Adv Data Anal Classif
DOI 10.1007/s11634-010-0058-3

REGULAR ARTICLE

Methods for merging Gaussian mixture components

Christian Hennig

Received: 16 June 2009 / Revised: 2 December 2009 / Accepted: 8 January 2010

Abstract The problem of merging Gaussian mixture components is discussed in situations where a Gaussian mixture is fitted but the mixture components are not separated enough from each other to interpret them as "clusters". The problem of merging Gaussian mixtures is not statistically identifiable, therefore merging algorithms have to be based on subjective cluster concepts. Cluster concepts based on unimodality and misclassification probabilities ("patterns") are distinguished. Several different hierarchical merging methods are proposed for different cluster concepts, based on the ridgeline analysis of modality of Gaussian mixtures, the dip test, the Bhattacharyya dissimilarity, a direct estimator of misclassification and the strength of predicting pairwise cluster memberships. The methods are compared by a simulation study and application to two real datasets. A new visualisation method of the separation of Gaussian mixture components, the ordered posterior plot, is also introduced.

Keywords Model-based cluster analysis · Multilayer mixture · Unimodality · Prediction strength · Ridgeline · Dip test

Mathematics Subject Classification (2000) 62H30

1 Introduction

The Gaussian mixture model is often used for cluster analysis (for an overview and references see Fraley and Raftery 2002; McLachlan and Peel 2000). This approach is based on the assumption that $\mathbb{R}^p$-valued observations $\mathbf{x}_1, \ldots, \mathbf{x}_n$ are i.i.d. according to the density

C. Hennig (✉)
Department of Statistical Science, UCL, Gower St., London WC1E 6BT, UK
e-mail: chrish@stats.ucl.ac.uk

Published online: 22 January 2010

$$f(\mathbf{x}) = \sum_{j=1}^{s} \pi_j \varphi_{\mathbf{a}_j,\Sigma_j}(\mathbf{x}), \tag{1}$$

where $\pi_j > 0\ \forall j,\ \sum_{j=1}^{s} \pi_j = 1,\ \varphi_{\mathbf{a},\Sigma}$ is the density of the p-dimensional Gaussian distribution $\mathcal{N}(\mathbf{a}, \Sigma)$ with mean vector $\mathbf{a}$ and covariance matrix Σ. Given a fixed s, the parameters can be estimated by Maximum Likelihood using the EM algorithm. The data points can then be classified to the mixture components by maximizing the estimated a posteriori probability that $\mathbf{x}_i$ was generated by mixture component j,

$$\hat{P}(\gamma_i = j|\mathbf{x}_i = \mathbf{x}) = \frac{\hat{\pi}_j \varphi_{\hat{\mathbf{a}}_j,\hat{\Sigma}_j}(\mathbf{x})}{\sum_{l=1}^{s} \hat{\pi}_l \varphi_{\hat{\mathbf{a}}_l,\hat{\Sigma}_l}(\mathbf{x})}, \tag{2}$$

where γ_i is defined by the two-step version of the mixture model where

$$P(\gamma_i = j) = \pi_j, \quad \mathbf{x}_i|(\gamma_i = j) \sim \varphi_{\mathbf{a}_j,\Sigma_j}, \quad i = 1, \ldots, n, \text{ i.i.d.} \tag{3}$$

A standard method (though not the only one) to estimate the number of components s is the Bayesian Information Criterion (BIC, Schwarz 1978), see Fraley and Raftery (2002) for details.

In order to restrict the number of parameters, various constraints on the Σ_j can be chosen. The BIC can be used to select an optimal model for the covariance matrices (Fraley and Raftery 2003). Note that some of these models, including the unconstrained one, have an issue with a potentially unbounded likelihood function if an eigenvalue of a covariance matrix is allowed to converge to zero. These issues are ignored here, because the introduced methodology can be applied to the outcome of the EM algorithm, which is a (hopefully) suitable finite local optimum of the likelihood under any covariance matrix model. For the present paper, the Gaussian mixture model has been fitted using the default options of the add-on package MCLUST version 3 (see Fraley and Raftery, Technical Report no. 504, Department of Statistics, University of Washington 2006) of the statistical software R (www.R-project.org). This is denoted by EM/BIC in case that the number of components was estimated by the BIC, and EM/s in case that it was fixed as s.

In cluster analysis usually every mixture component is interpreted as a cluster, and pointwise maximization of (2) defines the clustering. The idea is that a mixture formalizes that the underlying distribution is heterogeneous with several different populations, each of which is modelled by a homogeneous Gaussian distribution. Keeping in mind that there is no unique definition of a “true cluster”, and not necessarily assuming that the Gaussian mixture model assumption holds precisely, it could be said that this method employs the Gaussian distribution as the prototype shape of clusters to look for.

From a practical point of view, perhaps the most important problem with this approach is that for most applications Gaussian distributions are too restricted to formalize the cluster shapes one is interested in. For example, mixtures of two (or more) Gaussian distributions can be unimodal, and in such distributions there is no gap (and in this sense no separation) between the different Gaussian subpopulations.

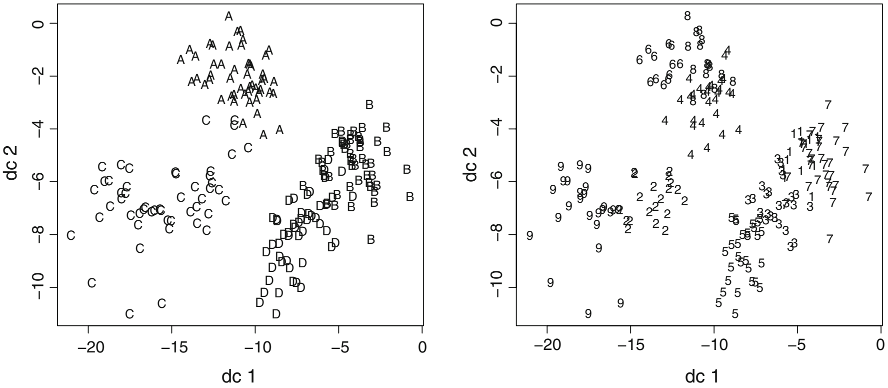

Fig. 1 Fisher's discriminant coordinates for crabs data based on MCLUST clustering. In the *left* plot, the true classes are shown. *A* and *B* denote male and *C* and *D* female crabs; the two subspecies are *A*/*C* and *B*/*D*. The numbers in the *right* plot denote the MCLUST clusters

In many applications where the number of clusters is not known, the EM algorithm together with the BIC yields a larger optimal number of mixture components than what seems to be a reasonable number of clusters when looking at the data.

