Adv Data Anal Classif (2019) 13:379–404
https://doi.org/10.1007/s11634-018-0308-3

REGULAR ARTICLE

Mixture model modal clustering

José E. Chacón[1]

Received: 2 January 2017 / Revised: 3 January 2018 / Accepted: 5 January 2018 /
Published online: 13 January 2018

Abstract The two most extended density-based approaches to clustering are surely mixture model clustering and modal clustering. In the mixture model approach, the density is represented as a mixture and clusters are associated to the different mixture components. In modal clustering, clusters are understood as regions of high density separated from each other by zones of lower density, so that they are closely related to certain regions around the density modes. If the true density is indeed in the assumed class of mixture densities, then mixture model clustering allows to scrutinize more subtle situations than modal clustering. However, when mixture modeling is used in a nonparametric way, taking advantage of the denseness of the sieve of mixture densities to approximate any density, then the correspondence between clusters and mixture components may become questionable. In this paper we introduce two methods to adopt a modal clustering point of view after a mixture model fit. Examples are provided to illustrate that mixture modeling can also be used for clustering in a nonparametric sense, as long as clusters are understood as the domains of attraction of the density modes. Finally, a simulation study reveals that the new methods are extremely efficient from a computational point of view, while at the same time they retain a high level of accuracy.

Keywords Mixture modeling · Modal clustering · Component merging · Mean shift algorithm

Mathematics Subject Classification Main 62H30; Secondary 68T10 · 91C20

✉ José E. Chacón
jechacon@unex.es

[1] Departamento de Matematicas, Universidad de Extremadura, 06006 Badajoz, Spain

1 Introduction

Classical clustering algorithms are mainly based on inter-point distances (e.g., hierarchical clustering) or on partitioning the space around a pre-fixed number of central points (these are usually called partitioning methods, and include K-means clustering, for instance). In the recent times, however, there is a growing body of researchers that advocate that "density needs to be incorporated in the clustering procedures" (Carlsson and Mémoli 2013).

Two very different density-based approaches to clustering are mixture model clustering (McLachlan and Basford 1988) and clustering based on high density regions (Hartigan 1975). The former, in a parametric context, starts by modeling the distribution density f as a mixture of densities in a pre-specified parametric family, that is, $f(x) = \sum_{g=1}^{G} \pi_g f_g(x)$ where the mixing weights $\pi_g > 0$ are such that $\sum_{g=1}^{G} \pi_g = 1$ and the density components $f_1, \ldots, f_G$ can be written as $f_0(x|\theta_1), \ldots, f_0(x|\theta_G)$ for a fixed parametric distribution $f_0(x|\theta)$ and different parameter values $\theta_1, \ldots, \theta_G$. If this mixture model is identifiable, it seems natural to associate different clusters to each of the distribution components. In practice, a density estimate $\widehat{f}$ within this model is obtained by estimating the parameters and mixing weights by maximum likelihood, and selecting the number of components using the Bayesian information criterion (BIC) (details can be found in Fraley and Raftery 2002), leading to $\widehat{f}(x) = \sum_{g=1}^{\widehat{G}} \widehat{\pi}_g f_0(x|\widehat{\theta}_g)$. Then, through Bayes theorem, any point x in the space can be assigned to the component that makes it more probable by looking at the value of $g \in \{1, \ldots, \widehat{G}\}$ that maximizes $\widehat{\pi}_g f_0(x|\widehat{\theta}_g)$.

The principles of clustering based on high density regions are quite different. In this context, clusters are understood as regions of tight concentration of probability mass, separated by each other by regions where the probability mass is more dispersed. There are two ways to formalize this. Hartigan (1975) proposed to focus on the region where the density is above some pre-specified level (density level sets) and defined clusters as the connected components of this region. This clearly captures the notion of a high density region (the density must be above some level) separated by regions of lower density (this happens where the region consists of more than one connected component). The main disadvantages of this definition are: first, since it concerns the region where the density is above some level, it may leave a substantial number of points with no cluster assigned; second, the whole cluster structure of the distribution may not be noticeable at a fixed, single level (see, for instance, Figure 1 in Rinaldo et al. 2012); and third, the obtention of the connected components of a density level set is not a computationally easy task (see Cuevas et al. 2001, or Azzalini and Torelli 2007). The first and second issues can be amended by considering the cluster tree (Stuetzle 2003), which represents how the density level clusters evolve as the level varies (in a close connection to persistent homology techniques, see Edelsbrunner and Harer 2008). This solution, unfortunately, emphasizes the aforementioned computational issue, since the construction of the cluster tree involves computing the connected components of not just one, but several density level sets.

An alternative formalization of high density clusters is through Morse theory tools. Clusters are defined as the domains of attraction of the density modes; i.e., a cluster

is made of all the points that are eventually taken to a given local maximum of the density, when moved through the flow line defined by the density gradient field (Chacón 2015). Having the clusters closely connected to the density modes, this approach is commonly known as modal clustering. It is related to, but different from the notion based on the connected components of density level sets, and avoids the above noted drawbacks of the latter. This definition results in a partition of the whole space into clusters, provided the density is sufficiently regular, with the boundaries of the partition components made of density valleys (regions of lower density), it does not require the choice of a level parameter and it is computationally tractable though the adaptation of numerical optimization methods to this setting, such as the mean shift algorithm (Fukunaga and Hostetler 1975), a variant of the gradient ascent algorithm for function maximization. See also Schnell (1964) and Bock (1974, Chapter 6).

Any of the two high density clustering approaches is described above in population terms, that is, in terms of the true density f. If a d-variate sample $X_1, \ldots, X_n$ from f is given, then empirical, data-based clusters are obtained by replacing the unknown underlying density by a density estimate $\widehat{f}$. Since no parametric model is assumed in this setting, it is common to adopt a nonparametric viewpoint here and use a kernel density estimator $\widehat{f}_h(x) = n^{-1} \sum_{i=1}^{n} K_h(x - X_i)$, where the kernel K is a unimodal, radially symmetric density, the bandwidth h is a positive number and the notation $K_h(x) = K(x/h)/h^d$ represents the scaled kernel.

This paper explores a new methodology that arises as a blend of the two previous density-based clustering schemes. There are two possibilities for this blending. As a first option, noting that the kernel density estimator is a mixture density itself, one could try to apply mixture model clustering to such particular mixture density estimator. However, this makes little sense, since following the principles of mixture model clustering would lead us to declare that each of the n "mixture components" $K_h(x - X_i)$ forms a separate cluster, which besides contains a single data point (X_i). The second possibility, the other way round, involves applying the modal clustering methodology when the density estimate is obtained as the result of fitting a mixture model to the data, and indeed this makes perfect sense.

So, the main goal of this paper is to illustrate how the principles of modal clustering can be combined with mixture modeling. A related recent paper by Scrucca (2016) precisely shows how this can be done when high density clusters are understood as connected components of density level sets. Here, on the contrary, we focus on the notion of high density clusters as domains of attraction of the density modes, expanding on some ideas previously presented in Chacón (2012). Even if the two methodologies lead to very similar results in practice, the main advantage of the latter is that it is much simpler from a computational point of view, especially for high dimensional data.

The rest of the paper is organized as follows. In Sect. 2, both approaches, mixture model clustering and modal clustering, are compared to each other, and the pros and the cons of the two methodologies are exemplified through the analysis of synthetic and real data sets. In Sect. 3, two methods are introduced with the aim of producing a clustering from a modal point of view, but starting from a mixture model fit. Both methods rely on the use of the mean shift algorithm for normal mixture densities, and a new representation of this algorithm as a quasi-Newton optimization method is provided. Section 4 includes a simulation study comparing the new proposals with

the existing methodologies, both in terms of accuracy and computational efficiency, and two detailed real data examples to explore these different approaches in practice. The paper finishes with a discussion section, posing some related open problems for future research.

2 Mixture model clustering versus modal clustering

In principle, mixture model clustering and modal clustering aim at very different goals. It is not that one of these views is right and the other one is wrong. They just seek after different notions of cluster. And in fact, the two clusterings look very similar in "non-problematic" situations, that is, when different mixture components correspond to different, well-separated unimodal distributions. As an example, Fig. 1 shows a bivariate trimodal 3-component normal mixture density, with the population clusters depicted according to mixture model clustering (left) and modal clustering (right).

Differences arise precisely when there is not a one-to-one correspondence between mixture components and density modes.

Consider the class $\mathcal{P}_G(f_0)$ of mixture distributions with G components based on a fixed parametric distribution $f_0(\cdot|\theta)$ and denote $\mathcal{P}_0 = \bigcup_{G\in\mathbb{N}} P_G(f_0)$. If the true distribution of the data is indeed in $\mathcal{P}_0$, then mixture model clustering allows to distinguish more subtle situations than modal clustering. For instance, it notices when there are two populations with the same center but different dispersion, or even different centers, but close enough to result in a unimodal distribution. The left and right panels of Fig. 2, respectively, represent these two scenarios. Modal clustering, on the contrary, only observes the overall resulting density (not its components) and since the two components are not sufficiently separated, it cannot detect two separate groups and notices only one cluster in these distributions.

