A Comparison of Model-Based and Fuzzy Clustering Methods

Un Confronto tra Metodi di Clustering di tipo Model-Based e Fuzzy

Marco Alfó, Maria Brigida Ferraro, Paolo Giordani, Luca Scrucca, and Alessio Serafini

Abstract Model-based and fuzzy clustering methods represent widely used approaches for soft clustering. In the former approach, it is assumed that the data are generated by a mixture of probability distributions where each component represents a different group or cluster. Each observation unit is ex-post assigned to a cluster using the so-called posterior probability of component membership. In the latter case, no probabilistic assumptions are made and each observation unit belongs to a cluster according to the so-called fuzzy membership degree. The aim of this work is to compare the performance of both approaches by means of a simulation study.

Abstract *I metodi basati sugli approcci model-based e fuzzy rappresentano i piú comuni approcci di soft clustering. Nel primo approccio si assume che i dati siano generati da una mistura di distribuzioni di probabilitá nella quale ciascuna componente individua un gruppo. Le osservazioni sono assegnate ai gruppi ex-post con le cosiddette probabilitá a posteriori (di appartenenza alle componenti). Nell'altro approccio, che non prevede alcuna assunzione probabilistica, le osservazioni ven-*

Marco Alfó
Department of Statistical Sciences, Sapienza University of Rome, P.le Aldo Moro, 5, 00185 Rome, e-mail: marco.alfo@uniroma1.it

Maria Brigida Ferraro
Department of Statistical Sciences, Sapienza University of Rome, P.le Aldo Moro, 5, 00185 Rome, e-mail: mariabrigida.ferraro@uniroma1.it

Paolo Giordani
Department of Statistical Sciences, Sapienza University of Rome, P.le Aldo Moro, 5, 00185 Rome, e-mail: paolo.giordani@uniroma1.it

Luca Scrucca
Department of Economics, Finance and Statistics, University of Perugia, Via A. Pascoli, 20, 06123 Perugia, e-mail: luca.scrucca@unipg.it

Alessio Serafini
Department of Statistical Sciences, Sapienza University of Rome, P.le Aldo Moro, 5, 00185, Rome e-mail: alessio.serafini@uniroma1.it

gono assegnate ai gruppi con i cosidetti gradi di appartenenza fuzzy. L'obiettivo di questo lavoro é confrontare le performance dei due approcci mediante uno studio di simulazione.

Key words: Cluster analysis, Model-based approach, Fuzzy approach

1 Introduction

In the last years, model-based and fuzzy clustering methods have received a great deal of attention. The two classes of methods are very different from a theoretical point of view. In the model-based framework, probabilistic assumptions are made. The data are generated by a mixture of known probability distributions (usually Gaussian). Each component of the mixture describes a cluster and, therefore, each cluster can be mathematically represented by a parametric distribution. In practice, the observation units are allocated to clusters via the so-called posterior probabilities of component membership and, for each observation unit, the cluster assignment is carried out by looking at the maximum posterior probability. In the fuzzy approach to clustering, the clusters are no longer represented in terms of parametric distributions. The observation units belong to the clusters according to the so-called membership degree, taking values in [0,1]. From a practical point of view, it is quite obvious that such two classes of clustering methods share similar features. Both of them produce a soft partition of the observation units and the posterior probability of component membership may play a role similar to the membership degree. Nevertheless, as far as we know, a thorough comparison between model-based and fuzzy clustering methods has never been carried out except for a few limited cases (see, e.g., [6, 10]). The aim of this work is to fill this gap by comparing the performances of four clustering methods, two from each class, in a simulation experiment.

