Assignment 2: Latent Variable Models

Unsupervised Learning

Maneesh Sahani

Due: Thurs Nov 2, 2006

Note: The Matrix Inversion Lemma

$$(A + XBX^\top)^{-1} = A^{-1} - A^{-1}X(B^{-1} + X^\top A^{-1}X)^{-1}X^\top A^{-1}$$

is a useful tool to know, and may be useful for some of these questions.

[10 points] Part I. Latent Variable Models

50% Describe a real-world data set which you believe could be modelled using factor analysis. Argue why factor analysis is a sensible model for this data. What do you expect the factors to represent? How many factors do you think there would be? Are the linearity and Gaussianity assumptions reasonable, and if not, how would you modify the model?

50% Describe a real-world data set which you believe could be modelled using a mixture model (do not use the example in Part III). Argue why a mixture model is a sensible model for your real world data set. What do you expect the mixture components to represent? How many components (or clusters) do you think there would be? What parametric form would each component have?

[10 points] Part II. Principal Components Analysis

In Probabilistic Principal Components Analysis

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{0}, I)$$

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}\left(\Lambda\mathbf{y}, \sigma^2 I\right)$$

and the principal components are assumed to be orthonormal: $\Lambda^\top\Lambda = I$. Derive the mean and covariance of $p(\mathbf{y}|\mathbf{x})$ in the PCA limit, $\sigma^2 \to 0$.

[80 points] Part III. EM for Binary Data

Consider a data set of binary (black and white) images. Each image is arranged into a vector of pixels by concatenating the columns of pixels in the image. The data set has N images $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\}$ and each image has D pixels, where D is (number of rows × number of columns) in the image. For example, image $\mathbf{x}^{(n)}$ is a vector $(x_1^{(n)}, \ldots, x_D^{(n)})$ where $x_d^{(n)} \in \{0, 1\}$ for all $n \in \{1, \ldots, N\}$ and $d \in \{1, \ldots, D\}$.

5% Explain why a multivariate Gaussian is not an appropriate model for this data set of images.

Assume that the images were modelled as independently and identically distributed samples from a D-dimensional **multivariate Bernoulli distribution** with parameter vector $\mathbf{p} = (p_1, \ldots, p_D)$, which has the form

$$P(\mathbf{x}|\mathbf{p}) = \prod_{d=1}^{D} p_d^{x_d}(1 - p_d)^{(1-x_d)}$$

where both $\mathbf{x}$ and $\mathbf{p}$ are D-dimensional vectors

5% How many bits would it take on average to code this data set?

5% What is the equation for the maximum likelihood (ML) estimate of $\mathbf{p}$ (recall assignment 1)? Note that you can solve for $\mathbf{p}$ directly.

10% Assuming independent Beta priors on the parameters p_d

$$P(p_d) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p_d^{\alpha-1}(1-p_d)^{\beta-1}$$

and $P(\mathbf{p}) = \prod_d P(p_d)$ What is the maximum a posteriori (MAP) estimate of $\mathbf{p}$? Hint: maximise the log posterior with respect to $\mathbf{p}$.

Download the data set `binarydigits.txt` from the course website, which contains $N = 100$ images with $D = 64$ pixels each, in an $N \times D$ matrix. These pixels can be displayed as 8×8 images by rearranging them. View them in Matlab by running `bindigit.m` (almost no Matlab knowledge required to do this).

10% Write code to learn the ML parameters of a multivariate Bernoulli from this data set and display these paramteres as an 8×8 image. Hand in your code and the learned parameter vector. (Matlab or Octave code is preferred, but C or Java are acceptable).

5% Modify your code to learn MAP parameters with $\alpha = \beta = 3$. What is the new learned parameter vector for this data set? Explain why this might be better or worse than the ML estimate.

Mixture Models:

10% Write down the likelihood for a model consisting of a mixture of K multivariate Bernoulli distributions. Use the parameters $\pi_1, \ldots, \pi_K$ to denote the mixing proportions ($0 \leq \pi_k \leq 1; \sum_k \pi_k = 1$) and arrange the K Bernoulli parameter vectors into a matrix $\mathbf{P}$ with elements p_{kd} denoting the probability that pixel d takes value 1 under mixture component k.

Just like in a mixture of Gaussians we can think of this model as a latent variable model, with a discrete hidden variable $s^{(n)} \in \{1, \ldots, K\}$ where $P(s^{(n)} = k|\boldsymbol{\pi}) = \pi_k$.

5% Write down the expression for the responsibility of mixture component k for data vector $\mathbf{x}^{(n)}$, i.e. $r_{nk} \equiv P(s^{(n)} = k|\mathbf{x}^{(n)}, \boldsymbol{\pi}, \mathbf{P})$

20% Implement the EM algorithm for a mixture of K multivariate Bernoullis. The algorithm should take as input K, a matrix Y containing the data set, and a number of iterations. The algorithm should run for that number of iterations or until the log likelihood converges (does not increase by more than a very small amount). Beware of numerical problems as likelihoods can get very small, it is better to deal with log likelihoods. Also be careful with numerical problems when computing responsibilities — it might be necessary to multiply the top and bottom of the equation for responsibilities by some constant to avoid problems. Hand in code and a high level explanation of what you algorithm does.

15% Run your algorithm on the data set for varying $K = 2, 3, 4$. Verify that the log likelihood increases at each step of EM. Report the log likelihoods obtained (measured in *bits*) and display the parameters found.

10% Comment on how well the algorithm works, whether it finds good clusters (look at the responsibilities and try to interpret them), and how you might improve the model.