MIXTURE MODELS AND NEURAL NETWORKS FOR CLUSTERING

Geoff McLachlan and David Peel,

Department of Mathematics,
University of Queensland, Brisbane, Queensland 4072, AUSTRALIA

KEYWORDS: Finite mixture models; Maximum likelihood; EM algorithm; Likelihood ratio test

ABSTRACT

We focus here on a mixture model-based approach to the clustering of multivariate data. For independent observations, a normal mixture model can be fitted by maximum likelihood via the EM algorithm. It leads to a probabilistic clustering of the data in terms of the fitted posterior probabilities of component membership of the mixture model. The performance of this approach is compared to methods more closely related to neural networks. This comparison is undertaken over a set of benchmark data sets in the classification literature.

1. INTRODUCTION

Classification is an area in which neural networks have been extensively applied in recent years. The term 'classification' is used broadly in the literature on discriminant and cluster analyses. Discriminant analysis as a whole is concerned with the relationship between a categorical variable and a set of interrelated variables. More precisely, suppose there is a finite number, say g, of distinct populations, categories, classes, or groups, which we shall denote here by $G_1, \ldots, G_g$. We shall henceforth refer to the G_i as groups. Note that in discriminant analysis the existence of the groups is known *a priori*. An entity of interest is assumed to belong to one (and only one) of the groups. We let the categorical variable z denote the group membership of the entity, where $z = i$ implies that it belongs to group G_i $(i = 1, \ldots, g)$. Also, we let the p-dimensional vector $\boldsymbol{x} = (x_1, \ldots, x_p)^T$ contain the measurements on p available features of the entity. In the case of $g > 2$ groups, it is convenient to replace the categorical variable z by a vector of g zero-one indicator variables, $\boldsymbol{z} = (z_1, \ldots, z_g)^T$, where $z_i = 1$ or 0, according as the entity does or does not belong to group G_i $(i = 1, \ldots, g)$.

A basic assumption in discriminant analysis is that in order to estimate the unknown group-conditional densities of the feature vector $\boldsymbol{X}$, there are entities of known origin on which the feature vector $\boldsymbol{X}$ has been recorded on each. We let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ denote these recorded feature vectors and $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$ the corresponding vectors of zero-one indicator variables defining the known group of origin of each. In statistical pattern recognition, the formation of an allocation rule from training data of known origin is referred to as supervised learning. To avoid any possible confusion here, we shall reserve the use of classification to describe the original definition of the underlying groups. Hence by a classified entity we mean an entity whose group of origin is known.

We are concerned here with the unsupervised case where the data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are unclassified; that is, the $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$ are unknown. The problem is to cluster the n entities into homogenous groups on the basis of their associated feature vectors, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. In the typical cluster analysis problem of this nature, the number of groups g also has to be inferred from the data.

In this paper, we shall compare the mixture model-based approach to clustering with some cluster analysis techniques related to neural networks.

2. NORMAL MIXTURE MODELS

With a mixture model-based approach to clustering [1], it is assumed that the data to be clustered are from a mixture of an initially specified number g of groups in some unknown proportions $\pi_1, \ldots, \pi_g$. That is, each data point is taken to be a realization of the mixture probability density function (p.d.f.),

$$f(\boldsymbol{x};\, \boldsymbol{\Psi}) = \sum_{i=1}^{g} \pi_i c_i(\boldsymbol{x};\, \boldsymbol{\theta}_i), \tag{1}$$

where the g components correspond to the g groups. Here the vector $\boldsymbol{\Psi}$ of unknown parameters consists of the mixing proportions π_i and the elements of the $\boldsymbol{\theta}_i$ known *a priori* to be distinct. On specifying a parametric form for each component p.d.f. $c_i(\boldsymbol{x};\, \boldsymbol{\theta}_i)$, $\boldsymbol{\Psi}$ can be estimated by maximum likelihood (or some other method).