2 Model-based clustering

Model-based clustering is a popular family of unsupervised learning methods for data classification. Such methods assume that the data are generated by a statistical model and try to recover it from the data. Let $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n) \in \mathbb{R}^p$ be a random sample of *i.i.d.* observations, where p denotes the number of variables. The random vector $\mathbf{x}_i$ is assumed to arise from a finite mixture of probability density functions:

$$f(\mathbf{x}_i;\boldsymbol{\theta}) = \sum_{k=1}^{g} \pi_k f(\mathbf{x}_i|\boldsymbol{\theta}_k), \tag{1}$$

where π_k, $k = 1, \ldots, g$, such that $\pi_k > 0$ and $\sum_{k=1}^{g} \pi_k = 1$, are the mixing proportions, g is the number of components, $f(\mathbf{x}|\boldsymbol{\theta}_k)$ is the component density $(k = 1, \ldots, g)$

and $\boldsymbol{\Phi} = (\pi_1, \pi_2, \ldots, \pi_g, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_g)$ is the parameter vector [20]. Each mixture component density belongs to a specific parametric class and represents a group or cluster. Even thought it is not necessary that each mixture component density arises from the same parametric distribution family, we will focus only on the case where the parametric distribution family is the same for each mixture component.

Maximum likelihood estimation of model parameters in $\boldsymbol{\Phi}$ is done by applying the Expectation-Maximization (EM) algorithm [11]. It is an iterative procedure to estimate the parameters of a finite mixture model by maximizing the expected value of the complete data log-likelihood by alternating two different steps. The Expectation step (E-step) computes the expected value of the complete data log-likelihood, and the Maximization step (M-step) maximises the expected value previously computed with respect to $\boldsymbol{\Phi}$. The log-likelihood can be derived as follows:

$$\ell(\boldsymbol{\Phi}) = \log \prod_{i=1}^{n} \sum_{k=1}^{g} \pi_k f(\mathbf{x}_i|\boldsymbol{\theta}_k) = \sum_{i=1}^{n} \log \sum_{k=1}^{g} \pi_k f(\mathbf{x}_i|\boldsymbol{\theta}_k). \tag{2}$$

Suppose to have an unobservable process $\mathbf{z} = (\mathbf{z}_1, \ldots, \mathbf{z}_n)$. We refer to this new dataset $(\mathbf{x}, \mathbf{z})$ as the complete data with density $f(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})$. The complete data log-likelihood is given by

$$\ell_c(\boldsymbol{\Phi}) = \log \prod_{i=1}^{n} f(\mathbf{x}_i, \mathbf{z}_i|\boldsymbol{\theta}) = \sum_{i=1}^{n} \log\{f(\mathbf{z}_i|\boldsymbol{\theta}) f(\mathbf{x}_i|\mathbf{z}_i, \boldsymbol{\theta})\}. \tag{3}$$

The E-step computes the expected value of the complete-data log-likelihood in (3) with respect to the missing part:

$$Q(\boldsymbol{\Phi}|\boldsymbol{\Phi}^t) = \mathbb{E}_{\boldsymbol{\Phi}^t}[\ell_c(\boldsymbol{\Phi})] = \mathbb{E}_{\boldsymbol{\Phi}^t}\left[\sum_{i=1}^{n} \log f(\mathbf{z}_i|\boldsymbol{\theta}^t) f(\mathbf{x}_i|\mathbf{z}_i, \boldsymbol{\theta}^t)\right]. \tag{4}$$

The M-step maximises equation in (4), such that:

$$\boldsymbol{\Phi}^{t+1} = \arg\max_{\boldsymbol{\Phi}} Q(\boldsymbol{\Phi}|\boldsymbol{\Phi}^t). \tag{5}$$

The procedure is iterated until some convergence criterion is satisfied. The EM algorithm guaranteesthat the observed log-likelihood is nondecreasing and, under fairly general conditions, the sequence converges to at least a local maximum [19]. Further details can be found in, e.g., [19, 20].

