UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
Department of Electrical and Computer Engineering

ECE 544NA Pattern Recognition
Fall 2007

Exam 2

Friday, December 14, 2007

- This is a CLOSED BOOK exam, but you may use TWO PAGES, BOTH SIDES of hand-written notes
- Calculators are permitted, but will probably not be useful. The answer "ln(2)" is preferable to the answer "0.693147."
- You must SHOW YOUR WORK to get full credit.

Problem	Score
1	
2	
3	
4	
5	
6	
Total	

Name: ____

Problem 1 (25 points)

Consider the problem of training a multi-class perceptron. Tokens $\vec{x}_1, \ldots, \vec{x}_n$ are drawn from classes $z_1, \ldots, z_n$, where each class label is an integer such that $1 \le z_i \le J$. The perceptron classification function may then be defined in terms of discriminant vectors $A = [\vec{a}_1, \ldots, \vec{a}_J]$ to be

$$h(\vec{x}) = \arg\max_{1 \le j \le J} \vec{a}_j^T \vec{x} \tag{1}$$

The multi-class perceptron error metric may be defined as

$$J(A) = \sum_{i=1}^{n} \max_{1 \le j \le J} \left(\vec{a}_j^T \vec{x}_i - \vec{a}_{z_i}^T \vec{x} \right). \tag{2}$$

Consider the following sub-problems:

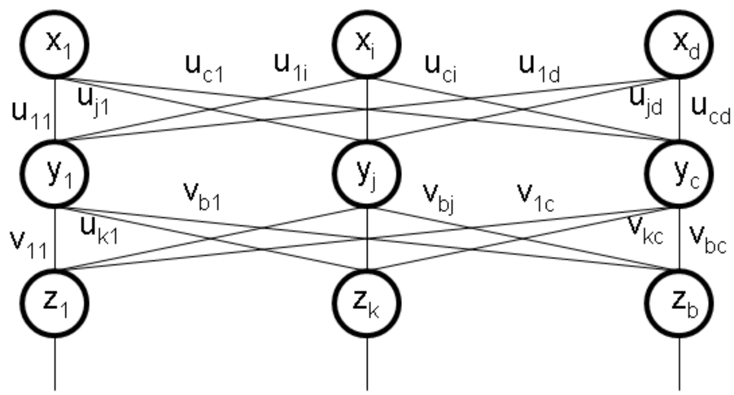
(a) Let $\mathcal{R}_j = \{\vec{x} : h(\vec{x}) = j\}$. Prove that $\mathcal{R}_j$ is a convex region with piece-wise linear boundaries.

(b) Prove that the error metric $J(A)$ is non-negative.

(c) Find the gradient of $J(A)$ with respect to $\vec{a}_4$, the discriminant vector for the fourth class.

(d) Based on your answer to part (c), devise an on-line training algorithm for the multi-class perceptron. How many of the $\vec{a}_j$ vectors are updated in response to a correctly classified training token? How many of the $\vec{a}_j$ vectors are updated in response to an incorrectly classified token?

Problem 2 (10 points)



Consider the neural network shown above. The output nodes are linear, but the hidden nodes use a cosine nonlinearity:

$$z_k = \sum_{j=1}^{c} v_{kj} y_j \tag{3}$$

$$y_j = \cos\left(\sum_{i=1}^{d} u_{ji} x_i\right) \tag{4}$$

The error metric is sum-squared error, i.e.,

$$J(U,V) = \frac{1}{2} \sum_{n=1}^{N} \sum_{k=1}^{b} |z_{kn} - t_{kn}|^2 \tag{5}$$

for targets $\vec{t}_n = [t_{1n}, \ldots, t_{bn}]^T$ corresponding to the training vectors $\vec{x}_n = [x_{1n}, \ldots, x_{dn}]^T$. Write $\partial J / \partial u_{pq}$ explicitly in terms of variables shown in the figure.

Problem 3 (15 points)

Suppose that $J(\vec{w})$, the error metric of a neural network, has a local minimum at $\vec{w} = 0$. Within the attractor basin for this local minimum, suppose that

$$J(\vec{w}) \approx \vec{w}^T H \vec{w} + J^* \tag{6}$$

Suppose that you are using a line search algorithm. Beginning with an initial weight vector $\vec{w}_1$, the following steps are iterated for $t = 1, \ldots$:

- Choose a search direction $\vec{v}_t$
- Choose α to minimize $J(\vec{w}_{t+1})$, where $\vec{w}_{t+1} = \vec{w}_t + \alpha \vec{v}_t$.

Suppose that, by wonderful good luck, you choose an initial search direction $\vec{v}_1$ that happens to be the first eigenvector of the Hessian matrix.

(a) Find $\vec{w}_2$.

(b) Assume that all future search directions are chosen to be negative gradients of J, i.e., $\vec{v}_t = -\nabla J(\vec{w}_t)$ for $t \geq 2$. Prove that $\vec{v}_1^T H \vec{w}_t \approx 0$ for all $t \geq 2$.

