Journal of Classification 20:263-286 (2003)
DOI: 10.1007/s00357-003-0015-3

SOFTWARE REVIEW

Enhanced Model-Based Clustering, Density Estimation, and Discriminant Analysis Software: MCLUST

Chris Fraley

University of Washington, Seattle

Adrian E. Raftery

University of Washington, Seattle

Abstract: `MCLUST` is a software package for model-based clustering, density estimation and discriminant analysis interfaced to the S-PLUS commercial software and the R language. It implements parameterized Gaussian hierarchical clustering algorithms and the EM algorithm for parameterized Gaussian mixture models with the possible addition of a Poisson noise term. Also included are functions that combine hierarchical clustering, EM and the Bayesian Information Criterion (BIC) in comprehensive strategies for clustering, density estimation, and discriminant analysis. `MCLUST` provides functionality for displaying and visualizing clustering and classification results. A web page with related links can be found at http://www.stat.washington.edu/mclust.

Keywords: Clustering software; Model-based clustering; Mixture models; Cluster analysis; Discriminant analysis; Density estimation; Supervised classification; Unsupervised classification.

Thanks go to Ron Wehrens for many valuable comments and suggestions.

Funded by National Institutes of Health grant 1 R01 CA94212-01 and by Office of Naval Research contract N00014-01-10745.

Authors' Address: Chris Fraley and Adrian E. Raftery, Department of Statistics, University of Washington, Box 354322, Seattle, WA 98195-4322 USA.

1. Introduction

The MCLUST software, with offerings originally limited to model-based hierarchical clustering, was extended to include EM for parameterized Gaussian mixture models, as well as a clustering strategy in which the model and number of clusters are simultaneously selected via the Bayesian Information Criterion (Fraley and Raftery 1999). This manuscript describes a substantial upgrade to MCLUST, which includes the following among its new features:

- EM for four diagonal covariance mixture models.
- Density estimation via parameterized Gaussian mixtures.
- Simulation from parameterized Gaussian mixtures.
- Discriminant analysis via MclustDA.
- Methods for one dimensional data.
- Enhanced displays, including uncertainty plots and random projections.

MCLUST is interfaced to the S-PLUS commercial software[1] and the R language[2]. A user's guide for MCLUST including code examples is available via the web as a technical report (Fraley and Raftery 2002b). For a comprehensive treatment of the methods used in MCLUST, see Fraley and Raftery (2002a).

2. Models

In MCLUST, each cluster is represented by a Gaussian model

$$\phi_k(\mathbf{x} \mid \mu_k, \Sigma_k) = (2\pi)^{-\frac{p}{2}} |\Sigma_k|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\mathbf{x}_i - \mu_k)^T \Sigma_k^{-1} (\mathbf{x}_i - \mu_k)\right\}, \quad (1)$$

where $\mathbf{x}$ represents the data, and k is an integer subscript specifying a particular cluster. Clusters are ellipsoidal, centered at the means μ_k. The covariances Σ_k determine their other geometric features.

Each covariance matrix is parameterized by eigenvalue decomposition in the form

$$\Sigma_k = \lambda_k D_k A_k D_k^T,$$

where D_k is the orthogonal matrix of eigenvectors, A_k is a diagonal matrix whose elements are proportional to the eigenvalues of Σ_k, and λ_k is a scalar. The orientation of the principal components of Σ_k is determined by D_k, while A_k determines the shape of the density contours; λ_k specifies the volume of the

[1] Insightful Corp., Seattle, WA USA - http:ww.insightful.com/splus

[2] The Comprehensive R Archive Network - http://lib.stat/cmu.edu/R/CRAN. MCLUST port by Ron Wehrens, University of Nijmegen.

Table 1. Parameterizations of the covariance matrix Σ_k currently available in MCLUST for hierarchical clustering (HC) and/or EM for multidimensional data. ('•' indicates availability).

identifier	Model	HC	EM	Distribution	Volume	Shape	Orientation
EII	λI	•	•	Spherical	equal	equal	NA
VII	$\lambda_k I$	•	•	Spherical	variable	equal	NA
EEI	λA		•	Diagonal	equal	equal	coordinate axes
VEI	$\lambda_k A$		•	Diagonal	variable	equal	coordinate axes
EVI	λA_k		•	Diagonal	equal	variable	coordinate axes
VVI	$\lambda_k A_k$		•	Diagonal	variable	variable	coordinate axes
EEE	λDAD^T	•	•	Ellipsoidal	equal	equal	equal
VVV	$\lambda_k D_k A_k D_k^T$	•	•	Ellipsoidal	variable	variable	variable
EEV	$\lambda D_k A D_k^T$		•	Ellipsoidal	equal	equal	variable
VEV	$\lambda_k D_k A D_k^T$		•	Ellipsoidal	variable	equal	variable

corresponding ellipsoid, which is proportional to $\lambda_k^d |A_k|$, where d is the data dimension. Characteristics (orientation, volume and shape) can vary between clusters, or be constrained to be the same across clusters (Murtagh and Raftery 1984; Banfield and Raftery 1993; Celeux and Govaert 1995). This parameterization includes but is not restricted to well-known models such as equal-volume spherical variance ($\Sigma_k = \lambda I$) which gives the sum of squares criterion (Ward 1963), constant variance (Friedman and Rubin 1967), and unconstrained variance (Scott and Symons 1971).

In one dimension, there are just two models: `E` for equal variance and `V` for varying variance. In more than one dimension, the model identifiers code geometric characteristics of the model. For example, `EVI` denotes a model in which the volumes of all clusters are equal (`E`), the shapes of the clusters may vary (`V`), and the orientation is the identity (`I`). Clusters in this model have diagonal covariances with orientation parallel to the coordinate axes. Parameters associated with characteristics designated by `E` or `V` are determined from the data. Table 1 shows the various multivariate model options currently available in `MCLUST` for hierarchical clustering (denoted HC) and EM.

3. Hierarchical Clustering

`MCLUST` provides functions `hc` for model-based hierarchical agglomeration, and `hclass` for determining the resulting classifications. A classification maximum likelihood approach is used in `hc` to determine which two groups to merge at each stage (Banfield and Raftery 1993; Celeux and Govaert 1993; Fraley 1998). The function `hc` starts by default with every observation of the data in a cluster by itself, and continues until all observations are merged into a

single cluster. Arguments `partition` and `minclus` can be used to initialize the process at a chosen nontrivial partition, and to stop it before it reaches the final stage of merging.

4. EM for Mixture Models

`MCLUST` also provides functions implementing EM (Expectation-Maximization) methods for maximum likelihood clustering with parameterized Gaussian mixture models (Dempster, Laird and Rubin, 1977; McLachlan and Krishnan 1997). In this application, an iteration of EM consists of an 'E'-step, which computes a matrix z such that z_{ik} is an estimate of the conditional probability that observation i belongs to group k given the current parameter estimates, and an 'M-step', which computes maximum likelihood parameter estimates given z. In the limit, the parameters usually converge to the maximum likelihood values for the Gaussian mixture model

$$\prod_{i=1}^{n} \sum_{k=1}^{G} \tau_k \, \phi_k(\mathbf{x}_i \mid \mu_k, \Sigma_k),$$

and the column means of z converge to the mixing proportions τ_k. Here G is the number of groups in the data, which is fixed in the EM algorithm. The parameterizations of Σ_k currently available for EM in `MCLUST` are listed in Table 1. They are a subset of the parameterizations discussed in Celeux and Govaert(1995), which gives details of the EM algorithm for these models.

