CS 410/510
Machine Learning
Spring, 2007

Homework 6:
Neural Networks and Dimensionality Reduction

Discussion of results-so-far in class: Tuesday, May 22.

Final version due Tuesday, May 29.

In this assignment, you will build and train an auto-associator neural network, and use the results for a hand-written-digits classification problem.

For this assignment, you may work in teams of two people, if you choose.

Data: A large set of 32x32 binary images of handwritten digits, from

`ftp://ftp.ics.uci.edu/pub/machine-learning-databases/optdigits/`

Divide into three sets: training, validation, and test.

I. Auto-associator network:

- **Input layer**: 1024 binary-valued input units, representing pixels in the image, plus one bias unit. Convert data from $\{0, 1\}$ to $\{-1, 1\}$.
- **Hidden layer**: 32 hidden units
- **Output layer**: 1024 binary valued output units.
- **Connection topology**: Fully connected, feed-forward.
- **Target activation for output units**: 0.9 if this pixel was 1 in input; 0.1 otherwise.
- **Activation function:** sigmoid.
- **Training method:** Backpropagation, with learning rate 0.3 and momentum 0.

Train the auto-associator network until the sum-squared error on validation set is less than a desired value. Run on the test data.

II. Perceptron classifier:

- **Input layer**: 4 input units plus one bias unit.
- **Output layer**: 10 output units, each corresponding to a digit in [0,9].
- **Connection topology**: Fully connected, feed-forward.
- **Target activation for output units**: 1.0 if this unit represents the correct classification; -1.0 otherwise.
- **Activation function:** Threshold: out = sgn (weights x inputs)
- **Training method**: Delta rule with learning rate 0.3, momentum 0.

For each instance in the training set, use the values of the 32 hidden units from the trained auto-associator network as input to the perceptron. Train the perceptron until the sum squared value of outputs on validation set is below a desired value. Run on the test data.

If error is still too high after a large number of epochs, experiment with: (1) different numbers of hidden units in auto-associator network, (2) different values for learning rate and momentum.

What each team should hand in.

Electronic: Tarball with your neural network code and instructions on how to run it.

Hard Copy: Accuracy on test and training sets for all experiments (i.e., on training and test data for auto-associator network, perceptron, and on variations on number of hidden units, learning rate, and momentum).