

Homework #3

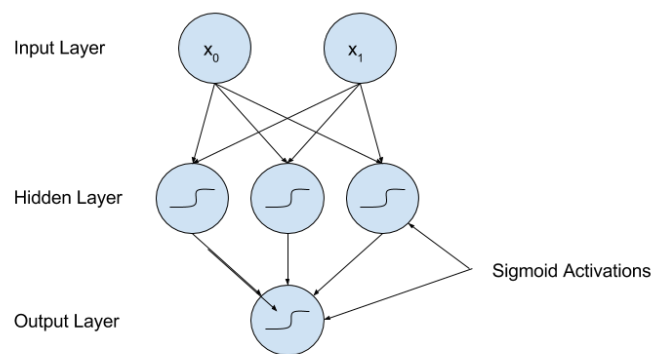
MTH 9899 Baruch College
DATA SCIENCE II: Machine Learning

Due: May 7, 2017 - 18:00

Notes

- Code for this **MUST** be written in **Python 3.x**.
- Do NOT use 3rd Party Packages for the regression functions unless specified.
- **The Due Date is Sunday night, not at the beginning of class**

Problem 1 In class, we went through the details of manually calculating backprop on a simple network. For this question, we're going to do backpropagation using numpy, and no 3rd party packages. The goal is to train a simple network with 2 input features and 3 hidden layers. Both the hidden layers and the output layer should use a sigmoid activation function. For loss, please use mean squared error (MSE). Below is a picture of the network and a listing of code to generate sample and test data. Please train your network on 50,000 input rows in a single training batch. Graph the loss function vs the iteration. Repeat this graph using different values of η , the learning rate. How does this affect the loss?



```

def get_test_data(num_pts):
    centers = [(0,0,1), (1,1,-1), (1,-1,1), (-2,0,-1), (4,1,1)]

    X = np.random.rand(num_pts, F)*10 - 5

    def get_y(x):
        distances = [np.sqrt(np.sum((x - c[:2])**2)) for c in centers]
        which = np.argmin(distances)
        return centers[which][2] * distances[which]

    Y = np.array([get_y(x) for x in X])
    return X,Y

```