightarrow \sum_i h_i(w_0 v_i + w_1 h_i - d_i) = 0$$

(w_0, w_1 unrelated to i)

$$\sum_i v_i^2 w_0 + v_i h_i w_1 - v_i d_i = 0 \qquad \sum_i v_i h_i w_0 + h_i^2 w_1 - h_i d_i = 0$$

$$\underbrace{\left(\sum_i v_i^2\right)}_{V} w_0 + \underbrace{\left(\sum_i v_i h_i\right)}_{A} w_1 - \underbrace{\left(\sum_i v_i d_i\right)}_{B} = 0$$

$$\underbrace{\left(\sum_i v_i h_i\right)}_{A} w_0 + \underbrace{\left(\sum_i h_i^2\right)}_{H} w_1 - \underbrace{\left(\sum_i h_i d_i\right)}_{C} = 0$$

$$Vw_0 + Aw_1 = B$$

$$Aw_0 + Hw_1 = C$$

$$\begin{bmatrix} V & A \\ A & H \end{bmatrix}\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} B \\ C \end{bmatrix}$$

$$\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} V & A \\ A & H \end{bmatrix}^{-1}\begin{bmatrix} B \\ C \end{bmatrix} \qquad M$$

$$\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \frac{1}{VH - A^2}\begin{bmatrix} H & -A \\ -A & V \end{bmatrix}\begin{bmatrix} B \\ C \end{bmatrix}$$

$M \cdot \text{adj}(M) = \det(M) I$

$M^{-1} = \frac{1}{\det(M)}\text{adj}(M)$

$\text{adj}(M) = \text{Cofactor}(M)^T$

$$\begin{bmatrix} +|H| & -|A| \\ -|A| & +|V| \end{bmatrix}^T$$

$$\begin{bmatrix} V & A \\ A & H \end{bmatrix}^{-1} = \frac{1}{VH - A^2}\begin{bmatrix} H & -A \\ -A & V \end{bmatrix}$$

$$\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \frac{1}{VH - A^2}\begin{bmatrix} HB - AC \\ -AB + VC \end{bmatrix}$$

$$w_0 = \frac{\left(\sum h_i^2\right)\left(\sum v_i d_i\right) - \left(\sum v_i h_i\right)\left(\sum h_i d_i\right)}{\left(\sum v_i^2\right)\left(\sum h_i^2\right) - \left(\sum v_i h_i\right)^2}$$

$$w_1 = \frac{\left(\sum v_i^2\right)\left(\sum h_i d_i\right) - \left(\sum v_i h_i\right)\left(\sum v_i d_i\right)}{\left(\sum v_i^2\right)\left(\sum h_i^2\right) - \left(\sum v_i h_i\right)^2}$$

IV. $\Rightarrow$ $w_0 = \dfrac{(\sum h_i^2)(\sum v_i d_i) - (\sum v_i h_i)(\sum h_i d_i)}{(\sum v_i^2)(\sum h_i^2) - (\sum v_i h_i)^2}$ { from III

(V)

$$w_1 = \frac{(\sum v_i^2)(\sum h_i d_i) - (\sum v_i h_i)(\sum v_i d_i)}{(\sum v_i^2)(\sum h_i^2) - (\sum v_i h_i)^2}$$

$$\cdots \overline{(v_0^2+v_1^2)(h_0^2+h_1^2) - (v_0 h_0 + v_1 h_1)^2\}\}}$$

$$v_0^2 h_0^2 + v_0^2 h_1^2 + v_1^2 h_0^2 + v_1^2 h_1^2 - v_0^2 h_0^2 - v_1^2 h_1^2 - 2 v_0 v_1 h_0 h_1$$

$$(v_0 h_1 - v_1 h_0)^2 \quad (I)$$

w_1 &

$$(v_0^2+v_1^2)(h_0 d_0 + h_1 d_1) - (v_0 h_0 + v_1 h_1)(v_0 d_0 + v_1 d_1)$$

$$v_0^2 h_0 d_0 + v_0^2 h_1 d_1 + v_1^2 h_0 d_0 + v_1^2 h_1 d_1 - v_0^2 h_0 d_0 - v_0 v_1 h_0 d_1 - v_0 v_1 h_1 d_0 - v_1^2 h_1 d_1$$