The relevant `MCLUST` functions are `em`, `me` (iterated M-step followed by E-step), `estep` and `mstep`. Functions `estep` and `mstep` implement the individual steps of the EM iteration. Conditional probabilities z and the loglikelihood can be recovered from parameters via `estep`, while parameters can be recovered from conditional probabilities z using `mstep`. These functions can also be used for discriminant analysis (see section 10.1).

5. Bayesian Information Criterion

The Baysian Information Criterion or BIC (Schwarz 1978) is the value of the maximized loglikelihood with a penalty on the number of parameters in the model, and allows comparison of models with differing parameterizations and/or differing numbers of clusters. In general the larger the value of the BIC, the stronger the evidence for the model and number of clusters (see, e.g. Fraley and Raftery 2002a). Other possible criteria are described in Chapter 6 of McLachlan and Peel (2000). `MCLUST` includes a function `bic` to compute the BIC given the maximized loglikelihood and the data dimensions. The next section describes functions that combine hierarchical clustering, EM, and BIC in a comprehensive model-based clustering strategy.

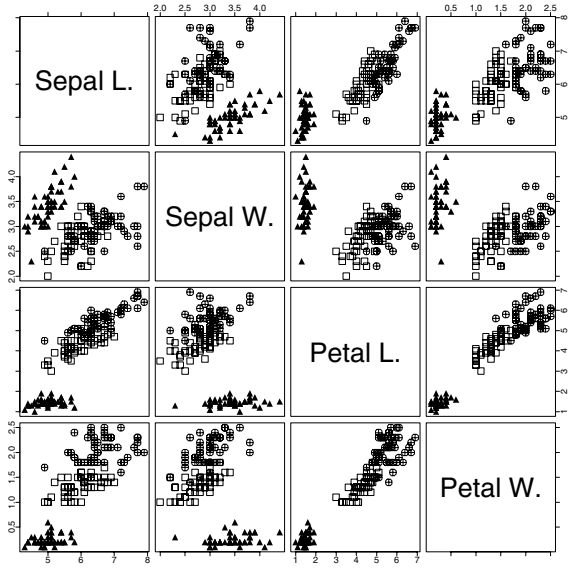

Figure 1. Pairs plot of Fisher's iris data showing classifcation into species. The plot was created with the MCLUST function clPairs, which uses different symbols to distinguish the three species.

6. Cluster Analysis

6.1 Model-based Clustering

MCLUST provides a function called EMclust for cluster analysis combining hierarchical clustering, EM, and BIC. The input to EMclust is the data, a list of models to apply in the EM phase, the desired numbers of groups to consider, and a hierarchical clustering in the same format as the output of hc for model-based hierarchical clustering (the default is to apply hc for the unconstrained model VVV to the data). EMclust returns BIC values for all of the chosen models and number of clusters, together with auxiliary information that is used by the corresponding summary method for recovering parameter values and clustering results.

For clustering examples, we will use Fisher's iris data (Fisher 1936). Figure 1 is a pairs plot of the iris data in which the three species are differentiated by symbol.

Figure 2 is a display of BIC values for Fisher's iris data, which is the result of applying plot to the EMclust output. The best model among those fitted by EMclust is the equal-shape model VEV, with 2 clusters. The same

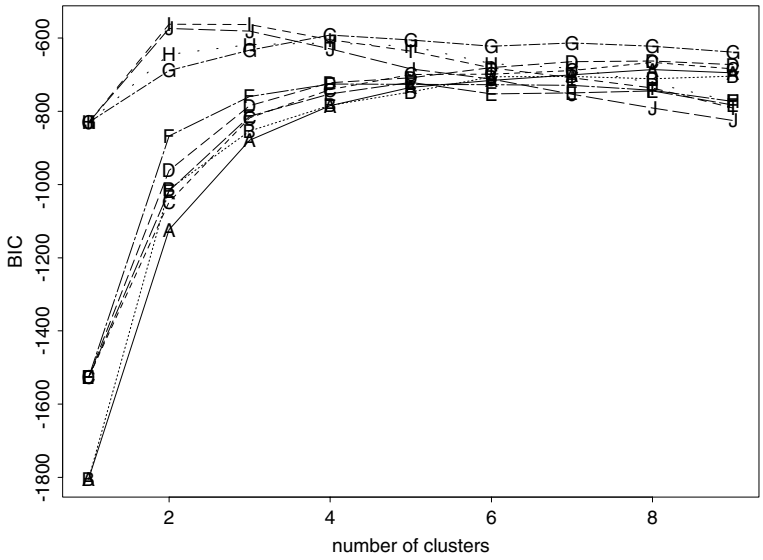

Figure 2. MCLUST plot of BIC values from EMclust for the models A - EII, B - VII, C - EEI, D - VEI, E - EVI, F - VVI, G - EEE, H - EEV, I - VEV, J - VVV applied to Fisher's iris data.

model with 3 clusters has a BIC value that is little different from the maximum; the conclusion is that there are either 2 or 3 clusters in the data under these models. The 2 cluster EMclust result separates one species from the other two while the 3 cluster result nearly separates the three species (there are 5 misclassifications out of 150).

Optimal parameter and z values are available through the summary function associated with EMclust objects, which has arguments allowing the summarizing information to be restricted to a subset of the number of clusters and models. The best classification (according to BIC) is recovered from summary by default.

6.2 Uncertainty

The uncertainty in the classification associated with the conditional probabilities z computed in the E-step of EM can be obtained by subtracting the probability of the most likely group for each observation from 1. The uncertainty of an observation i is equal to $1 - \max_k z_{ik}$ (Bensmail et al. 1997).

When groups intersect, it should be noted that uncertain classifications would be expected in the overlapping regions. When a true classification is

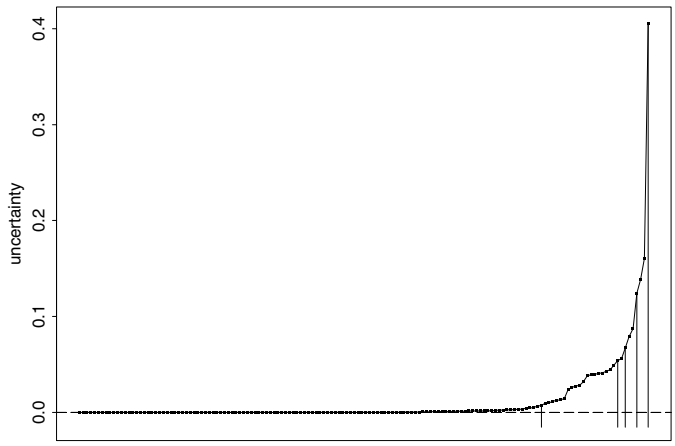

Figure 3. Uncertainty plot created with MCLUST function uncerPlot for the 3-cluster classification of Fisher's iris data via EM based on the equal-shape VEV model. The horizontal axis represents the individual observations in order of increasing uncertainty, while the vertical axis gives the value of the uncertainty. The vertical lines indicate misclassified observations.

known, the relative uncertainty of misclassified observations can be displayed by the function uncerPlot. An uncertainty plot for Fisher's iris data is shown in Figure 3.

