Minimum Entropy Data Partitioning

Stephen J. Roberts, Richard Everson & Iead Rezek

Intelligent & Interactive Systems Group
Department of Electrical & Electronic Engineering
Imperial College of Science, Technology & Medicine
Exhibition Road, London SW7 2BT, UK

Abstract

Problems in data analysis often require the unsupervised partitioning of a data set into clusters. Many methods exist for such partitioning but most have the weakness of being model-based (most assuming hyper-ellipsoidal clusters) or computationally infeasible in anything more than a 3-dimensional data space. We re-consider the notion of cluster analysis in information-theoretic terms and show that minimisation of partition entropy can be used to estimate the number and structure of probable data generators. The resultant analyser may be regarded as a Radial-Basis Function classifier.

1 Introduction

Many problems in data analysis, especially in signal and image processing, require the *unsupervised* partitioning of data into a set of 'self-similar' clusters or regions. An ideal partition unambiguously assigns each datum to a single cluster and one thinks of the data as being generated by a number of data generators, one for each cluster. Many algorithms have been proposed for such analysis and for the estimation of the optimal number of partitions. The majority of popular and computationally feasible techniques rely on assuming that clusters are hyper-ellipsoidal in shape. In the case of Gaussian mixture modelling [8, 3, 4] this is explicit; in the case of dendogram linkage methods (which typically rely on the L_2 norm) it is implicit [5]. For some data sets this leads to an over-partitioning. Alternative methods, based on valley seeking [3] or maxima-tracking in scale-space [7] for example, have the advantage that they are free from such assumptions. They can be, however, computationally intensive, sensitive to noise (in the case of valley seeking approaches) and unfeasible in high-dimensional spaces (indeed these methods can become prohibitive in even a 3-dimensional data space).

In this paper we re-consider the issue of data partitioning from an information-theoretic viewpoint and show that minimisation of entropy, or maximisation of partition certainty, may be used to evaluate the most probable set of data generators. The approach does not assume cluster convexity and is shown to partition a range of data structures and to be computationally efficient. We may regard the final system as a Radial-Basis Function (RBF) classifier in which the number of output nodes is determined from the data in an unsupervised manner.

2 Theory

The idea underlying this approach is that the observed dataset is generated by a number of data generators (classes). We first model the unconditional probability density function (pdf) of the data and then seek a number of partitions whose combination yields the data pdf. Densities and classifications conditioned on this partition set are then easily obtained.

2.1 Partition entropy

Consider a set of $k = 1..K$ partitions. The probability density function of a single datum $\mathbf{x}$, conditioned on this partition set, is given by:

$$p(\mathbf{x}) = \sum_{k=1}^{K} p(\mathbf{x} \mid k)p(k) \qquad (1)$$

We consider the overlap between the contribution to this density function of the k-th partition and the density $p(\mathbf{x})$. This overlap may be measured by the Kullback-Liebler measure between these two distributions. The latter is defined, for distributions $p(x)$ and $q(x)$ as:

$$KL\left(p(x) \parallel q(x)\right) = \int p(x) \ln\left(\frac{p(x)}{q(x)}\right) dx \qquad (2)$$

Note that this measure reaches a minimum of zero if, and only if, $p(x) = q(x)$. For any other case it is strictly positive and increases as the overlap between the two distributions *decreases*. What we desire, therefore, is that the KL measure be maximised as this implies that the overlap between two distributions is minimised. We hence write our overlap measure as:

$$v_k = -KL\left(p(\mathbf{x} \mid k)p(k) \parallel p(\mathbf{x})\right) \qquad (3)$$

As the this measure is strictly non-positive we may define a total overlap as the summation of all v_k:

$$V = \sum_k v_k \qquad (4)$$

An 'ideal' data partitioning separates the data such that overlap between partitions is minimal. We therefore seek the partitioning for which V is a minimum. Using Bayes' theorem we may combine Equations 2,3 & 4 such that:

$$V = \int H(\mathbf{x})p(\mathbf{x})d\mathbf{x} \qquad (5)$$

in which $H(\mathbf{x})$ is the Shannon entropy, given datum $\mathbf{x}$, over the set of partition posteriors, i.e. $H(\mathbf{x}) = -\sum_k p(k \mid \mathbf{x}) \ln p(k \mid \mathbf{x})$. Minimising V is hence equivalent to minimising the expected entropy of the partitions over all observed data. It is this objective which we will use to form minimum-entropy partitions. It is noted that this is achieved by having, for each datum, some partition posterior close to unity which conforms to our objective for ideal partitioning.

2.2 Partitions as mixture models

We restrict ourselves in this paper to considering a set of kernels or basis functions which model the probability density function (pdf) of the data and thence of each data partition.