An example for this is the crabs dataset (Campbell and Mahon 1974) available in the R-package MASS (www.R-project.org). The dataset consists of five morphological measurements (frontal lobe size, rear width, carapace length, carapace width, body depth; all in mm) on 200 crabs, 50 each of two color forms (subspecies) and both sexes, of the species *Leptograpsus variegatus* collected at Fremantle, W. Australia. Therefore, four classes are pre-defined in this dataset and the original intention behind collecting it was not clustering, but it makes sense to use a dataset with known true classes for illustration. Note that this information alone does not automatically mean that the correct number of clusters is four because the true classes themselves may be heterogeneous so that there may be more clusters to find, and on the other hand some true classes may not be distinguishable or at least not separable properly, so that it may also be possible to argue that there are fewer than four "true clusters". However, from the given information at least the straightforward expectation is that there should be four clusters.

EM/BIC estimates $s = 9$ clusters with equal covariance matrices for these data (note that nine is the default maximum for s in MCLUST, but increasing this to 20 still yields $s = 9$). Figure 1 shows the best two Fisher's discriminant coordinates derived from the nine clusters. Taking into account these (and some not shown) graphical diagnostics, it seems hardly reasonable to postulate as many as nine clusters in this dataset.

This does not mean that the MCLUST result is wrong in any sense and that there is any evidence for a Gaussian mixture with fewer than nine components. Nine Gaussian components are needed to fit these data according to the BIC, even with some distance to the second best solution. In terms of fitting the underlying density, this solution is much better than fitting the data with four Gaussian components. It only illustrates that it can be misleading to identify the number of Gaussian components with the number of clusters.

The present paper is about methods to decide whether and which Gaussian mixture components should be merged in order to interpret their union as cluster. At least some of the methods introduced later produce four clusters by merging the components 1/3/5, 2/9 and 4/6/8 while component 7 remains a cluster on its own. This is the best possible solution to the merging problem in terms of minimizing the misclassification rate for the four true classes, given the estimated nine-component Gaussian mixture.

The merging problem has drawn some attention recently in the literature. Tantrum et al. (2003) suggested some graphical diagnostics and proposed a hierarchical method to merge Gaussian components based on the dip test for unimodality (Hartigan and Hartigan 1985).

The hierarchical principle for merging Gaussian components works as follows:

1. Start with all components of the initially estimated Gaussian mixture as current clusters.
2. Find the pair of current clusters most promising to merge.
3. Apply a stopping criterion to decide whether to merge them to form a new current cluster, or to use the current clustering as the final one.
4. If merged, go to 2.

Two criteria are needed, namely in step 2 a rule to find the pair of clusters best to merge and the stopping rule in step 3. Tantrum, Murua and Stuetzle use the minimum difference in log-likelihood (assuming that a single Gaussian component is used to fit the merged mixture) in step 2 and a significant rejection of unimodality by the dip test as the stopping rule in step 3.

All of the methods suggested in the present paper use the hierarchical principle as well, because all of the criteria proposed later can either only with a heavy computational burden, or not at all, be applied to sets of more than two current clusters at a time.

Li (2004) suggested two non-hierarchical methods to fit what she calls a “multilayer mixture” (mixture of Gaussian mixtures). One of the methods is to apply k-means clustering to the s component means, the other one is based on the classification maximum likelihood. For these methods, the number of clusters k has to be fixed, though the number s of Gaussian components can be estimated. Fixing k, however, means that these methods cannot solve the problem of deciding whether two or more Gaussian components should be merged or not, and what the final number of clusters should be, which is of interest in the present paper.

Other papers hint at the merging problem without treating it explicitly. Biernacki et al. (2000) argue that the number of Gaussian components estimated by the BIC may not be the optimal number for clustering, which is smaller in many situations. To take this into account, they proposed ICL, a different criterion to estimate the number of mixture components. Related ideas are more directly applied to the merging problem in a recent Technical Report by Baudry et al. (2008). Ray and Lindsay (2005) investigate the topology of Gaussian mixtures with a view on modality, mentioning in their discussion that their results could be used for deciding whether Gaussian components may be merged. The related idea of modelling classes in discriminant analysis by a mixture of more than one Gaussians has been around in the literature for a bit longer (Hastie and Tibshirani 1996). Ueda et al. (2000) use a component merging and splitting algorithm on Gaussian mixtures to find a better local maximum of the mixture likelihood for fixed s.

In Sect. 2 of the present paper, the nature of the merging problem is discussed and the need for a decision about the cluster concept of interest is emphasized. In Sect. 3, two methods based on modality diagnostics are introduced, one based on the paper of Ray and Lindsay (2005), the other one, which is a modification of the suggestion of Tantrum et al. (2003), based on the dip test. In Sect. 4, three methods based on misclassification probabilities (modelling a cluster concept different from unimodality) are introduced. The first one is based on the Bhattacharyya distance (Fukunaga 1990), the second one on directly estimating misclassification probabilities, and the third one on the prediction strength approach to estimate the number of clusters of a general clustering method (Tibshirani and Walther 2005). In Sect. 5 a new graphical diagnostic is introduced. The different methods are compared by a simulation study in Sect. 6, illustrating that the choice of method should depend on the desired cluster concept. Another simulation sheds some light on the choice of the involved tuning constants. The crabs data are revisited in Sect. 7 along with the Wisconsin cancer data, another real data example. A concluding discussion is given in Sect. 8.

2 The nature of the problem

The merging problem looks as follows. Given a Gaussian mixture with s components as below, find $k \leq s$ and mixtures $f_1^*, \ldots, f_k^*$ of components of the original mixture so that each original Gaussian component appears in exactly one out of $f_1^*, \ldots, f_k^*$, and

$$f(\mathbf{x}) = \sum_{i=1}^{s} \pi_i \varphi_{\mathbf{a}_i, \Sigma_i}(\mathbf{x}) = \sum_{j=1}^{k} \pi_j^* f_j^*(\mathbf{x}), \tag{4}$$

where π_j^* is the sum of the π_i of the Gaussian components assigned to f_j^*. For datasets, clustering can be done by maximizing estimated posterior probabilities

$$\hat{P}(\gamma_i^* = j | \mathbf{x}_i = \mathbf{x}) = \frac{\hat{\pi}_j^* \hat{f}_j^*(\mathbf{x})}{\sum_{i=1}^{s} \hat{\pi}_i^* \hat{f}_i^*(\mathbf{x})}, \tag{5}$$

by analogy to (2) with $\gamma_1^*, \ldots, \gamma_n^*$ defined by analogy to (3). Estimators $\hat{\pi}_j^*$ can be obtained straightforward by summing up the $\hat{\pi}_m$ of the Gaussian member components of the mixture of mixtures j.

From (4), however, $f_1^*, \ldots, f_k^*$ are not identifiable. Imagine $s = 3$ and $k \leq 3$. In terms of the density and therefore the likelihood, it does not make a difference whether f_1^* is a mixture of the first two Gaussian components and f_2^* equals the third one, or f_1^* mixes the first and the third Gaussian component and f_2^* equals the second one, or any other admissible combination.