6.3 Clustering with Noise and Outliers

MCLUST uses a mixture model which has a single term representing noise as a first order Poisson process to handle noisy data:

$$\prod_{i=1}^{n}\left[\frac{\tau_0}{V}+\sum_{k=1}^{K}\tau_k\phi_k(\mathbf{x}_i \mid \theta_k)\right], \tag{2}$$

in which V is the hypervolume of the data region, and $\tau_k \geq 0$; $\sum_{k=0}^{G}\tau_k = 1$.

This model has been used successfully in a number of applications (Banfield and Raftery 1993; Dasgupta and Raftery 1998; Campbell et al. 1997, 1999; Bensmail and Meulman 2003).

The basic model-based clustering method needs to be modified when the data contains noise. First, a good initial noise estimate must be obtained. Some possible methods for denoising include a Voronoï method (Allard and Fraley 1997) and a nearest-neighbor method (Byers and Raftery 1998). Next,

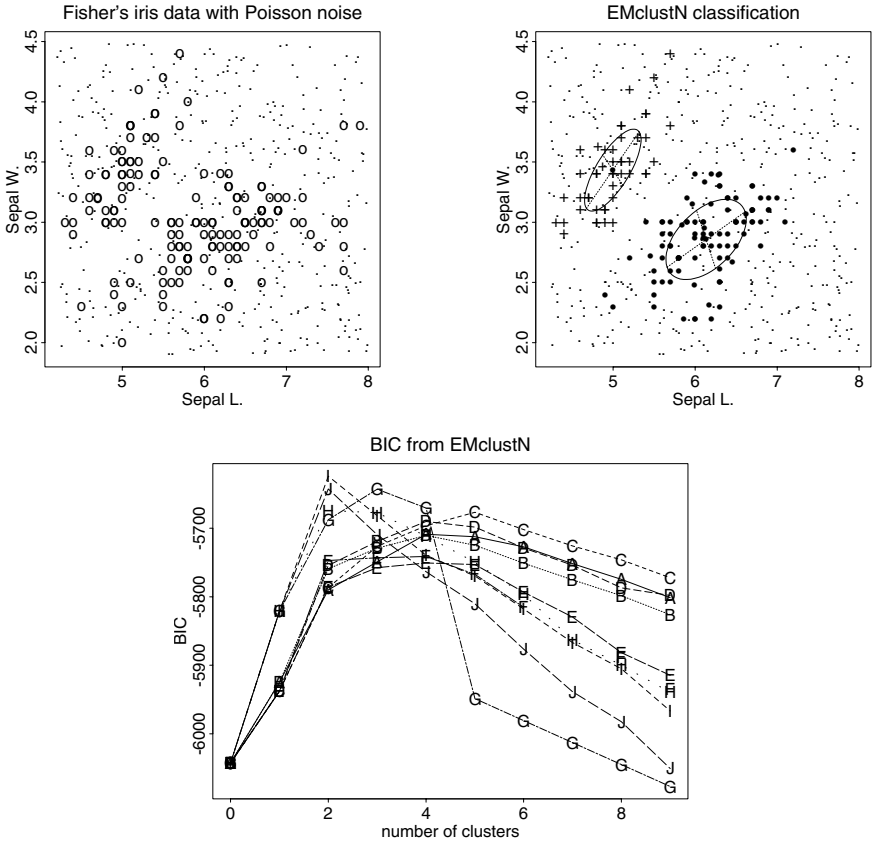

Figure 4. Cluster analysis of Fisher's iris data with added Poisson noise. Upper Left: A projection Fisher's iris data (circles) with 450 Poisson noise points (small dots). Upper Right: `EMclustN` classification. The ellipses are the projections of the standard deviation of each Gaussian component. Lower: BIC from `EMclustN`.

hierarchical clustering is applied to the denoised data. Finally, EM based on the Gaussian model with the added noise term (2) is applied to the entire data set, with the data removed in the denoising process as the initial noise estimate.

`MCLUST` provides a function `EMclustN` for model-based clustering with noise. Figure 4 shows an example of an analysis with `EMclustN` in which Poisson noise has been added to Fisher's iris data. A random initial estimate was used for the noise.

6.4 Further Considerations

For a complete analysis, it may be desirable to try varying models, initialization strategies for EM and values for the convergence tolerance, as well

as using permutations or subsets of the observations, and/or perturbations of the data, to see if the classification remains stable. Scaling or otherwise transforming the data may also affect the results. It is advisable to examine the data beforehand, in case (for example) the dimensions can be reduced due to highly correlated variables.

Finally, it is important to take into account numerical issues in cluster analysis. The EM computations break down when the covariance corresponding to one or more components becomes ill-conditioned (singular or nearly singular). In general they cannot proceed if clusters contain only a few observations or if the observations they contain are very nearly colinear. If EM for a model having a certain number of components is applied to a mixture in which there are actually fewer groups, then it may fail due to ill-conditioning. The EM functions in `MCLUST` compute and monitor the conditioning of the covariances, and an error condition is issued when the associated covariance appears to be nearly singular, as determined by a threshold that can be specified by the user.

7. Simulation from Mixture Densities

Because the cluster analysis strategy described in Section 6 is a model-fitting procedure, it has uses other than grouping observations. Given the parameters for a mixture model, data can be simulated from that model for evaluation and verification. The function `sim` allows simulation from mixture models generated by `MCLUST` functions. Besides the model, `sim` allows a seed as input for reproducibility. Projections of simulated data from the 2 and 3 class equal-shape models produced by `EMclust` for Fisher's iris data in Section 6.1 are shown in Figure 5.

8. Density Estimation

The clustering capabilities of `MCLUST` can also be viewed as a general strategy for multivariate density estimation. `MCLUST` can be used for density estimation by first using `EMclust` to get a model for the data, and then using `dens` to get the density of a given point relative to that model.

As an example, we look at density estimation for the location of maple trees in the Lansing Woods (Gerrard 1969; Kaluzny et al. 1998) (see Figure 6). The density estimate for the Lansing Woods maples over a 100×100 grid is shown Figure 7. For information on creating classification and density displays for model-based clustering results, see Section 9.

Probably the most common application for density estimation is discriminant analysis, which is discussed in Section 10.

