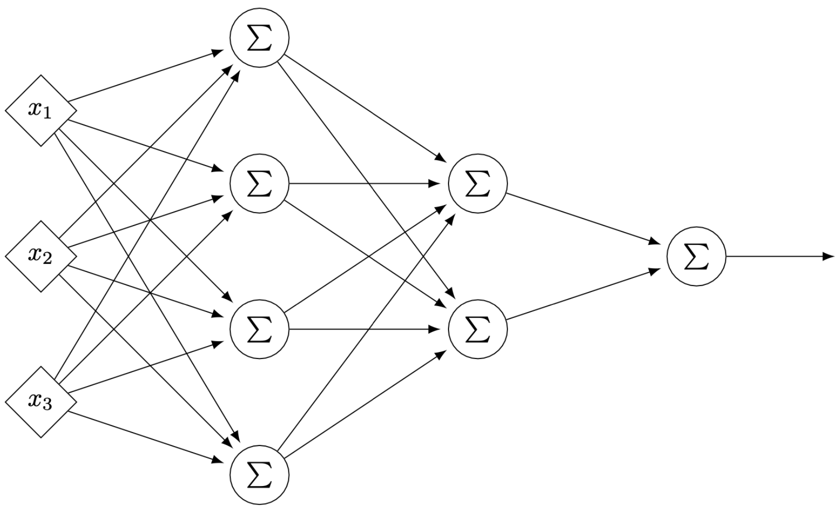
DSC 140A - Homework 05

Due: Wednesday, February 15

Write your solutions to the following problems by either typing them up or handwriting them on another piece of paper. Unless otherwise noted by the problem's instructions, show your work or provide some justification for your answer. Homeworks are due via Gradescope at 11:59 p.m.

Problem 1.

Consider the neural network shown below.



The weights and biases of the network are as follows:

$$W^{(1)} = \begin{pmatrix} -4 & 9 & 4 & 0 \\ -3 & -4 & 8 & 0 \\ 0 & -7 & -3 & -8 \end{pmatrix} \qquad W^{(2)} = \begin{pmatrix} -9 & 1 \\ -5 & -9 \\ -10 & 1 \\ 1 & 6 \end{pmatrix} \qquad W^{(3)} = \begin{pmatrix} -1 \\ 5 \end{pmatrix}$$

$$\vec{b}^{(1)} = \begin{pmatrix} 4 & 4 & 8 & 1 \end{pmatrix}^T \qquad \vec{b}^{(2)} = \begin{pmatrix} 9 & -8 \end{pmatrix}^T \qquad \vec{b}^{(3)} = \begin{pmatrix} -6 \end{pmatrix}$$

All neurons use linear activation functions.

You're encouraged to write code to do any and all parts of this problem. If you do, please show your code.

a) What is the output of the network if it is given $\vec{x} = (1, 2, 3)^T$ as input?

Solution: 119.

Assuming that the matrices above have been defined, we can compute the answer with the following code:

```
def H_1(z):
    return W1.T @ z + b1

def H_2(z):
    return W2.T @ z + b2
```

```python
def H_3(z):
    return W3.T @ z + b3

def H(z):
    return H_3(H_2(H_1(z)))

H((1, 2, 3))
```

Here, we're defining functions for each layer. The overall network is composed of all of these functions.

You could also do this one line as follows: `W3.T @ (W2.T @ (W1.T @ (1, 2, 3) + b1) + b2) + b3`

b) Recall from lecture that a neural network using linear activation functions is a linear predictor.

More formally, this means that the neural network above should be equivalent to a function $\tilde{H}(\vec{x}) = \vec{w} \cdot \text{Aug}(\vec{x})$, for some $\vec{w}$.

Find this $\vec{w}$. That is, find a parameter vector $\vec{w}$ so that the output of the network above is equal to $\vec{w} \cdot \text{Aug}(\vec{x})$ for all $\vec{x} \in \mathbb{R}^2$.

Hint: if $H(\vec{x})$ is the neural network above, write $H(\vec{x})$ using matrix multiplications (you might have already done this in the last part), then simplify as much as possible to get into the form $\vec{a} \cdot \vec{x} + c$. You can check your work by making sure that $\vec{w} \cdot (1, 1, 2, 3)^T$ is the same number you got in the last part.

Solution:

$\vec{w} = (-10, -356, 238, 3)^T$.

There's two ways of doing this: the easy way, and the hard way.

First, the hard way. Using the fact that a neural network is the composition of layer outputs:

$$H(\vec{x}) = [W^{(3)}]^T \left([W^{(2)}]^T \left([W^{(1)}]^T \vec{x} + \vec{b}^{(1)} \right) + \vec{b}^{(2)} \right) + \vec{b}^{(3)}$$

Multiplying this out:

$$\begin{aligned} &= [W^{(3)}]^T \left([W^{(2)}]^T [W^{(1)}]^T \vec{x} + [W^{(2)}]^T \vec{b}^{(1)} + \vec{b}^{(2)} \right) + \vec{b}^{(3)} \\ &= \underbrace{[W^{(3)}]^T [W^{(2)}]^T [W^{(1)}]^T}_{\vec{a}} \vec{x} + \underbrace{[W^{(3)}]^T [W^{(2)}]^T \vec{b}^{(1)} + [W^{(3)}]^T \vec{b}^{(2)} + \vec{b}^{(3)}}_{c} \end{aligned}$$

Next we need to calculate. In code, we calculate $\vec{a}$: `W3.T @ W2.T @ W1.T`. The result is $(-356, 238, 3)^T$.

Then we calculate c: `W3.T @ W2.T @ b1 + W3.T @ b2 + b3`. The result is -10.

All together, the solution is $\vec{w} = (-10, -356, 238, 3)^T$.

Now, the "easy" way. Recognize that if H is supposed to be a linear function of the form $w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3$, we can calculate w_0 by computing $H((0, 0, 0)^T)$. Using our code from part A, we find $H(0) = -10$.

Next, note that $H((1, 0, 0)^T) = w_0 + w_1$. If we compute $H((1, 0, 0)^T)$ from our code above, we get -366. But since this is $w_0 + w_1$, we need to subtract $w_0 = -10$, which gives $w_1 = -356$.

This can be repeated to get the other two weights.