Exercise 11

Machine Learning I WORK IN PROGRESS

11A-2.

Just plug in the identity $h(a) = a$ and calculate.

$$\begin{aligned}
y_k &= \sum_{j=1}^{M} w_{kj}^{(2)} h\left(\sum_{i=0}^{D} w_{ji}^{(1)}\, x_i\right) \\
&= \sum_{j=1}^{M} w_{kj}^{(2)} \sum_{i=0}^{D} w_{ji}^{(1)}\, x_i \\
&= \sum_{j=1}^{M} \sum_{i=0}^{D} w_{kj}^{(2)} w_{ji}^{(1)}\, x_i \\
&= \sum_{i=1}^{D} \sum_{j=0}^{M} w_{kj}^{(2)} w_{ji}^{(1)}\, x_i \\
&= \sum_{i=0}^{D} x_i \underbrace{\sum_{j=1}^{M} w_{kj}^{(2)} w_{ji}^{(1)}}_{\beta_i} \\
&= \sum_{i=0}^{D} x_i \beta_i \\
&= \sum_{i=1}^{D} x_i \beta_i + \beta_0
\end{aligned}$$

(CHECK RESULT)

11A-3.

Even though it is written in the task, it always helps to write a truth table:

X_0	X_1	$X_0 \oplus X_1$
-1	-1	0
-1	1	1
1	-1	1
1	1	0

In addition, let us define the weights as follows:

$$w_{1,1}^{(1)} = 1, \qquad w_{1,2}^{(1)} = 1, \qquad w_{1,0}^{(1)} = -1 \; w_{2,1}^{(1)} = -1, \qquad w_{2,2}^{(1)} = -1, \qquad w_{2,0}^{(1)} = -1$$

$$w_{1,1}^{(2)} = -2, \qquad w_{2,1}^{(2)} = -2, \qquad w_{2,0}^{(12)} = 1$$

Let h_1, h_2 be two hidden layer output functions and h_{out} be the last layer output function:

$$h_1 = RELU(1x_1 + 1x_2 - 1)$$

$$h_2 = RELU(-1x_1 - 1x_2 - 1)$$

$$h_{out} = -2h_1 - 2h_2 + 1$$

Drawn, the layout looks like this:

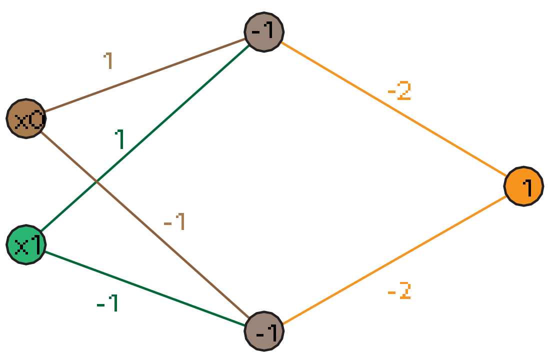

Figure 1 XOR Function realized with the wanted network. The values in the nodes represent the bias term $w_{i,0}^{l}$ of each node.

Inputting the values into f we notice:

$$f(-1,-1) = -2[RELU(-1-1-1)] - 2[RELU(1+1-1)] + 1 = -1$$

$$f(-1,1) = -2[RELU(-1+1-1)] - 2[RELU(1-1-1)] + 1 = 1$$

$$f(1,-1) = -2[RELU(1-1-1)] - 2[RELU(-1+1-1)] + 1 = 1$$

$$f(1,1) = -2[RELU(1+1-1)] - 2[RELU(-1-1-1)] + 1 = -1$$

Why does it work?

The two interior nodes act as AND gates, that output *true* if $x_0 = x_1$. We then negate their input and receive the corresponding output.

If $x_0 \neq x_1$, both AND gates are shut. Because the last bias $w_{1,0}^{2} = 1$ is constantly true, we will output *true* if both inputs are different.