The posterior probability that the entity belongs to the ith component with $\boldsymbol{x}_j$ having been observed on it, is given by

$$\tau_i(\boldsymbol{x}_j;\, \boldsymbol{\Psi}) = \pi_i c_i(\boldsymbol{x}_j;\, \boldsymbol{\theta}_i)/f(\boldsymbol{x}_j;\, \boldsymbol{\Psi}) \qquad (2)$$

for $i = 1, \ldots, g$; $j = 1, \ldots, n$). If we let $r_B(\boldsymbol{x};\, \boldsymbol{\Psi})$ denote the Bayes or optimal rule of allocation, then

$$r_B(\boldsymbol{x};\, \boldsymbol{\Psi}) = \arg\max_i \tau_i(\boldsymbol{x};\, \boldsymbol{\Psi}), \qquad (3)$$

where $r_B(\boldsymbol{x};\, \boldsymbol{\Psi}) = i$ implies that $\boldsymbol{x}$ is assigned to the ith group G_i ($i = 1, \ldots, g$). The Bayes rule can be estimated by the so-called plug-in rule, $r_B(\boldsymbol{x};\, \hat{\boldsymbol{\Psi}})$, where $\hat{\boldsymbol{\Psi}}$ denotes the estimate of the unkown parameter vector $\boldsymbol{\Psi}$. This approach where the component p.d.f.'s (that is, the group-conditional p.d.f.'s) are directly modelled for use in the formation of the posterior probabilities of group membership is called the sampling approach. Another approach to the estimation of the Bayes rule is to model these posterior probabilities directly, as with the logistic model; see McLachlan [2, Chapter 8]. With this approach, called the diagnostic paradigm, the interest is not on what the component p.d.f.'s of the feature vector $\boldsymbol{X}$ look like, but on the distribution of the component-label vector $\boldsymbol{Z}$ for the observed feature data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ and similar values. This is the main approach with neural networks [3]. However, the diagnostic paradigm is limited to the case where there are classified data available.

A common assumption in practice is to take the component densities to be multivariate normal, and to proceed on the basis that any nonnormal features in the data are due to some underlying group structure. Further, often clusters in the data are essentially elliptical, so that it is reasonable to consider fitting mixtures of elliptically symmetric component densities. Within this class of component densities, the multivariate normal density is a convenient choice given its computational tractability. In those cases where the underlying population consists of groups in which the feature vector is unable to be modelled by a single normal distribution but needs a normal mixture formulation, the components in the fitted g-component normal mixture model (**??**) and the consequent clusters will correspond to g subgroups rather than to the smaller number of actual groups represented in the data.

In the case of multivariate normal components, the ith component p.d.f. $c_i(\boldsymbol{x};\, \boldsymbol{\theta}_i)$ has mean $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$, and so $\boldsymbol{\theta}_i$ denotes the parameter vector consisting of the elements of $\boldsymbol{\mu}_i$ and the $\frac{1}{2}p(p+1)$ distinct elements of $\boldsymbol{\Sigma}_i$ ($i = 1, \ldots, g$). The vector $\boldsymbol{\Psi}$ containing all the unknown parameters is given now by

$$\boldsymbol{\Psi} = (\pi_1, \ldots, \pi_{g-1}, \boldsymbol{\theta}^T)^T,$$

where $\boldsymbol{\theta}$ is the vector containing the elements of $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_g$ known *a priori* to be distinct. For example, in fitting normal mixture models, the covariance matrices in the component densities may be taken to be equal. Less restrictive constraints can be imposed by a reparameterization of the component-covariance matrices in terms of their eigenvalue decompositions.

Under the assumption that $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are independent realizations of the feature vector $\boldsymbol{X}$, the log likelihood function for $\boldsymbol{\Psi}$ is given by

$$\log L(\boldsymbol{\Psi}) = \sum_{j=1}^{n} \log \sum_{i=1}^{g} \pi_i c_i(\boldsymbol{x}_j\,;\, \boldsymbol{\theta}_i). \qquad (4)$$

With the maximum likelihood approach to the estimation of $\boldsymbol{\Psi}$, an estimate is provided by an appropriate root of the likelihood equation,

$$\partial \log L(\boldsymbol{\Psi})/\partial \boldsymbol{\Psi} = \boldsymbol{0}. \qquad (5)$$

It is straightforward to find solutions of (**??**) using the EM algorithm [4, 5]. With mixture models in general, the likelihood has multiple local maxima. Typically, the desired root corresponds to the global maximizer of the likelihood function in those situations where the likelihood function is bounded over the parameter space. But with mixtures of normal components with unequal variances in the univariate case or unequal covariance matrices in the multivariate case, the likelihood function is unbounded. In this case, the choice of root of the likelihood equation is not as obvious as in the bounded case and so requires careful consideration in practice. However, it is not suggested that the clustering of a data set should be based solely on a single solution of the likelihood equation, but rather on the various solutions considered collectively as above.