2.1 Finite mixtures of Gaussian densities

Due to its flexibility and mathematical tractability, the most popular model for clustering postulates that the data follow a Gaussian mixture distribution, i.e. $f(\mathbf{x}_i, |z_{ik} = 1, \boldsymbol{\theta}_k) \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ [15]. The finite mixture of Gaussian densities is then given by

$$f(\mathbf{x}_i;\boldsymbol{\theta}) = \sum_{k=1}^{g} \pi_k \phi(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \tag{6}$$

where $\boldsymbol{\theta} = \{\pi_1, \pi_2, \dots, \pi_{k-1}, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_k\}$ denotes the parameter set for the finite mixture model and $\phi(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ the underlying component-specific density function with parameters $\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$, $k = 1, \dots, g$.

Thus, model in (6) generates ellipsoidal clusters centred at the mean vector $\boldsymbol{\mu}_k$, with $\boldsymbol{\Sigma}_k$ controlling the other geometrical properties of each cluster. Parsimonious parametrisations of the cluster covariance matrices can be obtained through the eigendecomposition $\boldsymbol{\Sigma}_k = \lambda_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k^\top$ [4, 9], where λ_k is a scalar controlling the volume of the ellipsoid, $\mathbf{A}_k$ is a diagonal matrix controlling its shape and $\mathbf{D}_k$ is an orthogonal matrix controlling the orientation of the ellipsoid. Such an eigendecomposition generates a class of models with different geometrical properties. For some covariance parametrisations a closed formula for the M-step in the EM algorithm can be obtained [9]. The estimation for each of the 14 different models resulting from the eigendecomposition of the within clusters covariance matrices is implemented in the R package **mclust** [23].

The number of clusters and the parametrisation of the covariance matrices are selected using selection criteria, such as the Bayesian information criterion (*BIC*) [14, 22].

2.2 Finite mixtures of t densities

In [21], a heavy-tailed alternative to the component-specific density (6) is proposed by replacing the Gaussian distribution with a *t* distribution as follows:

$$f(\mathbf{x}_i;\boldsymbol{\theta}) = \sum_{k=1}^{g} \pi_k f_t(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k), \tag{7}$$

where $\boldsymbol{\theta} = \{\pi_1, \pi_2, \dots, \pi_{k-1}, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_k, \nu_1, \dots, \nu_k\}$ and $f_t(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k)$ is the multivariate *t* distribution with mean $\boldsymbol{\mu}_k$, covariance matrix $\boldsymbol{\Sigma}_k$ and ν_k degrees of freedom. As in the Gaussian case, the EM algorithm is employed for maximum likelihood estimation of $\boldsymbol{\theta}$ [2, 21]. In the present context, a parsimonious parametrization can be based on the same eigendecomposition of the covariance matrices as in the Gaussian case and constraining the degrees of freedom to be equal or not across groups [1]. This produces a class of finite mixture models with *t*-distributed components called *tEIGEN*. The *tEIGEN* family is implemented in the R package **teigen** [1, 2].

3 Fuzzy clustering

As opposed to the model-based framework, no probabilistic assumptions are made in the fuzzy approach to clustering. The complexity of the clustering process is managed in terms of fuzziness [24]. The observation units are assigned to the clusters according to the so-called fuzzy membership degree, taking values in [0,1]. This is inversely related to the dissimilarity between the observation units and the cluster prototype. A membership degree approaching 1 implies that the observation unit is close to the corresponding prototype and therefore it can be clearly assigned to the cluster. In the literature, there exist several fuzzy clustering methods. Among them, the most common one is the Fuzzy k-Means (FkM) algorithm [5]. In the following subsections, we will recall the FkM algorithm and the closely related Gustafson-Kessel variant [12]. The algorithms are implemented in the R package **fclust** [13].