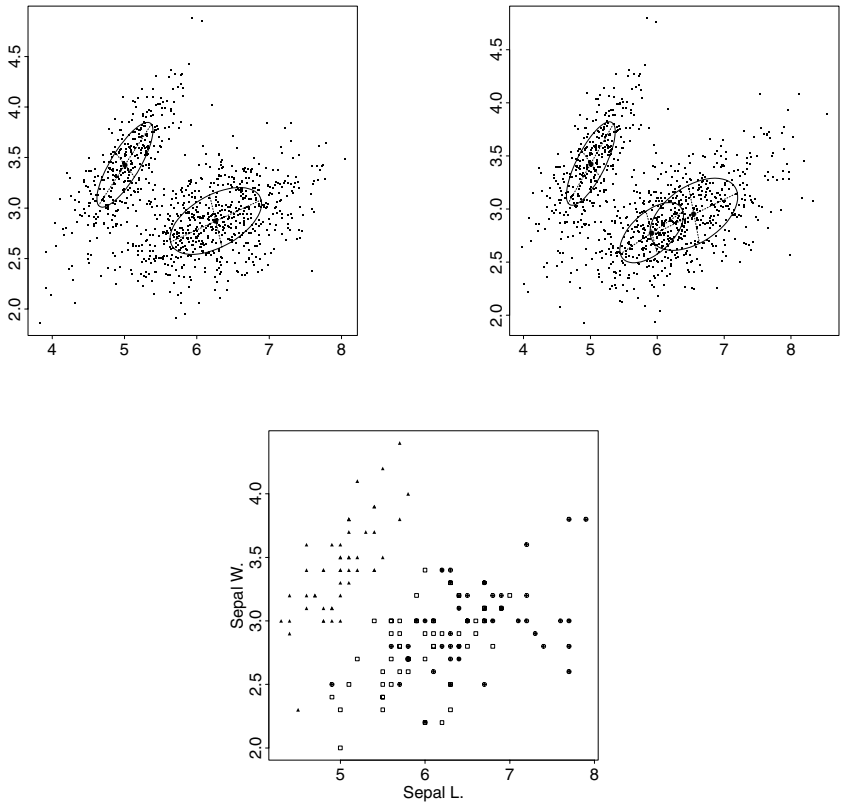

Figure 5. A projection of data simulated from `EMclust` models for Fisher's iris data. Upper Left: 1000 data points simulated from the 2-component `VEV` model. Upper Right: 1000 data points simulated from the 3-component `VEV` model. Lower: Fisher's iris data showing classfication. The ellipses shown are projections of the standard deviation of the mixture components.

9. Displays

Once parameters values are available, projections of the data showing the means and standard deviations of the corresponding clusters may be plotted. In the two-dimensional case, density and uncertainty surfaces may also be plotted.

9.1 Plotting Two-Dimensional Results

The function `mclust2Dplot` may be used for displaying the classification, uncertainty, or classification errors for `MCLUST` models of two-dimensional data. Classification and uncertainty plots for the Lansing Woods maples exam-

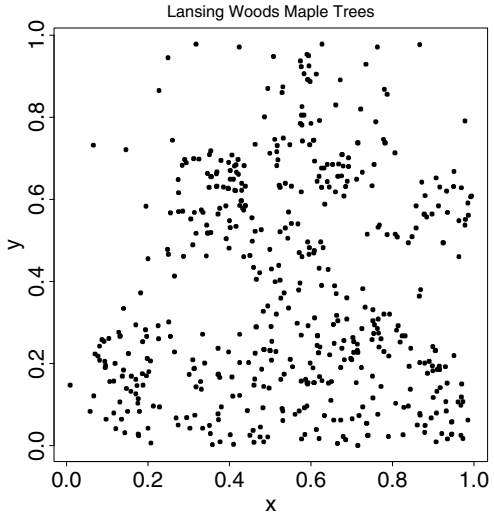

Figure 6. Lansing Woods maple trees.

ple of Section 8 are displayed in Figure 8.

The function `surfacePlot` may be used for displaying the density or uncertainty for `MCLUST` models of two-dimensional data. It also returns the grid coordinates and corresponding surface values. Figure 9 shows the density and uncertainty surfaces for the Lansing Woods maples example.

9.2 Plotting Multidimensional Results

9.2.1 Coordinate Projections

The `MCLUST` function `coordProj` is for plotting coordinate projections. If the plot type is not specified, `coordProj` will offer a menu of options if there is more than one possibility with the given input. Plots of the `EMclust` result for Fisher's iris data (Section 6.1) are displayed using a coordinate projection of the data in Figure 10.

9.2.2 Random Projections

Projections with randomly generated coordinates are often useful in visualizing multidimensional data. The `MCLUST` function `randProj` is for plotting random projections. Plots of the `EMclust` result for the iris data are displayed using random projections in Figure 11.

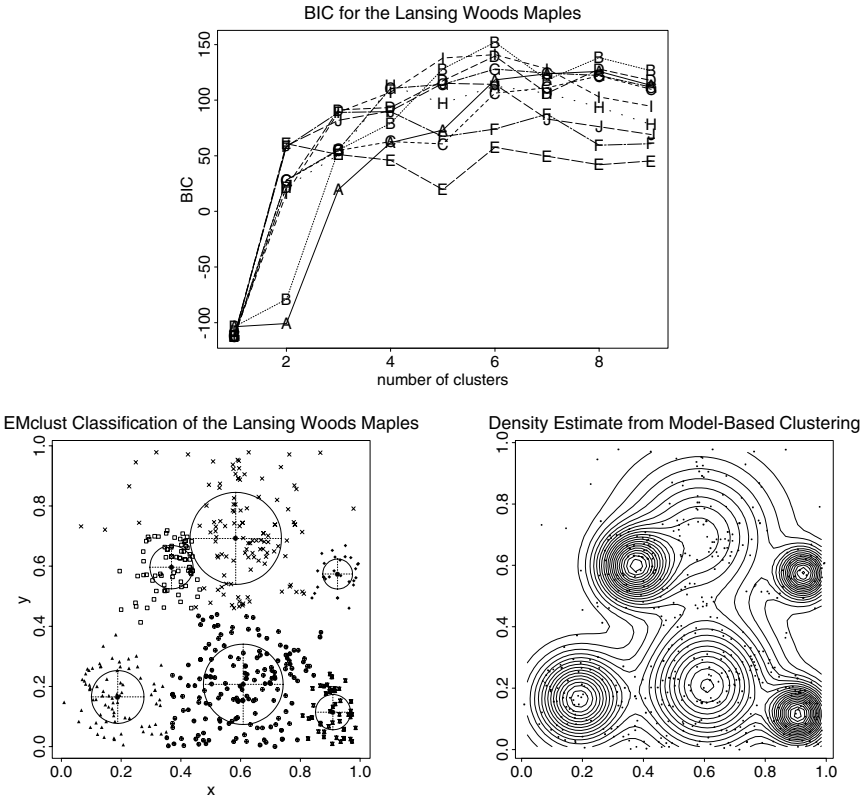

Figure 7. MCLUST density estimate for the Lansing Woods maples. Top: BIC from model-based clustering. Left: EMclust classification with circles/ellipses indicating the standard deviation of each component. Right: Density contours with the location of the maples superimposed.

10. Discriminant Analysis

In discriminant analysis, observations of known classification are used to classify others. MCLUST provides a number of functions that can be used for discriminant analysis. We demonstrate some possible methods applied to the Lansing Woods data (Gerrard 1969; Kaluzny et al. 1998), which gives the spatial location of maple and hickory trees (see Figure 12).

10.1 Discriminant Analysis Using mstep and estep

MCLUST functions mstep and estep implementing the individual steps of the EM algorithm for Gaussian mixtures can be used for discriminant analy-

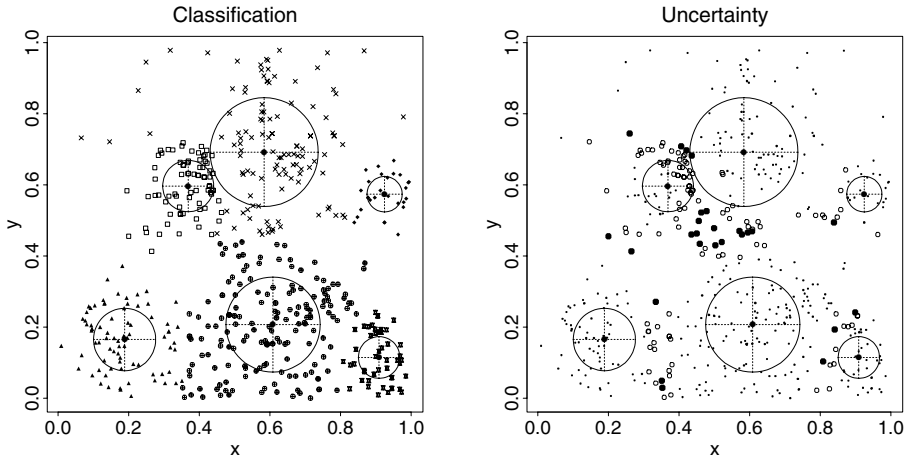

Figure 8. Classification (left) and uncertainty (right) plots created with `mclust2Dplot` for the `EMclust` modeling of the Lansing Woods maples. The circles shown are the standard deviations of each mixture component. In the classification plot, points in different classes are indicated by different symbols. In the uncertainty plot, the symbols have the following meaning: large filled symbols, 95% quantile of uncertainty; smaller open symbols, 75–95% quantile; small dots, first three quartiles of uncertainty.

sis. The idea is to produce a density estimate for the training data which is a mixture model, in which each known class is modeled by a single Gaussian term.