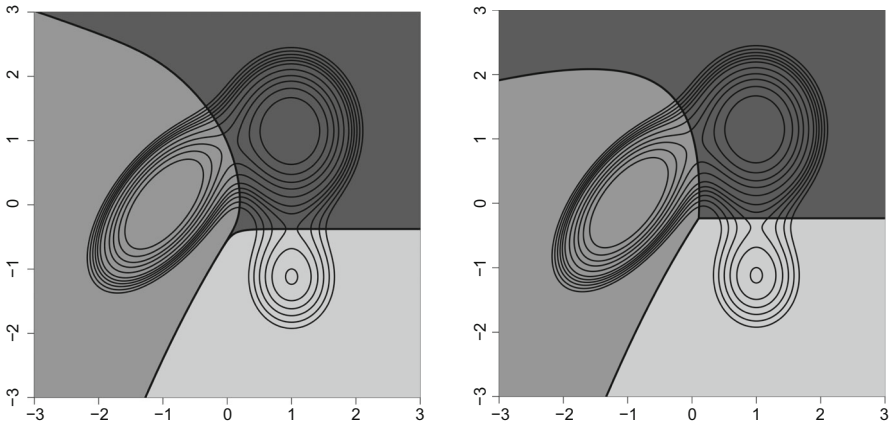

Fig. 1 Bivariate trimodal 3-component normal mixture density. Left, normal mixture model clustering. Right, modal clustering. The boundaries of the modal clusters are necessarily perpendicular to the contour lines, since they are flow lines of the gradient field

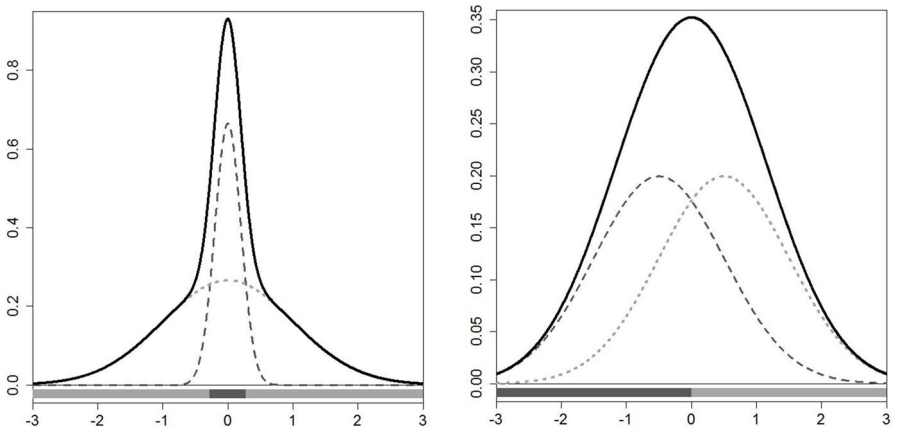

Fig. 2 Two examples of 2-component Gaussian mixture densities with only one mode. The thick black line is the mixture density, the dotted and dashed grey lines are the (weighted) density components, and the bottom line represents with the different grey levels the mixture model clusters defined by the density components

Nevertheless, since the class $\mathcal{P}_0$ is dense in the set of all density functions under the L_1 metric (see, e.g., Li and Barron 2000), in fact mixture modeling can be used in a nonparametric way to estimate any density, either belonging to $\mathcal{P}_0$ or not (Priebe 1994). However, even if mixture modeling is thus useful for nonparametric density estimation, mixture model clustering should be used with caution when the true distribution of the data does not belong to the class $\mathcal{P}_0$ employed to estimate the density, because in this case, it may be misleading to identify mixture components with clusters (Baudry et al. 2010; Hennig 2010).

For example, Fig. 3 shows a bivariate skew-normal distribution and a normal mixture density estimate for this skew-normal density based on a sample of size $n = 500$. The density estimate is reasonably close to the true density, but for that it is necessary to use a 3-component normal mixture, as suggested by the BIC. A blind application of mixture model clustering would result in a partition of the data into 3 clusters. But that seems quite artificial, since the same data are best fit using a single-component skew-normal mixture density (Lin 2009). In this case, adopting a modal clustering perspective after mixture-model density estimation yields one cluster, even with a mixture of normal components, since the obtained density estimate is unimodal.

Another classical real data example where a similar situation occurs is the Old Faithful data set (Azzalini and Bowman 1990), which records the variables "eruption time" and "waiting time" regarding $n = 272$ eruptions of the Old Faithful geyser in Yellowstone National Park. As noted in Scrucca (2016), normal mixture density estimation results in a 3-component mixture for this data set, albeit the density estimate only shows two separate regions of high density (see Fig. 4). Indeed if the more general class of skew-normal mixture densities is employed, then the best fit is obtained using only two components.

A solution frequently proposed in the literature to amend this problem is to "merge" some components into the same cluster. The exhaustive paper by Hennig (2010) pro-

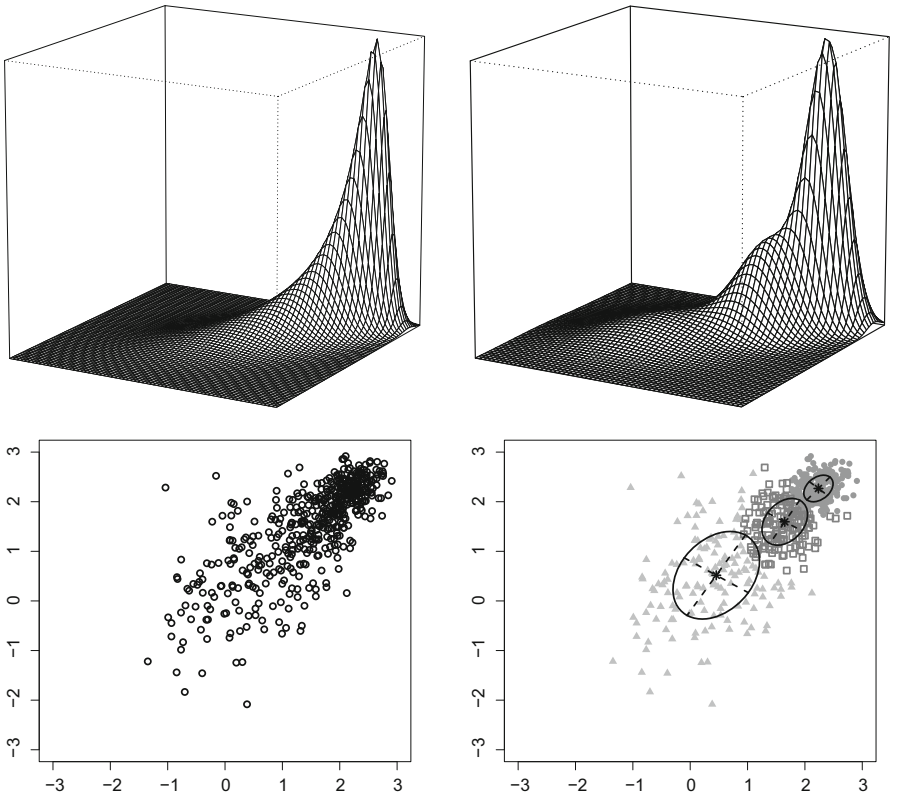

Fig. 3 Normal mixture model clustering applied to a skew-normal distribution. Top left: true distribution density. Bottom left: $n = 500$ data points from this distribution. Top right: normal mixture density estimate from these data. Bottom right: mixture model clustering for these data

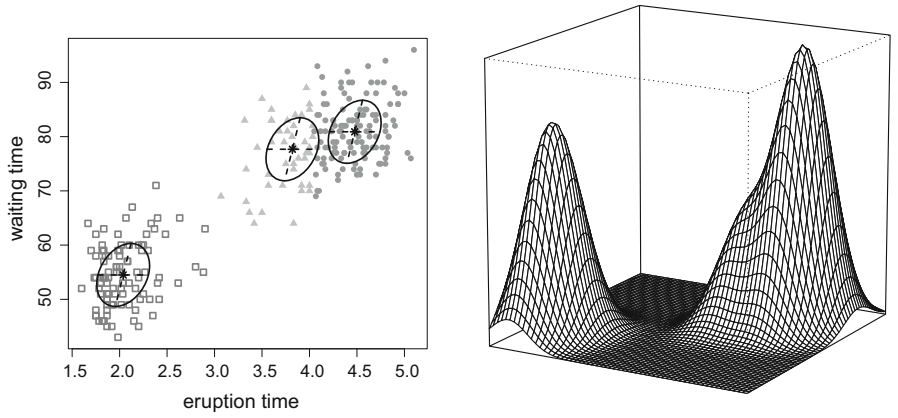

Fig. 4 Old Faithful data example. Left, data set and normal mixture clustering. Right, normal mixture density estimate

vides an excellent review and comparison of the existing techniques to date, and introduces some new ones based on modality arguments. One of the proposals examined in this paper (Method 1 below) is indeed a further addition to this list, with a view towards simplicity and computational easiness.

In all the previous examples the disagreement between mixture model clustering and modal clustering is mainly due to the fact that the number of components is greater than the number of modes. This seems to be the most common case in practice. However, it is worth mentioning that the opposite situation may also occur; that is, a 2-component mixture may have more than two modes. These examples, however, appear to be quite less frequent in practice, and to produce them a thorough search of the mixture parameters is needed. Ray and Lindsay (2005) showed a 2-component normal mixture density with three modes and Carreira-Perpiñán and Williams (2003b) found a 3-component isotropic normal mixture with four modes. For the latter, the exhibited phenomenon occurs only for a small range of values of the scale parameter of the isotropic components (see also Edelsbrunner et al. 2013).

For the class of normal mixture densities, the number of extra modes is quite controlled, since Carreira-Perpiñán and Williams (2003a) showed that for $d = 1$ the number of modes cannot be larger than the number of components, and for $d > 2$ Ray and Ren (2012) showed that a d-variate 2-component normal mixture can have at most $d + 1$ modes. However, for mixture densities based on a distribution different from the normal one, the situation can get much more complicated and more bizarre examples can be found, as shown in Walther (2003). For this reason, we concentrate on normal mixture densities henceforth.

3 Modal clustering after mixture modeling

As announced in the previous sections, the goal of this paper is to illustrate how modal clustering can be applied after mixture model density estimation. For the reasons explained above, the density estimation step will be performed within the class of normal mixture distributions. That is, as a first step, by applying the expectation maximization (EM) algorithm to find the maximum likelihood estimates of the parameters and mixing weights, and the BIC to select the number of components (see Fraley and Raftery 2002 fordetails), a density of the type $\widehat{f}(x) = \sum_{g=1}^{\widehat{G}} \widehat{\pi}_g \phi(x|\widehat{\mu}_g, \widehat{\Sigma}_g)$ is fitted to the data $X_1, \ldots, X_n$. Here,

$$\phi(x|\mu, \Sigma) = (2\pi)^{-d/2}|\Sigma|^{-1/2} \exp\left\{ -\tfrac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\right\}$$

is the density of the $N(\mu, \Sigma)$ distribution.