From a purely frequentist perspective, modelling f as the underlying data generating distribution and the $(\pi_i, \mathbf{a}_i, \Sigma_i),\ i = 1, \ldots, s$, as fixed parameters, there is no

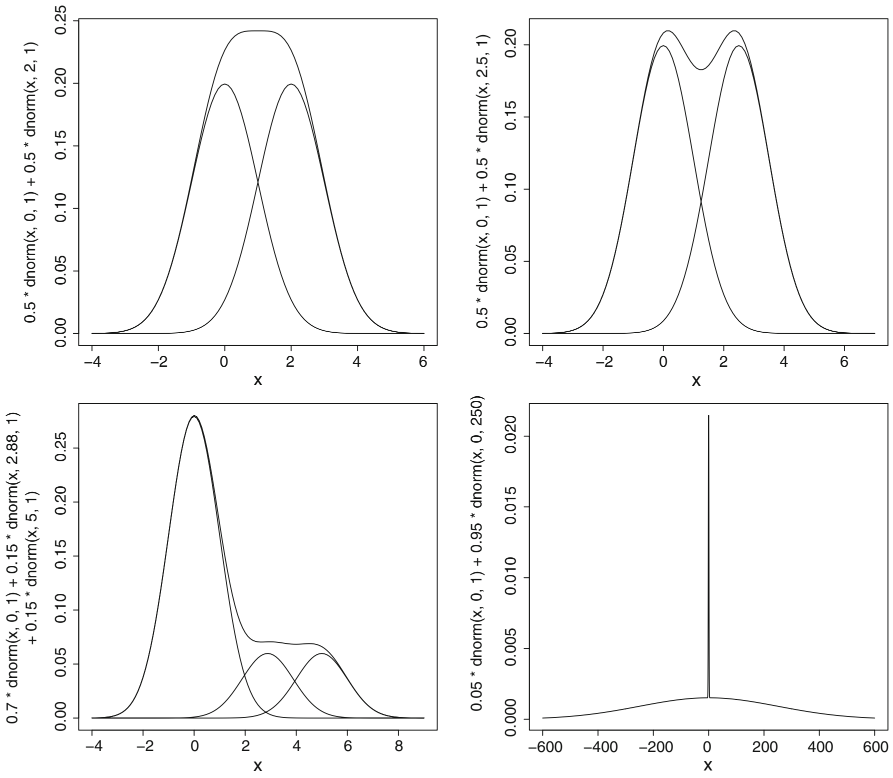

Fig. 2 Four one-dimensional Gaussian mixtures

statistical way to distinguish "true" cluster mixtures $f_1^*, \ldots, f_k^*$ from any others. It rather has to be decided by the statistician under which conditions different Gaussian mixture components should be regarded as a common cluster. This cannot be estimated from the data, but needs to take into account the (usually subject-matter dependent) cluster concept of interest. It is well known that there is no objective unique definition of what a "true cluster" is, so there is necessarily a subjective component in this decision.

The situations in Fig. 2 illustrate that essentially different cluster concepts may be of interest. Particularly the role of unimodality may be controversial. Some researchers may find it intuitive to identify a cluster with a set of points surrounding a density mode, and in most situations the unimodal mixture at the top left of Fig. 2 may be regarded as generating a single cluster (except if there are strong reasons to believe that "true clusters" should at least be approximately Gaussian, be mixtures unimodal or not, i.e., not demanding any "separation" between clusters). However, in some applications the unimodal mixtures at the bottom of Fig. 2 may not be regarded as a single cluster, because the modes in these examples are surrounded by dense "patterns" of the data that seem to be separated from what goes on in the tails, which is caused by other Gaussian components. But it is not clear that these mixtures in any case should

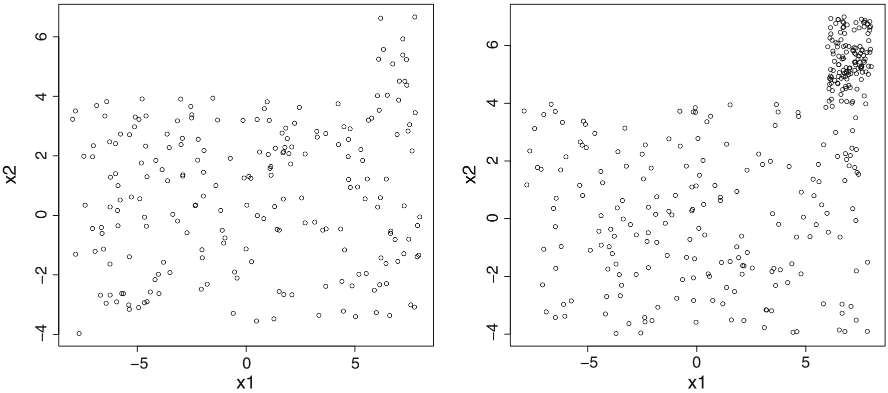

Fig. 3 Data from two unimodal mixtures of two uniform distributions

not be merged into a single cluster, because there is no separating "gap" between them, which may be required to speak of "clusters". Another illustration of the ambiguity is given in Fig. 3. Both examples in that figure are generated by unimodal mixtures of uniforms, and in both cases it is not straightforward to decide whether there are one or two clusters. My experience is that most statisticians tend to see two clusters in the right plot (this could also be interpreted as one dense cluster and some systematic non-clustered noise) but just one in the left one. Note, however, that one-dimensional projections of the data onto the line between the cluster means are bimodal in both cases.

On the other hand, multimodal mixtures may also be accepted as single clusters if the modes are not properly separated as in the upper right plot in Fig. 2. Note that the true density gap between the two apparently not very strongly separated clusters on the left side of Fig. 6 is much stronger than the one on the upper right side of Fig. 2. Note also that ML-estimation of Gaussian mixtures applied to data generated from uniform distributions such as the mixture components in Fig. 3 tends to come up with multimodal Gaussian mixtures. I even presume that under some suitable conditions the best Gaussian mixture approximation with any given finite s to a single uniform distribution in - not only - Kullback-Leibler sense is not unimodal.

Therefore, in order to define a suitable method for merging normals, the statistician has to decide

- whether only gaps in the density are accepted as separation between different clusters ("modality based cluster concept") or whether a dense data subset around a mode should be separated from clearly less dense data subsets even if the latter cannot be assigned to another mode ("pattern based cluster concept"),
- how strong the separation between different clusters should at least be (regardless of which of the two concepts is chosen, though the meaning of "separation" differs between them to some extent),
- what the role of the number of points in a cluster is, i.e., how strongly "cluster-shaped" small data subsets should be in order to constitute a cluster on their own.

The present paper offers a range of methods to deal with various decisions in these respects.

Obviously, the decisions have to depend on the requirements of the practical situation. For example, in a medical setup, a group of patients that could be seen as a cluster of interest may be characterized by a large variance of a measurement without necessarily differing too much from another more homogeneous cluster (“measurements in the normal healthy range”) on average, so that the pattern based cluster concept is needed to separate them. Another example for such a situation are quasars in astronomy. On the other hand, for example in biology it may be of interest to find communities such as species or subspecies that are separated by genetic or morphological gaps. Such applications rather need the modality based cluster concept.