First, the parameterization giving the best model fit to the training data must be chosen. Most commonly, this would be done by leave-one-out cross validation. Leaving out one training observation at a time, the `MCLUST` function `cv1EMtrain` fits each model using `mstep`, then classifies the observation that was left out using `estep`. It is an implementation the model-selection step of *Eigenvalue Decomposition Discriminant Analysis* or *EDDA* developed by Bensmail and Celeux (1996) that uses the 10 models currently available in `MCLUST` out of 14 possible models. The output of `cv1EMtrain` is the error rate for each model; that is, the fraction of left-out observations correctly classified by the model fit to the remaining observations.

For the Lansing Woods data with the odd-numbered observations as a training set, the error rate for the odd-numbered data is 33%, while for the even-numbered data it is 31%. The results for this analysis are displayed in the left hand plot of Figure 13.

Another option for model selection that is much quicker to compute than crossvalidation is to use `mstep` to fit each model to the training data, then select the best fitting model via BIC. A function `bicEMtrain` is provided in

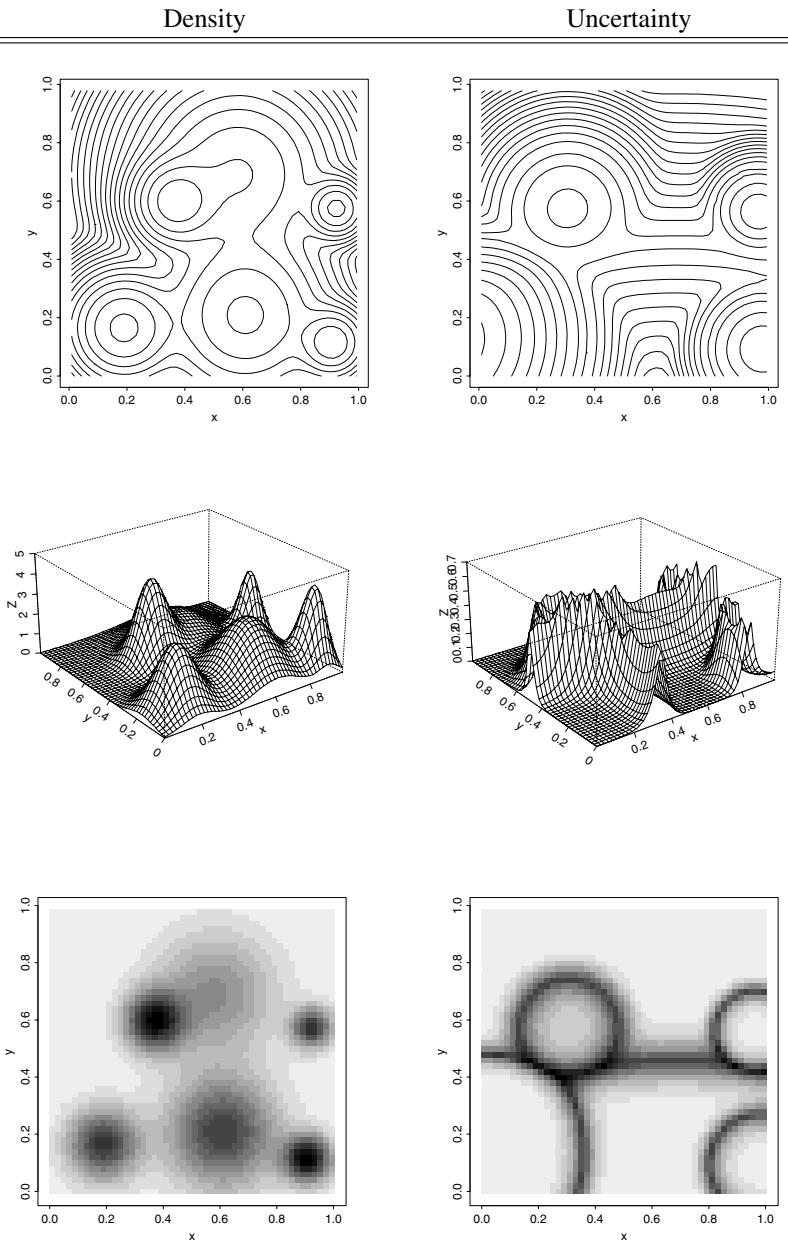

Figure 9. Density (left column) and uncertainty (right column) surfaces for the Lansing Woods maples. Contour, perspective, and image plots are displayed in the first, second, and third rows respectively. A logarithmic scale was used for the contour plots.

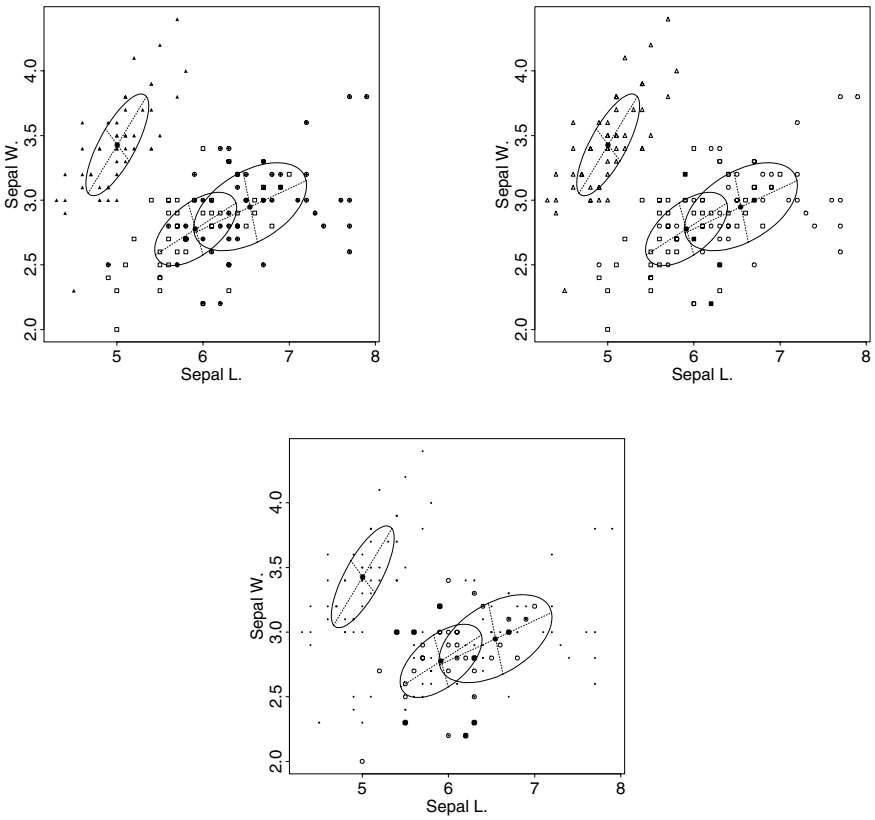

Figure 10. Projection of the first and second coordinates of Fisher's iris data created with `coordProj`. Plots show the 3 cluster classification (top left), with the associated classification errors (top right) and uncertainty (bottom) for the equal-shape Gaussian mixture model (`VEV`).