In a second step, since the density estimate $\widehat{f}$ is a normal mixture density, we can make use of specifically designed mode-finding algorithms to investigate its modality features (see Carreira-Perpiñán 2000). This is the main difference with the proposal in Scrucca (2016): while modal clustering based on connected components density level sets involve a high computational load, hindered by the need to compute the Delaunay triangulation of the data points, the mean-shift algorithm or one of its accelerated

variants (Carreira-Perpiñán 2006, 2007) provides an efficient tool to perform modal clustering analysis from a normal mixture density estimate, even in high dimensions.

Depending on how we use the mean shift algorithm, it leads to two different modal clustering methods after mixture model density estimation: on one hand, a method for merging components, quite naive but extremely fast even for in high dimensional data; and, on the other hand, a clustering method that does not necessarily coincide with a merging of the mixture components.

3.1 The non-isotropic mean shift algorithm as a quasi-Newton optimization method

First we derive the formulation of the mean shift algorithm for a normal mixture density, introduced in Carreira-Perpiñán (2000) as a fixed-point iterative scheme to numerically find the modes of a normal mixture density.

The gradient of a normal mixture density $f(x) = \sum_{g=1}^{G} \pi_g \phi(x|\mu_g, \Sigma_g)$ with respect to x is easily shown to be

$$\mathsf{D} f(x) = \sum_{g=1}^{G} \pi_g \phi(x|\mu_g, \Sigma_g) \Sigma_g^{-1} (\mu_g - x), \tag{1}$$

so to find the critical points of f we would solve $\mathsf{D} f(x) = 0$ for x to obtain $x = T(x)$ with $T : \mathbb{R}^d \to \mathbb{R}^d$ given by

$$T(x) = \left\{ \sum_{g=1}^{G} \pi_g \phi(x|\mu_g, \Sigma_g) \Sigma_g^{-1} \right\}^{-1} \sum_{g=1}^{G} \pi_g \phi(x|\mu_g, \Sigma_g) \Sigma_g^{-1} \mu_g. \tag{2}$$

Thus, from any initial point $y_0 \in \mathbb{R}^d$, a sequence $\{y_j\}_{j=1}^{\infty}$ to approximate the critical points numerically can be iteratively constructed by setting $y_{j+1} = T(y_j)$. This iterative procedure is known as the mean shift algorithm for normal mixture densities. Li et al. (2007) showed, under mild conditions, that the sequence $\{f(y_j)\}_{j=1}^{\infty}$ is non-decreasing and that $\{y_j\}_{j=1}^{\infty}$ is convergent for any initial y_0 (see also Aliyari Ghassabeh 2015 and references therein).

Nonetheless, the mean shift algorithm (Fukunaga and Hostetler 1975) was initially motivated as a gradient clustering algorithm, by application of a (normalized) gradient ascent maximization algorithm to an arbitrary initial point y_0, to obtain a sequence $y_{j+1} = y_j + a_j \mathsf{D} f(y_j)/f(y_j)$ for some step size $a_j > 0$. Arias-Castro et al. (2016) showed not only that the resulting mean shift sequence converges to a mode of f, but also that the polygonal line defined by linear interpolation of two consecutive steps provides a consistent estimator of the flow lines of the density gradient field.

Next we show that the previous iterative scheme $y_{j+1} = T(y_j)$ can also be obtained as a quasi-Newton optimization method of the form $y_{j+1} = y_j + B_j \mathsf{D} f(y_j)/f(y_j)$ for a positive definite matrix B_j, which makes it closer to the original mean shift idea. Starting from (1) and considering the weights $w_g(x) = \pi_g \phi(x|\mu_g, \Sigma_g)/f(x)$,

which are positive and add to one (in fact, $w_g(x)$ can be recognized as the a posteriori probability of the g-th mixture component, given x), reasoning as in Comaniciu (2003) it is clear that

$$\mathsf{D}f(x)/f(x) = \sum_{g=1}^{G} w_g(x)\Sigma_g^{-1}\mu_g - \overline{\Sigma}(x)^{-1}x,$$

where $\overline{\Sigma}(x) = \{\sum_{g=1}^{G} w_g(x)\Sigma_g^{-1}\}^{-1}$ is a weighted harmonic mean of the variance matrices of the normal mixture. Therefore, by taking $B_j = \overline{\Sigma}(y_j)$ it follows that $T(y_j) = y_j + B_j\mathsf{D}f(x)/f(x)$, and hence the iterative scheme $y_{j+1} = T(y_j)$ can also be seen as a quasi-Newton maximization algorithm.

In the isotropic case, multiplication by the matrix B_j is simplified to multiplication by a constant a_j, so the procedure has the form of a gradient ascent algorithm, as in the original formulation of the mean shift algorithm.

3.2 The two methods

After fitting a normal mixture density $\widehat{f}(x) = \sum_{g=1}^{\widehat{G}} \widehat{\pi}_g\phi(x|\widehat{\mu}_g, \widehat{\Sigma}_g)$ to the data, there are two possibilities to obtain a modal clustering:

3.2.1 Method 1: modal merging of mixture components

The $\widehat{G}$ whole-space clusters obtained from the normal mixture fit are $\widehat{\mathcal{C}}_1, \ldots, \widehat{\mathcal{C}}_{\widehat{G}}$, where

$$\widehat{\mathcal{C}}_g = \left\{x \in \mathbb{R}^d : \widehat{\pi}_g\phi(x|\widehat{\mu}_g, \widehat{\Sigma}_g) \geq \widehat{\pi}_j\phi(x|\widehat{\mu}_j, \widehat{\Sigma}_j),\ \forall j \neq g\right\} \tag{3}$$

for $g = 1, \ldots, \widehat{G}$. If the goal is just to cluster the data set $X_1, \ldots, X_n$, then each of this data points should be assigned to the cluster $\widehat{\mathcal{C}}_g$ where it belongs.

But, as it was illustrated in Sect. 2, it could happen that two or more of these clusters represent in fact a single unimodal distribution. So in that case, from a modal clustering point of view, it would be advisable to merge into a single cluster all the components that give rise to the same unimodal distribution. This is easy to do with the naked eye, after plotting the resulting density estimate, if the data dimension is one or two. But for higher dimensional data it is necessary to have an automated algorithm to do so.

In Hennig (2010), two methods are proposed to achieve this goal, based on the concept of ridgeline introduced in Ray and Lindsay (2005). They consist in examining if every possible pair of fitted mixture components leads to a unimodal distribution or not (in the second method, a tuning parameter is introduced to allow merging two components even if the resulting mixture is not unimodal, provided the valley in the ridgeline that identifies the two modes is not too deep). A disadvantage of this methodology, as noted by the author, is that since the comparisons are made two by two components, it induces a hierarchical merging that may cause trouble when the algorithm is re-run again to look for further mergings.

From a slightly different point of view, notice that if the mixture of two or more components result in a unimodal density region, then their means belong to the domain

of attraction of that same mode (because if one belonged to the domain of attraction of another mode, there should be a valley in the density separating it from the other components, and then their mixture could not be unimodal). Therefore, our proposal for merging based on modal clustering is to apply the mean shift method in Section 3.1 starting from each of the estimated component means $\widehat{\mu}_1, \ldots, \widehat{\mu}_{\widehat{G}}$, and merge all the components whose estimated means converge to the same mode of $\widehat{f}$. In contrast with the methods based on the ridgeline, here the merging process is not pairwise nor hierarchical; rather, all the components are dealt with at the same time. As a result we obtain a new clustering $\widetilde{\mathcal{C}}_1, \ldots, \widetilde{\mathcal{C}}_{\widehat{M}}$, where $\widehat{M}$ is the number of modes of $\widehat{f}$ and $\widetilde{\mathcal{C}}_m$ is made of the union of those clusters out of $\widehat{\mathcal{C}}_1, \ldots, \widehat{\mathcal{C}}_{\widehat{G}}$ whose component means converge to the m-th mode of $\widehat{f}$. The whole process is summarized in Algorithm 1.

Algorithm 1: Modal merging of mixture components (`modmerge`)

Input : data $X_1, \ldots, X_n$
Output: clustering $\widetilde{\mathcal{C}}_1, \ldots, \widetilde{\mathcal{C}}_{\widehat{M}}$, where $\widehat{M}$ is the number of estimated modes

1. Fit a normal mixture $\widehat{f}$ to the data
2. Find the mixture component clusters $\widehat{\mathcal{C}}_1, \ldots, \widehat{\mathcal{C}}_{\widehat{G}}$, defined as in (3)
3. Run the mean shift algorithm with all the estimated component means $\widehat{\mu}_1, \ldots, \widehat{\mu}_{\widehat{G}}$ as initial values
4. Build up $\widetilde{\mathcal{C}}_m$ as the union of those clusters out of $\widehat{\mathcal{C}}_1, \ldots, \widehat{\mathcal{C}}_{\widehat{G}}$ whose component means converge to the m-th mode of $\widehat{f}$, for $m = 1, \ldots, \widehat{M}$

The merging stage of the previous algorithm is simple and fast. Notice that once the mixture density is fitted, the mean shift algorithm only needs to be run with $\widehat{G}$ initial values, the estimated component means, which besides are typically not far from the modes (Ray and Lindsay 2005). As a consequence, convergence is guaranteed after a reasonably small number of iterations, even in high dimensions.