The cluster concept in Tantrum et al. (2003) paper is obviously modality based. Li (2004) does not discuss the underlying cluster concept of her paper, but applying k-means to cluster means cannot separate scale mixtures.

Further considerations regarding the cluster concept of interest concern the question whether large within-cluster distances or small between-cluster distances should be allowed and whether the possible shapes of clusters should be restricted. In the present paper, however, large within-cluster or small between-cluster distances will always be allowed and there is no explicit restriction of cluster shapes. It may be argued that if different cluster concepts in these respects are of interest, Gaussian mixtures (with flexible covariance matrices) may not be a suitable basis anyway and other clustering methods (e.g., complete linkage or mixtures of other families of distributions) should be used.

An important concern about merging Gaussian mixture components is whether clustering should be based on a Gaussian mixture in the first place if clusters of interest are not necessarily Gaussian. An argument in favour of fitting and merging Gaussian mixtures is that the Gaussian mixture model with estimated k is a very versatile method to approximate any density (see Fraley and Raftery 2002). While this holds for mixtures of other parametric families as well, using a Gaussian prototype for “homogeneity” often at least makes sense in order to not separate data points that are estimated as belonging to the same Gaussian component, even though several Gaussian components may be merged. Furthermore, most clustering methods that are not based on mixture distributions have difficulties to find patterns with different covariance structures that are not separated by gaps such as in scale mixtures. Gaussian mixtures with a flexible enough covariance matrix model are best for this task among the most popular clustering methods.

An important reason for the problem that often more Gaussian mixture components are estimated than there are “true clusters” in the data is that the true clusters could actually be non-Gaussian (though homogeneous in some sense). In these cases, a mixture with more than one Gaussian component is needed to approximate homogeneous non-Gaussian distributions such as the uniform or Gamma. This implies that in applications where such distributions should be identified with a cluster, it may be needed to merge several Gaussian component to interpret them as a single cluster.

However, a true Gaussian mixture with a large enough number of components cannot be distinguished by the data from a homogeneous non-Gaussian distribution that it approximates properly, and therefore the issue that a mixture of several estimated

Gaussian components may be interpreted as a single cluster does not only arise because the Gaussian mixture assumption is questionable. The flexibility of Gaussian mixtures actually means that the question whether the Gaussian mixture assumption is really fulfilled is beyond observability and therefore meaningless, but in the context of cluster analysis it is meaningful to decide whether a (sub)mixture of Gaussians as a whole is homogeneous or not.

A referee noted that when fitting mixtures of t-distributions (as discussed in Sect. 7 of McLachlan and Peel 2000) often only a single component suffices for fitting subsets of the data for which a Gaussian scale mixture with two or more components would be needed. This means that identifying clusters with t-mixture components corresponds to a different cluster concept than identifying clusters with Gaussian mixture components, and some unimodal clusters that need to be fitted by more than one Gaussian component occur naturally as single t-components (on the other hand, if the interest is in a pattern based cluster concept, it may be desired to keep scale mixtures separated). However, mixtures of two or more t-distributions can be unimodal as well, so if clusters are associated with modes, merging would still be needed for mixture components of t-distributions.

3 Methods based on modality

3.1 The ridgeline unimodal method

Under a strong version of the modality based cluster concept, the "strong modality merging problem" is to find a partition of the mixture components so that all resulting clusters are unimodal but any further merging would result in a cluster that is no longer unimodal. This requires an analysis of the modality of Gaussian mixtures. The most advanced paper on this topic, to my knowledge, is Ray and Lindsay (2005). They showed that for any mixture f of s Gaussian distributions on $\mathbb{R}^p$ there is an $(s-1)$-dimensional manifold of $\mathbb{R}^p$ so that all extremal points of f lie on this manifold.

For $s = 2$, this manifold is defined by the so-called "ridgeline",

$$\mathbf{x}^*(\alpha) = [(1-\alpha)\Sigma_1^{-1} + \alpha\Sigma_2^{-1}]^{-1}[(1-\alpha)\Sigma_1^{-1}\mathbf{a}_1 + \alpha\Sigma_2^{-1}\mathbf{a}_2], \tag{6}$$

and all density extrema (and therefore all modes, which may be more than 2 in some situations) can be found for $\alpha \in [0, 1]$.

Unfortunately, for $s > 2$, Ray and Lindsay's result does not yield a straightforward method to find all nor even the number of modes. Therefore, their results can in general only be used to solve the strong modality merging problem approximately. The **ridgeline unimodal method** is defined by the hierarchical principle as follows.

1. Start with all components of the initially estimated Gaussian mixture as current clusters.
2. Using the mean vectors and covariance matrices of the current clusters (initially the Gaussian components), for any pair of two current clusters use the 2-component Gaussian mixture derived from these parameters on the ridgeline (6) to check whether it is unimodal.

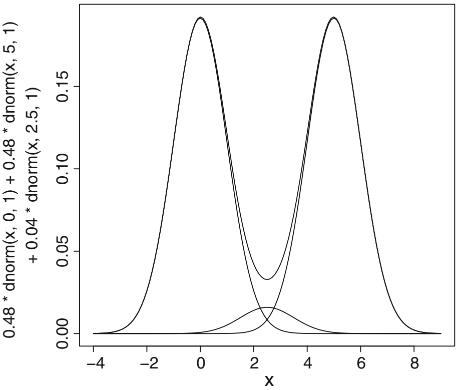

Fig. 4 Bimodal Gaussian mixture in which the mixtures of both the *left* and the *right* component with the *middle* one are unimodal

3. If none of these is unimodal, use the current clustering as the final one.
4. Otherwise,
   (a) merge all of the pairs leading to unimodal mixtures and consisting of current clusters only involved in a single unimodal mixture,
   (b) if there are cliques of more than two current clusters so that every pair of them leads to a unimodal mixture not involving any other current cluster, merge them all,
   (c) in situations where a current cluster is involved in different unimodal mixtures with other current clusters of which further pairwise mixtures are not unimodal, merge only the unimodal pair with the closest mean vectors (see Fig. 4 for an illustration that it is not sensible to merge all current clusters involved in intersecting unimodal pairs in such a case),
   (d) go to step 2.

Approximation enters in two ways. First, instead of checking unimodality of mixtures of more than two current clusters at a time, clusters are merged hierarchically. Second, current clusters consisting of a mixture of two or more Gaussians are treated as single Gaussians in step 2, by plugging their mean vectors and covariance matrices into Ray and Lindsay's theory for pairs of Gaussian components. This may in some rare situations lead to assessments of unimodality that are wrong if applied to the resulting Gaussian mixture of more than two components.

In order to apply this principle to data, means and covariance matrices are replaced by their ML-estimators for Gaussian mixture components. For mixtures of two or more Gaussians appearing as current clusters in the hierarchy, mean vectors and covariance matrices can be computed using the weights of points in the current cluster computed by summing up the weights (2) for all involved mixture components (dividing by the sum of all involved weights corresponding to ML for weighted Gaussian data).