$$v_0^2 h_1 d_1 + v_1^2 h_0 d_0 - v_0 v_1 (h_0 d_1 + h_1 d_0)$$

$$\Rightarrow w_1 = \frac{v_0^2 h_1 d_1 + v_1^2 h_0 d_0 - v_0 v_1 (h_0 d_1 + h_1 d_0)}{(v_0 h_1 - v_1 h_0)^2} = \frac{(v_0 h_1 - v_1 h_0)(v_0 d_1 - v_1 d_0)}{(v_0 h_1 - v_1 h_0)^2}$$

$$w_0 = \frac{h_0^2 v_1 d_1 + h_1^2 v_0 d_0 - h_0 h_1 (v_0 d_1 + v_1 d_0)}{(v_0 h_1 - v_1 h_0)^2} = \frac{(v_0 h_1 - v_1 h_0)(h_1 d_0 - h_0 d_1)}{(v_0 h_1 - v_1 h_0)^2}$$

$$w_0 = \frac{h_1 d_0 - h_0 d_1}{v_0 h_1 - v_1 h_0} \quad (V)$$

$$\Rightarrow (v_0 h_1 - v_1 h_0) \neq 0 \Longrightarrow \quad w_1 = \frac{v_0 d_1 - v_1 d_0}{v_0 h_1 - v_1 h_0} \quad (V)$$

* hypothesis: $\det\begin{bmatrix} v_0 & h_0 \\ v_1 & h_1 \end{bmatrix} \neq 0$

$$w_0 v_0 + w_1 h_0 - d_0 = \frac{-v_0 h_0 d_1 + v_0 h_1 d_0 + v_0 h_0 d_1 - v_1 h_0 d_0}{v_0 h_1 - v_1 h_0} - d_0 = 0 \quad (V)$$

$$w_0 v_1 + w_1 h_1 - d_1 = \frac{-v_1 h_0 d_1 + v_1 h_1 d_0 + v_0 h_1 d_1 - v_1 h_1 d_0}{v_0 h_1 - v_1 h_0} - d_1$$

$$= 0 \quad (V)$$

* Continued on next page

IV. for $I \geq 3$

$$w_0 = \frac{(\Sigma h_i^2)(\Sigma v_i d_i) - (\Sigma v_i h_i)(\Sigma h_i d_i)}{(\Sigma v_i^2)(\Sigma h_i^2) - (\Sigma v_i h_i)^2}$$

$$w_1 = \frac{(\Sigma v_i^2)(\Sigma h_i d_i) - (\Sigma v_i h_i)(\Sigma v_i d_i)}{(\Sigma v_i^2)(\Sigma h_i^2) - (\Sigma v_i h_i)^2}$$

$$w_i = \frac{\cdots}{v_1^2 h_1^2 + v_1^2 h_2^2 + v_1^2 h_3^2 \cdots - (v_1 h_1 + v_2 h_2 + v_3 h_3)^2}$$

$$v_1^2 h_2^2 + v_1^2 h_3^2 + v_2^2 h_1^2 + v_2^2 h_3^2 + v_3^2 h_1^2 + v_3^2 h_2^2 - 2(v_1 h_1 v_2 h_2 + v_1 h_1 v_3 h_3 + v_2 h_2 v_3 h_3)$$

$$= \frac{\cdots}{(v_1 h_2 - v_2 h_1)^2 + (v_1 h_3 - v_3 h_1)^2 + (v_2 h_3 - v_3 h_2)^2}$$

$$w_0 = \frac{(v_1 h_2 - v_2 h_1)(h_2 d_1 - h_1 d_2) + (v_1 h_3 - v_3 h_1)(h_3 d_1 - h_1 d_3) + (v_2 h_3 - v_3 h_2)(h_3 d_2 - h_2 d_3)}{(v_1 h_2 - v_2 h_1)^2 + (v_1 h_3 - v_3 h_1)^2 + (v_2 h_3 - v_3 h_2)^2}$$

$$w_1 = \frac{(v_1 h_2 - v_2 h_1)(v_2 d_1 - v_1 d_2) + (v_1 h_3 - v_3 h_1)(v_3 d_1 - v_1 d_3) + (v_2 h_3 - v_3 h_2)(v_3 d_2 - v_2 d_3)}{(v_1 h_2 - v_2 h_1)^2 + (v_1 h_3 - v_3 h_1)^2 + (v_2 h_3 - v_3 h_2)^2}$$

now we could see that as the number of datapoints is increased, w_1 and w_0 would become more complicated and $\hat{R}=0$ would be almost infeasable and -
- also $\hat{R}$ would increase because we are approximating a non-linear function with a linear function!!

