

---

# Tutorial Questions | Week 3

## COSC2779 - Deep Learning

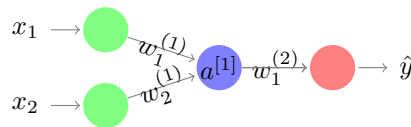
---

This tutorial is aimed at reviewing feed forward neural networks. Please try the questions before you join the session.

1. For a fully-connected deep network with one hidden layer, increasing the number of hidden units should have what effect on bias and variance?

**Solution:** Adding more hidden units should decrease bias and increase variance. In general, more complicated models will result in lower bias but larger variance, and adding more hidden units certainly makes the model more complex.

2. Consider the following one hidden layer network.



$$a^{[1]} = \sigma_1 \left( w_1^{(1)} x_1 + w_2^{(1)} x_2 + b^{(1)} \right)$$

$$y = \sigma_2 \left( w_1^{(2)} a^{[1]} + b^{(2)} \right)$$

Show that if  $\sigma_1$  is linear, the network can be represented by one layer perceptron.

**Solution:** Linear activation means  $\alpha z = \sigma(z)$

$$y = \sigma_2 \left( w_1^{(2)} a^{[1]} + b^{(2)} \right)$$

$$y = \sigma_2 \left( w_1^{(2)} \alpha \left( w_1^{(1)} x_1 + w_2^{(1)} x_2 + b^{(1)} \right) + b^{(2)} \right)$$

$$y = \sigma_2 \left( \alpha w_1^{(2)} w_1^{(1)} x_1 + \alpha w_1^{(2)} w_2^{(1)} x_2 + \alpha w_1^{(2)} b^{(1)} + b^{(2)} \right)$$

If we define

$$\omega_1 = \alpha w_1^{(2)} w_1^{(1)}$$

$$\omega_2 = \alpha w_1^{(2)} w_2^{(1)}$$

$$\beta = \alpha w_1^{(2)} b^{(1)} + b^{(2)}$$

$$y = \sigma_2 (\omega_1 x_1 + \omega_2 x_2 + \beta)$$

3. You want to map every possible image of size  $64 \times 64$  to a binary category (cat or non-cat). Each image has 3 channels and each pixel in each channel can take an integer value between (and including) 0 and 255. How many bits do you need to represent this mapping?

**Solution:**  $256^{3 \times 64 \times 64}$

4. The mapping from question (3) clearly can not be stored in memory. Instead, you will build a classifier to do this mapping. You will use a single hidden layer of size 100 for this task. Each weight in the two weight matrices can be represented in memory using a float of size 64 bits. How many bits do you need to store your two layer neural network?

**Solution:**  $64(100(64 \times 64 \times 3 + 1) + (100 + 1))$

5. One of the difficulties with the logistic activation function is that of saturated units. Briefly explain the problem, and whether switching to tanh fixes the problem. Recall:

$$\sigma(z) = \frac{1}{1 + \exp(-z)}$$

$$\tanh(z) = \frac{\exp(z) - \exp(-z)}{\exp(z) + \exp(-z)}$$

**Solution:** No, switching to tanh does not fix the problem. The derivative of  $\sigma(z)$  is small for large negative or positive  $z$ . The same problem persists in  $\tanh(z)$ . Both function has a sigmoidal shape. We can see tanh is effectively a scaled and translated sigmoid function:  $\tanh(z) = 2\sigma(2z) - 1$ .

6. You are asked to develop a NN to identify if a given images contain a cat and/or a dog. *Note that some images may contain both a cat and a dog.* What will be a possible output activation and loss function?

**Solution:**

Option 1: Have 4 outputs (none, cat, dog, both) and softmax. use categorical-cross-entropy.

Option 2: Have two sigmoid activated units. Use combined loss function of two binary-cross-entropy summed.