3.2.2 Method 2: modal clustering of the mixture density estimate

The second proposal to perform modal clustering after normal mixture modeling is not inspired by component merging. Instead, in a more straightforward manner, it consists in obtaining the clusters as the domains of attraction of the modes of the fitted normal mixture $\widehat{f}$. To be precise, consider the curve $\widehat{\gamma}_x : \mathbb{R} \to \mathbb{R}^d$ defined as the solution of the initial value problem

$$\widehat{\gamma}_x'(t) = \mathsf{D}\widehat{f}\big(\widehat{\gamma}_x(t)\big), \ \widehat{\gamma}_x(0) = x.$$

Then, as in Chacón (2015), define the whole-space clustering associated to $\widehat{f}$ as the partition of the space with clusters $\widetilde{\mathcal{D}}_1, \ldots, \widetilde{\mathcal{D}}_{\widehat{M}}$, given by

$$\widetilde{\mathcal{D}}_m = \Big\{x \in \mathbb{R}^d : \lim_{t\to\infty} \widehat{\gamma}_x(t) = \widehat{\tau}_m\Big\} \tag{4}$$

for $m = 1, \ldots, \widehat{M}$, where $\widehat{\tau}_1, \ldots, \widehat{\tau}_{\widehat{M}}$ are the modes of $\widehat{f}$. In practice, again we use the mean shift algorithm described in Sect. 3.1, and its consistency as an estimator of the gradient flow lines (Arias-Castro et al. 2016), to approximate $\lim_{t\to\infty} \widehat{\gamma}_x(t) \approx \lim_{j\to\infty} y_j$, where $y_{j+1} = \widehat{T}(y_j)$ with initial $y_0 = x$, and $\widehat{T}$ the same as in (2), but with the parameters replaced by their estimates. See Algorithm 2.

Algorithm 2: Modal clustering with a mixture density estimate (`modclust`)

Input : data $X_1, \ldots, X_n$
Output: clustering $\widetilde{\mathcal{D}}_1, \ldots, \widetilde{\mathcal{D}}_{\widehat{M}}$, where $\widehat{M}$ is the number of estimated modes

1. Fit a normal mixture $\widehat{f}$ to the data
2. Apply the iterative process $y_{j+1} = \widehat{T}(y_j)$ to all the initial values y_0 that need to be clustered to approximate the modal clusters $\widetilde{\mathcal{D}}_1, \ldots, \widetilde{\mathcal{D}}_{\widehat{M}}$ defined as in (4)

This second option is more meaningful from a modal clustering perspective, since instead of merging mixture component clusters, it is based on directly identifying the modal clusters of $\widehat{f}$. Yet, another difference is that it could have a higher computational cost. For instance, if the goal (as usual) is to cluster only the data $X_1, \ldots, X_n$, then the mean shift algorithm has to be run with all this data points in the role of the initial value y_0, whereas for Algorithm 1 the mean shift procedure is applied only to the $\widehat{G}$ estimated component means as initial values, and in practice we usually have $\widehat{G} \ll n$.

4 Numerical comparisons

4.1 Simulation study

A simulation study was carried out to compare the new proposals with the existing methods designed to perform modal clustering after normal mixture modeling.

The seven methods involved in the study were:

- Methods 1 and 2 introduced here (labeled `modmerge` and `modclust`, respectively).
- The recent proposal of Scrucca (2016) to perform modal clustering based on identifying the connected components of high density regions of a normal mixture density estimate (labeled `gmmhd`).
- The methods `ridgeuni`, `ridgeratio` and `dipuni` introduced in Sections 3.1, 3.2 and 3.3 of Hennig (2010), respectively, with the recommended default parameters.
- The entropy-based merging methodology introduced in Baudry et al. (2010), here labeled `entmerge`.

As mentioned before, `ridgeuni` and `ridgeratio` are based on the analysis of the ridgeline connecting two mixture components (Ray and Lindsay 2005), a curve joining the two component means which necessarily contains all the critical points of the mixture having such two components. If two modes are found along this ridgeline, then the `ridgeuni` method does not merge the two mixture components; the

`ridgeratio` method proceeds in the same way, but it can merge the two components even if they give rise to two different modes, as long as the gap between these two modes is not strong enough. Finally, the `dipuni` method makes the decision of merging two components (or not) based on a unimodality test. See Hennig (2010) for further details.

In contrast, `entmerge` markedly differs from the rest of the methods considered here, since its merging mechanism does not have a modality motivation. Instead, a sequential merging scheme is produced in which the initial mixture model clusters are successively combined until all observations are assigned to a single cluster. At each step, the procedure chooses to merge the pair of clusters from the previous iteration that yields the minimum possible entropy of the resulting new clustering. Here, the optimal combination of clusters was chosen by examining the elbow of a piecewise regression estimate of the entropy reduction function, as suggested in Baudry et al. (2010).

An R script with the implementation of the `modmerge` and `modclust` methods is available from the author's webpage http://matematicas.unex.es/~jechacon. The R package `fpc` (Hennig 2015) includes efficient implementations of the `ridgeuni`, `ridgeratio` and `dipuni` methods, which were used in this study. Also, version 5.3 of the R package `mclust` (Scrucca et al. 2016) offers ready-to-use routines for both the `entmerge` and `gmmhd` methods, although it should be remarked that, for the preliminar versions of this paper, an R script with the implementation of the `gmmhd` procedure was kindly provided by Professor Luca Scrucca.

It should be noted that two out of the methods in the study (`modclust` and `gmmhd`) are not based on merging components of the mixture density estimate, but on directly identifying the modal regions of the density estimate. Hence, they should be expected to have a higher computational load, since the job of the remaining methods is "only" to decide if two given components should be merged or not.

There exist other closely related density-based clustering methods in the literature, like DBSCAN (Ester et al. 1996) and its variants (see Ester 2014), where the density is estimated using nearest neighbours (in a sense), and also those based on kernel density estimators (Azzalini and Torelli 2007; Chacón and Duong 2013). They were not included in this study for two reasons: first, because the main focus of this paper is to investigate methods to perform modal clustering after mixture modeling, and such methods do not fall exactly within this category; and second, because the performance of these other methods relies heavily on the choice of their tuning parameters and, even if there is a variety of proposals for making these choices automatically, including them in this study would necessarily involve comparing also the different tuning parameter methodologies, which might distract us from the main goal of this paper.

The models over which the previous methods were tested cover a wide range of cluster shapes, to study the impact of skewness, heavy tails and non-elliptical groups in the different methodologies. They are:

- M1, *uniform on* $[0, 1] \times [0, 1]$ Included by suggestion of an anonymous reviewer, this is a model where the notion of modal clustering is not expected to fit well. As noted in Hennig (2010, p. 9) normal mixture modeling applied to this distribution "tends to come up with multimodal" density estimates, so it is likely that in this

case the number of modes of the mixture density estimate does not even provide a consistent estimator for the number of modes of this distribution.

- M2, *mixture of three skew-normal components* Included to explore the effect of having skew components, this mixture has three equal-weight restricted skew-normal density components, each having a parametrization $rSN(\mu, \Sigma, \lambda)$ as given in Lin et al. (2016), with parameters

$$\mu_1 = (4, -4), \quad \Sigma_1 = \begin{pmatrix} 1 & -0.1 \\ -0.1 & 1 \end{pmatrix}, \quad \lambda_1 = (3, 3)$$
$$\mu_2 = (3.5, 4), \quad \Sigma_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \lambda_2 = (1, 5)$$
$$\mu_3 = (0, 0), \quad \Sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \lambda_3 = (-3, 1)$$

- M3, *mixture of three t distributions*. Included to measure the influence of heavy-tail components, it also has three equal-weight $t_\nu(\mu, \Sigma)$ components, with the same μ_g and Σ_g parameters as in M2 for $g = 1, 2, 3$ and degrees of freedom $\nu_1 = 3$, $\nu_2 = \nu_3 = 5$.
- M4, *skew mixture of normal components*. This is a 12-component normal mixture distribution with only 4 modes, so it represents a scenario where modal clustering after mixture modeling should be useful. The first of the 4 modal regions consists of a pear-like skewed distribution made of a 3-component normal mixture, with weights $\pi_1 = 1/2$, $\pi_2 = 2/5$ and $\pi_3 = 1/10$ and parameters

$$\mu_1 = (2, 0), \quad \Sigma_1 = \begin{pmatrix} 0.5^2 & 0 \\ 0 & 0.5^2 \end{pmatrix}$$
$$\mu_2 = (3, 0), \quad \Sigma_2 = \begin{pmatrix} 0.5^2 & 0 \\ 0 & 0.35^2 \end{pmatrix}$$
$$\mu_3 = (4, 0), \quad \Sigma_3 = \begin{pmatrix} 0.35^2 & 0 \\ 0 & 0.35^2 \end{pmatrix}$$

  Then, this distribution is replicated three times after a rotation 90, 180 and 270 degrees, and each of these four leaves is assigned the same weight into the final 12-component mixture.
- M5, *normal mixture with overlapping components*. This distribution was introduced in Baudry et al. (2010, Section 4.1). It is a 6-component normal mixture with 4 modes. Two of the modes are associated with two corresponding unimodal distributions with elipsoidal contours, while the other two modes are related to two, less usual, unimodal distributions with cross-shaped contours. The mixing proportions, means and variances defining this distribution, as extracted from Baudry (2010, Appendix A.2), are

$$\pi_1 = \pi_2 = \pi_3 = \pi_4 = 0.2, \ \pi_5 = \pi_6 = 0.1,$$
$$\mu_1 = (0, 0), \ \mu_2 = (8, 5), \ \mu_3 = \mu_4 = (1, 5), \ \mu_5 = \mu_6 = (8, 0),$$
$$\Sigma_1 = RAR^\top, \ \Sigma_2 = R^\top AR, \ \Sigma_3 = B, \ \Sigma_4 = A, \ \Sigma_5 = B, \ \Sigma_6 = A,$$

  where $A = \begin{pmatrix} 1 & 0 \\ 0 & 0.1 \end{pmatrix}$, $B = \begin{pmatrix} 0.1 & 0 \\ 0 & 1 \end{pmatrix}$ and $R = \frac{1}{2}\begin{pmatrix} 1 & -\sqrt{3} \\ \sqrt{3} & 1 \end{pmatrix}$.

- M6, *broken ring distribution*. This distribution is described in detail in Chacón and Duong (2013, Section 5.1). It has five modal clusters: a spherical bump in the middle, surrounded by four crescent-shaped clusters with different orientation.
- M7, *bimodal mixture of three normal components*. This is Density (J) in Wand and Jones (1993), where it is called Trimodal II. This denomination is somewhat misleading, since it is a mixture of three components with only two modes. It represents a distribution in which modal clusters cannot be simply obtained by merging mixture components.

The density contours of all these distributions, along with their population modal clusterings, are depicted in Fig. 5.

