Machine Learning 2007
(Home work 2)

June 29, 2007

- Due on Wednesday, July 20, before 4 p.m.
- Late submissions will not be accepted.
- Submit hard copy of the results, plots and your workings
- Submit a printed copy of the codes also.
- You may save time if you use MATLAB for the computations and plots.
- Please do not hesitate to contact me if you do not understand the problems.

1. **[10 +15 points] K-Means Algorithm**

   (a) In the course website you will find four files named, band1.irs, band2.irs, band3.irs and band4.irs. Download these files. These files contain a satellite image of Kolkata in 4 different bands. Use the k-means algorithm to cluster this data into 2, 4 and 6 clusters. Display the clusters suitably.

   (b) Download the file lena.pgm from the course website. This file contains a gray level picture of size 256×256. Our objective here is to compress this image using the k-means algorithm. Break this image into 4×4 blocks and consider these blocks as your data. Cluster this data into k clusters. Now, for each block b_i, replace the block b_i by the mean of the cluster in which b_i belong. Do this for $k = 100, 50$ and 10. Display the corresponding images. Comment on the amount of compression achieved in each case.

2. **[10 + 15 points] Multilayer Perceptrons**

   (a) In the class we considered training an MLP using an error function for data $\boldsymbol{x}^{(i)}$ of the form:

$$E(\boldsymbol{x}^{(i)}) = \frac{1}{2}\sum_{k=1}^{K}(y_k^{(i)} - z_k)^2,$$

where z_k represents the output of the k^{th} output node. This error function is called the sum of square error function. Instead of the sum of square error cost function consider the cross entropy function given by

$$E(\boldsymbol{x}^{(i)}) = \sum_{k=1}^{K} \ln \frac{y_k^{(i)}}{z_k}.$$

Find the update equation for the two sets of weights (input to hidden layer and hidden to output layer) using the above cost function.

(b) Write a program to implement an MLP with one hidden layer using the above error function. Train the network with trgIris.dat (the same data you used in homework 1). Do a 10 fold cross validation on the data. Repeat the experiment 10 times with different initializations and report the mean and standard deviation of the error.

3. **[10 + 15 points] Radial Basis Function Networks**

(a) Derive the update equation of the weights from the hidden to output layer of a radial basis function network considering the mean and spread of the hidden nodes fixed.

(b) Implement a radial basis function network. Use the k-means algorithm to fix the centers of the hidden nodes. Apply the method discussed in class to find the spreads. Train the network with trgIris.dat (the same data you used in homework 1). Do a 10 fold cross validation on the data. Repeat the experiment 10 times with different initializations and report the mean and standard deviation of the error.