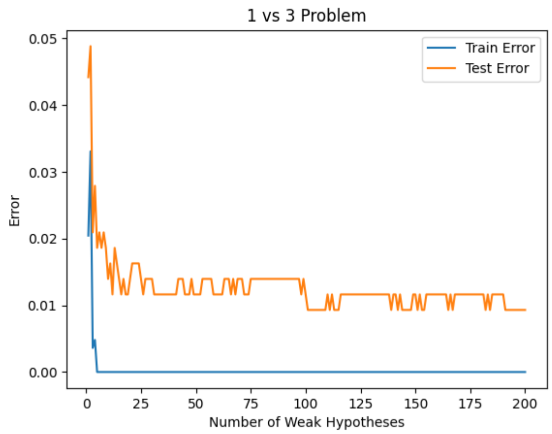
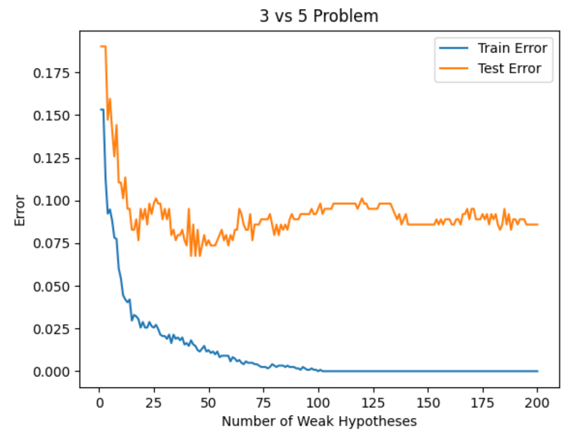
1)

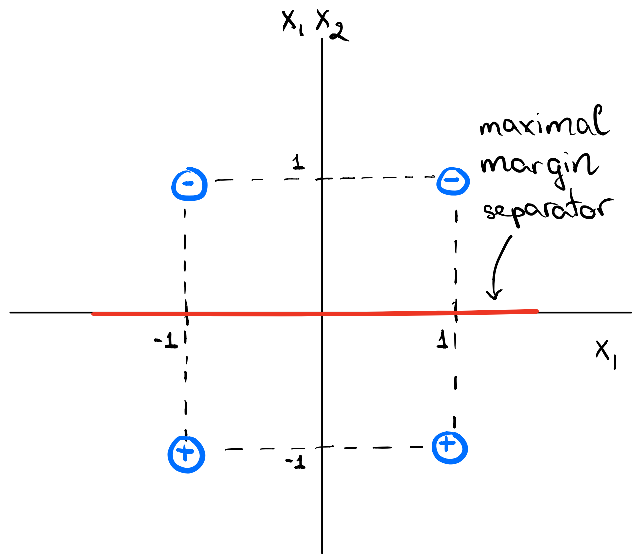




As the number of weak hypothesis increases, both train & test error decreases for both 1 vs 3 and 3 vs 5 Problems. As I have concluded from previous homework and the 2 above graphs, 3 vs 5 problem seems to be more difficult to classify since its test error is higher compared to the 1 vs 3 Problem. I noticed that in both problems, when number of weak hypothesis increases, train error reaches 0 and stay 0 and test error sometimes goes up (especially in the 3 vs 5 Problem) so it seems like Adaboost is overfitting as the number of weak learners increases.