3.1 Fuzzy k-Means

The Fuzzy k-Means (FkM) clustering algorithm [5] aims at grouping n observation units in k clusters by solving the following constrained optimization problem:

$$\begin{array}{ll} \min\limits_{\mathbf{U},\mathbf{H}} J_{FkM} & = \sum_{i=1}^{n} \sum_{k=1}^{g} u_{ik}^{m} d^2\left(\mathbf{x}_i, \mathbf{h}_k\right), \\ \text{s.t.} & u_{ik} \geq 0, \quad i=1,\ldots,n, \quad k=1,\ldots,g, \\ & \sum_{k=1}^{g} u_{ik} = 1, \quad i=1,\ldots,n. \end{array} \tag{8}$$

In (8), the term u_{ik} denotes the membership degree of observation unit i to cluster k, as a generic element of the matrix $\mathbf{U}$ of order $(n \times g)$. The row-wise sum of $\mathbf{U}$ is equal to 1. Furthermore, $\mathbf{h}_k = \left[h_{k1}, \ldots, h_{kp}\right]$, the k-th row of the prototype matrix $\mathbf{H}$ of order $(g \times p)$, is the prototype for cluster k, $k = 1, \ldots, g$. Finally, $d^2\left(\mathbf{x}_i, \mathbf{h}_k\right)$ is the squared Euclidean distance between observation unit i and prototype k, while $m > 1$ is the fuzziness parameter which tunes the level of fuzziness of the obtained partition. The higher the values of m, the fuzzier the partition with membership degrees tending to $\frac{1}{k}$. When m is close to 1, the FkM solution approaches that of the standard (non-fuzzy or hard) k-means [18] with membership degrees equal to either 0 or 1. The standard choice is $m = 2$.

The solution of (8) is carried out through an iterative optimization algorithm by updating the elements of $\mathbf{U}$ as follows

$$u_{ik} = \frac{1}{\sum_{k'=1}^{g} \left(\frac{d^2(\mathbf{x}_i, \mathbf{h}_k)}{d^2\left(\mathbf{x}_i, \mathbf{h}_{k'}\right)} \right)^{\frac{1}{m-1}}}, \quad i=1,\ldots,n, \quad k=1,\ldots,g, \tag{9}$$

and the rows of $\mathbf{H}$ as

$$\mathbf{h}_k = \frac{\sum_{i=1}^{n} u_{ik}^{m} \mathbf{x}_i}{\sum_{i=1}^{n} u_{ik}^{m}}, \quad k=1,\ldots,g. \tag{10}$$

In order to select the optimal number of clusters, several cluster validity indexes can be adopted. A popular choice is the Fuzzy Silhouette index [8].

3.2 Gustafson-Kessel variant of F*k*M

The major limitation of the F*k*M algorithm is that the obtained clusters are defined to be spherical. Therefore, F*k*M may be inadequate whenever the clusters have different geometrical shapes. In such cases, the so-called Gustafson-Kessel variant of the F*k*M algorithm can be applied, hereinafter GK-F*k*M [12]. The main difference between F*k*M and GK-F*k*M is that, in the latter, a Mahalanobis-type dissimilarity is considered, that is, $d^2(\mathbf{x}_i, \mathbf{h}_k)$ is replaced by

$$d_M^2(\mathbf{x}_i, \mathbf{h}_k) = (\mathbf{x}_i - \mathbf{h}_k)^\top \mathbf{M}_k (\mathbf{x}_i - \mathbf{h}_k), \tag{11}$$

with $\mathbf{M}_k$ symmetric and positive definite. The GK-F*k*M can then be formulated as

$$\begin{array}{lll} \min\limits_{\mathbf{U},\mathbf{H},\mathbf{M}_1,\ldots,\mathbf{M}_g} J_{GK-FkM} & = \sum_{i=1}^n \sum_{k=1}^g u_{ik}^m d_M^2(\mathbf{x}_i, \mathbf{h}_k), \\ \text{s.t.} & u_{ik} \geq 0, \quad i = 1, \ldots, n, \quad k = 1, \ldots, g, \\ & \sum_{k=1}^g u_{ik} = 1, \quad i = 1, \ldots, n, \\ & |\mathbf{M}_k| = \rho_k > 0 \quad k = 1, \ldots, g. \end{array} \tag{12}$$

