Engineering Tripos Part IIB FOURTH YEAR

Module 4F10: STATISTICAL PATTERN RECOGNITION

Examples Paper 1

Straightforward questions are marked †
Tripos standard (but not necessarily Tripos length) questions are marked ∗

Bayes Risk

1. In many pattern classification problems, one has the option either to assign the pattern to one of the c classes, or to reject it as being unrecognizable. If the cost to reject is not too high, rejection may be a desirable action. Let the cost of classification be defined as

$$\lambda(\omega_i|\omega_j) = \begin{array}{lll} 0 & \omega_i = \omega_j & (i.e.\ \text{(Correct classification)} \\ \lambda_r & \omega_i = \omega_0 & (i.e.\ \text{Rejection}) \\ \lambda_s & \text{Otherwise} & (i.e.\ \text{Substitution Error}) \end{array}$$

   Show that for minimum risk classification, the decision rule should associate a test vector $\boldsymbol{x}$ with class ω_i, if $P(\omega_i|\boldsymbol{x}) \geq P(\omega_j|\boldsymbol{x})$ for all j **and** $P(\omega_i|\boldsymbol{x}) \geq 1 - \lambda_r/\lambda_s$, and reject otherwise.

EM and Mixture Models

2. † For d-dimensional data compare the computational cost of calculating the log-likelihood with a diagonal covariance matrix Gaussian distribution, a full covariance matrix Gaussian distribution and an M-component diagonal covariance matrix Gaussian mixture models. Clearly state any assumptions made.

3. A 1-dimensional 2-component mixture distribution has a common fixed known variance = 1 and initial mean values $\mu_1 = 0$ $\mu_2 = 2$ and mixture weights $c_1 = c_2 = 0.5$.

   There is a data set of 9 training data points provided

$$-1.5,\ -0.5,\ 0.1,\ 0.3,\ 0.9,\ 1.3,\ 1.9,\ 2.3,\ 3.0$$

   (a) Calculate the log likelihood of the training data for the mixture distribution with the initial parameters.

   (b) Calculate updated values for the mean and mixture weights for 1 iteration of the E-M algorithm.

4. A series of n independent, noisy, measurements are taken, $x_1, \ldots, x_n$. The noise is known to be Gaussian distributed with zero mean and unit variance. The "true" data is also known to be Gaussian distributed.

(a) Find the maximum likelihood estimates of the mean, μ, and variance, σ^2, of the "true" data by equating the gradient to zero.

(b) A latent variable z_i is introduced. It is the value of the noise for observation x_i. Show that the posterior probability of z_i given the current model parameters is

$$p(z_i|x_i,\theta) = \mathcal{N}\left(z_i; \frac{(x_i-\mu)}{(1+\sigma^2)}, \frac{\sigma^2}{(1+\sigma^2)}\right)$$

Using the expectation-maximisation algorithm derive re-estimation formulae for the mean, μ, and variance, σ^2. Show that the iterative estimation scheme for the mean converges to the correct answer, you may assume that the variance is of the true data is known and fixed at σ^2.

Discuss the merits of the two optimisation schemes for this task and for optimisation tasks in general.

5. Consider an M component mixture model of d-dimensional binary data $\mathbf{x}$ of the form

$$p(\mathbf{x}) = \sum_{m=1}^{M} P(\omega_m) p(\mathbf{x}|\omega_m)$$

where the j^{th} component PDF has parameters $\lambda_{j1}, \ldots, \lambda_{jd}$ and

$$p(\mathbf{x}|\omega_j) = \prod_{i=1}^{d} \lambda_{ji}^{x_i} (1-\lambda_{ji})^{1-x_i}$$

A set of training samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ are used to train the mixture model. Using the standard form of EM with mixture models show that the maximum likelihood estimate for the "new" parameters, $\hat{\lambda}_{ji}$, is given by

$$\hat{\lambda}_{ji} = \frac{\sum_{k=1}^{n} P(\omega_j|\mathbf{x}_k) x_{ki}}{\sum_{k=1}^{n} P(\omega_j|\mathbf{x}_k)}$$

where $P(\omega_j|\mathbf{x}_k)$ is obtained using the "old" model parameters.