`MCLUST` for this purpose. Using this approach on the Lansing Woods data, the error rate for the training [odd-numbered] observations is about 35%, while for the test [even numbered] observations it is 33%. The results for this analysis are displayed in the right hand plot of Figure 13.

Although the error rate for the test data is somewhat higher for BIC than for crossvalidation, it should be noted that it took more than 10 minutes to execute `cv1EMtrain` on this training data, while `bicEMtrain` executed in about half a second.[3]

[3] This is not a precise algorithmic timing comparison because crossvalidation can be accomplished more efficiently for these models using updating schemes, and because the training code could be made more efficient if more of it were written in a compiled language rather than in S-PLUS.

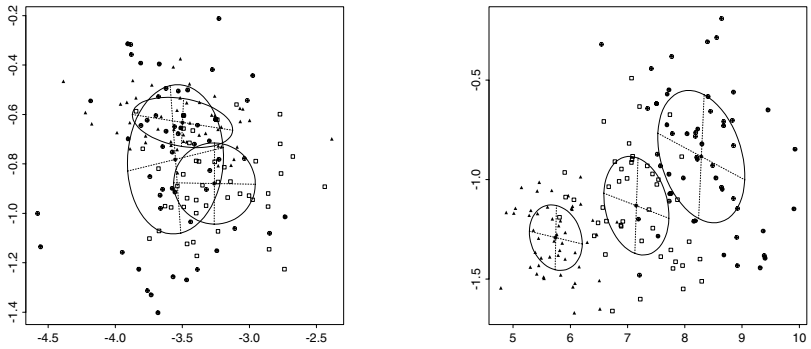

Figure 11. Two random projections of Fisher's iris data created with `randProj`. Plots show the 3-cluster classification for the equal-shape Gaussian mixture model (`VEV`).

10.2 Mixture Discriminant Analysis via MclustDA

In Section 10.1, discriminant analysis was accomplished modeling the training data by a mixture density with a single Gaussian component for each class. That section also showed how to choose the appropriate cross-cluster constraints to give the lowest training error rate using either leave-one-out cross-validation or BIC. An alternative is to use model-based clustering to fit a Gaussian mixture model as a density estimate for each class in the training set. This extends a method for discriminant analysis described in Hastie and Tibshirani (1996) to include a range of models for the covariance matrices, and BIC to select the model and number of clusters. `MCLUST` provides functions `mclustDAtrain` and `mclustDAtest` for this purpose.

Function `mclustDAtrain` also allows users to choose training model parameterizations, selecting from among all available models as a default. The output of `mclustDAtrain` is a list, each element being the model for each class.

By default, `mclustDAtrain` will fit up to nine components for each possible model. For the odd-numbered observations in the Lansing Woods data, `mclustDAtrain` chooses the 7-class `EEV` model for the hickory training class and the 4-class `EEE` model for the maple training class.

The density of the test data under the training models can be obtained using `mclustDAtest`, while the classification and posterior probabilities of the test data can be recovered from the `summary` function for `mclustDAtest`. The error rates are about 23% and 25% for the training [odd-numbered] and test [even-numbered] data, respectively. Figure 14 shows the models selected

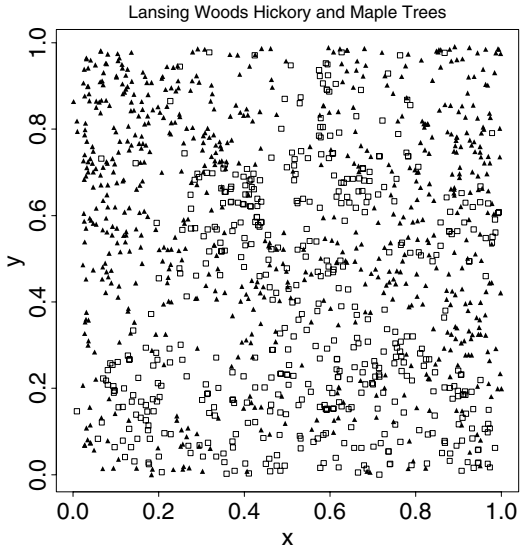

Figure 12. Location of Hickory and Maple Trees in the Lansing Woods. There are 703 hickories (triangles) and 514 maples (squares) in the data set.

for each class of the training data, as well as the discriminant curve delineating the two classes and the misclassified (training and test) data points.

11. One Dimensional Data

The `MCLUST` functions for clustering, density estimation and discriminant analysis work on one-dimensional as well as multidimensional data. Analysis is somewhat simplified since there are only two possible models — equal variance (`E`) or varying variance (`V`).

11.1 Clustering

Cluster analysis for one-dimensional data can be carried out as for two and higher dimensions, except that there is a special plotting function `mclust-1Dplot`. As an example, we use simulated data consisting of two clusters of size 300 with variance 1 centered at -9 and 9, respectively, and one cluster of size 400 with variance 4 centered at 0. Figure 15 shows the BIC, classification, uncertainty, and density for this simulated example.

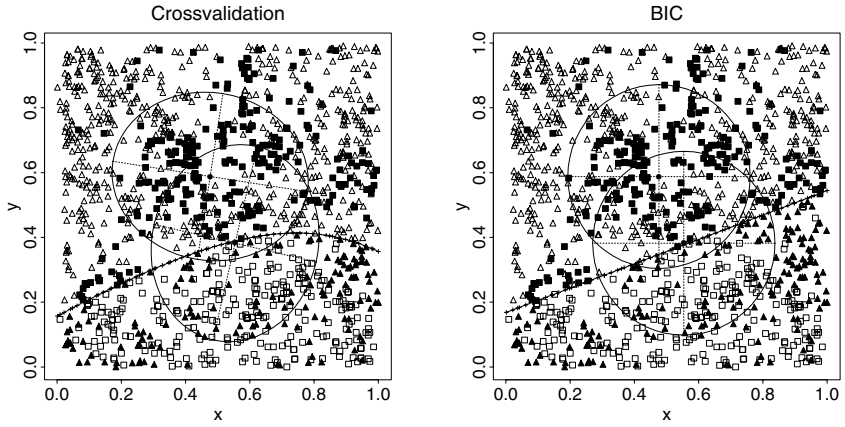

Figure 13. Discriminant analysis using `mstep` and `estep`. The plots show the ellipses corresponding to the standard deviation of each of the two Gaussians used to fit the training data, as well as the discriminant curve (hatched line) where the two groups contribute equally to the mixture density. Filled symbols represent misclassied training [odd-numbered] and test [even-numbered] observations. Left: Model `EEV` selected by leave-one-out crossvalidation. Right: Model `EII` selected via BIC.

11.2 Discriminant Analysis

Discriminant analysis via EM (Section 10.1) or `MclustDA` (Section 10.2) is also possible in in one dimension; the same functions are used as in the multidimensional case.

