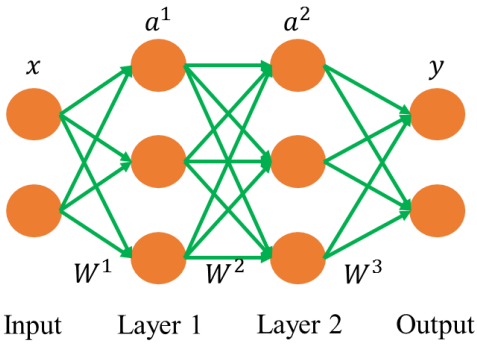
CS 5984: Deep Learning

Homework 1

Due Date: October 5th, 2017 (5:00PM) **Total: 100 Points**

Note: All implementations are required to be accomplished and submitted using the *Jupiter notebook*. Show your results and necessary comments in the notebook. *No handwritten homework is accepted.*

Consider the following deep feedforward neural network with two hidden layers.



Assumptions:

a. We have zero biases in each layer, i.e., $z^1 = W^1 x$, $z^2 = W^2 a^1$, $z^3 = W^3 a^2$.
b. The activation function for two hidden layers are sigmoid function, i.e., $a^1 = \sigma(z^1)$ and $a^2 = \sigma(z^2)$, where $\sigma(z) = \frac{1}{1+e^{-z}}$.
c. The activation function for the output layer is a softmax function, i.e., $y_i = \frac{e^{z_i^3}}{\sum_j e^{z_j^3}}$.
d. The loss function is given by $L = \|y - \hat{y}\|^2 = (y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2$.

Questions:

1. Using backpropagation method and the computational graph learned in the class to derive $\frac{\partial L}{\partial W^1}, \frac{\partial L}{\partial W^2}, \frac{\partial L}{\partial W^3}$. Show the details of your work and solutions.
2. Implement the stochastic gradient decent (SGD) algorithm learned in the class to train the above neural network.
   a. The weights W^1, W^2, W^3 are randomly initiated.
   b. Define a learning rate η for the SGD and set it as a small number.
   c. In each training step, print the value of loss L. Print out the final weights.
   d. The training data has been provided in a separate file.

   In this question, you cannot call the optimization functions in the tensorflow, theano or other deep learning frameworks.

3. Train the above the neural network on the same training data using the deep learning frameworks. You are free to choose any python package, but your code and results should presented in Jupiter Notebook.
4. Evaluate models that obtained in question 2 and 3 with the testing data using metrics *precision, recall and f-score*. Show the details of your work.