CS-419m: Practice question set 4

1. A feed forward neural network can represent any function. Our goal is to design a neural network to multiply two integers $\mathbf{x}^1, \mathbf{x}^2$ expressed as k bit binary vectors $x_1^1, \ldots, x_k^1$ and $x_1^2, \ldots, x_k^2$ respectively. For example, if $k = 3, \mathbf{x}^1 = [1\ 0\ 1], \mathbf{x}^1 = [0\ 1\ 0]$, then the network should output 10. Show the layers and the optimal values of each parameter. Use ReLU as the activation function. [Your solution cannot involve a multiplication of two hidden outputs or two inputs. You are only allowed linear transforms followed by ReLU in standard feed forward networks.] At the first layer, we maintain k^2 units. The i,j-th unit computes $h_{i,j}^1 = ReLU(2^i 2^j + Mx_j^1 + Mx_i^2 - 2M)$ where M is greater than 2^{2k}. When $x_i^1 = x_j^2 = 1$ the value of $h_{i,j}^1 = 2^i 2^j$ Else it is zero. The next layer simply adds these values.

2. Let W_{jk}^l denote the weight connecting jth output of layer $l-1$ to kth unit of layer l in a 2 hidden-layer feed-forward network with a last linear layer and a least square loss. The activation unit at each hidden layer is a ReLU. For an example $(\mathbf{x}, y)$, let h_k^l denote the kth output of the l-th layer o denote the output from the network. Write the expression for the gradient of loss on this example wrt W_{12}^1. Let the number of hidden units be m in each of the two hidden layers. Your answer should be in the full expanded form and not in recursive form like in Backpropagation.

   $-(y-o)\sum_{j=1}^m W_{j1}^3 \delta(h_j^2 > 0) W_{2j}^2 \delta(h_2^1 > 0) x_1$