11.3 `"mclust"` Option for S-PLUS `density` Function

`MCLUST` includes an augmented version of the S-PLUS function `density` for computing the density of a one-dimensional data set. A `method` argument has been added, with the option to specify `method = "mclust"` to have the density computed via model-based clustering, instead of the default kernel estimate. It has been shown in simulation studies that mixtures of normals with equal variance can give substantially better density estimates than kernel-based methods for one-dimensional data (Roeder and Wasserman 1997).

12. Extensions

12.1 Large Data Sets

The clustering strategy used in the function `EMclust` is not suitable for large data sets because of the use of model-based hierarchical clustering to ini-

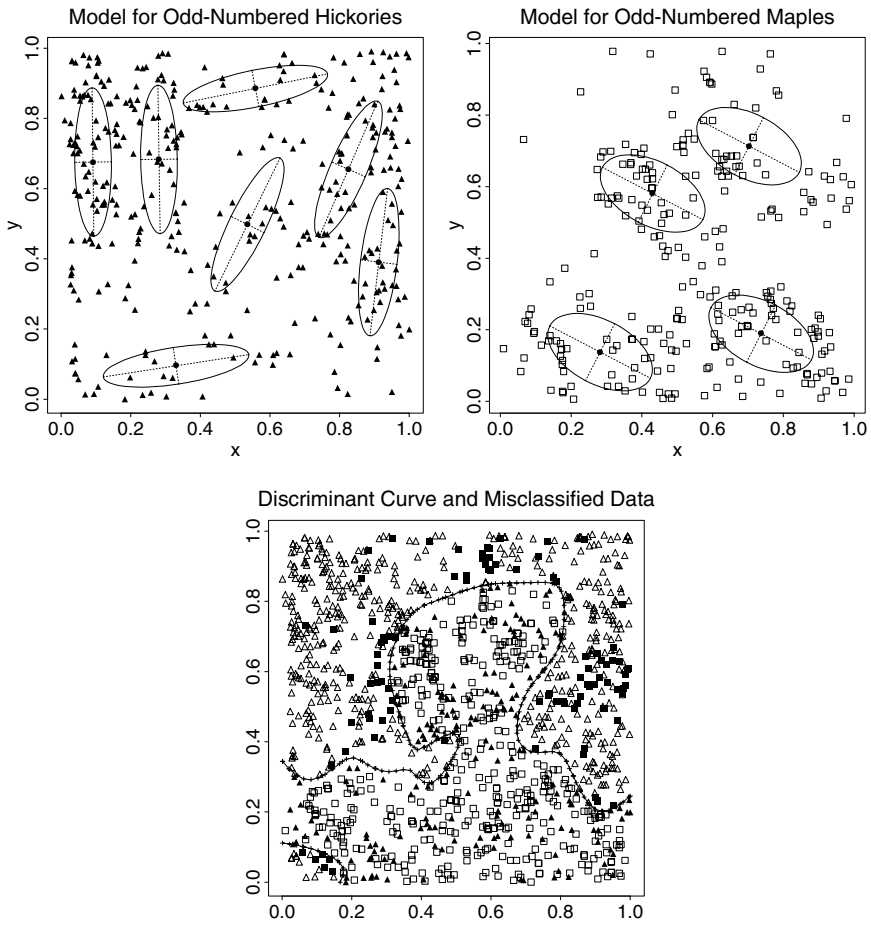

Figure 14. `mclustDAtrain` models and discriminant curve for the Lansing Woods data. Top Left: the hickory training [odd-numbered] class with its 7-group `EEV` model. Top Right: the maple training [odd-numbered] class with its 4-group `EEE` model. Bottom: discriminant curve with training and test errors (filled symbols).

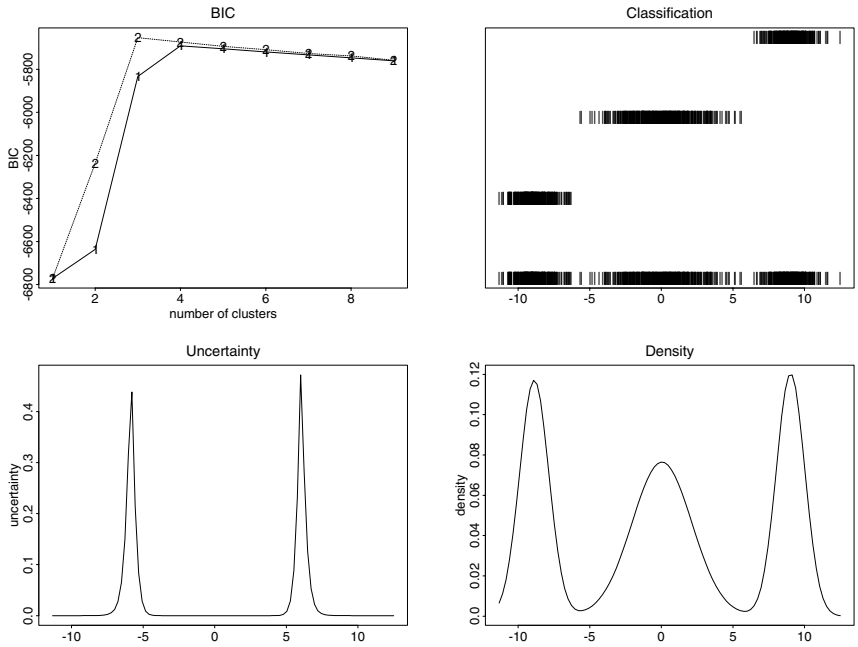

Figure 15. MCLUST plots for model-based clustering of one-dimensional data. Clockwise from upper left: BIC, classification, uncertainty, and density from EMclust applied to the simulated one-dimensional example. In the classification plot at the top right, all of the data is displayed at the bottom, with the separated classes shown different levels above.

tialize EM. However, if hierarchical clustering is applied to a sample of the data, there are a number of possibilities for clustering large data sets. These include clustering the sample, and then either extending the classification to the full data set via discriminant analysis, or using mixture parameters obtained for the sample as starting values for EM applied to the full data set. The function EMclust includes a provision for using a subset of the data in the hierarchical clustering phase. These and other alternatives for large data sets are discussed in Wehrens et al. (2003).

12.2 High Dimensional Data

Models in which the orientation is allowed to vary between clusters (EEV, VEV, EVV, VVV) have $\mathcal{O}(d^2)$ parameters per cluster, where d is the dimension of the data. For this reason, MCLUST may not work well or may otherwise be inefficient for these models when applied to high-dimensional data. It may still be possible to analyze such data with MCLUST by restriction to models with

fewer parameters (e.g. spherical or diagonal models), or else by applying a dimension-reduction technique such as principal components.

Some of the more parsimonious models (e.g. spherical, diagonal, or fixed covariance) can be applied to datasets in which the number of observations is smaller than the data dimension.

13. Function Summary

13.1 Hierarchical Clustering

`hc`	Merge sequences for model-based hierarchical clustering.
`hclass`	Classifications corresponding to `hc` results.

13.2 Parameterized Gaussian Mixture Models

`em`	EM algorithm (starting with E-step).
`me`	EM algorithm (starting with M-step).
`estep`	E-step of the EM algorithm.
`mstep`	M-step of the EM algorithm.
`mvn`	One-component fit.

13.3 Density Computation for Parameterized Gaussian Mixtures

`cdens`	Component density (without mixing proportions).
`dens`	Mixture density.

