

Instructor: Prof. R. Newcomb

TA: Cagdas Dirik cdirik@glue.umd.edu

TA Office Hours: Tu. 13.00:13.50 EGL 1153/1164

Homework # 2(due 02/21/2002)

(3rd and 5th questions will be graded, each 50 pts.)

0. Read Neural Network Toolbox Manual Chapter 3. Pay attention to perceptron learning rules (starting section 3-12, page 82). Also take a look at section 3-35, page 105 to get overview of the perceptron architecture and be familiar with 3 basic formulas used in training of perceptrons.

1. Write a matlab script that creates a 2-neuron perceptron (implementing hardlim function) and simulates it. Inputs to this matlab script are:

- W: Weight matrix
- b: bias value
- P: matrix of inputs (each column corresponds to an input to the perceptron. So this will be a $2 \times p$ matrix where p is the number of inputs)

Output should be simulation of your inputs stored in matrix A

2. Practice perceptron-learning rules by running demo program `nnd4pr`. Write a matlab script that does a similar thing. Assume a single neuron perceptron with 2 inputs. Get 1 input-target pair as input argument to your matlab code; find the weight matrix and bias value for your perceptron to satisfy given targets. There are two options you may follow:

- You may either use `learnp` command of matlab
- Or you may write the code yourself following the three steps of learning (Case 1,2,3 of section 3.13, page 83 of matlab manual)

3. Now assume the same perceptron architecture modify your matlab code so that it gets 4 input-target pairs and trains the perceptron accordingly. Again you have two paths to follow: (50 pts)

- You may either use `train` command of Matlab
- Or you may modify your code by adding `for` loops so that you follow the 3 steps of learning in each loop for a specific input-target pair

4. Practice the GUI Network/Data Manager to design perceptron models. The manual discusses the details of how to use GUI by building a AND network.

5. Assuming a 2-neuron perceptron create a network such that one neuron simulates an ANDNet while the other simulates an ORNet. Also by hand or using matlab calculate number of epochs needed to train your perceptron and verify your results with the output graph, which shows the training results.

You are welcome to try to train your perceptron such that it responds as a XOR, NAND etc logic. (50 pts)