3.2 The ridgeline ratio method

Even if the cluster concept is modality based, in some situations the statistician may want to allow clusters that deviate from unimodality as long as the gap between the modes is not strong enough to interpret them as belonging to two separated clusters. One reason for this is that clusters may be required to be strongly separated. Another reason is that, as a result of too small sample size or particular instances of non-normality, data from unimodal underlying distributions may be approximated by EM/BIC by a multimodal Gaussian mixture. As the simulations in Sect. 6.2 reveal, the latter is by no means exceptional.

A straightforward method to deal with this is to replace the demand of unimodality in the previous session by a cutoff value r^* for the ratio r between the minimum of the mixture density f along the ridgeline (6) and the second largest mode in case that there is more than one (note that because the ridgeline is defined only for $\alpha \in [0, 1]$, the density minimum does not become arbitrarily small).

The reason for choosing the second largest mode is that if two Gaussian components are mixed, to measure the “gap” between the two, the minimum has to be compared with the smaller of the modes corresponding to the two mixture components, but not taking into account even smaller modes in those special cases in which there are more than two modes; see Ray and Lindsay (2005) for examples of such situations. Here is the resulting **ridgeline ratio** algorithm:

1. Choose a tuning constant $r^* < 1$.
2. Start with all components of the initially estimated Gaussian mixture as current clusters.
3. Using the mean vectors and covariance matrices of the current clusters, for any pair of current clusters use the 2-component Gaussian mixture derived from these parameters on the ridgeline (6) to compute r.
4. If $r < r^*$ for all pairs of current clusters, use the current clustering as the final one.
5. Otherwise, merge the pair of current clusters with maximum r (in case of ties, proceed by analogy to the ridgeline unimodal method) and go to step 3.

The cutoff value r^* formalizes the degree of separation between clusters. If in a practical situation clusters are required to be strongly separated, r^* has to be chosen by subject-matter considerations and subjective decisions. If, on the other hand, the ridgeline ratio method is rather preferred to the ridgeline unimodal method because the statistician wants to keep the probability low of splitting up data by chance that are generated by unimodal underlying distributions, the choice of r^* could be guided by simulating data from borderline unimodal distributions, see Sect. 6.1.

3.3 The dip test method

As explained in the introduction, Tantrum et al. (2003) defined a hierarchical merging algorithm for the modality based cluster concept, the stopping rule of which is a sufficiently small p-value of the Hartigan and Hartigan (1985) dip test for unimodality. To use a significance test for unimodality is intuitively appealing if the statistician wants to merge components if the resulting mixture cannot be statistically distinguished from a unimodal distribution. However, Tantrum, Murua and Stuetzle’s use of the minimum difference in log-likelihood (assuming that a single Gaussian component is used to fit the merged mixture) to find the pair of current clusters to be tested for unimodality seems to be somewhat inconsistent, because it is not directly related to the modality based cluster concept. Furthermore, the Gaussian log-likelihood can only be expected to be a useful measurement of the goodness of fit of a merged mixture if the mixture is sufficiently close to a single Gaussian which cannot be expected to hold in many situations in which it is desired to merge components, such as for example shown in the left panel of Fig. 3.

Therefore, as a modification of Tantrum, Murua and Stuetzle's method, I suggest to replace the log-likelihood difference by the ridgeline ratio r defined in Sect. 3.2. Here is the proposed **dip test method** in more detail:

1. Choose a tuning constant $p^* < 1$.
2. Start with all components of the initially estimated Gaussian mixture as current clusters.
3. Using the mean vectors and covariance matrices of the current clusters, use the 2-component Gaussian mixture derived from these parameters on the ridgeline (6) to compute r for any pair of current clusters.
4. Consider the data subset $\mathbf{x}^*$ of points classified to the union of the pair of current clusters maximizing r by maximizing (5).
5. Let $\mathbf{x}^{*1}$ be the projection of $\mathbf{x}^*$ onto the discriminant coordinate based on the pooled covariance matrix of the two involved current clusters, separating the two current cluster means (this is necessary because the dip test operates on one-dimensional data).
6. If the p-value of the dip test applied to $\mathbf{x}^{*1}$ is $\leq p^*$, use the current clustering as the final one.
7. Otherwise merge this pair of current clusters and go to step 3.

The dip statistic is defined as the Kolmogorow distance between the empirical distribution and the closest unimodal distribution. Hartigan and Hartigan (1985) suggested to compute p-values from a uniform distribution. These can be obtained from the R-package DIPTEST. Tantrum, Murua and Stuetzle obtained their p-values differently from simulations from the closest unimodal distribution (which I call "simulation version of the dip test"). I recommend Hartigan and Hartigan's original proposal because it is computationally much more demanding to carry out data dependent simulations every time. I found results from the "simulation version" often to be unstable in simulated situations from Sect. 6.2, though it may perform better in situations where the method based on Hartigan and Hartigan's original suggestion merges too strongly.

Two issues with the dip test method remain with the present proposal as well as with Tantrum, Murua and Stuetzle's original one. The first one is that there are multiple tests and they are chosen in a data dependent way (determined by the initial result of EM/BIC), so that the resulting p-values violate the assumptions of a standard statistical hypothesis test. This means that their usual interpretation is invalid. It would only be valid under the assumption that only the data belonging to the sub-mixture tested for unimodality in the current step were observed. Keeping the violation of this assumption in mind, the p-values should still be a suitable (ordinal) measurement of the strength of evidence against unimodality for sub-mixtures, so that the multiple testing issue can be tolerated in the framework of the merging problem, though it is not straightforward to choose a suitable value for p^*.

The second one is that linear one-dimensional projections of unimodal data may be multimodal. To see this, consider again a unimodal two-dimensional uniform distribution like the one generating the data in the left panel of Fig. 3. Imagine the data as partitioned into two clusters, one consisting of all the points with $x_{i2} > 4$, the other one with all other points. The distribution of the projection of the data onto the line connecting the two mean vectors is bimodal; there is no unimodal way to reflect the L-shape of the data when projecting onto a single dimension. Therefore the dip test method is based on a cluster concept based on unimodality of one-dimensional projections. It is up to the statistician to decide whether this is acceptable in a practical situation.

An alternative to the use of the ridgeline ratio for the decision which pair of current clusters is to be tested for merging is to carry out the dip test for every pair of current clusters and to merge the pair with the highest p-value first. I favour the ridgeline ratio because I do not think that it is meaningful to say that of two high p-values such as 0.9 and 0.7, the larger one indicates a clearly stronger support of unimodality.

4 Methods based on misclassification probabilities

The methods introduced in this section formalize versions of the pattern based cluster concept as opposed to the modality based one. Misclassification probabilities provide an intuitive possibility to formalize separation between different clusters in a different way than density gaps. For example the two components of the scale mixture on the lower right side of Fig. 2 are not separated in the sense that there are no gaps between them, but nevertheless the misclassification probability between them is low. Obviously, the misclassification probability would be low as well in case of a strong density gap between components, so that in many clear cut situations both concepts arrive at the same clustering.