13.4 Model-based Clustering / Density Estimation

`EMclust`	BIC computation; clusters and models through `summary`.
`Mclust`	Combines `EMclust` and its `summary` (fewer options).
`densityMclust`	S-PLUS one-dimensional density function with `method = "mclust"` option.

13.5 Discriminant Analysis

Class Densities as Mixture Components

`cv1EMtrain`	Training via leave-one-out crossvalidation.
`bicEMtrain`	Training via BIC.
`estep`	E-step of the EM algorithm.
`mstep`	M-step of the EM algorithm.

Parameterized Gaussian Mixture for Class Densities (MclustDA)

`mclustDAtrain`	MclustDA training.
`mclustDAtest`	MclustDA density; classification via `summary`.
`mclustDA`	Combines `mclustDAtrain` and `mclustDAtest`(fewer options).

13.6 Support for Modeling and Classification

`.Mclust`	vector of default values.
`map`	Convert conditional probabilities to a classification.
`unmap`	Convert a classification to indicator variables.
`bic`	BIC for parameterized Gaussian mixture models.
`sim`	Simulate data from a parameterized Gaussian mixture model.
`compareClass`	Compare two classifications.
`mapClass`	Mapping between two classifications.
`classError`	Classification error.
`classErrors`	Identification of classification errors.
`sigma2decomp`	Convert mixture covariances to decomposition form.
`decomp2sigma`	Convert decomposition form to mixture covariances.

13.7 Plotting Functions

13.7.1 One-Dimensional Data

`mclust1Dplot` Classification, uncertainty, density and/or classification errors.

13.7.2 Two-Dimensional Data

`mclust2Dplot` Classification, uncertainty, and/or classification errors.

`surfacePlot` Contour, image, or perspective plot of either density or uncertainty.

13.7.3 More than Two Dimensions

Classification, uncertainty, and/or classification errors.

`coordProj`	coordinate projections
`randProj`	random projections
`spinProj`	random projection followed by reflection or rotation.

13.7.4 Other Plotting Functions

`clPairs` pairs plot showing classifications
`uncerPlot` relative uncertainty of misclassified observations.

References

ALLARD, D., and FRALEY, C.(1997), "Nonparametric maximum likelihood estimation of features in spatial point processes using Voronoï tessellation," *Journal of the American Statistical Association 92*, 1485–1493. (available as technical report no. 293R at http://www.stat.washington.edu/www/research/reports).

BANFIELD, J. D., and RAFTERY, A. E. (1993), "Model-based Gaussian and non-Gaussian clustering," *Biometrics 49*, 803–821.

BENSMAIL, H., and CELEUX, G. (1996), "Regularized Gaussian discriminant analysis through eigenvalue decomposition," *Journal of the American Statistical Association 91*, 1743–1748.

BENSMAIL, H., CELEUX, G., RAFTERY, A. E., and ROBERT, C. P. (1997), "Inference in model-based cluster analysis," *Statistics and Computing 7*, 1–10.

BENSMAIL, H., and MEULMAN, J. J. (2003), "Model-based clustering with noise: Bayesian inference and estimation," *Journal of Classification 20*, 49–76.

BYERS, S. D., and RAFTERY, A. E. (1998), "Nearest neighbor clutter removal for estimating features in spatial point processes," *Journal of the American Statistical Association 93*, 577–584.

CAMPBELL, J. G., FRALEY, C., MURTAGH, F., and RAFTERY, A. E. (1997), "Linear flaw detection in woven textiles using model-based clustering," *Pattern Recognition Letters 18*, 1539–1548. (available as technical report no. 314 at http://www.stat.washington.edu/www/research/reports).

CAMPBELL, J. G., FRALEY, C., STANFORD, D., MURTAGH, F., and RAFTERY, A. E. (1999), "Model-based methods for real-time textile fault detection," *International Journal of Imaging Systems and Technology 10*, 339–346.

CELEUX, G., and GOVAERT, G. (1993), "Comparison of the mixture and the classification maximum likelihood in cluster analysis," *Journal of Statistical Computation and Simulation 47*, 127–146.

CELEUX, G., and GOVAERT, G. (1995), "Gaussian parsimonious clustering models," *Pattern Recognition 28*, 781–793.

DASGUPTA, A., and RAFTERY, A. E. (1998), "Detecting features in spatial point processes with clutter via model-based clustering," *Journal of the American Statistical Association 93*, 294–302.

DEMPSTER, A. P., LAIRD, N. M. and RUBIN, D. B. (1977), "Maximum likelihood for incomplete data via the EM algorithm (with discussion)," *Journal of the Royal Statistical Society, Series B 39*, 1–38.

FISHER, R. A. (1936), "The use of multiple measurements in taxonomic problems," *Annals of Eugenics 7*, 179–188.

FRALEY, C. (1998), "Algorithms for model-based Gaussian hierarchical clustering," *SIAM Journal on Scientific Computing 20*, 270–281. (available as technical report no. 311 at http://www.stat.washington.edu/www/research/reports).

FRALEY, C., and RAFTERY, A. E. (1999), "MCLUST: Software for model-based cluster analysis," *Journal of Classification 16*, 297–306.

FRALEY, C., and RAFTERY, A. E. (2002a, October), MCLUST: Software for model-based clustering, density estimation, and discriminant analysis, Technical Report 415, University of Washington, Department of Statistics. (available at http://www.stat.washington.edu/www/research/reports).

FRALEY, C., and RAFTERY, A. E. (2002b), "Model-based clustering, discriminant analysis and density estimation," *Journal of the American Statistical Association 97*, 611–631. (available as technical report no. 380 at http://www.stat.washington.edu/www/research/reports).

FRIEDMAN, H. P., and RUBIN, J. (1967), "On some invariant criteria for grouping data.," *Journal of the American Statistical Association 62*, 1159–1178.

GERRARD, D. J. (1969). Competition quotient: a new measure of the competition affecting individual forest trees, Research Bulletin No. 20, Agricultural Experimental Station, Michigan State University.

HASTIE, T., and TIBSHIRANI, R. (1996), "Discriminant analysis by Gaussian mixtures," *Journal of the Royal Statistical Society, Series B 58*, 155–176.

KALUZNY, S. P., VEGA, S. C., CARDOSO, T. P., and SHELLY, A. A. (1998), *S+SpatialStats: User's manual for Windows and UNIX*, Springer.

MCLACHLAN, G. J., and KRISHNAN, T. (1997), *The EM Algorithm and Extensions*, Wiley.

MCLACHLAN, G. J., and PEEL, D. (2000), *Finite Mixture Models*, Wiley.

MURTAGH, F., and RAFTERY, A. E. (1984), "Fitting straight lines to point patterns," *Pattern Recognition 17*, 479–483.

ROEDER, K., and WASSERMAN, L. (1997), "Practical Bayesian density estimation using mixtures of normals," *Journal of the American Statistical Association 92*, 894–902.

SCHWARZ, G. (1978), "Estimating the dimension of a model," *The Annals of Statistics 6*, 461–464.

SCOTT, A. J., and SYMONS, M. J. (1971), "Clustering methods based on likelihood ratio criteria," *Biometrics 27*, 387–397.

WARD, J. H. (1963), "Hierarchical groupings to optimize an objective function," *Journal of the American Statistical Association 58*, 234–244.

WEHRENS, R., BUYDENS, L., FRALEY, C., and RAFTERY, A. E. (2003, February), Model-based clustering for image segmentation and large datasets via sampling. Technical Report 424, University of Washington, Department of Statistics.