Single Layer Perceptrons

6. The standard single layer perceptron is used to discriminate between two classes. There are two simple techniques for generalising this to a K class problem. The first is to build a set of pairwise classifiers i.e. ω_i versus ω_j, $j \neq i$. The second is to build a set of classifiers of each class versus all other classes i.e. ω_i versus $\{\omega_1, \ldots, \omega_{i-1}, \omega_{i+1}, \omega_K\}$. Compare the two forms of classifier in terms of training and testing computational cost. By drawing a specific example with $K = 3$ show that both forms of classifier can result in an "ambiguous" region i.e. no decision can be made. Describe how multiple binaries classifiers may be trained so that no ambiguous regions exist.

Multi-Layer Perceptrons

7. † A multi-layer perceptron consists of d inputs, L hidden layers with M hidden units in each hidden layer, and K output nodes. Write down an expression for the total number of weights (including biases) in the network. Describe the factors that influence the number of hidden layers, the activation functions on the output layer, and the number of hidden units.

8. † For the logistic regression function, $\phi(z)$, show that

$$\frac{\partial}{\partial z}\phi(z) = \phi(z)(1 - \phi(z))$$

How does the nature of the activation function affect the computational cost of the error-back propagation algorithm?

9. Consider the optimisation of a set of weights where the gradient of the error function with respect to the weight space is approximately constant. The following update rule is used

$$\mathbf{w}[\tau + 1] = \mathbf{w}[\tau] + \mathbf{\Delta w}[\tau]$$

where

$$\mathbf{\Delta w}[\tau] = -\eta \, \boldsymbol{\nabla} E(\mathbf{w})|_{\mathbf{w}[\tau]} + \alpha \mathbf{\Delta w}[\tau - 1]$$

Show that the effect of the momentum term is to increase the effective learning rate from η to $\frac{\eta}{1-\alpha}$.

What is the effective learning rate for a region where the gradient descent scheme is oscillating about the real solution?

10. $*$ The Hessian is a useful matrix for use in the optimisation of the weights of multi-layer perceptrons.

    (a) Describe how the Hessian may be used for optimising the weights of a multi-layer perceptron. Discuss the limitations for the practical implementation of such schemes.

    (b) For the least squares error function

$$E = \frac{1}{2}\sum_{p=1}^{n}(y(x_p) - t(x_p))^2$$

show that the elements of the Hessian matrix can be expressed as

$$\frac{\partial^2 E}{\partial w_{ij}\partial w_{lk}} = \sum_{p=1}^{n}\frac{\partial y(x_p)}{\partial w_{ij}}\frac{\partial y(x_p)}{\partial w_{lk}} + \sum_{p=1}^{n}(y(x_p) - t(x_p))\frac{\partial^2 y(x_p)}{\partial w_{ij}\partial w_{lk}}$$

For the case of well trained, sufficiently powerful, network, with an infinitely large training set, show that at the minimum the second term may be ignored. This is called the *outer-product* approximation.

(c) The Hessian after the N^{th} data point is approximated by

$$\mathbf{H}_N = \sum_{p=1}^{N} \mathbf{g}^{(p)} (\mathbf{g}^{(p)})'$$

where

$$\mathbf{g}^{(p)} = \boldsymbol{\nabla} E|_{\mathbf{w}[\tau]}$$

By using the equality

$$(\mathbf{A} + \mathbf{BC})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{I} + \mathbf{CA}^{-1}\mathbf{B})^{-1}\mathbf{CA}^{-1}$$

where $\mathbf{I}$ is the identity matrix, show that

$$\mathbf{H}_{N+1}^{-1} = \mathbf{H}_N^{-1} - \frac{\mathbf{H}_N^{-1}\mathbf{g}^{(N+1)}(\mathbf{g}^{(N+1)})'\mathbf{H}_N^{-1}}{1 + (\mathbf{g}^{(N+1)})'\mathbf{H}_N^{-1}\mathbf{g}^{(N+1)}}$$

Why is this a useful approximation to estimate the inverse Hessian during multi-layer perceptron training.

Answers

3. (a) total likelihood of data = 2.262e-07; (b) $\hat{\mu}_1 = -0.0426$; $\hat{\mu}_2 = 1.878$; $\hat{c}_1 = 0.5266$; $\hat{c}_2 = 0.4734$

M.J.F. Gales
P.C. Woodland
Oct 2003 - Jan 2007