Write a Python program that trains a single layer neural network

with sigmoid activation. You may use numpy. Your input is in dense

liblinear format which means you exclude the dimension and include 0's.

Let your program command line be:

python single\_layer\_nn.py <train> <test> <n>

where n is the number of nodes in the single hidden layer.

For this assignment you basically have to implement gradient

descent.

Test your program on the XOR dataset:

1 0 0

1 1 1

-1 0 1

-1 1 0

1. Does your network reach 0 training error?

2. Can you make your program into stochastic gradient descent (SGD)?

3. Does SGD give lower test error than full gradient descent?

4. What happens if change the activation to sign? Will the same algorithm

work? If not what will you change to make the algorithm converge to a local

minimum?