Consider a set of partitions of the data. We may model the density function of the data, conditioned on the i-th partition, via a semi-parametric mixture model of the form:

$$p(\mathbf{x} \mid i) = \sum_{j=1}^{J} p^*(\mathbf{x} \mid j)\pi^*(j) \qquad (6)$$

where J is the number of kernels forming the mixture and π_j^* are a set of (unknown) mixture coefficients which sum to unity. Each mixture component may be, for example, a Gaussian kernel and hence each candidate partition of the data is represented in this approach as a mixture of these kernels. The use of the 'star' notation, i.e. $p^*(j)$, is to denote that this set of probabilities is evaluated over the kernel representation, rather than over the set of data partitions. Equation 6 may be written, via Bayes' theorem, as a linear transformation to a set of partition *posteriors* of the form:

$$\mathbf{p} = \mathbf{W}\mathbf{p}^* \tag{7}$$

where $\mathbf{p}$ is the set of partition posterior probabilities (in vector form), $\mathbf{W}$ is some transform, or mixing, matrix (not assumed to be square) and $\mathbf{p}^*$ is the set of kernel posteriors (in vector form). Hence the i-th partition posterior may be written as:

$$p_i \equiv p(i \mid \mathbf{x}) = \sum_j W_{ij} p^*(j \mid \mathbf{x}) \tag{8}$$

If we are to interpret $\mathbf{p}$ as a set of posterior probabilities we require:

$$p_i \in [0,1]\ \forall i \text{ and } \sum_i p_i = 1 \tag{9}$$

As $p^*(j \mid \mathbf{x}) \in [0,1]$ so the first of these conditions is seen to be met if each $W_{ij} \in [0,1]$. The second condition is met when each column of $\mathbf{W}$ sums to unity.

2.3 Entropy minimisation

Given that we represent each partition by a fixed set of kernels, we wish to adjust the elements of the matrix $\mathbf{W}$ such that the entropy over the partition posteriors is minimised. We must also, however, take into account the constraints on the elements of $\mathbf{W}$ (that they are bounded in $[0,1]$ and the sum down each column of $\mathbf{W}$ is unity). We may achieve this by introducing a set of dummy variables, which will be optimised, such that $\mathbf{W}$ is represented by a generalised logistic function (the so-called 'softmax' function) of the form:

$$W_{ij} = \frac{\exp(\theta_{ij})}{\sum_{i'} \exp(\theta_{i'j})} \tag{10}$$

The gradient of the entropy with respect to each dummy variable, θ_{ij}, is given by the chain rule as

$$\frac{\partial H}{\partial \theta_{ij}} = \sum_{i'} \frac{\partial H}{\partial W_{i'j}} \cdot \frac{\partial W_{i'j}}{\partial \theta_{ij}} \tag{11}$$

The summation term is easily evaluated noting that

$$\frac{\partial W_{i'j}}{\partial \theta_{ij}} = W_{i'j}\delta_{i'i} - W_{i'j}W_{ij} \tag{12}$$

where $\delta_{i'i} = 1$ if $i = i'$ and zero otherwise. The term $\partial H / \partial W_{i'j}$ is evaluated by writing the expectation of the entropy (of Equation 5) as a sample mean over all the data, i.e.

$$\frac{\partial H}{\partial W_{i'j}} = \left\langle \frac{\partial H(\mathbf{x}_n)}{\partial p(i' \mid \mathbf{x}_n)} \cdot \frac{\partial p(i' \mid \mathbf{x}_n)}{\partial W_{i'j}} \right\rangle \tag{13}$$

As $p(i' \mid \mathbf{x}_n) = \sum_j W_{i'j} p^*(j \mid \mathbf{x}_n)$ the above is easily evaluated. In all the experiments reported in this paper we optimise $\mathbf{W}$ using the above formalism via the BFGS quasi-Newton method [6].

2.3.1 A RBF interpretation

Note that the form of the analysis is identical to that of a Radial-Basis Function (RBF) classifier in which the hidden-layer representation consists of a set of Gaussians which are optimised to model the data density, followed by a layer in which a set of weights are optimised so as to minimise an entropy function. In the case of a *supervised* RBF system, this is the *cross*-entropy function between the class-posteriors and the targets whereas in our case it is the posterior entropy.