3. CLUSTERING VIA MIXTURE MODELS

After we fit the g-component mixture model to obtain the estimate $\hat{\boldsymbol{\Psi}}$ of the vector of unknown parameters in the mixture model, we can give a probabilistic clustering of the n feature observations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ in terms of their fitted posterior probabilities of component membership. For each $\boldsymbol{x}_j$, the g probabilities $\tau_1(\boldsymbol{x}_j;\, \hat{\boldsymbol{\Psi}}), \ldots, \tau_g(\boldsymbol{x}_j;\, \hat{\boldsymbol{\Psi}})$

give the estimated posterior probabilities that this observation belongs to the first, second, ..., and gth component, respectively, of the mixture ($j = 1, \ldots, n$). We can give an outright or hard clustering of these data by assigining each $\boldsymbol{x}_j$ to the component of the mixture to which it has the highest posterior probability of belonging.

4. k-MEANS ALGORITHM

As several cluster analysis techniques using neural network models are similar to the k-means algorithm, we shall first describe the latter clustering algorithm before proceeding to discuss approaches using neural networks. The k-means algorithm is implemented in various forms. Darken and Moody [6] distinguish between two basic versions of k-means. One, due to Fisher [7], is a "batch" algorithm, whereby the cluster means updated exactly following presentation of the one, some, or all of the feature vectors. This exact updating can be effected by assigning an observation to the cluster to whose mean it is closest in Euclidean distance. If the updating of the cluster means is done only after a complete pass through the observations, then it corresponds to using the so-called classification likelihood approach to fit a mixture in equal proportions of normal components with a common covariance matrix of spherical form, $\sigma^2 \boldsymbol{I}_p$, where $\boldsymbol{I}_p$ denotes the $p \times p$ identity matrix. A clustering under the classification mixture approach can be obtained by using zero-one weights in place of the fractional weights given by the current posterior probabilities of component (group) membership on the E-step of the EM algorithm. On the M-step, the constraints $\pi_1 = 1/g$ and $\boldsymbol{\Sigma}_i = \sigma^2 \boldsymbol{I}_p$ ($i = 1, \ldots, g$) must be imposed.

The second basic version of the k-means algorithm distinguished by in [6] is an adaptive approach due to McQueen [8]. If on the $(m+1)$th iteration of the process, the jth feature vector $\boldsymbol{x}_j$ is assigned to the ith group, then the mean of the latter is updated as

$$\boldsymbol{\mu}_i^{(m+1)} = \boldsymbol{\mu}_i^{(m)} + \eta_{ij}^{(m)}(\boldsymbol{x}_j - \boldsymbol{\mu}_i^{(m)}),$$

where the learning rate $\eta_{ij}^{(k)}$ is a small fraction. In the "batch" version of k-means, $\eta_{ij}^{(m)}$ would be set equal to

$$\eta_{ij}^{(m)} = 1/(n_i^{(m)} + 1),$$

where $n_i^{(m)}$ denotes the number of the feature data $\boldsymbol{x}_j$ assigned to the ith group at the end of the mth iteration. There are other variants of the k-means algorithm [9, 10, 11, 12].

5. NEURAL NETWORKS FOR CLUSTERING

With most neural network models in a discriminant analysis context, consideration is given directly to the fitting of the posterior probabilities of group membership through use of the classified data; for example, as with the multilayer perception. With the model based on radial basis functions, the classified data are used only on the final layer, as the hidden layer of outputs is obtained by using only the feature data without their known classification labels. In the absence of classified data as in a typical cluster analysis, a common theme of several neural network models is of the "self-organization" type as with the self-organizing maps (SOM) of Kohonen [13, 14, 15], the adaptive resonance theory (ART) models of Carpenter [16, 17, 18], and simple competitive learning or vector quantization (VQ) and its generalizations as considered in [19, 20]. As discussed, for example, in [21, 22, 23, 24], the updating procedures used in these methods are similar to the k-means algorithm. In particular, the SOM method can be viewed as an adaptive k-means algorithm of the sort described above, with an additional topological relationship which representationally constrains the cluster means. Murtagh and Hernández-Pajaras [22] found that the k-means algorithm with random initial guesses and a varying number of clusters provided fairly consistent results compared to those obtained with the SOM method.

