Journal of Classification 16:297-306 (1999)

Software Abstract

MCLUST: Software for Model-Based Cluster Analysis

Chris Fraley
University of Washington

Adrian E. Raftery
University of Washington

MCLUST is a software package for cluster analysis implementing parameterized Gaussian hierarchical clustering algorithms (Murtagh and Raftery 1984; Banfield and Raftery 1993; Fraley 1998) and the EM algorithm for parameterized Gaussian mixture models with the possible addition of a Poisson noise term. **MCLUST** also includes functions that combine hierarchical clustering, EM and the Bayesian Information Criterion (BIC) in a comprehensive clustering strategy (Dasgupta and Raftery 1998; Fraley and Raftery 1998). Methods of this type have shown promise in a number of practical applications, including character recognition (Murtagh and Raftery 1984), tissue segmentation (Banfield and Raftery 1993), minefield and seismic fault detection (Dasgupta and Raftery 1998), identification of textile flaws from images (Campbell, Fraley, Murtagh, and Raftery 1997), and classification of astronomical data (Celeux and Govaert 1995; Mukerjee, Feigelson, Babu, Murtagh, Fraley, and Raftery 1998). A web page with related links can be found at http://www.stat.washington.edu/fraley/mclust

---

Funded by the Office of Naval Research under contracts N00014-96-1-0192 and N00014-96-1-0330.

Authors' Address: C. Fraley and A. Raftery, Department of Statistics, Box 354322, University of Washington, Seattle, WA 98195-4322 USA;
e-mail: fraley/raftery@stat.washington.edu

1. Models

In **MCLUST**, each cluster is represented by a Gaussian model

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

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

$$\Sigma_k = \lambda_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k^T,$$

where $\mathbf{D}_k$ is the orthogonal matrix of eigenvectors, $\mathbf{A}_k$ 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 $\mathbf{D}_k$, while $\mathbf{A}_k$ determines the shape of the density contours; λ_k specifies the volume of the corresponding ellipsoid, which is proportional to $\lambda_k^d \mid \mathbf{A}_k \mid$, where d is the data dimension. Characteristics (orientation, volume, and shape) of distributions are usually estimated from the data, and can be allowed to vary between clusters, or constrained to be the same for all 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 uniform 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).

Table 1 shows the various model options currently available in **MCLUST** for hierarchical clustering (denoted HC) and EM ('×' in the appropriate column indicates availability). The model identifiers code geometric characteristics of the model. For example, **EFV** denotes a model in which the volumes of all clusters are equal (**E**), the shapes of all clusters are fixed (**F**) in advance by the user, and the orientation is allowed to vary (**V**) among the clusters. Parameters associated with characteristics designated by **E** or **V** are determined from the data.

Table 1

Parameterizations of the covariance matrix Σ_k in MCLUST.

ID	Model	HC	EM	Distribution	Volume	Shape	Orientation
EI	$\lambda\mathbf{I}$	×	×	Spherical	equal	equal	NA
VI	$\lambda_k\mathbf{I}$	×	×	Spherical	variable	equal	NA
EEE	$\lambda\mathbf{D}\mathbf{A}\mathbf{D}^T$	×	×	Ellipsoidal	equal	equal	equal
VVV	$\lambda_k\mathbf{D}_k\mathbf{A}_k\mathbf{D}_k^T$	×	×	Ellipsoidal	variable	variable	variable
EFV	$\lambda\mathbf{D}_k\hat{\mathbf{A}}\mathbf{D}_k^T$	×		Ellipsoidal	equal	fixed	variable
EEV	$\lambda\mathbf{D}_k\mathbf{A}\mathbf{D}_k^T$		×	Ellipsoidal	equal	equal	variable
VFV	$\lambda_k\mathbf{D}_k\hat{\mathbf{A}}\mathbf{D}_k^T$	×		Ellipsoidal	variable	fixed	variable
VEV	$\lambda_k\mathbf{D}_k\mathbf{A}\mathbf{D}_k^T$		×	Ellipsoidal	variable	equal	variable

2. Obtaining and Using MCLUST

MCLUST is written in FORTRAN and interfaced to the S-PLUS commercial software package[1] It can be obtained via the world wide web from the S-archive of StatLib

http://www.stat.cmu.edu/S/mclust

or else at

http://www.stat.washington.edu/fraley/mclust/soft.html

which also includes a link to an expanded version of this document. **MCLUST** is also available via anonymous ftp from ftp.u.washington.edu in the directory public/mclust. A file giving installation instructions and examples of usage is included.