2)

x_1	x_2	$x_1 x_2$	$XOR(x_1, x_2)$
+1	+1	+1	-1
+1	-1	-1	+1
-1	+1	-1	+1
-1	-1	+1	-1

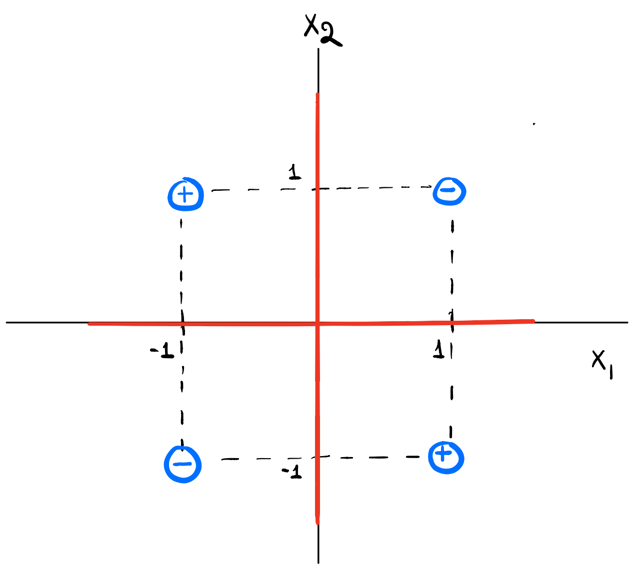


↳ Margin = 1

Linear separator : $h(\vec{x}) = \text{sign}(\vec{w}^T \vec{x}) = \text{sign}(-x_1 x_2)$

We can observe that $x_1 x_2 = 0$ (horizontal separator)

$$\Rightarrow \begin{cases} x_1 = 0 \\ x_2 = 0 \end{cases}$$



3)

$$d\left(\phi(\vec{x}_i), \phi(\vec{x}_j)\right) = \left(\phi(\vec{x}_i) - \phi(\vec{x}_j)\right)^T \left(\phi(\vec{x}_i) - \phi(\vec{x}_j)\right)$$

$$= \left(\phi(\vec{x}_i)^T - \phi(\vec{x}_j)^T\right)\left(\phi(\vec{x}_i) - \phi(\vec{x}_j)\right)$$

$$= \phi(\vec{x}_i)^T \phi(\vec{x}_i) - \phi(\vec{x}_i)^T \phi(\vec{x}_j) - \phi(\vec{x}_j)^T \phi(\vec{x}_i) + \phi(\vec{x}_j)^T \phi(\vec{x}_j)$$

$$= \phi(\vec{x}_i)^T \phi(\vec{x}_i) - 2\phi(\vec{x}_i)^T \phi(\vec{x}_j) + \phi(\vec{x}_j)^T \phi(\vec{x}_j)$$

$$= K(\vec{x}_i, \vec{x}_i) - 2K(\vec{x}_i, \vec{x}_j) + K(\vec{x}_j, \vec{x}_j)$$

4)

