CSci 4270 and 6270
Computational Vision,
Spring Semester, 2021
Lecture 15 Exercise — Neural Network Feed Forward Computation
Due: Saturday, March 20 at 11:59pm EST

In this exercise you will implement the basic feed-forward computation of a neural network. You will be given two files for a feed-forward network, one containing the weights and biases of the network and the other containing bottom layer input vectors (the **x** vectors). The network is input in the form of a NumPy .npz file, while the input vectors are in a Numpy .npy file. We've provided functions for you to do this input. In particular, the network is returned to you as a list of weight matrices and bias vectors. If we call these lists `w_list` and `b_list` then `w_list[i]` and `b_list[i]` are the weight matrix and bias vectors for layer `i+1` of the network as described in lecture. The input vector list will contain just one 2d array. Each column of the array will be a different input **x** vector.

Your first output should be a Python list given the number of neurons at each layer.

For each input vector you must compute the feed forward propagation of the network to the last layer. Apply the ReLU activation function at the end of each layer's computation. Then output the index and value of both the minimum activation and the maximum activation. The index of the maximum activation is the "decision" of the network. These outputs should be accurate to three decimal places.

You will need to write a for loop over the indices of the layers, but interestingly, you do not need a for loop over the input vectors, at least not during the feed forward computation. You will not be penalized if you do not do this, however.

The command line will be simply

```
lec15_ex root
```

where if `root == 'foo'` then the network will be read from `foo_nn.npz` and the input data vectors from `foo_x.npy`.

Finally, note that there is no meaning to the examples given here: the data were generated randomly.