To evaluate the performance of the methods, a distance between whole-space clusterings is needed, since the goal is to compare the achievements of the methods at the time of estimating the true population clustering. To that aim a choice was made to use the distance in measure introduced in Chacón (2015). This distance is defined as follows: given two clusterings $\mathcal{C} = \{C_1, \ldots, C_r\}$ and $\mathcal{D} = \{D_1, \ldots, D_s\}$ of a probability distribution P, with $r \leq s$, set

$$d_P(\mathcal{C}, \mathcal{D}) = \frac{1}{2} \min_{\sigma \in \mathcal{P}_s} \sum_{i=1}^{s} P(C_i \triangle D_{\sigma(i)}), \tag{5}$$

where $\mathcal{P}_s$ denotes the set of permutations of $\{1, 2, \ldots, s\}$, the partition $\mathcal{C}$ has been enlarged by adding $s-r$ empty sets $C_{r+1} = \cdots = C_s = \varnothing$ if necessary, and $\triangle$ denotes the symmetric difference between two sets, namely $C \triangle D = (C \cap D^c) \cup (C^c \cap D)$. Other choices of distance between clusterings are possible, but d_P has a natural interpretation since it represents the smallest probability mass that needs to be moved to transform clustering $\mathcal{C}$ into clustering $\mathcal{D}$.

To compute the distance between a data-based clustering $\widehat{\mathcal{C}}$, obtained from any of the aforementioned methods, and the true population clustering $\mathcal{D}$, a discretization scheme was followed, much as in Chacón and Monfort (2014):

1. Take a fine enough grid over a large rectangle chosen to contain at least 0.999 probability mass of the distribution.
2. Rule this grid in rectangles by considering a tiny rectangle centered at each grid point with its sides of length half the distance to the next grid point in each coordinate direction.
3. Assign each grid point to its cluster by running a population version of the mean shift algorithm (i.e., using the true density and density gradient). This results in a discretization of the true population clustering $\mathcal{D}$, since every cluster can be approximated by the union of tiny rectangles surrounding the grid points that are labeled to belong to it.
4. Compute the probability mass of each tiny rectangle in the discretization of the population clustering.
5. Run the corresponding method in the simulation study to construct a data-based clustering $\widehat{\mathcal{C}} = \{\widehat{C}_1, \ldots, \widehat{C}_r\}$ and assign each of the grid points to a cluster in $\widehat{\mathcal{C}}$. If necessary, enlarge one of the two clusterings ($\widehat{\mathcal{C}}$ or $\mathcal{D}$) by attaching empty sets so that both have the same number of clusters (say, s). By adding up the

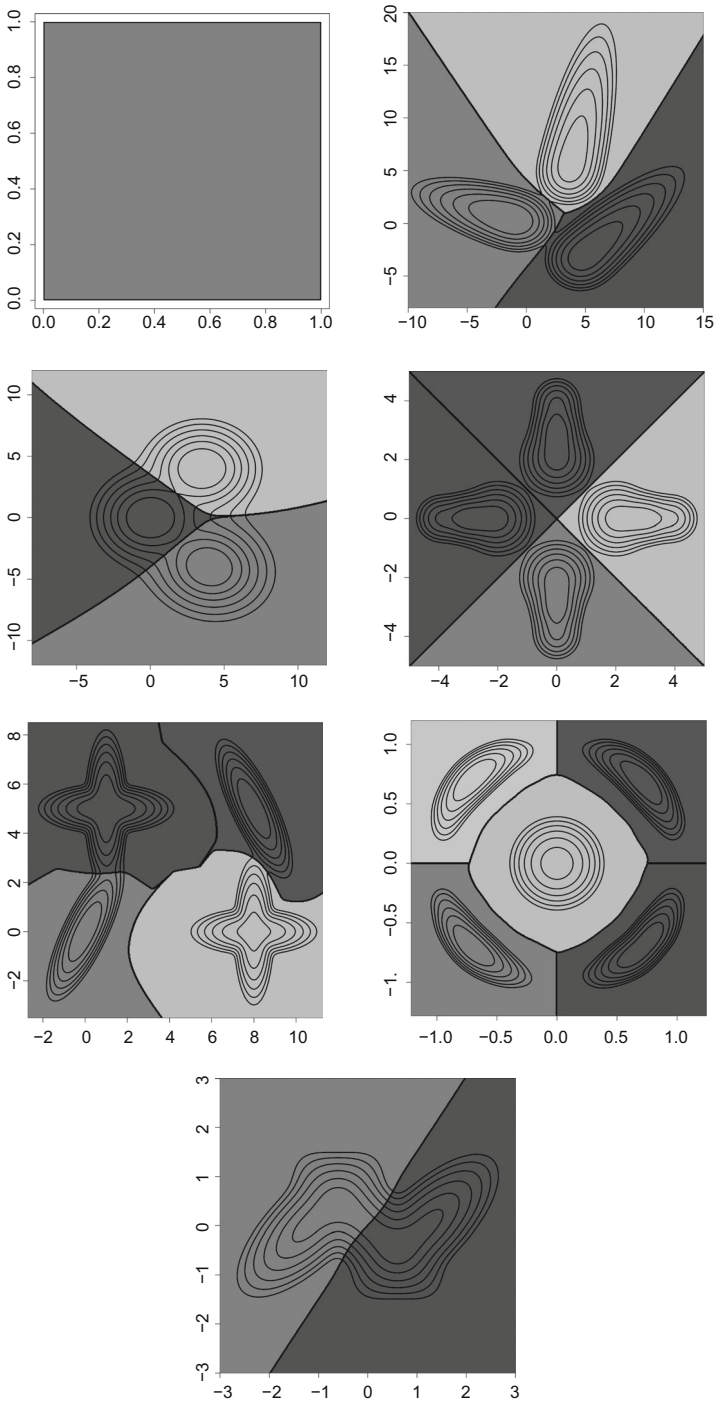

Fig. 5 The seven density models included in the simulation study. Each picture contains the density contour plot plus the population modal clusterings indicated with colours

Table 1 Mean computation time, in seconds, for the seven methods in the study along the 100 simulation runs from the seven models M1–M7

	modmerge	modclust	gmmhd	ridgeuni	ridgeratio	dipuni	entmerge
$n = 500$							
M1	0.01	0.95	14.60	2.83	2.82	4.01	0.04
M2	0.01	0.52	29.07	1.80	1.79	2.55	0.02
M3	0.01	0.35	42.08	1.70	1.69	2.49	0.02
M4	0.00	0.36	28.17	1.60	1.60	1.64	0.02
M5	0.01	0.54	21.46	4.91	4.91	7.61	0.08
M6	0.03	1.17	20.86	7.47	7.47	11.35	0.13
M7	0.02	1.28	59.04	0.79	0.79	1.08	0.01
$n = 2000$							
M1	0.03	1.94	136.09	8.54	8.54	11.14	0.55
M2	0.03	0.83	236.61	4.50	4.50	7.18	0.25
M3	0.01	0.38	401.98	2.66	2.65	4.39	0.14
M4	0.08	2.13	282.82	6.98	6.97	11.33	0.43
M5	0.01	0.48	210.89	7.04	7.03	11.65	0.44
M6	0.09	2.27	350.15	13.66	13.65	22.62	1.03
M7	0.02	1.11	735.85	0.80	0.79	1.15	0.04

Shorter times indicate more efficient computational performance

contributions corresponding to the rectangles that approximate each symmetric difference we obtain an approximation of $P(\widehat{C}_i \triangle D_j)$ for each $i, j = 1, \ldots, s$. Finally, finding the minimum over all the permutations $\mathcal{P}_s$ in Equation (5) is a linear sum assignment problem, and efficient algorithms to solve it are shown, e.g., in Papadimitriou and Steiglitz (1982).

Notice that steps 1–4 only need to be executed once for every model in the simulation study.

For the study, sample sizes $n = 500$ and $n = 2000$ were considered, and 100 samples of each size were drawn from each of the six density models. For each of these samples a normal mixture density was fitted by combining the EM algorithm and the BIC, and each of the grid points (not the sample points) in the above discretization scheme was assigned a component in the fitted mixture based on its a posteriori probability.

Since one of the main motivations for the new methods presented here is computational efficiency, a first feature to examine is the computational time of each of the methods. Average computational times of the six methods, over the 100 replications for each model and sample size, when executed on a 3.10 GHz quad-core Intel(R) Xeon(R) CPU E3-1220 V2 with 10 GB RAM, are shown in Table 1.

The times reflect how long the studied methods take to assign its cluster to all the grid points in the above discretization scheme, once they are given the normal mixture density estimate and the components to which each of these grid points have been assigned by mixture model clustering. As noted before, this last piece of information

is not needed by the `modclust` and `gmmhd` methods, which are not based on merging components, but on producing a modal clustering directly from the mixture density estimate. Hence, their job is admittedly more onerous, since the other methods just need to relabel the cluster memberships given by the mixture model clustering assignment.

It is worth noting that a 150×150 grid was used for all the models, so Table 1 in fact shows the average times that each method takes to cluster $150^2 = 22{,}500$ points. In practice, for the more common goal of clustering just the points of the data sample, these times might be substantially shorter.

In view of Table 1, in terms of computational efficiency `modmerge` is the undeniable winner, closely followed by `entmerge` and then `modclust` which, despite not being a component-merging method, is faster (on average) than the remaining competitors. The method `gmmhd` seems to be considerably slower than the others, with a prominent slow-down as the sample size grows. As expected, both ridgeline-based methods share similar performance times, and `dipuni` seems to be the slowest among these three component-merging methods.

A different story follows by inspecting the methods in terms of their accuracy. Table 2 contains the means and standard deviations (multiplied by a factor of 100, for ease of reading) of the distance in measure to the population modal clustering for the seven methods across the seven density models. If, for instance, a method has an average score of 5% in distance in measure, the interpretation is that an average of 5% of the probability mass needs to be moved to transform the data-based clustering that it produces into the true population one. Notice that, if that were the case, then of course the data-based clustering produced with such method would not be significantly different than the true clustering.