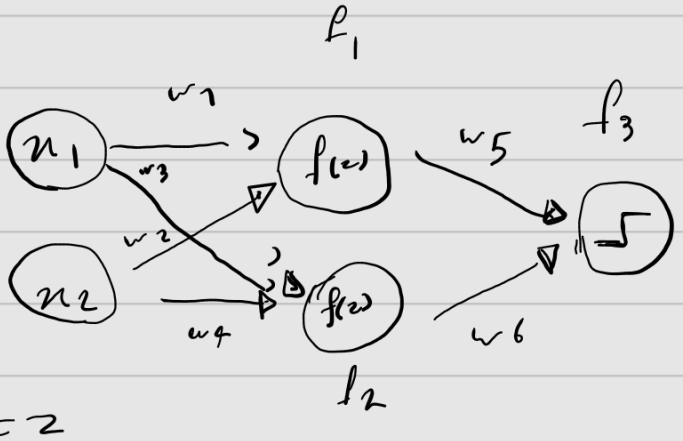
V. for w_t to converge in one step, we
- need $w^{t+1} \leftarrow w^t - \eta \nabla R$ to get to the optimal w in one step, meaning that
- both w_0 and w_1 would need to change
- to w_0' and w_1' in one step.

we could notice that because η is applied to both w_0, w_1 then the (I) initial values and their (II) distance from optimal w_0 and w_1 would need to be exactly the same so that a single η
- would make both optimal in one step
- which we know almost never happens!!

VI. If we use $\begin{bmatrix} \eta_1 & 0 \\ 0 & \eta_2 \end{bmatrix}$, then both η_1, η_2 could be fine tuned regardless of the other one so that w_0 and w_1 could converge to optimal values in one step. ✓

Q2

I.



$f(z) = z$

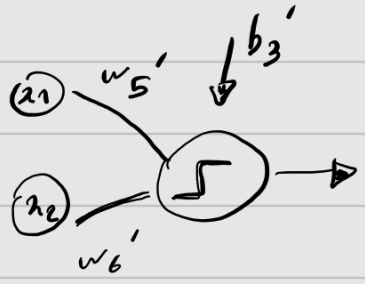
$f_1 \rightarrow w_1 x_1 + w_2 x_2 + b_1$

$f_2 \rightarrow w_3 x_1 + w_4 x_2 + b_2$

$\Rightarrow S\left((w_1 x_1 + w_2 x_2 + b_1) w_5 + (w_3 x_1 + w_4 x_2 + b_2) w_6 + b_3 \right)$

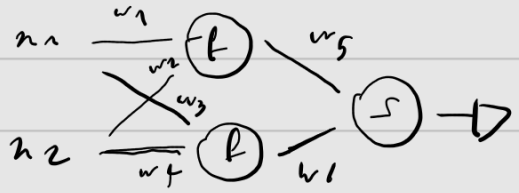
$= S\left((w_1 w_5 + w_3 w_6) x_1 + (w_2 w_5 + w_4 w_6) x_2 + w_5 b_1 + w_6 b_2 + b_3 \right)$

$S(w_5' x_1 + w_6' x_2 + b_3') =$



Q2.

II.