2.4 Model-order estimation

We evaluate the entropy change, per partition, as a result of observing the data set, X. This quantity is given as,

$$\Delta H(M_K \mid X) = H(M_K) - H(M_K \mid X) \tag{14}$$

where M_K is the K-partitions model. The first term on the right-hand side of the above Equation is simply the entropy of the model priors *before* data are observed and is the Shannon entropy taking the partition probabilities to be uniform and equal to $1/K$. The second term is the entropy associated with the posterior partition probabilities having observed X. Noting that $H(X) - H(X \mid M_K) = H(M_K) - H(M_K \mid X)$ and that $H(X \mid M_K)$ is the expectation of the negative log-likelihood of X given M_K so the likelihood (evidence) of X given M_K may be written as:

$$p(X \mid M_K) \propto \exp\{\Delta H(M_K \mid X)\} \tag{15}$$

in which the data entropy term, $H(X)$, is ignored as it is constant for all models. Choosing the model with the largest value of this likelihood is equivalent, via Bayes' theorem, to choosing the model with the highest probability, $p(M_K \mid X)$ if we assume flat prior beliefs, $p(M_K)$, for each model. By normalisation of Equation 15 we hence obtain a posterior belief, $p(M_K \mid X)$, for each candidate partitioning and it is this measure which we use to assess the model order, choosing the order K for which it is maximal.

3 Results

3.1 Simple data set

We first present results from a data set in which clusters are simple and distinct; the data are generated from four Gaussian distributed sources with 30 data drawn from each. Each component has the same (spherical) covariance. As an illustration of a simple kernel set, ten Gaussian components are fitted to the data using the EM algorithm. Figure 1 shows $P(M_K \mid X)$ on (a) log and (b) linear scales. Note that a set of four sources is clearly favoured. Choosing the $K = 4$ model we obtain, for this example, $\mathbf{W}$ as a 4×10 matrix. The resultant partitioning of the data set gives the results of Figure 2. There are no errors in the partitioning for this simple data set.

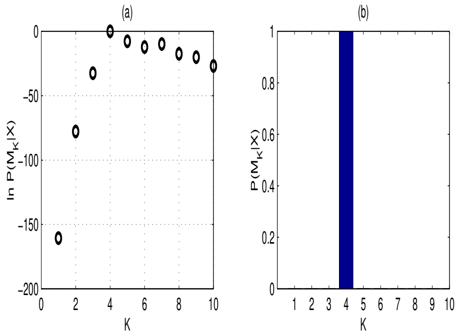

Figure 1: **Simple data set:** (a) $\ln p(M_K \mid X)$ and (b) $p(M_K \mid X)$. Note the clear maxima at $K = 4$.

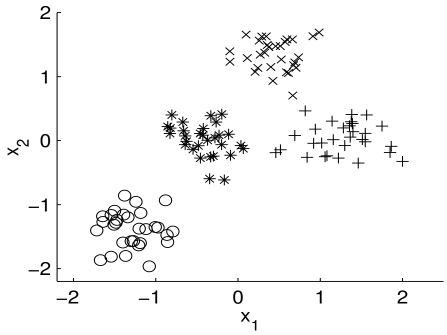

Figure 2: **Simple data set:** Data partitioning in the transformed space. For this simple example there are no errors.

3.2 Ring data

The next (synthetic) data set we investigate is drawn from two generator distributions; an isotropic Gaussian and a uniform 'ring' distribution. A total of 100 data points were drawn from each distribution (hence $N = 200$). A 20-kernel Gaussian mixture model was fitted to the data (using again the EM algorithm). Figure 3(a) shows that $p(M_K \mid X)$ gives greatest support for the two partition model. Plot (b) of the same figure depicts the resultant data partioning. For this example there are no errors. Figure 4 shows the maximum partition posterior and clearly depicts the decision boundary for this example. Note that, due to the pathological structure of this example, a Gaussian mixture model *per se* fails to estimate the 'correct' number of partitions and provide a reasonable data clustering.

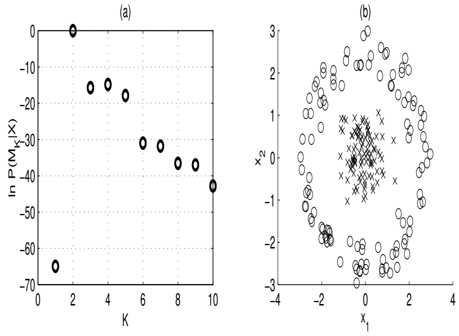

Figure 3: **Ring data set:** (a) $\ln P(M_K \mid X)$ & (b) resultant partitioning. For this example there are no errors.

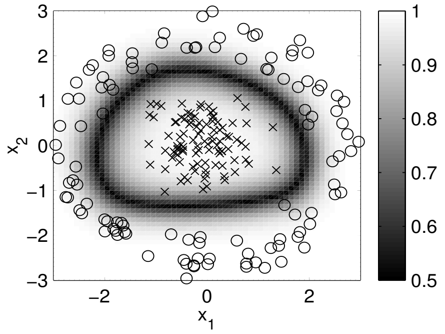

Figure 4: **Ring data set:** Maximum partition posterior.

