<u>Problem Set 5a</u>

MAS 622J/1.126J: Pattern Recognition and Analysis

Due Wednesday, 5 November 2008

[Note: All instructions to plot data or write a program should be carried out using either Python accompanied by the `matplotlib` package or Matlab. Feel free to use either or both, but in order to maintain a reasonable level of consistency and simplicity we ask that you do not use other software tools. Unless otherwise specified, please copy and paste all your codes in your solution document.]

Problem 1: Parameter Learning by Estimation and Maximization

Consider data, $D = \left\{ \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \begin{pmatrix} 4 \\ 4 \end{pmatrix}, \begin{pmatrix} 2 \\ * \end{pmatrix} \right\}$, sampled from a two-dimensional (separable) distribution, $p(x_1, x_2) = p(x_1)p(x_2)$, with

$$p(x_1) = \begin{cases} \frac{1}{\theta_1} e^{-x_1/\theta_1} & \text{if } x_1 \geq 0 \\ 0 & \text{otherwise} \end{cases} \qquad p(x_2) = \begin{cases} \frac{1}{\theta_2} & \text{if } 0 \leq x_2 \leq \theta_2 \\ 0 & \text{otherwise} \end{cases}$$

and a missing feature value, *.

a. Start with an initial estimate $\underline{\theta}^0 = \begin{pmatrix} 3 \\ 5 \end{pmatrix}$ and analytically calculate $Q(\underline{\theta}; \underline{\theta}^0)$. This is the *estimate* step in the EM algorithm.

b. Find the $\underline{\theta}$ that maximizes your $Q(\underline{\theta}; \underline{\theta}^0)$ – the *maximization* step of the EM algorithm.

Problem 2: Mixture of Gaussians

Implement the EM algorithm for estimating the parameters of a mixture of Gaussians with isotropic covariances $\Sigma_j = \sigma_j I$. Note: Download the data file from the course website. There are two datasets each of which is two-dimensional.

To solve this problem you can write your own code or use any MATLAB/Python toolboxes available for the purpose. In particular, there is a MATLAB mixture of Gaussians algorithm available for download here that would be good to explore:

`http://dataclustering.cse.msu.edu/`

- Experiment with the number of mixtures and comment on the tradeoff between the number of mixtures and goodness of fit (i.e. loglikelihood) of the data. Suggestion: Plot the loglikelihood as a function of the number of components of a mixture of Gaussians to support your argument.

- Find a fixed number of Gaussians that works well for each data set.

- Plot the estimated Gaussians as one-sigma countours of each mixing component on top of the training data.

- List mean, covariance and mixing weights of each mixture component.

- Include your source code.