3. Hierarchical Clustering

MCLUST provides functions **mhtree** for computing classification trees via model-based hierarchical agglomeration, and **mhclass** for

1. MathSoft, Inc., Seattle, WA, USA — http://www.mathsoft.com/splus

determining the resulting classifications. As an example of the use of **mhtree** and **mhclass**, consider Fisher's iris data (Fisher 1936), which is available as a data set in S-PLUS. We first transform the data from a three-dimensional array to a matrix in which the species information is lost, then apply the hierarchical clustering algorithm for non-uniform spherical variances (VI):

```
> iris.matrix <- matrix(aperm(iris,c(1,3,2) ),150,4,dimn=dimnames(iris)[1:2])
> cltree <- mhtree(iris.matrix, modelid = "VI")
```

The classification produced by **mhtree** for various numbers of clusters can be obtained with **mhclass**. For example, for the classifications corresponding to 2 and 3 clusters:

```
> cl <- mhclass(cltree, 2:3)
```

Classifications can be displayed with the data by means of the function **clpairs**:

```
> clpairs(iris.matrix, cl[,"3"])
```

Figure 1 shows the 3-cluster classification for the iris data with this model.

The function **mhtree** 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. However there is the option of initializing the process at a chosen nontrivial partition, and of stopping it before the final stage of merging.

4. EM for Mixture Models

MCLUST provides iterative EM or Expectation-Maximization methods (Dempster, Laird, and Rubin 1977; McLachlan and Krishnan 1997) for maximum likelihood clustering with parameterized Gaussian mixture models. EM iterates between an 'E-step', which computes a matrix $\mathbf{z}$ such that $\mathbf{z}_{ik}$ 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 $\mathbf{z}$. 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 sums of the columns of $\mathbf{z}$ converge to n times the mixing proportions τ_k, where n is the number of observations in the data. Here G is the number of groups in the data, which is assumed to be known for the purposes of the EM algorithm. The parameterizations of Σ_k currently available for EM in

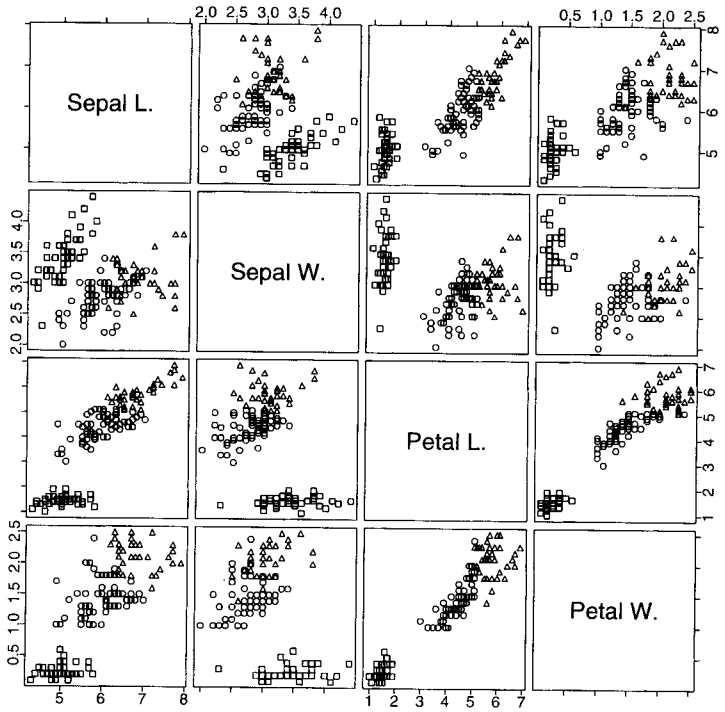

Figure 1. Pairs plot created with the function **clpairs** showing the 3-cluster classification of Fisher's iris data. This classification was produced by agglomerative hierarchical clustering using the criterion for a nonuniform spherical Gaussian model (VI).

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 iteration for these models.