As the cost function is linear with respect to the matrices $\mathbf{M}_k$, a trivial solution with $\mathbf{M}_k = \mathbf{0}, k = 1, \ldots, g$ would be obtained. To avoid it, $\mathbf{M}_k$ must be constrained. A way to do it is to consider volume constraints such that the determinant of $\mathbf{M}_k$ is positive. Note that the most common choice is $\rho_k = 1$, $k = 1, \ldots, g$.

An iterative solution of (12) can be found by updating $\mathbf{U}$ and $\mathbf{H}$ according to (9) and (10) provided that d^2 is replaced by d_M^2 and $\mathbf{M}_k$ by

$$\mathbf{M}_k = [\rho_k |\mathbf{V}_k|]^{\frac{1}{n}} \mathbf{V}_k^{-1}, \quad k = 1, \ldots, g, \tag{13}$$

where $\mathbf{V}_k$ is the fuzzy covariance matrix for cluster k given by

$$\mathbf{V}_k = \frac{\sum_{i=1}^n u_{ik}^m (\mathbf{x}_i - \mathbf{h}_k)(\mathbf{x}_i - \mathbf{h}_k)^\top}{\sum_{i=1}^n u_{ik}^m}, \quad k = 1, \ldots, g. \tag{14}$$

To avoid possible numerical problems for updating $\mathbf{M}_k$, a computational improvement has been proposed in [3] where the condition number of $\mathbf{M}_k$ is constrained to be higher than a prespecified threshold. Note that this condition is similar to that imposed to covariance matrices in finite mixture models with either Gaussian or t components [16, 17].

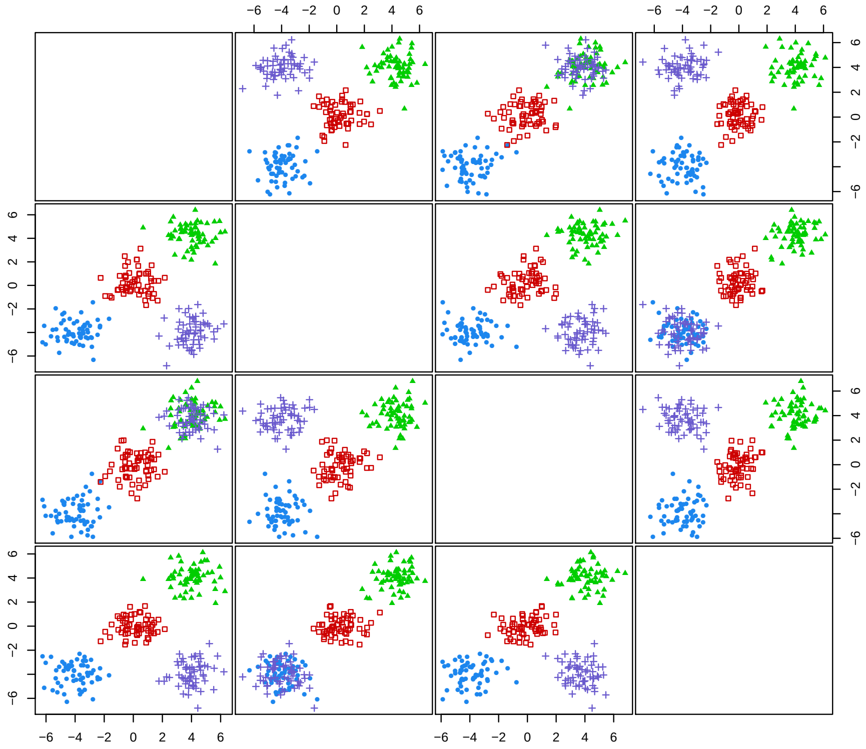

Fig. 1 Example of simulated data

4 Simulation study

A simulation study has been carried out to compare the previously described clustering methods. Simulated data sets have been generated randomly in a full factorial design; an example of generated data is displayed in Figure 1. In the simulation study, the focus lies on assessing the performance of the methods in recovering the cluster structure and checking whether the design variables influence the differential performance of the methods. The results will be presented at the meeting.