4.1 The Bhattacharyya distance method

The Bhattacharyya distance is a general distance between two distributions related to the overall Bayes misclassification probability for the 2-class problem with arbitrary class probabilities. This is bounded from above by $\exp(-d)$, where d is the Bhattacharyya distance (Matusita 1971). For two Gaussian distributions with mean vectors and covariance matrices $\mathbf{a}_j, \Sigma_j, j = 1, 2$, the Bhattacharyya distance is (Fukunaga 1990)

$$d = \frac{(\mathbf{a}_1 - \mathbf{a}_2)^t \bar{\Sigma}^{-1}(\mathbf{a}_1 - \mathbf{a}_2)}{8} + \frac{1}{2}\log\left(\frac{|\frac{1}{2}(\Sigma_1 + \Sigma_2)|}{\sqrt{|\Sigma_1||\Sigma_2|}}\right). \tag{7}$$

For data, the parameters can of course be replaced by their estimators.

The Bhattacharyya distance between two mixtures of Gaussians cannot be computed in a straightforward way. Therefore, for hierarchical merging, I again suggest to represent mixtures of Gaussians by their overall mean vector and covariance matrix and in this sense to treat them as single Gaussians. The **Bhattacharyya distance method** looks as follows:

1. Choose a tuning constant $d^* < 1$.
2. Start with all components of the initially estimated Gaussian mixture as current clusters.
3. Compute the (in case of mixtures with more than one component approximately) estimated Bhattacharyya distances d between all pairs of current clusters from their mean vectors and covariance matrices.
4. If $\exp(-d) < d^*$ for all pairs of current clusters, use the current clustering as the final one.
5. Otherwise, merge the pair of current clusters with maximum d and go to step 3.

Here d^* formalizes the degree of separation between clusters. By analogy to r^* above, d^* could be chosen by subject matter considerations or by simulations from borderline distributions, see Sect. 6.1.

Note that $\exp(-d)$ can overestimate the misclassification probability quite severely (over-estimation tends to be stronger for location mixtures with equal covariance matrices than for scale mixtures with equal means), but d can be interpreted as a useful dissimilarity measurement between Gaussian distributions in its own right, linked to the Mahalanobis distance and Fisher's discriminant function in case of equal covariance matrices, see Fukunaga (1990).

Equation (7) does not take into account the component probabilities. This differs from all the other methods proposed in the present paper as well as from the theoretical misclassification probabilities between the two involved distributions. This means that the cluster concept implied by the Bhattacharyya distance method separates data subsets with different enough mean vectors and/or covariance matrices even if one of them contains much fewer points than the other one. A mixture component with very few points may remain unmerged under this approach even if another, larger, mixture component (with quite different mean vector and/or covariance matrix, though) generated relatively many points close to the domain of the former component.

4.2 Directly estimated misclassification probabilities

Instead of estimating the Bhattacharyya distance, misclassification probabilities $p_{ij} = P(\tilde{\gamma}_1^* = i | \gamma_1^* = j) = \frac{P(\tilde{\gamma}_1^* = i,\ \gamma_1^* = j)}{\pi_j^*}$ between components of a mixture distribution can also be estimated directly from the results of the EM algorithm. Here γ_1^* denotes the mixture component number that generated the first data point (or any other point according to the i.i.d. assumption, as long as only probabilities are of interest), and $\tilde{\gamma}_1^*$ is the mixture component to which the point is classified by maximizing the population version of (5), i.e., by the Bayes rule with true parameters. $1(\bullet)$ denotes the indicator function.

π_j^* can be estimated by $\hat{\pi}_j^*$. Note that

$$\hat{P}(\tilde{\gamma}_1^* = i,\ \gamma_1^* = j) = \frac{1}{n} \sum_{h=1}^{n} \hat{P}(\gamma_h^* = j | \mathbf{x}_h) 1(\hat{\gamma}_h^* = i) \tag{8}$$

is a consistent estimator of $P(\tilde{\gamma}_1^* = i,\ \gamma_1^* = j)$, where $\hat{\gamma}_h^*$ denotes the data based classification of data point $\mathbf{x}_h$, estimating $\tilde{\gamma}_h^*$, by maximizing (5), which also defines $\hat{P}(\gamma_h^* = j|\boldsymbol{x}_h)$.

Therefore,

$$\hat{p}_{ij} = \frac{\hat{P}(\tilde{\gamma}_1^* = i,\ \gamma_1^* = j)}{\hat{\pi}_j^*}$$

is a consistent estimator of p_{ij}. This works regardless of whether the mixture components are Gaussian distributions or mixtures of Gaussians. Therefore it is not needed to represent mixtures by their mean vectors and covariance matrices in order to compute $\hat{p}_{ij}$. The method of directly estimated misclassification probabilities (**DEMP method**) below therefore does not treat mixtures of Gaussians as single Gaussians in any way.

1. Choose a tuning constant $q^* < 1$.
2. Start with all components of the initially estimated Gaussian mixture as current clusters.
3. Compute $q = \max(\hat{p}_{ij}, \hat{p}_{ji})$ for all pairs of current clusters.
4. If $q < q^*$ for all pairs of current clusters, use the current clustering as the final one.
5. Otherwise, merge the pair of current clusters with maximum q and go to step 3.

It is more reasonable to work with $\max(\hat{p}_{ij}, \hat{p}_{ji})$ than with the estimated overall misclassification probability $\hat{\pi}_j^* \hat{p}_{ij} + \hat{\pi}_i^* \hat{p}_{ji}$, because it makes sense to merge if just one of the two misclassification probabilities is large; the other one may only be small because one of the current clusters is much larger than the other one and therefore misclassifications from the former one into the latter one may be unlikely regardless of the separation between them. The situation is somewhat different from the related discussion in Sect. 4.1, where in case of two current clusters of very different size (ignored by the Bhattacharyya distance) it is at least required that their estimated parameters are similar enough in order to merge them.

While asymptotically correct, for finite samples $\hat{p}_{ij}$ is a somewhat optimistic estimator of p_{ij} because it equates $\hat{\gamma}_h^*$ with $\tilde{\gamma}_h^*$, implying that $\hat{P}(\gamma_h^* = j|x_h)$ has to be small enough to allow (5) to be maximized by the "correct" mixture component i, and assuming further that EM/BIC did not get the initial Gaussian mixture too wrong. The effect of this bias can be quite severe, and therefore, unless the dataset is very large and clusters are not expected to be small, q^* should be chosen smaller than the true misclassification probability the researcher is willing to accept between "separated clusters". Of course, simulating from a borderline distribution is again possible to obtain a suggestion for q^*.

Note that it would be cumbersome to compute the theoretical misclassification probabilities between Gaussian mixtures, and therefore to estimate them by plugging in estimated parameters.