6. APPLICATIONS TO REAL DATA SETS

The performance of the normal mixture likelihood approach to clustering is compared with available results in [22, 24, 28] of various neural network-related clustering methods on three real world data sets. These data sets are available from the machine learning repository of the University of California, Irvine [25]. The MIXFIT algorithm [26] was used to cluster each data set by fitting a mixture of normal components with unrestricted covariance matrices. For a given fit of the normal mixture model, the implied clusters corresponding to its components were identified with the true grouping. In those cases where the number of components was greater than the number of groups, a cluster was associated with the group having the majority of feature vectors represented in it. The accuracy of the mixture likelihood-based approach was taken to be the overall proportion of correctly allocated feature observations. These results are reported in Table 1. In this table, we have listed also the results for some other clustering methods as reported in Joshi et al. [28]. The CLUSTER, FASTCLUS, and VARCLUS algorithms are routines from the SAS package, while Simpson's [29, 30] and the multiresolution algorithm [28] are two hybrid neuro-fuzzy algorithms. The AutoClass algorithm, which uses the AutoClass C++ procedure [31], is based on a fully Bayesian version of the normal mixture model with the simplifying assumption that the component covariance matrices are diagonal. It can be seen from Table 1 that the clustering produced by the normal mixture likelihood-based approach as applied above is superior to the other clustering methods for each of the three data sets listed.

In other work on the clustering of one of these sets (the Iris data), Murtagh and Hernández-Pajaras [22] found that the SOM method misallocated 8 of the 150 observations; that is, it had an overall accuracy rate of 94.67%. Also, Mao and Jain [24] clustered this data set using the k-means algorithm with $k = 3$ and reported an overall error of 16 misallocations (89.33% accuracy rate). This inferior result compared to that in Table 1 for k-means in [28] can be explained by k being taken to be greater than 3 in the latter and then identifying more than one cluster with each true group. The normal mixture model-based clustering, which gives 5 misallocations, coincides with the k-means procedure in [24] using Mahalanobis instead of Euclidean distance and implemented via a feedforward two-layer neural network.

REFERENCES

[1] McLachlan, G.J. and Basford, K.E. (1988). *Mixture Models: Inference and Applications to Clustering.* New York: Marcel Dekker.

[2] McLachlan, G.J. (1992). *Discriminant Analysis and Statistical Pattern Recognition.* New York: Wiley.

[3] Ripley, B.D. (1996). Pattern Recognition and Neural Networks. Cambridge: Cambridge University Press.

[4] Dempster, A.P., Laird, N.M., and and Rubin, D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society B* **39**, 1–38.

[5] McLachlan, G.J. and Krishnan, T. (1997). *The EM Algorithm and Extensions.* New York: Wiley.

[6] Darken, C. and Moody, J. (1990). Fast adaptive k-means clustering: some empirical results. *Proceedings International Conference on Neural Networks*, Vol. II. New York: IEEE Computer Science Press, pp. 233–238.

[7] Lloyd, S.P. (1957). Least squares quantization in PCM. Bell Labs. Technical Report. Reprinted in *IEEE Transactions on Information Theory* **28** (1982), 129–137.

[8] MacQueen J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability* Vol. 1. Berkeley, California: University of California Press, pp. 281–297.

[9] Forgy, E.W. (1965). Cluster analysis of multivariate data: efficiency vs interpretability of classifications. *Biometrics* **21**, 768–769.

[10] Hall, D.J. and Ball, G.B. (1965). ISODATA: a novel method of data analysis and pattern classification. *Technical Report.* Menlo Park, California: Stanford Research Institute.

[11] Bezdek, J.C. (1974). Cluster validity with fuzzy sets. *Journal of Cybernetics* 3, 58–72.

[12] Hartigan, J. A. (1975). *Clustering Algorithms.* New York: Wiley.