References

1. Andrews, J.L., McNicholas, P.D.: Model-based clustering, classification, and discriminant analysis via mixtures of multivariate t-distributions. Stat. Comput. **22**, 1021–1029 (2012)
2. Andrews, J.L., McNicholas, P.D.: teigen: Model-based clustering and classification with the multivariate t-distribution, R package version 2 (2015)

3. Babuška, R., van der Veen, P.J., Kaymak, U.: Improved covariance estimation for Gustafson-Kessel clustering. In: IEEE International Conference on Fuzzy Systems, pp. 1081–1085 (2002)
4. Banfield, J.D., Raftery, A.E.: Model-based gaussian and non-gaussian clustering. Biometrics **49**, 803–821 (1993)
5. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithm. Plenum Press, New York (1981)
6. Bezdek, J.C., Hathaway, R.J., Huggins, V.J.: Parameter estimation for normal mixtures. Pattern Recognit. Lett. **3**, 79–84 (1985)
7. Biernacki, C., Celeux, G., Govaert, G.: Assessing a mixture model for clustering with the integrated completed likelihood. IEEE Trans. Pattern Anal. Mach. Intell. **22**, 719–725 (2000)
8. Campello, R.J.G.B., Hruschka, E.R.: A fuzzy extension of the silhouette width criterion for cluster analysis, Fuzzy Sets Syst. **157**, 2858–2875 (2006)
9. Celeux, G., Govaert, G.: Gaussian parsimonious clustering models. Pattern Recognit. **28**, 781–793 (1995)
10. Davenport, J.W., Bezdek, J.C., Hathaway, R.J.: Parameter estimation for finite mixture distributions. Comput. Math. Applic. **15**, 819–828 (1988)
11. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. Royal Stat. Soc. Series B **39**, 1–38 (1977)
12. Gustafson, E., Kessel, W.: Fuzzy clustering with a fuzzy covariance matrix. In: IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes, pp. 761–766 (1978)
13. Ferraro, M.B., Giordani, P.: A toolbox for fuzzy clustering using the R programming language. Fuzzy Sets Syst. **279**, 1–16 (2015)
14. Fraley, C., Raftery, A.E.: How many clusters? Which clustering method? Answers via model-based cluster analysis. Comput. J. **41**, 578–588 (1998)
15. Fraley, C., Raftery, A.E.: Model-based clustering, discriminant analysis, and density estimation. J. Am. Stat. Assoc. **97**, 611–631 (2002)
16. Greselin, F., Ingrassia, S.: Constrained monotone EM algorithms for mixtures of multivariate *t* distributions. Stat. Comput. 20, 9–22 (2010)
17. Ingrassia, S., Rocci, R.: Constrained monotone EM algorithms for finite mixture of multivariate Gaussians. Comput. Stat. Data Anal. 51, 5339–5351 (2007)
18. Mac Queen, J.B.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, pp. 281–297 (1967)
19. McLachlan, G., Krishnan, T.: The EM algorithm and extensions. Wiley, New York (2008)
20. McLachlan, G., Peel, D.: Finite mixture models. Wiley, New York (2000)
21. Peel, D., McLachlan, G.: Robust mixture modelling using the t distribution. Stat. Comput. **10**, 339–348 (2000)
22. Schwarz, G.: Estimating the dimension of a model. Ann. Stat. **6** 461–464 (1978)
23. Scrucca, L., Fop, M., Murphy, T.B., Raftery, A.E.: mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. R J. **8** 289–317 (2016)
24. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)