Regarding the results, it is clear that for the case of the uniform distribution, as expected from the above remarks, modal clustering does not fit well; the only notable exception is the `ridgeratio` method, which is able to merge clusters even if they correspond to different modes. Something similar happens for model M3, this time due to the heavy tails that may produce numerous spurious bumps in the less dense zones. For the rest of the models, `modclust` and `gmmhd` perform remarkably well, particularly when $n = 2000$, and the same applies to `modmerge`, `ridgeuni`, `ridgeratio` and `dipuni`, with the expected exception of model M7: for this model, the true clustering can not be obtained by merging components, and this causes much trouble to the component merging methods. Finally, the fact that `entmerge` does not have a modality motivation is also noticeable for this model M7, although its performance is quite acceptable despite its different rationale.

Overall, if we were to make a single recommendation, with the added value of computational efficiency, `modclust` is probably the method that should be advised for general use.

4.2 Multivariate models

Two additional synthetic models were considered to study the performance of the methods beyond the bivariate case. They are 6-dimensional and 4-dimensional variants of the former M2 and M5 models, respectively. Precisely:

Table 2 Means and (standard deviations) of the distance in measure to the population modal clustering, multiplied by a factor of 10^2, for the seven methods in the study along the 100 simulation runs from the seven models M1–M7

	modmerge	modclust	gmmhd	ridgeuni	ridgeratio	dipuni	entmerge
$n = 500$							
M1	76.63	76.15	71.83	76.63	0.00	45.96	34.99
	(6.43)	(4.82)	(6.44)	(6.43)	(0.00)	(24.71)	(15.66)
M2	2.02	1.55	0.94	2.35	0.93	2.34	9.42
	(2.87)	(2.38)	(3.10)	(4.25)	(4.66)	(10.35)	(14.63)
M3	66.02	65.00	58.88	66.62	69.98	68.06	67.26
	(11.42)	(7.67)	(11.22)	(11.78)	(11.77)	(10.66)	(10.28)
M4	0.17	0.20	0.33	0.17	0.09	0.09	34.28
	(0.88)	(1.20)	(1.38)	(0.88)	(0.06)	(0.06)	(13.57)
M5	0.19	0.16	1.22	0.82	4.89	2.23	0.89
	(0.51)	(0.42)	(2.98)	(3.46)	(8.56)	(5.48)	(2.57)
M6	10.14	9.27	8.66	10.14	12.50	7.93	7.32
	(7.12)	(7.29)	(9.98)	(7.12)	(19.31)	(13.92)	(7.98)
M7	20.61	14.82	12.94	29.30	49.96	28.89	16.25
	(9.55)	(14.08)	(12.33)	(15.32)	(0.04)	(16.72)	(3.35)
$n = 2000$							
M1	82.57	84.82	81.26	82.57	0.00	70.28	34.70
	(6.31)	(2.88)	(4.79)	(6.31)	(0.00)	(13.02)	(16.72)
M2	2.43	1.96	0.97	2.43	0.12	3.56	1.44
	(4.73)	(4.05)	(3.38)	(4.73)	(0.04)	(6.48)	(1.81)
M3	68.21	64.68	52.74	70.98	74.74	69.40	67.95
	(6.53)	(8.11)	(10.67)	(6.31)	(0.01)	(6.73)	(6.46)
M4	7.37	6.91	3.55	7.05	0.57	2.92	2.60
	(9.22)	(8.72)	(6.27)	(9.09)	(3.51)	(7.61)	(10.93)
M5	0.07	0.06	1.23	0.32	4.45	3.37	0.62
	(0.03)	(0.02)	(4.29)	(2.21)	(8.32)	(6.39)	(1.96)
M6	5.44	4.73	9.55	5.12	18.31	9.33	3.61
	(4.60)	(4.77)	(8.31)	(4.50)	(21.46)	(12.84)	(2.60)
M7	17.62	5.39	6.21	37.65	49.96	31.57	16.61
	(6.43)	(8.33)	(7.28)	(16.26)	(0.02)	(16.93)	(2.07)

Smaller distance indicates better performance

- 6-variate M2. This is a multivariate generalization of model M2, with three equally weighted skew-normal components with parameters

$$\mu_1 = (4, -4, 4, -4, 4, -4), \quad \Sigma_1 = I_6, \quad \lambda_1 = \mu_1$$
$$\mu_2 = (4, 4, 4, 4, 4, 4), \quad \Sigma_2 = I_6, \quad \lambda_2 = \mu_2$$
$$\mu_3 = (0, 0, 0, 0, 0, 0), \quad \Sigma_3 = I_6, \quad \lambda_3 = (-6, 0, 0, 0, 0, 0),$$

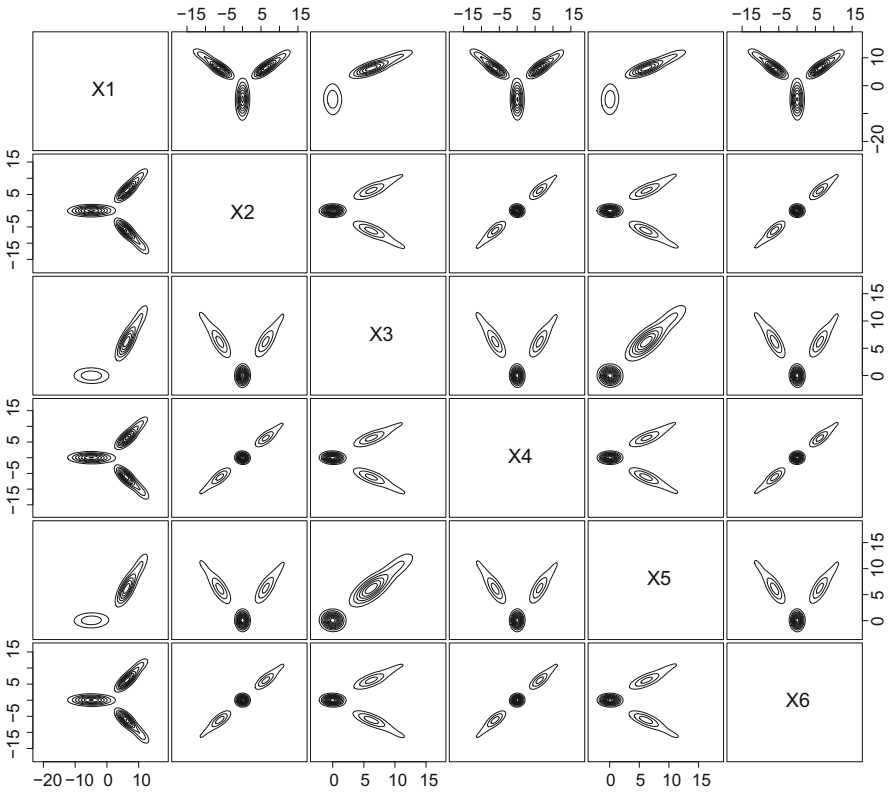

Fig. 6 Matrix showing the contour plots of all the pairwise marginals of the 6-dimensional extension of model M2

where I_d stands for the $d \times d$ identity matrix.

- 4-variate M5. This is a multivariate generalization of the bivariate model M5, with exactly that distribution for the first two coordinates and the remaining two representing normal noise.

The matrices with the contour plots for the pairwise marginals of each of these two models are included in Figs. 6 and 7, respectively. As before, Tables 3 and 4 contain the average computational times and average performance of each of the seven studied methods over these two multivariate models, obtained from 100 replications. In this multivariate context, since using a grid of 150 points for each coordinate would result in a very large number of total points to cluster to find a discretized version of the distance in measure ($150^6 \simeq 1.139 \times 10^{13}$ for the 6-dimensional model), we opted to cluster only the data points, thus approximating (5) by its empirical counterpart, which is obtained by replacing P in (5) with the sample probability distribution (i.e., the distribution that assigns mass $1/n$ to each of the data points). As noted in Chacón (2015), this empirical distance in measure coincides with a commonly used discrepancy measure between clusterings of finite data points, namely the transfer distance

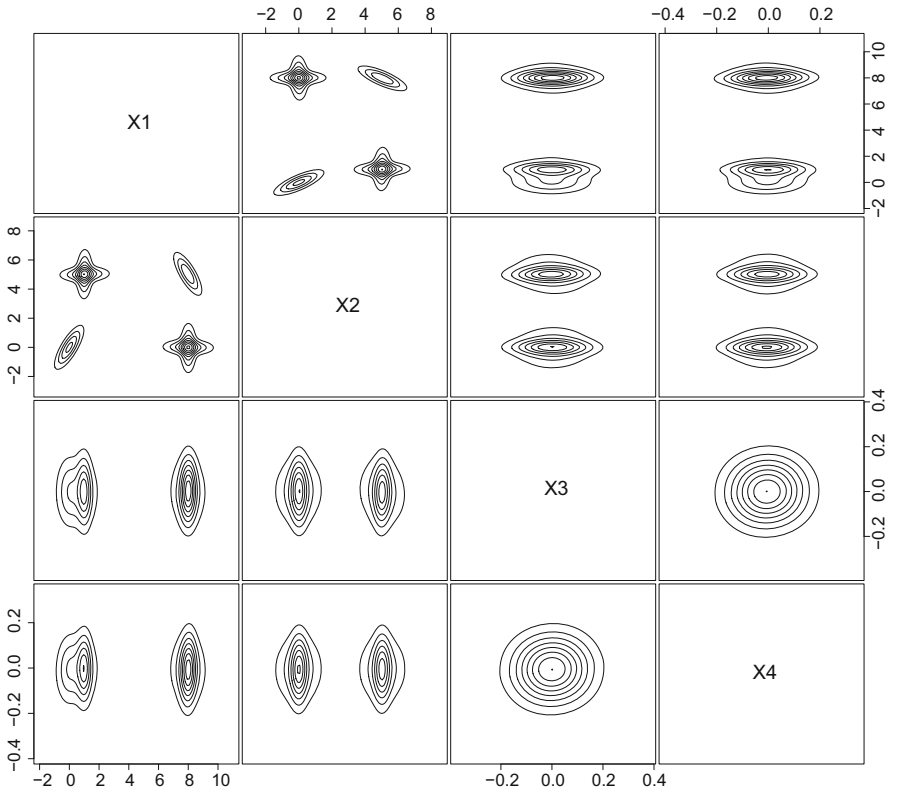

Fig. 7 Matrix showing the contour plots of all the pairwise marginals of the 4-dimensional extension of model M5

Table 3 Mean computation time, in seconds, for the seven methods in the study along the 100 simulation runs from the two multivariate models

	modmerge	modclust	gmmhd	ridgeuni	ridgeratio	dipuni	entmerge
$n = 500$							
M2 ($d = 6$)	0.00	0.10	15.69	1.21	1.21	1.44	0.01
M5 ($d = 4$)	0.01	0.09	30.90	4.10	4.09	6.01	0.06
$n = 2000$							
M2 ($d = 6$)	0.00	0.38	146.48	1.33	1.33	1.61	0.06
M5 ($d = 4$)	0.01	0.33	207.61	4.26	4.25	6.42	0.22

Shorter times indicate more efficient computational performance

(Denœud 2008), that records the minimal proportion of data points that would need to be re-labeled to transform one clustering into the other.