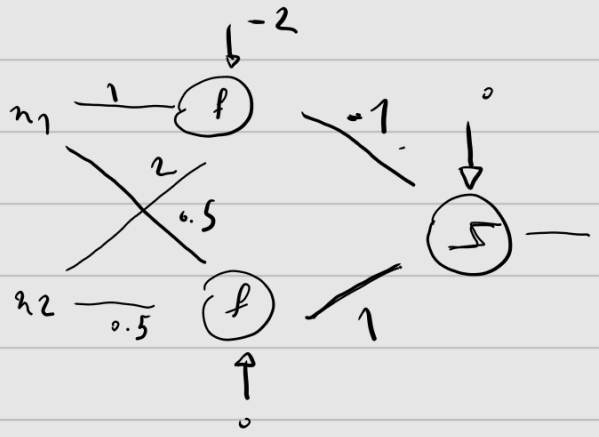
Lets try to create AND and OR with ReLU

AND → $w_1 = 1$ $w_2 = 2$ $b = -2$

OR → we can't but we create a semi-OR

$w_3 = 0,5$ $w_4 = 0,5$

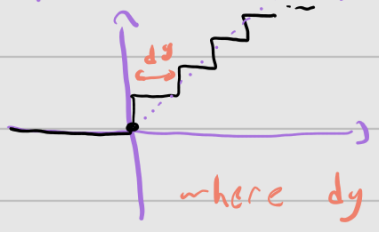
$b_2 = 0$

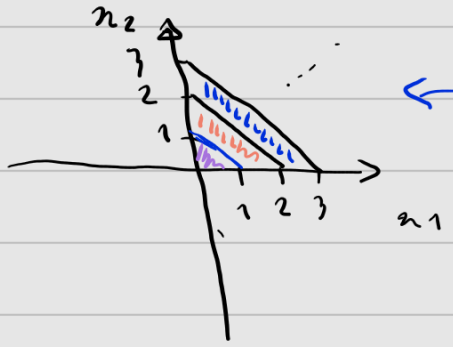


* what we try to do is we use ReLU as a small classifier like perceptron

* note → geometrically analyzing,

ReLU is basically like this



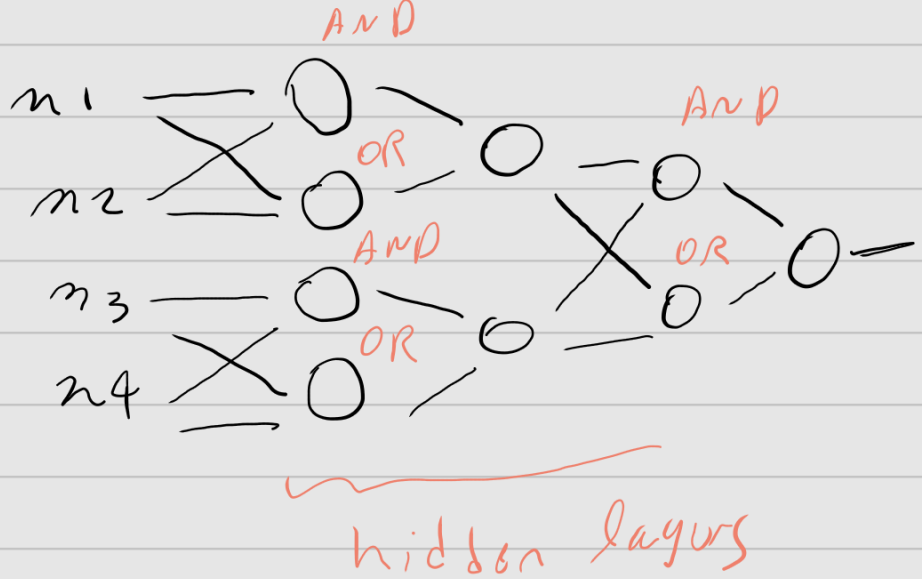


now if dy = 1

these areas show the classifications

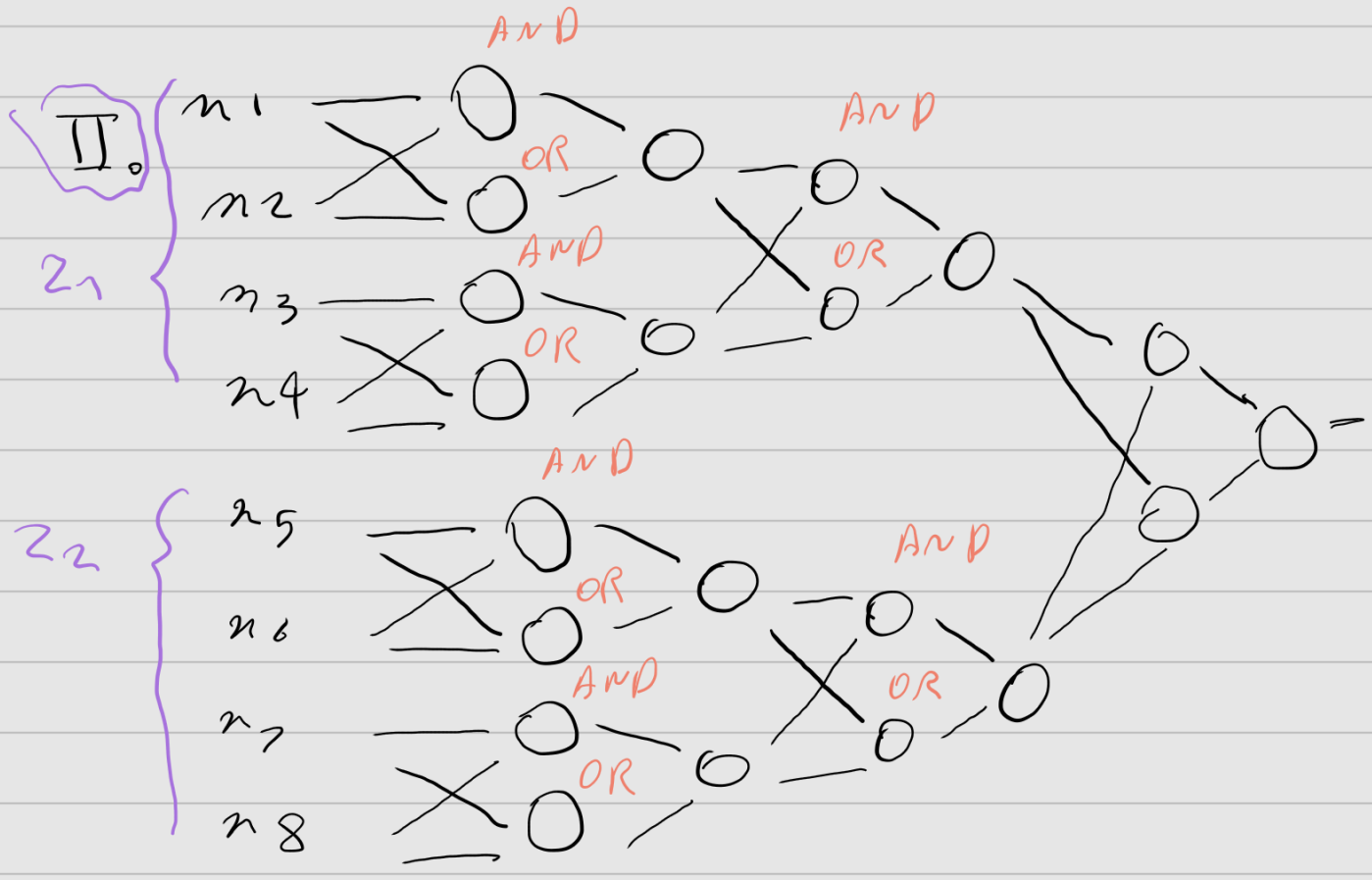
Q3

I.



hidden layers

II.



$$(x_1 \oplus x_2 \oplus x_3 \oplus x_4) \oplus (x_5 \oplus x_6 \oplus x_7 \oplus x_8)$$