4.3 The prediction strength method

The prediction strength approach by Tibshirani and Walther (2005) to estimating the number of clusters is based on a different concept of misclassification. Instead of assessing the classification of the data points to the clusters, it assesses how well it can be predicted whether pairs of points belong to the same cluster. Furthermore, instead of estimating the misclassification passively, by recomputing the clustering on subsamples, the approach does not only take into account the separation of the estimated clusters, but also the stability of the clustering solution.

The two-fold version of their original method, suggested for estimating the number of clusters k based on any chosen clustering method for fixed k, works as follows:

1. Choose a tuning constant $c^* < 1$ (Tibshirani and Walther 2005, suggest $c^* = 0.8$ or 0.9; the simulations presented later in this paper rather hint at $c^* = 0.75$ for the way the method is used here). For a suitable range of values of k, repeat m times:
2. Split the dataset in two halves. Cluster both parts.
3. Use the clustering $\mathcal{C}_1$ of the first half of the data to predict the cluster memberships (of clusters in $\mathcal{C}_1$) of the points of the second half of the data by assigning every point of the second half to the cluster in $\mathcal{C}_1$ with the closest mean vector.
4. For every cluster in the clustering $\mathcal{C}_2$ of the second half of the data, compute the proportion of correctly predicted co-memberships of pairs of points by the membership predictions of $\mathcal{C}_1$. Record the minimum over clusters $\tilde{c}$ of these proportions.
5. Repeat steps 3 and 4 exchanging the roles of the two halves.
6. Let c be the average of the $2m$ recorded values of $\tilde{c}$. Use the largest k with $c \geq c^*$ as the estimated number of clusters.

For the merging problem, a special version of the prediction strength method is required. The original method is based on a clustering algorithm for fixed k. This can be obtained for the merging problem by carrying out the DEMP method from Sect. 4.2, but instead of using $q < q^*$ as a stopping rule, clusters are merged until the number of k clusters is reached. Any other hierarchical method could be used as well, but the cluster concept of the DEMP method seems to be most compatible to the idea of prediction strength.

Furthermore, assigning points to the closest mean in step 3 of the original prediction strength method is not suitable for the pattern based cluster concept allowing scale mixtures, and therefore the classification should rather be done by maximizing (5). This leads to the following **prediction strength method** for the merging problem (assuming that $\hat{s}$ is the number of Gaussian mixture components estimated by EM/BIC):

1. Choose a tuning constant $c^* < 1$. For $k = 2, \ldots, \hat{s}$, repeat m times:
2. Split the dataset in two halves.
3. Cluster both halves as follows:
   (a) Apply EM/$\hat{s}$ (fixing the number of Gaussian components).
   (b) Apply the DEMP method to the solution, stopping at k clusters.

4. Use the clustering $\mathcal{C}_1$ of the first half of the data to predict the cluster memberships (of clusters in $\mathcal{C}_1$) of the points of the second half of the data by maximizing (5) for every point of the second half with respect to the mixture components in $\mathcal{C}_1$.
5. For every cluster in the clustering $\mathcal{C}_2$ of the second half of the data, compute the proportion of correctly predicted co-memberships of pairs of points by the membership predictions of $\mathcal{C}_1$. Record the minimum over clusters $\tilde{c}$ of these proportions.
6. Repeat steps 3 and 4 exchanging the roles of the two halves.
7. Let c be the average of the $2m$ recorded values of $\tilde{c}$. Use the largest k with $c \geq c^*$ as the estimated number of clusters.

The main problem with the prediction strength method is that clustering half of the data may be significantly less stable than clustering the whole dataset, so that c may be rather pessimistic. c^* could be chosen following Tibshirani and Walther's recommendation or by simulations from borderline distributions, see Sect. 6.1.

For some simulations (Sect. 6.1, Simple Uniform and Exponential setup in Sect. 6.2), $m = 25$ was chosen for computational reasons, otherwise $m = 50$. In real applications $m = 100$ or higher can be used; higher m improves the stability of the outcome.

5 Ordered posterior plots

There are several graphical tools to diagnose whether different mixture components as for example estimated by MCLUST are well enough separated in order to interpret them as clusters. Fisher's discriminant coordinates have already been used in Fig. 1. A problem with Fisher's discriminant coordinates is that it visualizes all differences between components means together in a single plot, which is usually not simultaneously optimal for each component and may therefore hide the separation of some components. The methods introduced in Hennig (2005) solve this problem by visualizing the separation of a single cluster at a time by finding suitable linear projections. An example is given on the bottom right of Fig. 5, illustrating that component 7 in the crabs data set (the only one that should not be merged with any other component in order to achieve the clustering with the minimal misclassification probability) is actually separated from the others, apart from a few observations that can be seen as somewhat outlying with respect to the component center.

Further graphical diagnostics for the merging problem were introduced by Tantrum et al. (2003). They showed component-wise rootograms of the estimated posterior probabilities and trees to visualize the merging process. It could also be helpful to visualize the "uncertainties" given out by the summary method in MCLUST.

A new diagnostic plot is proposed here. The ordered posterior plot is another possibility to visualize separation and overlap between components as indicated by the estimated posterior probabilities (2). This plot is again defined for each mixture component separately. For a given mixture component j, the points in the dataset are ordered according to their estimated posterior probability of having been generated by component j, i.e., the point with the highest $\hat{P}(\gamma_i = j|\mathbf{x}_i)$ gets rank 1 assigned and is therefore the most central point with respect to this component, and so on.