Again, `gmmhd` seems to be notably slower than the other methods. This was expected since it makes use of the Delaunay triangulation of the sample, whose computation is known to be problematic as the dimension increases. However, it looks

Table 4 Means and (standard deviations) of the empirical distance in measure to the population modal clustering, multiplied by a factor of 10^2, for the seven methods in the study along the 100 simulation runs from the two multivariate models

	`modmerge`	`modclust`	`gmmhd`	`ridgeuni`	`ridgeratio`	`dipuni`	`entmerge`
$n = 500$							
M2 ($d = 6$)	2.78	2.51	4.76	2.78	0.05	0.56	29.80
	(9.16)	(8.42)	(5.34)	(9.16)	(0.10)	(3.67)	(7.62)
M5 ($d = 4$)	0.94	0.91	1.21	0.94	0.39	1.54	3.44
	(2.72)	(2.64)	(2.79)	(2.72)	(1.91)	(4.18)	(7.84)
$n = 2000$							
M2 ($d = 6$)	2.31	2.10	4.26	2.31	0.06	0.99	31.14
	(9.73)	(9.10)	(6.41)	(9.73)	(0.06)	(6.14)	(6.13)
M5 ($d = 4$)	0.04	0.03	1.14	0.04	0.37	0.79	0.05
	(0.04)	(0.04)	(3.55)	(0.04)	(2.37)	(3.33)	(0.16)

Smaller distance indicates better performance

like the sample size represents a more difficult challenge than the dimensionality for the current implementation of this method, since increasing the sample size yields an considerable increment in its computation time.

In terms of performance, however, all the methods do a good job, with distances to the true population clustering below 5%. The only exception to this is the performance of `entmerge` for the 6-dimensional variant of model M2, which is significantly worse (but again, note that the population goal of `entmerge` is not the true population modal clustering).

4.3 Two additional real data examples

4.3.1 DLBCL data set

Aghaeepour et al. (2013) considered a data set regarding biopsies of 30 patients with Diffuse Large B-cell Lymphoma (DLBCL). In this experiment, each sample was stained with three fluorochrome-conjugated antibodies, CD3, CD5, and CD19. Within the flow cytometer, cells passed sequentially through laser beams that excite the fluorochromes. The emitted light, which is proportional to the antigen density, was then measured and recorded. The whole data set is available from the FlowRepository database at the address `https://flowrepository.org/id/FR-FCM-ZZYY`. Here, we analyze a subset of $n = 8183$ cells from one of the patients, that is contained in the R package `EMMIXuskew` (Lee and McLachlan 2013), using the seven methods studied in this section.

Figure 8 represents the matrix of pairwise scatterplots for this data set. Eight components were obtained as the result of a normal mixture fitting though EM and BIC. Once the distribution was thus fitted, the computation times to obtain the final cluster-

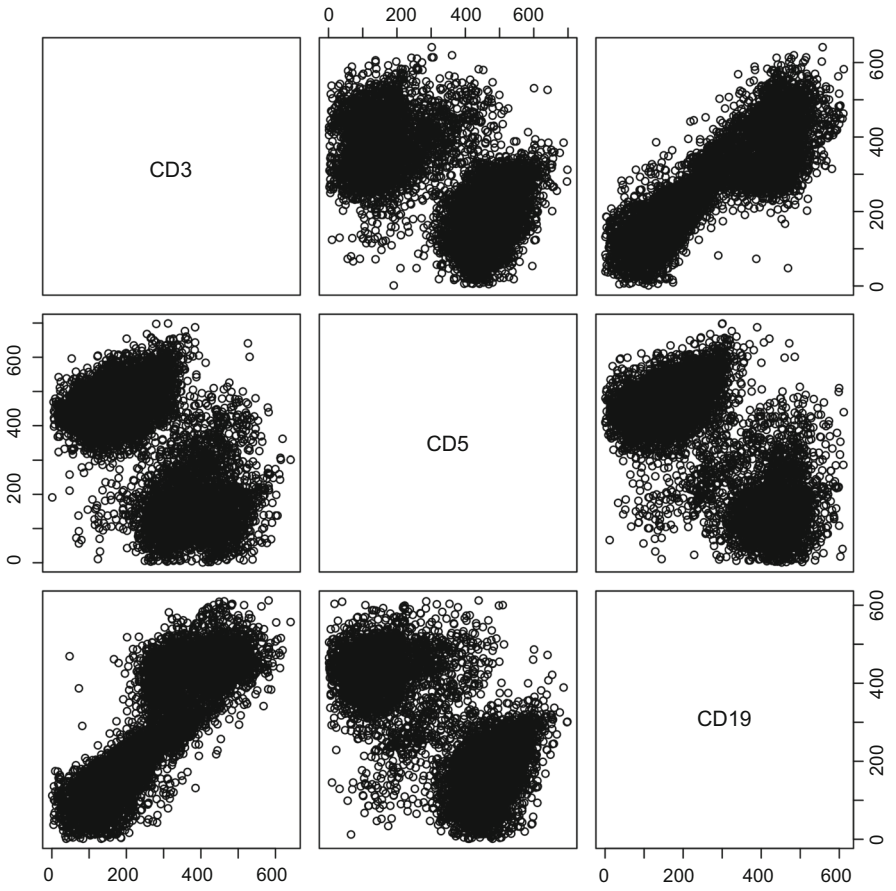

Fig. 8 Two by two variables scatterplot of the DLBCL data

Table 5 Computation times (in seconds), number of final clusters and accuracy (in terms of 100 times the empirical distance in measure) for the analysis of the DLBCL data with each of the methods in the study

	`modmerge`	`modclust`	`gmmhd`	`ridgeuni`	`ridgeratio`	`dipuni`	`entmerge`
Comp. time	0.30	30.70	4751.78	13.76	13.80	21.53	2.75
No. of clusters	3	3	4	3	1	5	5
Accuracy	9.39	9.12	12.10	19.52	39.62	18.45	12.73

ing were registered, and they are shown in Table 5. Again, `gmmhd` looks considerably slower than the other methods.

The true cluster labels, obtained by manual gating, are also included in the EMMIXuskew package. Manual gating led to five different clusters, although one of them appears critically smaller than the other four, since it contains only 62 observations (0.76% of the total data). Using this information about the true labels, it is

Table 6 Computation times (in seconds), number of final clusters and accuracy (in terms of 100 times the empirical distance in measure) for the analysis of the wine data with each of the methods in the study

	`modmerge`	`modclust`	`gmmhd`	`ridgeuni`	`ridgeratio`	`dipuni`	`entmerge`
Comp. time	0.01	0.07	8.55	3.15	3.14	4.20	0.03
No. of clusters	3	3	3	3	2	3	3
Accuracy	1.69	1.69	2.81	1.69	33.71	1.69	1.69

possible to compute the empirical distance in measure (or transfer distance) to the true clustering for each of the automatic clustering algorithms. The results are also listed in Table 5. Once again, `modclust` seems to be the best method in terms of accuracy. Despite producing three clusters instead of five, it would suffice to change the label of only 7.58% of the data points to transform the result of this method into the true classification. The procedure `modmerge` comes second in this ranking, and `gmmhd` is the third best choice. Modal merging through `ridgeuni` also led to three clusters, but with a significantly lower accuracy, and `ridgeratio` suggested merging all the components, thus yielding the worse performance. Finally, both `dipuni` and `entmerge` found five clusters, but their performances are far from the best for this data set: even if they found the correct number of clusters, they incurred in a higher overall transfer rate.

4.3.2 Wine data set

This data set was introduced in Forina et al. (1986). It contains measurements of $d = 13$ chemical variables on a set of $n = 178$ Italian wines from three different cultivars (Barolo, Grignolino, Barbera), which are assumed to represent the true cluster labels. This example provides a situation where the data are higher-dimensional but the sample size is relatively small.

After scaling, the normal mixture fit for this data set resulted in four components, leading to a transfer distance of 15.73% to the true clustering. Table 6 shows the computation times, number of clusters and accuracies of all the methods in the study, when applied after obtaining the normal mixture fit. Again, `gmmhd` appears as the slowest of them all, although it took a reasonable amount of time for this example, which seems to indicate that the computation time for this method is more severely affected by the sample size rather than by the data dimensionality. Regarding the accuracy, all the methods but `gmmhd` and `ridgeratio` obtained a remarkable transfer distance of 1.69%, corresponding to only 3 misclassified observations. Its two-cluster proposal led `ridgeratio` to a poor performance of 33.71%, and it should be noted that the transfer distance 2.81% of `gmmhd` regards the application of this method without the dimension reduction recommended in Scrucca (2016); if `gmmhd` is applied after the proposed dimension reduction then its transfer distance to the true clustering reduces to 0.56% (that is, only one misclassified observation).

5 Discussion

This paper illustrates how mixture modeling can be useful even if the final goal is to cluster the data according to a modal approach, instead of by mixture component assignment. Two different proposals are introduced for this task: one based on merging mixture components (Method 1, `modmerge`) and a second one which uses the mean shift algorithm on the fitted normal mixture density to find the domains of attraction of the estimated density modes (Method 2, `modclust`).

