CIS6930/4930 Intro to Computational Neuroscience Fall 2008
Home Work Assignment 4:
Due Tuesday 11/18/08 before class

1. (Asymptotic Equipartition) Consider the alphabet $\{$A,B$\}$ with probabilities p(A)=0.27 and p(B)=0.73. Verify that the entropy of this distribution is $H = 0.841464$. Now consider all 1048576 sequences of length 20 generated out of this alphabet. Choose a small ϵ, say 0.001, and find the number of sequences that have probabulities between $2^{-20*(H+\epsilon)}$ and $2^{-20*(H-\epsilon)}$ (when the sequences are generated i.i.d). How many bits would be necessary to encode for any one of these sequences assuming that you have a simple index table for these sequences? Now compute the sum of the probabilities of these sequences. Assuming that you use 20 bits to encode for the remaining sequences, how many bits *on average* would be necessary to encode a string of length 20?

2. Code and test a feed forward net of sigmoidal nodes with two input units, ten hidden units and one output unit that learns the concept of a circle in 2D space. The concept is: $\langle x, y\rangle$ is labeled "+" if $(x - a)^2 + (y - b^2) < r^2$ and is labeled "-" otherwise. Set $a = 2$, $b = 3$, $r = 1$.

   Generate 100 random samples uniformly distributed from a cell in $\mathbb{R}^2$ to train the network using error backpropagation and 100 random samples to test it. Repeat the procedure multiple times and with multiple initial weights. Report the changing accuracy and the hyperplanes corresponding to the hidden nodes.

3. Get 20 images from the internet and reduce them to black and white dithered images of size 50×50 pixels. Now train a Hopfield type recurrent net to retrieve these images. Finally, initialize the net with noisy versions of each of these images (randomly inverted pixels or cropped versions of the image) and see whether the net converges to the original images.

   Images can be displayed easily in pbm format. For example, copy the next few lines into a file called test.pbm and use your favourite viewer (irfanview, xv, eog) to view it.

```
P1
# Width, height of image. Pixel values follow: < 70 chars per line
10 20
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
```

0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1