MCLUST provides functions **me** (iterated M-step followed by E-step), **estep** and **mstep**, implementing the EM algorithm for the parameterized Gaussian mixtures. Given the data, an initial estimate of **z**, and the model specification, **me** produces the value of **z** associated with maximum likelihood parameters. An initial estimate for **z** can be obtained from a discrete classification, resulting in a matrix that has only 0,1 entries with exactly one 1 per row. For example, **me** can be started with a classification produced by **mhtree**:

```
> cltree <- mhtree( iris.matrix, modelid = "VVV")      # unconstrained model
> cl <- mhclass(cltree, 3)                             # 3-group mhtree classification
> z <- me( iris.matrix, modelid = "VVV", ctoz(cl))     # optimal z
```

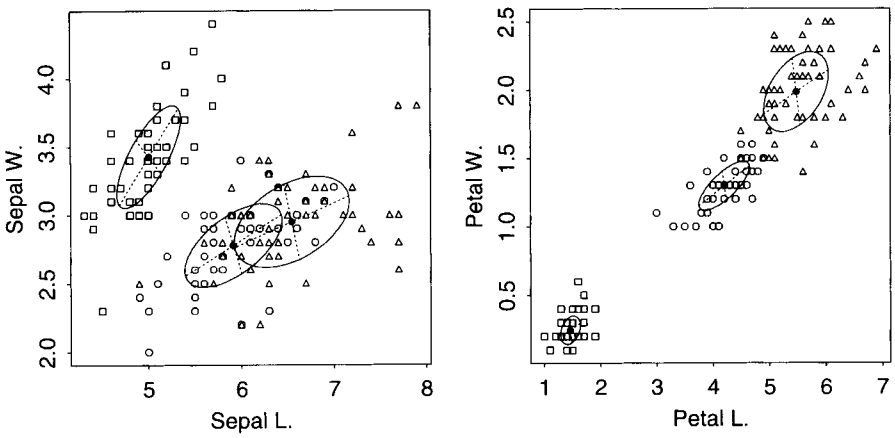

Figure 2. Plots created with the function **mixproj** showing the 3-cluster classification from EM for Fisher's iris data using an unconstrained Gaussian model (VVV).

The function **ctoz** converts a discrete classification into the corresponding **z** matrix. In general, the models used in **mhtree** and **me** need not be the same. It may in some cases be desirable to use one of the faster methods in **mhtree** (e.g. spherical or unconstrained models), followed by specification of a more complex model for EM.

Maximum likelihood parameters can be recovered from the **z** produced by **me** by means of the function **mstep**:

```
> pars <- mstep( iris.matrix, modelid = "VVV", z)
```

Once values of the parameters are available, projections of the data showing the means and standard deviation of the corresponding clusters (and optionally the partition or classification) may be plotted using the function **mixproj**:

```
> mixproj(iris.matrix, ms=pars, partition=ztoc(z), dimens = c(1,2))
> mixproj(iris.matrix, ms=pars, partition=ztoc(z), dimens = c(3,4))
```

The function **ztoc** coverts **z** to its nearest discrete classification, assigning each observation to the group represented by the column in which the **z** value for that observation is maximized. The resulting plots are displayed in Figure 2.

The EM iteration can be started with parameter estimates rather than an estimate of **z**. These can be supplied as starting values to **estep**, whose result in turn can be used to start **me**.

5. Bayesian Information Criterion

MCLUST provides a function **bic** to compute the Bayesian Information Criterion or BIC (Schwarz 1978) given the data and a model along with conditional probability estimates. This 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. A standard convention for calibrating BIC differences is that differences of less than 2 correspond to weak evidence, differences between 2 and 6 to positive evidence, differences between 6 and 10 to strong evidence, and differences greater than 10 to very strong evidence (Jeffreys 1961; Kass and Raftery 1995).

6. Cluster Analysis

MCLUST provides two functions, **emclust** and **emclust1**, for cluster analysis with BIC. Both initialize EM using hierarchical clustering for various parameterizations of the Gaussian model. The input to **emclust** is the data, the desired numbers of groups, and a list of models to apply in the EM phase (initialized with hierarchical clustering using the unconstrained model). It returns the 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. The following is an example of the use of **emclust** with Fisher's iris data:

```
> bicvals <- emclust( iris.matrix, nclus = 1:6, modelid = c("VVV","EEV","VEV"))
> bicvals
```

BIC:

	1	2	3	4	5	6
VVV	-829.9782	-574.0178	-580.8389	-628.9564	-683.8114	-711.5657
EEV	-829.9782	-644.5997	-610.0836	-645.9950	-621.6901	-669.7069
VEV	-829.9782	-561.7285	-562.5507	-589.3510	-635.2051	-681.2976

sample noise equal

```
    F       F       F
> plot(bicvals)
```

The BIC values for this example are shown in Figure 3.