Problem 4 (10 points)

Suppose that $J(\vec{w})$, the error metric of a neural network, has a local minimum at $\vec{w} = 0$. Within the attractor basin for this local minimum, suppose that

$$J(\vec{w}) \approx \vec{w}^T H \vec{w} + J^* \tag{7}$$

Suppose that the weight vector can be divided into two parts, i.e., $\vec{w} = [w_1, \vec{w}_2^T]^T$, where $\vec{w}_2$ contains all of the weights except w_1, i.e., $\vec{w}_2 = [w_2, \ldots, w_{(bc+dc)}]^T$. Notice that under this circumstance, $J(\vec{w})$ can be written as

$$J(\vec{w}) \approx w_1^2 h_{11} + 2w_1 \vec{h}_{12}^T \vec{w}_2 + \vec{w}_2^T H_{22} \vec{w}_2 + J^*, \tag{8}$$

where $\vec{h}_{12}^T = [h_{12}, \ldots, h_{1K}]$, and H_{22} is the remainder of the Hessian.

Suppose that w_1 is to be estimated using deterministic simulated annealing: you are going to fix all of the coefficients in vector $\vec{w}_2$, and compute $\hat{w}_1$, the new value of w_1, according to

$$\hat{w}_1 = E\left[w_1 | \vec{w}_2\right] \tag{9}$$

using the Boltzmann probability density $p(w_1|\vec{w}_2) \propto e^{-J(\vec{w})/T}$.

Solve for $\hat{w}_1$. Your answer should be a function of the temperature T, the fixed weights $\vec{w}_2$, and the elements of the Hessian.

Problem 5 (20 points)

Consider two decision trees, T_1 and T_2. Tree T_1 has leaf nodes N_k, $1 \le k \le K$. Tree T_2 has leaf nodes N_m, $1 \le m \le M$. Your colleague George Washington has proposed a function $d(T_1, T_2)$ that he believes can be used to measure the distance between the two trees:

$$d(T_1, T_2) = \sum_{k=1}^{K} \sum_{m=1}^{M} P(N_k, N_m) \left[\arg \max_{1 \le j \le J} P(\omega_j | N_k) \neq \arg \max_{1 \le j \le J} P(\omega_j | N_m) \right] \tag{10}$$

where:

- $P(N_k, N_m)$ is the probability that a vector $\vec{x}$ drawn from the evidence distribution $p(\vec{x})$ falls into node N_k of tree T_1, and also falls into node N_m of tree T_2.
- $[p]$ is the unit indicator function for proposition p, defined by

$$[p] = \begin{cases} 1 & p \text{ true} \\ 0 & p \text{ false} \end{cases} \tag{11}$$

(a) Is $d(T_1, T_2)$ non-negative?

(b) Is $d(T_1, T_2)$ reflexive?

(c) Is $d(T_1, T_2)$ symmetric?

(d) Does $d(T_1, T_2)$ satisfy the triangle inequality? Hint: write $d(T_1, T_2)$ as a probability.

Problem 6 (20 points)

The maximum-Gaussian density is similar to the mixture-Gaussian density, except that instead of adding weighted Gaussians, we compute the maximum:

$$\hat{p}(\vec{x}_i) = \max_{1\le j\le J} c_j \phi_j(\vec{x}_i), \tag{12}$$

where $\phi_j(\vec{x}_i)$ is the Gaussian PDF with mean vector $\vec{\mu}_j$ and covariance matrix Σ_j, and c_j are chosen so that $\int \hat{p}(\vec{x})d\vec{x} = 1$.

In both parts of this problem, please assume that the weights are constrained to be uniform ($c_j = c$) and that the covariance matrices are constrained to be identity ($\Sigma_j = I$), so that the only free parameters are $\theta = \{J, \vec{\mu}_1, \ldots, \vec{\mu}_J\}$.

Please also assume that the training database contains n unlabeled vectors, $\mathcal{D} = \{\vec{x}_1, \ldots, \vec{x}_n\}$.

(a) The evidence estimate $\hat{p}(\vec{x})$ may be trained using maximum likelihood, i.e., in order to maximize

$$\mathcal{L}(\vec{\mu}_1, \ldots, \vec{\mu}_J) = \sum_{i=1}^{n} \ln \hat{p}(\vec{x}_i) \tag{13}$$

Prove that the K-means clustering algorithm finds a local maximum of $\mathcal{L}$.

(b) What is the Kolmogorov description length $\mathcal{K}(\mathcal{D}, \hat{p})$?

- Assume that the mean vectors have d elements, $\vec{\mu}_j = [\mu_{j1}, \ldots, \mu_{jd}]^T$, and that B bits are required to quantize each element.
- Assume that $\vec{x}_i$ may be quantized using $-B \log_2 \hat{p}(\vec{x}_i)$ bits.