[13] Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics* **43**, 59–69.

[14] Kohonen, T. (1989). Self-Organization and Associative Memory. (3rd Edition). Berlin: Springer-Verlag.

[15] Kohonen, T. (1990). Statistical pattern recognition revisited. In *Advanced Neural Computers*, Eckmiller, R. (Ed.). Amsterdam: North-Holland, pp. 137–144.

[16] Carpenter, G. and Grossberg, S. (1986). Adaptive resonance theory; stable self-organization of neural recognition codes in response to arbitrary lists of input patterns. In *Proceedings of the 8th Annual Conference of the Cognitive Science Society*, pp. 45–62,

[17] Carpenter, G. and Grossberg, S. (1987). ART2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics* **26**, 4919–4930.

[18] Carpenter, G. and Grossberg, S. (1990). ART3: Hierarchical search using chemical transmitters is self-organizing pattern recognition architectures. *Neural Networks* **3**, 129–152.

[19] Pal, N.R., Bezdek, J.C., and Tsao, E.C.-K. (1993). Generalized clustering networks and Ko-

TABLE 1
The Performance (% Correctly Allocated) of Seven Clustering Algorithms

Algorithm	Iris	Ionosphere	Wine
MIXFIT	96.67	93.16	94.94
AutoClass	88	92.87	84.84
CLUSTER	89.33	92.3	94.09
FASTCLUS	89.33	92.87	84.09
VARCLUS	89.33	92.023	84.09
SIMPSON	90	91.737	82.57
MULTIRESOLUTION	91.33	92.87	84.09

honen's self-organizing scheme. *IEEE Transactions on Neural Networks* **4**, 549–557. Correction (1995). *IEEE Transactions on Neural Networks* **6**, 521.

[20] Gonzalez, A.I., Graña, and D'Anjou, A. (1995). An analysis of the GLVQ algorithm. *IEEE Transactions on Neural Networks* **6**, 1012–1016.

[21] Balakrishnan, P.V., Cooper, M.C., Jacob, V.S., and Lewis, P.A. (1994). A study of the classification capabilities of neural networks using unsupervised learning: a comparison with k-means clustering. *Psychometrika* **59**, 509–525.

[22] Murtagh, F. and Hernández-Pajaras (1995). The Kohonen self-organizing map method: an assessment. *Journal of Classification* **12**, 165–190.

[23] Murtagh, F. (1996). Neural networks for clustering. In *Clustering and Classification*, P. Arabie, L.J. Hubert, and G. De Soete (Eds.). River Edge, New Jersey: World Scientific Publ., 235–269.

[24] Mao, J. and Jain, A.K. (1996). Self-organizing network for hyperellipsoidal clustering (HEC). *IEEE Transactions on Neural Networks* **7**, 16–29.

[25] Murphy, P. M. and Aha, D. W. (1994). *UCI Repository of Machine Learning Databases.* Irvine, CA: University of California, Department of Information and Computer Science.

[26] McLachlan, G.J., Peel, D., Basford, K.E., and Adams, P. (1997). MIXFIT: an algorithm for the automatic fitting and testing of normal mixture models. Unpublished manuscript.

[27] McLachlan, G.J. (1987). On bootstrapping the likelihood ratio test statistic for the number of components in a normal mixture. *Applied Statistics* **36**, 318–324.

[28] Joshi, A., Ramakrishman, N., Houstis, E.N., and Rice, J.R. (1997). On neurobiological, neuro-fuzzy, machine learning, and statistical pattern recognition techniques. *IEEE Transactions on Neural Networks* **8**, 18–31.

[29] Simpson, P.K. (1992). Fuzzy min-max neural networks – Part 1: Classification. *IEEE Trans. Neural Networks* **3**, 776–786.

[30] Simpson, P.K. (1993). Fuzzy min-max neural networks – Part 2: Clustering. *IEEE Trans. Fuzzy Systems* **1**, 32–45.

[31] Cheeseman, P., and Stutz, J. (1995). Bayesian classification (AutoClass): theory and results. In *Advances in Knowledge Discovery and Data Mining*, U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy (Eds.). Menlo Park, California: The AAAI Press, pp. 61–83.