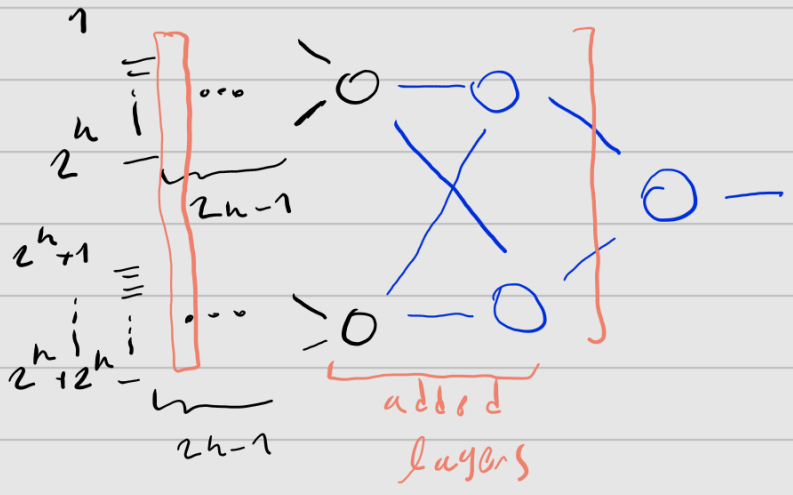
z_1 z_2

III, IV. If input $= N = 2^k$

hidden layers $= 2k-1$

proof: we know the base of induction ($k=1$) is true (1 hidden layer)

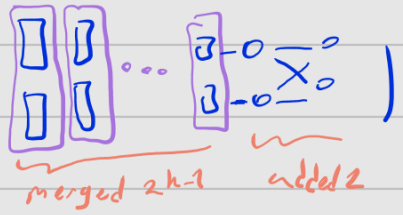
induction step: 2^n has $2n-1$ layers so 2^{n+1} has $2n-1+2$ hidden layers



$(x_1 \oplus x_2 \oplus \cdots \oplus x_{2^n})(x_{2^n+1} \oplus \cdots \oplus x_{2^n+2^n})$

needs $2n-1$ need $2n-1$

$2n-1$ first hidden layers merge (



)

so $n+1$ layers should have

$$2(n+1)-1 = (2n-1)+2$$

✓ proof complete

So we could conclude that having deep-NNs would decrease the number of needed neurons ($= N$) to a $\log(N)$ order