Project 3

for NeuroComputing/Theoretical Neuroscience 2016

This assignment is due before class on Thursday, March 3 by email to NeuroCompDal@gmail.com.

Part 1:

Use an attractor network as an autoassociator to recognize noisy versions of the letter patterns in file *pattern1* (the same patterns we used in the last project). How many letters can the network memorize? How does this compare to the theoretical capacity of the network?

Part 2 for NESC 4177:

a. Implement a multilayer perceptron that can translate the digital letters given in file *pattern1* into the corresponding binary ASCII representation. Plot a training curve and interpret your results.

c. Take the first 5 letters of the alphabet and produce several noisy versions of each letter where 20% of the bits are flipped. Out of these noisy examples form two datasets, one for training the network and one for testing the network. Plot training curve that shows the performance of the network in the test set.

Part 2 for CSCI 6508:

Implement an autoencoder for random binary pattern similar to the one discussed in class. The difference is that the random binary pattern should be made up of components 0 and 1 instead of the previous one with -1 and 1. Make sure that the network performs the same as the previous version by showing the load capacity of the network. Include your code in the report and comment on your findings.