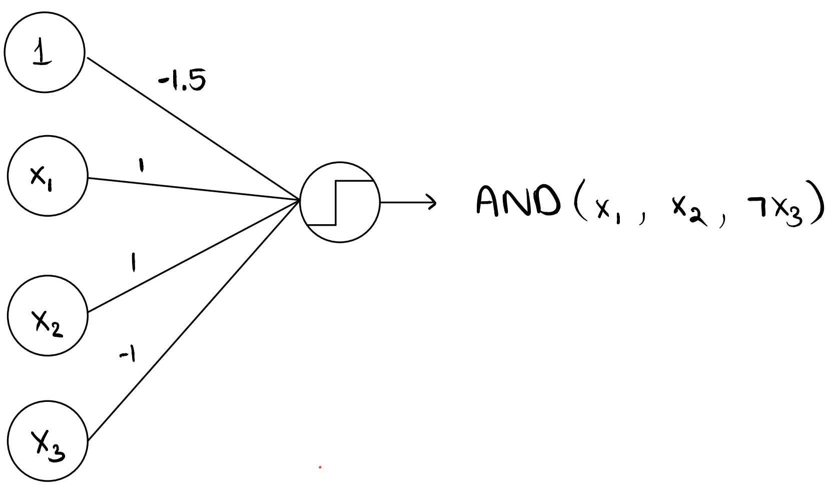
$$\mathrm{XOR}((x_1 \wedge x_2), x_3)$$

$$= ((x_1 \wedge x_2) \wedge \neg x_3) \vee (\neg(x_1 \wedge x_2) \wedge x_3)$$

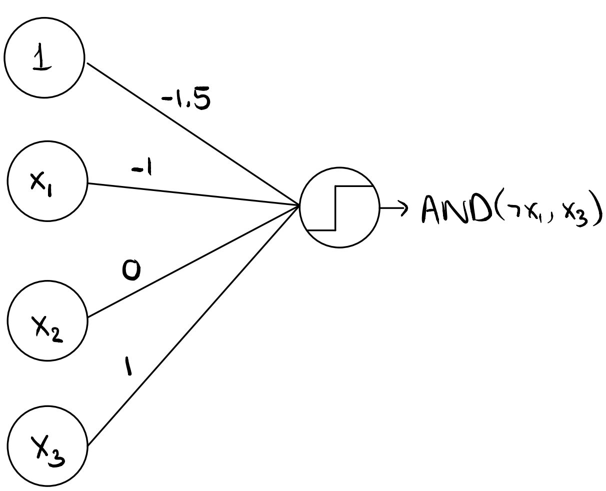
$$= (x_1 \wedge x_2 \wedge \neg x_3) \vee ((\neg x_1 \vee \neg x_2) \wedge x_3)$$

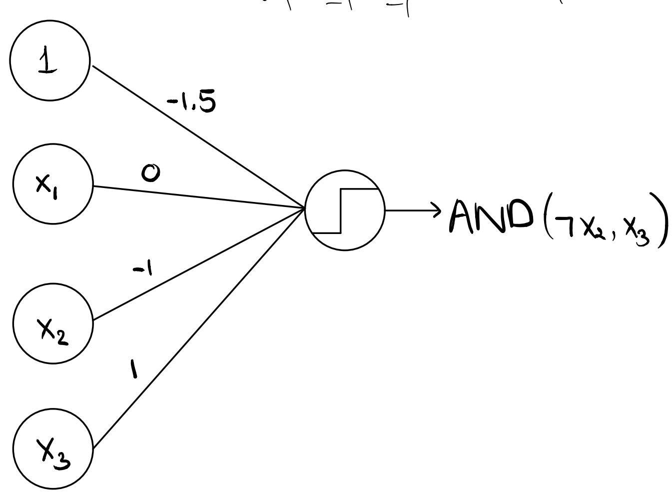
$$= (x_1 \wedge x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3) \vee (\neg x_2 \wedge x_3)$$

