$\boxed{1^\circ\ \neg x_1}$

$w = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} \quad x = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}$

$x_0 = 1$

$w_0 = 0.5$

$w_1 = -1$

2 cases

$1^\circ\ x_1 = 0 \Rightarrow f(x) = 1$

$f(x) = \text{sign}(0.5 \cdot 1 - 1 \cdot 0) = \text{sign}(0.5) = 1$

$2^\circ\ x_1 = 1 \Rightarrow \cancel{f(x) = 0}\ f(x) = -1$

$f(x) = \text{sign}(0.5 \cdot 1 - 1 \cdot 1) = \text{sign}(-0.5) = -1$

$\boxed{2^\circ\ x_1 \wedge \neg x_2}$

X_1	X_2	$\neg X_2$	← $X_1 \wedge \neg X_2$
0	0	1	0
0	1	0	0
1	0	1	1
1	1	0	0

$w = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} \quad x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$

$x_0 = 1 \quad w_0 = -0.5$

$w_1 = 1$

$w_2 = -1$

$\boxed{\text{4 cases}}$

$1^\circ\ x_1 = 0 \quad x_2 = 0 \Rightarrow f(x) = -1$

$f(x) = \text{sign}(-0.5 \cdot 1 + 1 \cdot 0 - 1 \cdot 0) = \text{sign}(-0.5) = -1$

$2^\circ\ x_1 = 0 \quad x_2 = 1 \Rightarrow f(x) = -1$

$f(x) = \text{sign}(-0.5 \cdot 1 + 1 \cdot 0 - 1 \cdot 1) = \text{sign}(-1.5) = -1$

$3^\circ\ x_1 = 1 \quad x_2 = 0 \Rightarrow f(x) = 1$

$f(x) = \text{sign}(-0.5 \cdot 1 + 1 \cdot 1 - 1 \cdot 0) = \text{sign}(0.5) = 1$

4° $x_1 = 1$ $x_2 = 1 \Rightarrow f(x) = -1$

$\cancel{f}(x) = \text{sign}(-0.5 \cdot 1 + 1 \cdot 1 - 1 \cdot 1) = \text{sign}(-0.5) = -1$

$\boxed{3^{\circ} \quad x_1 \text{ XOR } x_2}$

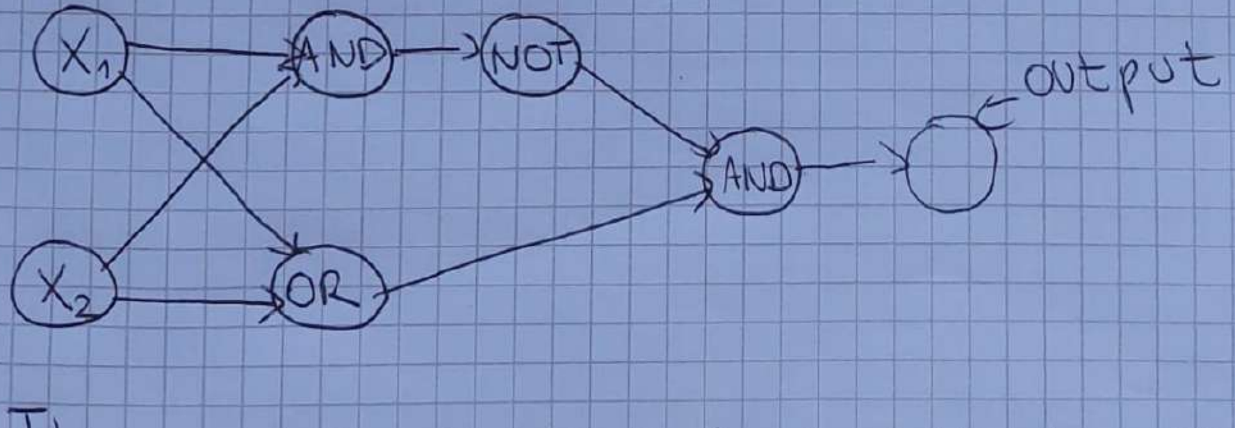
Since this problem is not linearly separable, I have to use multiple perceptron neurons. We already know that AND, OR and NOT operations can be computed by single perceptron neuron. I will use combination of those three operations in order to compute XOR.

x_1 XOR x_2 can be represented as: $\neg(x_1 \land x_2) \land (x_1 \lor x_2)$

- I will call this operation XOR'. (in the table)

x_1	x_2	$\land$	$\lor$	$\neg(\land)$	XOR	XOR'
0	0	0	0	1	0	0
0	1	0	1	1	1	1
1	0	0	1	1	1	1
1	1	1	1	0	0	0

- The table shows that XOR and XOR' operations are indeed the same. (give the same results).



- This graph is simplified, it doesn't show the summations, weights and other formulas. It is done for simplicity, since we have already showed those details in separate simple operations.