Application of the **summary** function to the result reveals further information:

```
> sumry <- summary(bicvals, iris.matrix)          # summary object for emclust()
> sumry
```

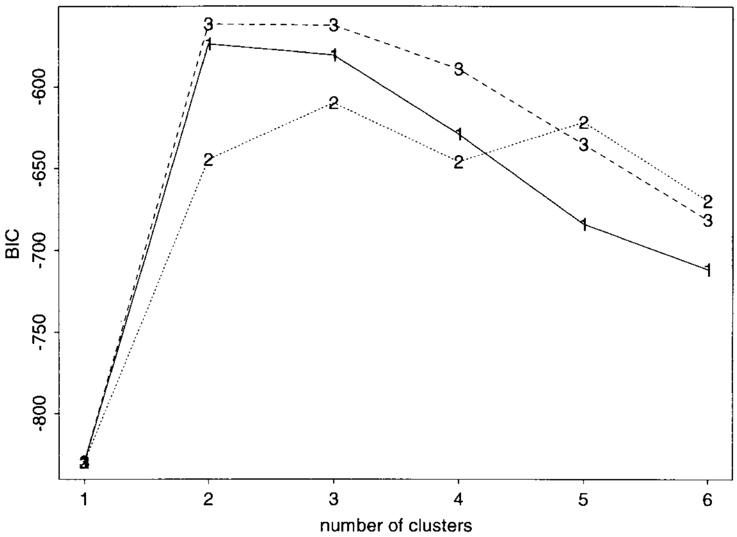

Figure 3. BIC values from **emclust** for the models 1 - VVV, 2 - EEV, and 3 - VEV with up to six clusters applied to Fisher's iris data.

best classification:

[1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
[38] 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[75] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[149] 2 2

uncertainty (quantiles):

0%	25%	50%	75%	100%
0	0	0	8.917506e-12	0.0002025599

best BIC values:

VEV,2	VEV,3	VVV,2
-561.7285	-562.5507	-574.0178

best model: uniform shape

sample noise equal

F F F

The best model among those fitted by **emclust** is the uniform shape model **VEV**, with 2 clusters. The same 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 EM result separates the first species from the other two, while the 3 cluster result nearly separates the three species (there are 5 misclassifications out of 150).

The function **emclust1** is similar to **emclust**, except that in addition to the data and the desired numbers of groups, it takes as input a pair of models, the first to be used in the initial hierarchical clustering phase, and the second to be used in the EM phase. For large data sets, there is the option of using only a subset of the data in the initial hierarchical clustering phase in **emclust/emclust1**. Because of the size of the objects involved, optimal parameter and **z** values are not available from **emclust/emclust1**. Instead, they can be obtained through **summary** functions, which have arguments allowing the summarizing information to be restricted to a subset of the number of clusters (and models, in the case of **emclust**).

For a complete analysis, it may be desirable to try various models, initialization strategies for EM, permutations or subsets of the observations, and/or to perturb 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 because of highly correlated variables.

References

BANFIELD, J. D., and RAFTERY, A. E. (1993), "Model-Based Gaussian and Non-Gaussian Clustering," *Biometrics, 49*, 803-821.

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.

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," *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.

FRALEY, C., and RAFTERY, A. E. (1998), "How Many Clusters? Which Clustering Method? — Answers via Model-Based Cluster Analysis," *Computer Journal, 41*, 578-588.

FRIEDMAN, H. P., and RUBIN, J. (1967), "On Some Invariant Criteria for Grouping Data," *Journal of the American Statistical Association, 62*, 1159-1178.

JEFFREYS, H. (1961), *Theory of Probability* (3rd ed.), Oxford: Clarendon Press.

KASS, R. E., and Raftery, A. E. (1995), "Bayes Factors," *Journal of the American Statistical Association, 90*, 773-795.

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

MUKERJEE, S., FEIGELSON, E. D., BABU, G. J., MURTAGH, F., FRALEY, C., and RAFTERY, A. E. (1998), "Three Types of Gamma Ray Bursts," *Astrophysical Journal, 508*, 314-327.

MURTAGH, F., and RAFTERY, A. E. (1984), "Fitting Straight Lines to Point Patterns," *Pattern Recognition, 17*, 479-483.

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. JR. (1963), "Hierarchical Groupings to Optimize an Objective Function," *Journal of the American Statistical Association, 58*, 234-244.