CS 6320 Computer Vision

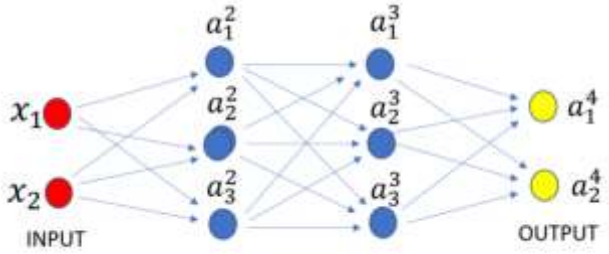
Homework 5 (Due Date – April 15th)

1. Install any deep learning package (Matconvnet, Caffe, tensorflow) and test the MNIST digit recognition program. You don't have to understand the details of the program, but try to change the number of layers or other parameters and observe the change in the accuracy of the digit recognition. The following websites will be useful:

   http://www.vlfeat.org/matconvnet/training/
   https://www.tensorflow.org/get_started/mnist/beginners
   http://caffe.berkeleyvision.org/gathered/examples/mnist.html

   Show the outputs for 2 different parameter settings. [25 points]

2. Use backpropagation to compute to compute all the gradients $\partial C/\partial w_{ij}^{l}$ and $\partial C/\partial b_{i}^{l}$.



Weights: $w^2 = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}, w^3 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, w^4 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$

Bias Terms: $b^2 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, b^3 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, b^4 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

Input: $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

Groundtruth output: $\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

`

The activation function to be used is RELU. In other words, the basic feedforward equations are given below:

$$z_j^2 = \sum_{k=1}^{2} w_{jk}^2 x_k + b_j^2$$

$$z_j^3 = \sum_{k=1}^{3} w_{jk}^3 a_k^2 + b_j^3$$

$$z_j^4 = \sum_{k=1}^{3} w_{jk}^4 a_k^3 + b_j^4$$

$$a^l = RELU(z^l) = \max(z^L, 0)$$

You can use the following derivatives for RELU:

$$\frac{\partial RELU(z)}{\partial z} = RELU'(z) = \begin{matrix} 1 \; if \; z > 0 \\ 0 \; otherwise \end{matrix}$$

We use a quadratic cost function as given below:

$$C = (y_1 - a_1^4)^2 + (y_2 - a_2^4)^2$$

To find the gradients, you will have to do the forward propagation once and then do the backpropagation once.

[25 points]

3. Use threshold activation functions to implement the following function $f(x, y)$ as shown below:

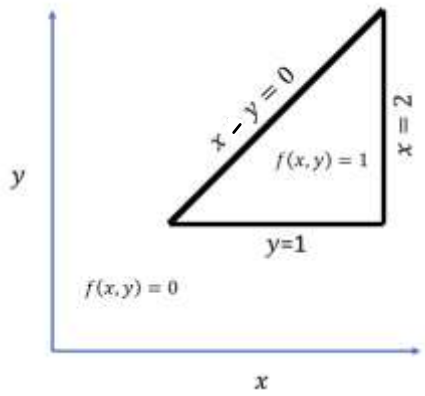


Figure 3: A simple function $f(x, y)$ that takes the value 1 inside the triangle and 0 otherwise.

Consider a linear threshold unit T. Let $\{x_1, x_2, \dots, x_n\}$ be the real inputs to the linear threshold unit and $\{w_1, w_2, \dots, w_n\}$ be the learned real weights and let b be the bias term. Then the output from T will be 1 if $\sum_{i=1}^{n} w_i x_i + b \geq 0$ and 0 otherwise. Consider a function $f(x, y)$ that takes two real inputs $\{x, y\}$ and gives Boolean output 1 or 0 as shown in Fig 3. Build the function using linear threshold units. You are free to use as many linear threshold units as you need. Manually come up with weights and biases for each linear threshold unit.

[25 points]

4. We are given a deep neural network DNN for alphabet classification. The first layer is the input layer with 784 neurons coming from 28x28 image pixels. There is only one channel in the input. The second layer is a convolution layer with 3 filters, each of size 7×7 with stride 2. Assume that the convolution layer uses bias terms. We do not use any padding for convolution. The output from the second layer passes through RELU activation unit (RELU(z) = max(z,0)). The third layer is a 2 × 2 max-pooling layer with no padding and stride 2. The fourth layer is a fully connected layer with 26 neurons for classifying the 26 alphabets. The final fifth layer is a softmax layer. The softmax layer produces 26 outputs, where each output could vary from 0 to 1. After training, an input of alphabet "a" would produce an output vector close to ([1,0,0,0,0,0,0,0,0,0,…]) for most of the time.

   (a) Show the dimensions of the layers 2 and 3.
   (b) What is the total number of parameters (weights and biases) in the network?
   (c) Can you think of having fewer than 26 output neurons to classify 26 alphabets? In that case, would you need a softmax layer?

   [25 points]