The issue of the convergence of the mean shift algorithm is still not a fully solved problem. Aliyari Ghassabeh (2015) points out the incompleteness of several existing convergence proofs. In this paper, a new representation of the mean shift algorithm for non-isotropic normal mixture densities is provided, which allows to cast it as a quasi-Newton optimization method. This could be useful to address the convergence issue once again, since it could be tackled using the tools that are normally employed to study the convergence of such optimization methods (see Dennis and Schnabel 1996 Chapter 6).

As another open problem, even if Li and Barron (2000) showed that the mixture density estimator is consistent under mild assumptions, that does not guarantee that the number of modes of the mixture density estimator be a consistent estimator for the true number of density modes. Or, looking further afield, it would be even better if it could be proved that the modal clustering that is obtained from the mixture density estimate (using any of the methods in Sect. 4) results in a consistent estimate of the true population modal clustering, in the sense indicated in Chacón (2015, Section 4.3).

Finally, our simulation study reveals that the two new proposals are quite efficient from a computational point of view, and perform similarly to or better than other existing methods in terms of clustering accuracy. So if computational efficiency is of great concern, any of the two proposals should be recommended to perform modal clustering after mixture modeling.

Acknowledgements The author is grateful to Professor Luca Scrucca (Università degli Studi di Perugia) for kindly sharing his code to perform modal clustering based on high density regions. The comments of the Associate Editor and two anonymous reviewers also contributed to improve the initial version of the manuscript. The author acknowledges the support of the Spanish Ministerio de Economía y Competitividad grants MTM2013-44045-P and MTM2016-78751-P and the Junta de Extremadura grant GR15013.

References

Aghaeepour, N., Finak, G., The FlowCAP Consortium, The DREAM Consortium, Hoos, H., Mosmann, T.R., Brinkman, R., Gottardo, R. and Scheuermann, R.H (2013) Critical assessment of automated flow cytometry analysis techniques. Nat Methods 10:228–238

Aliyari Ghassabeh Y (2015) A sufficient condition for the convergence of the mean shift algorithm with Gaussian kernel. J Multivar Anal 135:1–10

Arias-Castro E, Mason D, Pelletier B (2016) On the estimation of the gradient lines of a density and the consistency of the mean-shift algorithm. J Mach Learn Res 17:1–28

Azzalini A, Bowman AW (1990) A look at some data on the Old Faithful geyser. Appl Stat 39:357–365

Azzalini A, Torelli N (2007) Clustering via nonparametric density estimation. Stat Comput 17:71–80

Baudry J-P (2010) Sélection de Modèle pour la Classifcation Non Supervisée. Choix du Nombre de Classes. Ph.D. Thesis, Université Paris-Sud 11

Baudry J-P, Raftery AE, Celeux G, Lo K, Gottardo R (2010) Combining mixture components for clustering. J Comput Graph Stat 19:332–353
Bock H-H (1974) Automatische Klassifikation (Clusteranalyse). Vandenhoeck & Ruprecht, Göttingen
Brinkman RR, Gasparetto M, Lee S-JJ, Ribickas AJ, Perkins J, Janssen W, Smiley R, Smith C (2007) High-content flow cytometry and temporal data analysis for defining a cellular signature of Graft-versus-Host Disease. Biol Blood Marrow Transpl 13:691–700
Carlsson G, Mémoli F (2013) Classifying clustering schemes. Found Comput Math 13:221–252
Carreira-Perpiñán MÁ (2000) Mode-finding for mixtures of Gaussian distributions. IEEE Trans Pattern Anal Mach Intell 22:1318–1323
Carreira-Perpiñán MÁ (2006) Acceleration strategies for Gaussian mean-shift image segmentation. In: IEEE conference on computer vision and pattern recognition (CVPR 2006), pp 1160–1167
Carreira-Perpiñán MÁ (2007) Gaussian mean shift is an EM algorithm. IEEE Trans Pattern Anal Mach Intell 29:767–776
Carreira-Perpiñán MÁ, Williams CKI (2003a) On the number of modes of a Gaussian mixture. In: Scale-space methods in computer vision. Lecture notes in computer science, vol 2695, pp 625–640. Springer, Berlin
Carreira-Perpiñán MÁ, Williams CKI (2003b) An isotropic Gaussian mixture can have more modes than components. Technical report EDI-INF-RR-0185, School of Informatics, University of Edinburgh, UK
Chacón JE (2012) Identifying nonstandard group shapes in mixture model clustering through the mean shift algorithm. In: Programme and abstracts of the 5th international conference of the ERCIM working group on computing and statistics, p 122
Chacón JE (2015) A population background for nonparametric density-based clustering. Stat Sci 30:518–532
Chacón JE, Duong T (2013) Bandwidth selection for multivariate density derivative estimation, with applications to clustering and bump hunting. Electron J Stat 7:499–532
Chacón JE, Monfort P (2014) A comparison of bandwidth selectors for mean shift clustering. In: Skiadas CH (ed) Theoretical and applied issues in statistics and demography, pp 47–59. International Society for the Advancement of Science and Technology (ISAST), Athens
Comaniciu D (2003) An algorithm for data-driven bandwidth selection. IEEE Trans Pattern Anal Mach Intell 25:281–288
Comaniciu D, Meer P (2002) Mean shift: a robust approach toward feature space analysis. IEEE Trans Pattern Anal Mach Intell 24:603–619
Cuevas A, Febrero M, Fraiman R (2001) Cluster analysis: a further approach based on density estimation. Comput Stat Data Anal 36:441–459
Dennis JE, Schnabel RB (1996) Numerical methods for unconstrained optimization and nonlinear equations. SIAM, Philadelphia
Denœud L (2008) Transfer distance between partitions. Adv Data Anal Classif 2:279–294
Duong T, Cowling A, Koch I, Wand MP (2008) Feature significance for multivariate kernel density estimation. Comput Stat Data Anal 52:4225–4242
Edelsbrunner H, Fasy BT, Rote G (2013) Add isotropic Gaussian kernels at own risk: more and more resilient modes in higher dimensions. Discrete Comput Geom 49:797–822
Edelsbrunner H, Harer J (2008) Persistent homology—a survey. Contemp Math 453:257–282
Ester M (2014) Density-based clustering. In: Aggarwal CC, Reddy CK (eds) Data clustering: algorithms and applications. Chapman & Hall, Boca Raton, pp 111–126
Ester M, Kriegel H, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the 2nd international conference on knoledge discovery and data mining, pp 226–231. AAAI Press, Portland
Forina M, Armanino C, Castino M, Ubigli M (1986) Multivariate data analysis as a discriminating method of the origin of wines. Vitis 25:189–201
Fraley C, Raftery AE (2002) Model-based clustering, discriminant analysis, and density estimation. J Am Stat Assoc 97:611–631
Fraley C, Raftery AE, Scrucca L (2016) mclust: Gaussian mixture modelling for model-based clustering, classification, and density estimation. R package version 5:2
Fukunaga K, Hostetler LD (1975) The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Trans Inf Theory 21:32–40
Hartigan JA (1975) Clustering algorithms. Wiley, New York

Hennig C (2010) Methods for merging Gaussian mixture components. Adv Data Anal Classif 4:3–34
Hennig C (2015) fpc: flexible procedures for clustering. R package version 2.1-10. https://CRAN.R-project.org/package=fpc
Lee SX, McLachlan GJ (2013) EMMIXuskew: an R package for fitting mixtures of multivariate skew t distributions via the EM algorithm. J Stat Softw 55:1–22
Li JQ, Barron AR (2000) Mixture density estimation. In Solla SA, Leen TK, Mueller K-R (eds) Adv Neural Inf Process Syst 12:279–285
Li X, Hu Z, Wu F (2007) A note on the convergence of the mean shift. Pattern Recognit 40:1756–1762
Lin T-I (2009) Maximum likelihood estimation for multivariate skew normal mixture models. J Multivar Anal 100:257–265
Lin T-I, Ho HJ, Lee C-R (2014) Flexible mixture modelling using the multivariate skew-t-normal distribution. Stat Comput 24:531–546
Lin T-I, McLachlan GJ, Lee S-X (2016) Extending mixtures of factor models using the restricted multivariate skew-normal distribution. J Multivar Anal 143:398–413
Lo K, Brinkman RR, Gottardo R (2008) Automated gating of flow cytometry data via robust model-based clustering. Cytom A 73:321–332
McLachlan GJ, Basford KE (1988) Mixture models: inference and applications to clustering. Marcel Dekker Inc, New York
Papadimitriou C, Steiglitz K (1982) Combinatorial optimization: algorithms and complexity. Prentice Hall, Englewood Cliffs
Priebe CE (1994) Adaptive mixtures. J Am Stat Assoc 89:796–806
Ray S, Lindsay BG (2005) The topography of multivariate normal mixtures. Ann Stat 33:2042–2065
Ray S, Ren D (2012) On the upper bound of the number of modes of a multivariate normal mixture. J Multivar Anal 108:41–52
Rinaldo A, Singh A, Nugent R, Wasserman L (2012) Stability of density-based clustering. J Mach Learn Res 13:905–948
Schnell P (1964) Eine methode zur auffindung von gruppen. Biometrische Zeitschrift 6:47–48
Scrucca L (2016) Identifying connected components in Gaussian finite mixture models for clustering. Comput Stat Data Anal 93:5–17
Scrucca L, For M, Murphy TB, Raftery AE (2016) mclust 5: clustering, classification and density estimation using gaussian finite mixture models. R J 8:289–317
Stuetzle W (2003) Estimating the cluster tree of a density by analyzing the minimal spanning tree of a sample. J Classif 20:25–47
Walther G (2003) Bikernel mixture analysis. In: Misra JC (ed) Industrial mathematics and statistics. Narosa Publishing House, New Delhi, pp 586–604
Wand MP, Jones MC (1993) Comparison of smoothing parameterizations in bivariate kernel density estimation. J Am Stat Assoc 88:520–528