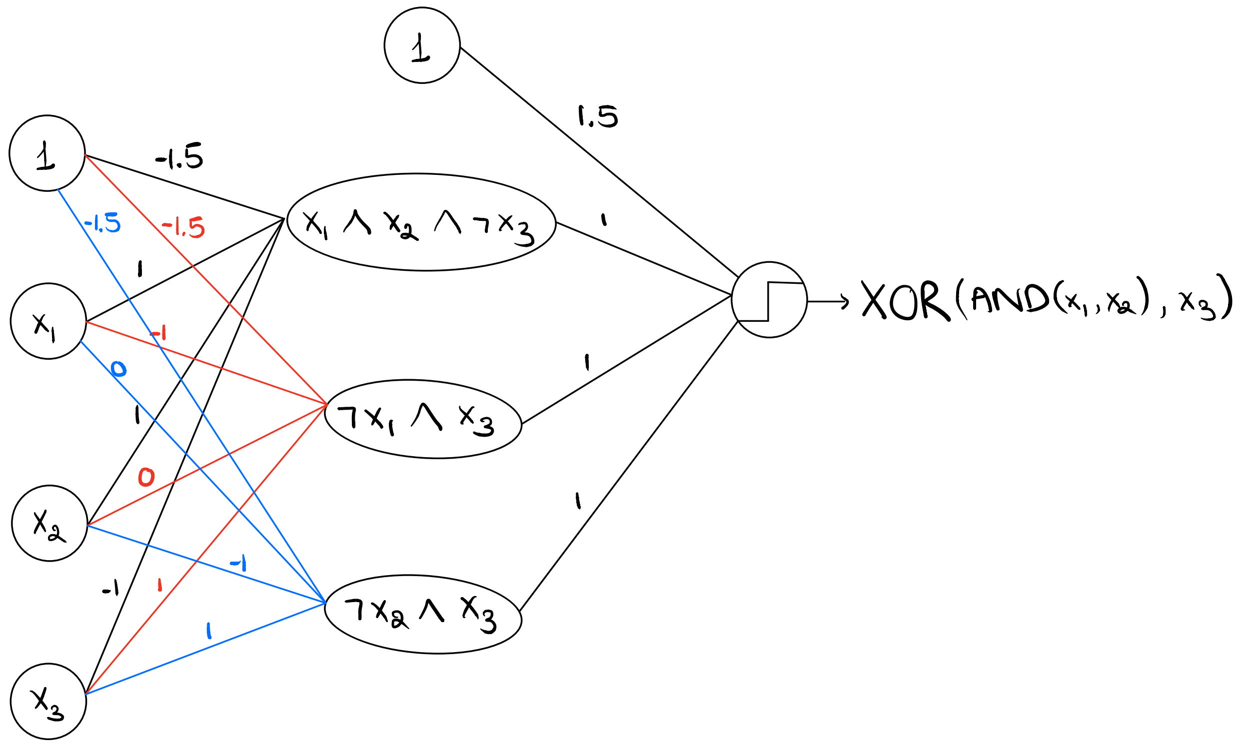
↳ 4 units in the hidden layer (1, $x_1 \wedge x_2 \wedge \neg x_3$, $\neg x_1 \wedge x_3$, $\neg x_2 \wedge x_3$)



x_1	x_2	x_3	$x_1 x_2 \neg x_3$
+1	+1	+1	-1
+1	+1	-1	+1
+1	-1	+1	-1
+1	-1	-1	-1
-1	+1	+1	-1
-1	+1	-1	-1
-1	-1	+1	-1
-1	-1	-1	-1







1
x1
x2
x3
1
-1.5
-1.5
-1.5
1
-1
0
1
0
-1
-1
1
1
x1 ∧ x2 ∧ ¬x3
¬x1 ∧ x3
¬x2 ∧ x3
1.5
1
1
1
XOR(AND(x1, x2), x3)

5, LFD Problem 7.10

If all weights are set to 0

↳ $W^{(\ell)} = 0 \quad , \ell = 1, \dots, L$

$$\Rightarrow \quad s^{(\ell)} = (W^{(\ell)})^T x^{(\ell-1)} = 0$$

$$x^{(\ell)} = \theta(s^{(\ell)}) = \theta(0) = \tanh(0) = 0$$

$$\Rightarrow \frac{\partial e}{\partial W^{(\ell)}} = x^{(\ell-1)} (\delta^{(\ell)})^T = 0 \quad \text{for } \ell = 2, \dots, L$$

$$\text{at } \ell = 1 \rightarrow \delta^{(1)} = \sum_{k=1}^{d^{(2)}} \underbrace{w_{jk}^{(2)}}_{=0} \delta_k^{(2)} \, \theta'(s_j^{(1)}) = 0 \quad \forall j$$

$$\Rightarrow \frac{\partial e}{\partial W^{(1)}} = 0 \quad \Rightarrow \frac{\partial e_n}{\partial W^{(1)}} = 0 \quad \text{for } \ell = 1, \dots, L$$

∴ It is not a good idea to initialize the weights to zero since the weights remain unchanged after each iteration $\left(W^{(\ell)} \leftarrow W^{(\ell)} - \eta \underbrace{\frac{\partial e_n}{\partial W^{(\ell)}}}_{=0} = W^{(\ell)} \right)$