HW #4 Due: 5/30/2018

1. Assuming that the following is a part of convolution neural networks. Compute the resultant values if it has two input planes (channels), stride of **two**, no zero-padding, and using the ReLU activation function.

Input plane (channel) 1

6	0	-4	0	1
4	4	0	2	1
3	-7	1	4	2
-2	2	1	-4	2
5	1	2	4	-1

Kernel

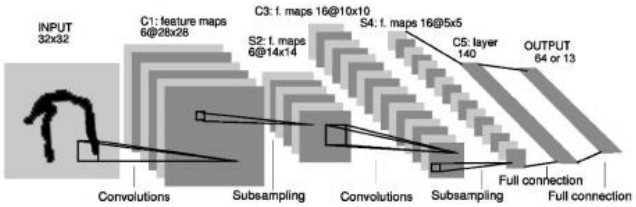
0	1	0
-2	0	-2
0	1	0

Input plane (channel) 2

3	1	-4	0	-1
3	0	3	4	-1
3	7	-2	-2	2
-5	-2	2	0	-2
2	1	-1	1	2

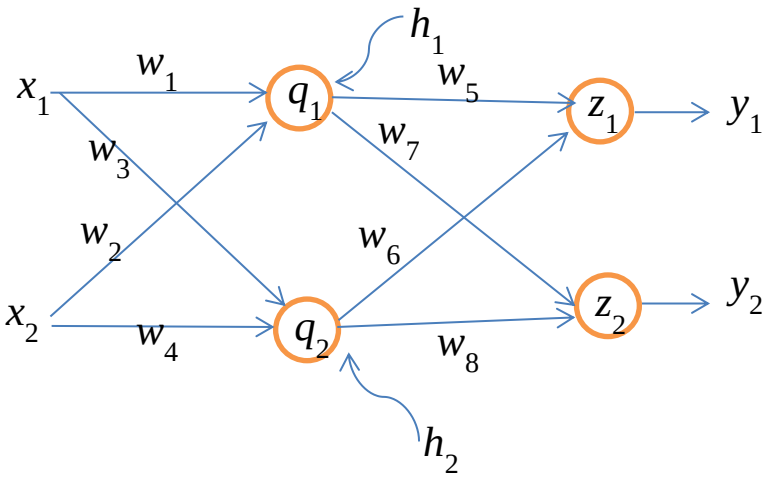
0	1	0
1	-4	1
0	1	0

2. The following shows the LeNet-5 architecture. If we use the weight-sharing approach for the convolutional layers, compute the number of connections and trainable weights of the network. To compute the results, you need the following parameters for the convolutional layers: kernel size = 5x5, stride = 1, and no zero-padding. To simplify the computations, ignore the bias weights.



3. For the neural network given below, the activation function from q_1 to h_1 and q_2 to h_2 is ReLU, the activation function at the output nodes remain sigmoid, and the cost function is still MSE. Let w_1 to w_8 be 0.5, $d_1 = 0.9$, $d_2 = 0.1$, and $\eta = 0.1$.
   i. Find y_1 and y_2 (forward computation) if $x_1 = x_2 = 0.5$.

ii. Find the value of $\Delta w_1 = \eta \frac{\partial J}{\partial w_1}$ and Δw_5 using back propagation if $x_1 = x_2 = 0.5$



4. Show the following is true. If $y_1 = \frac{1}{1+exp(-z_1)}$, then $\frac{\partial y_1}{\partial z_1} = y_1(1 - y_1)$.

5. Justify the following claim: The back propagation algorithm does NOT work if weights in the neural network are all set to zero initially.