3.3 Iris data

Anderson's 'iris' data set is well-known [2]. The data we analysed consisted of 50 samples for each of the three classes present in the data, *Iris Versicolor, Iris Virginica* and *Iris Setosa*. Each datum is four-dimensional and consists of measures of the plants morphology. Once more a 20-kernel model was fitted to the data set. Figure 5(a) shows the model-order measure, shown in this case on a linear y-scale. Although support is greatest for the $K = 3$ partitioning it is clear that a two-partition model has support. We regard this as sensible given the nature of the data set, i.e. it naturally splits into two partitions. As in previous figures plot(b) depicts the data partitioning. This plot shows the projection onto the first two principal components of the data set. The partitioning has three errors in 150 samples giving an accuracy of 98%. This is slightly better than that quoted in [4] and the same as that presented for Bayesian Gaussian mixture models in [8].

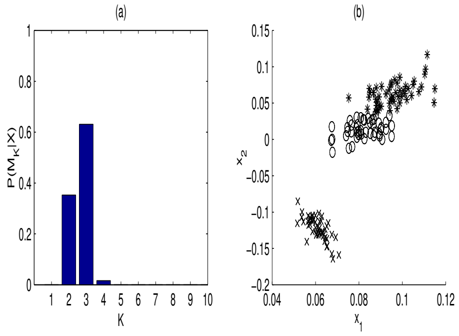

Figure 5: **Iris data set:** (a) $P(M_K \mid X)$ & (b) resultant partitioning. For this example there are three errors, corresponding to an accuracy of 98%.

3.4 Wine recognition data

As a final example we present results from a wine recognition problem. The data set consists of 178 13-dimensional exemplars which are a set of chemical analyses of three types of wine. Once more we fit a 20-kernel model and perform a minimum-entropy clustering. Figure 6(a) shows $\ln P(M_K \mid X)$. There is a clear maximum at the 'correct' partitioning ($K = 3$). Plot (b) shows this partitioning projected onto the first two components of the data set. For this example there are 4 errors, corresponding to an equivalent classification performance of 97.75%. This data set has not (to the authors' knowledge) been analysed using an *unsupervised* classifier, but *supervised* analysis has been reported. Our result is surprisingly good considering that *supervised* first-nearest neighbour classification achieves only 96.1%, and multivariate linear-discriminant analysis 98.9% [1]. It should be commented that the same partitioning is obtained via analysis of the first two data principal components *alone*, rather than the full 13-D data set.

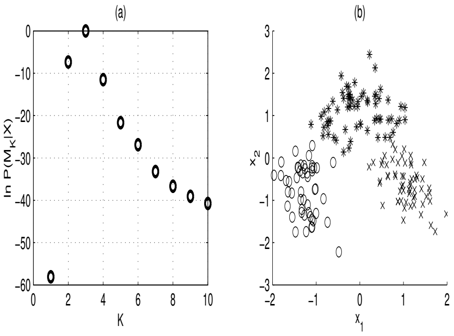

Figure 6: **Wine recognition data set:** (a) $\ln P(M_K \mid X)$ & (b) resultant partitioning. For this example there are four errors, corresponding to an accuracy of 97.75%.

4 Conclusions

We have presented a computationally simple technique for data partitioning based on a linear mixing of a set of fixed kernels. The technique is shown to give excellent results on a range of problems. For computational parsimony we have used an initial semi-parametric approach to kernel fitting although the results from a non-parametric analysis are near identical in all cases. The methodology is general and non-Gaussian kernels may be employed in which case the estimated partition-conditional densities will be mixture models of the chosen kernel functions. The method, furthermore, scales favourably with the dimensionality of the data space and the entropy-minimisation algorithm is efficient even with large numbers of samples.

5 Acknowledgements

IR and RE are funded respectively via grants from the commission of the European Community (project SIESTA, grant BMH4-CT97-2040) and British Aerospace plc. whose support we gratefully acknowledge. The *iris* and *wine* data sets are available from the UCI machine-learning repository.

References

[1] S. Aeberhard, D. Coomans, and O. de Vel. Comparative-Analysis of Statistical Pattern-Recognition Methods in High-Dimensional Settings. *Pattern Recognition*, 27(8):1065–1077, 1994.

[2] E. Anderson. The Irises of the Gaspe peninsula. *Bull. Amer. Iris Soc.*, 59:2–5, 1935.

[3] K. Fukunaga. *An Introduction to Statistical Pattern Recognition.* Academic Press, 1990.

[4] I. Gath and B. Geva. Unsupervised Optimal Fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):773–781, 1989.

[5] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, 1988.

[6] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C.* Cambridge University Press, 1991.

[7] S.J. Roberts. Parametric and non-parametric unsupervised cluster analysis. *Pattern Recognition*, 30(2):261–272, 1997.

[8] S.J. Roberts, D. Husmeier, I. Rezek, and W. Penny. Bayesian approaches to mixture modelling. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 20(11):1133–1142, 1998.