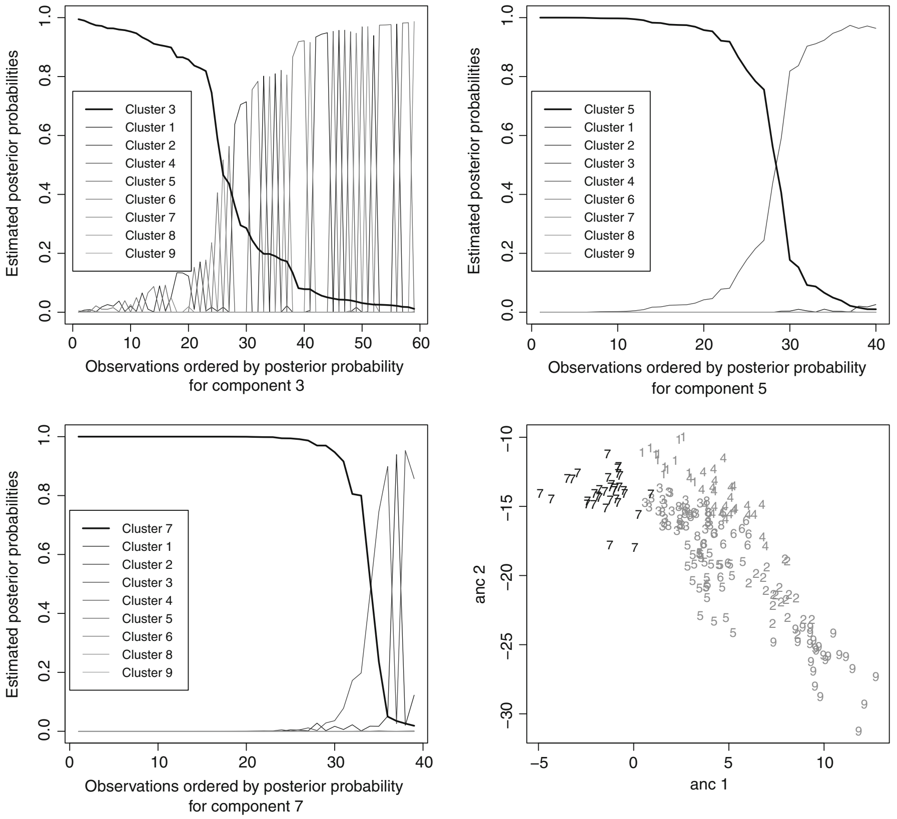

Fig. 5 *Top and bottom left*: ordered posterior plots for mixture components 3, 5 and 7 in the crab data. *Bottom right*: asymmetric neighborhood based coordinates plot for component 7, see Hennig (2005)

These ranks are plotted along the x-axis. It would be possible to use all points in the dataset, but those with a very small posterior probability for component no. j are quite uninformative, so only those observations with $\hat{P}(\gamma_i = j|\mathbf{x}_i) > 0.01$, say, are used (which therefore have the smallest ranks).

On the y-axis, $\hat{P}(\gamma_i = j|\mathbf{x}_i)$ is plotted using a bold line (in a particular color if this is desired). The posterior probabilities for the other components are plotted as well, but thinner, so that it can be seen how strongly component j is separated from the others in terms of the posterior probabilities. With colors it is possible furthermore to indicate which line belongs to which other component so that it can be seen which component is "most overlapping" to j (for illustration, in Fig. 5 different shades of grey have been used, which is not as good as colors). Three examples, for components 3, 5 and 7 in the crabs dataset, are shown in Fig. 5. From looking at all nine plots (not shown) it becomes clear that component no. 7 (together with 9) is the most and no. 3 the least well separated. No. 3 overlaps strongly with two other components (could be better seen with color). No. 5 is an example for a component that only noticeably overlaps with one other component. More examples are given in Fig. 7 in Sect. 7.2.

6 Simulations

6.1 Cutoff values

A possible approach to determine suitable cutoff values for the methods suggested above is to simulate data from "borderline distributions" that should "just" be regarded as a single cluster and to find out at what cutoff value a normal mixture fitted to data from these distributions would be merged into a single cluster in 95% of the cases, say. The results given in Table 1 have been obtained by simulating from the following two models, each defined for $p = 1, 2, 5, 10$ and three different values of n for each p, see Table 1:

1. The first variable is a borderline unimodal Gaussian mixture (any larger mean difference would lead to bimodality with these variances) $0.5 * \mathcal{N}(0, 1) + 0.5 * \mathcal{N}(2, 1)$,

Table 1 Cutoff values required so that in 95% of simulation runs all estimated mixture components would be merged

Distribution	p	n	ridgerat	dip	bhat	DEMP	predstr
Unimodal Gaussian mixture	1	50	0.413	0.187	0.201	0.043	0.830
	1	200	0.821	0.262	0.390	0.102	0.829
	1	500	0.940	0.360	0.490	0.124	0.868
	2	50	0.333	0.143	0.112	0.016	0.771
	2	200	0.926	0.270	0.450	0.119	0.798
	2	500	0.981	0.383	0.530	0.139	0.853
	5	100	0.890	0.154	0.395	0.107	0.729
	5	400	0.990	0.410	0.531	0.139	0.814
	5	1,000	0.998	0.326	0.559	0.149	0.860
	10	200	0.963	0.112	0.478	0.126	0.714
	10	400	0.990	0.235	0.529	0.138	0.773
	10	1,000	0.999	0.290	0.560	0.147	0.824
Uniform	1	50	0.123	0.000	0.070	0.000	0.859
	1	200	0.270	0.000	0.151	0.009	0.811
	1	500	0.303	0.000	0.113	0.016	0.771
	2	50	0.125	0.000	0.031	0.000	0.679
	2	200	0.293	0.000	0.124	0.023	0.664
	2	500	0.349	0.000	0.167	0.047	0.650
	5	100	0.142	0.000	0.012	0.000	0.827
	5	400	0.381	0.000	0.112	0.039	0.682
	5	1,000	0.437	0.000	0.205	0.066	0.637
	10	200	0.227	0.000	0.009	0.000	0.903
	10	400	0.375	0.000	0.040	0.013	0.791
	10	1,000	0.445	0.000	0.100	0.048	0.632

Methods: ridgeline ratio (ridgerat), dip test (dip), Bhattacharyya (bhat), DEMP, prediction strength (predstr). "0.000" means "< 0.0005"

as shown in the upper left of Fig. 2. If $p > 1$, the further variables are generated independently by homogeneous standard Gaussian distributions.

2. Uniform distribution on $[0, 1]^p$.

In order to save computing time, the number of components in MCLUST was fixed to be 2 for the normal mixture model and 6 for the uniform model (this is the median of the estimated cluster numbers for the uniform setup in Sect. 6.2). 200 simulation runs have been used for each combination of model, p and n except for the computationally more demanding prediction strength approach, for which 100 simulation runs have been used in each case with $m = 25$. For every simulation run and every method, merging was carried out until only one cluster remained, and the minimum value of the stopping criterion occurring during the merging process was recorded. In Table 1, the 5%-quantiles of these values over the simulation runs are given, except for the prediction strength method, where the (analogous) 95%-quantile of the maximum strength values is given. For the Bhattacharyya method, simulation results are given in terms of the upper bound $\exp(-d)$ on the misclassification probability instead of the Bhattacharyya distance d.

From Table 1 it can be seen that the Gaussian mixture leads to quite different values from the uniform. The Gaussian mixture is much "easier" to merge. This is not too surprising, because for a Gaussian mixture fulfilling the model assumptions perfectly, the EM/2 fit can be expected to be much better than piecing together the uniform out of six Gaussians (note that for large n even the 5%-quantile of ridgeline ratio comes very close to 1 for the Gaussian mixture and a unimodal ridgeline is estimated in a vast majority of simulation runs). Furthermore, only two components have to be merged. The only exception is that the prediction strength method merges the uniform more easily (as opposed to all other methods, for the prediction strength method a smaller value in the table indicates that merging is easier). The reason is that, because model assumptions are fulfilled for the Gaussian mixture, EM/2 yields more stable solutions than EM/6 for the uniform. It is particularly striking that when merging the uniform, the dip test method at least at some stage of the merging process almost always produces a very small p-value below 0.001.

Considering the Gaussian mixture with a theoretical misclassification probability of $Q = 0.159$, it can be seen that this is overestimated strongly by $\exp(-d)$ of Bhattacharyya but underestimated by DEMP. Note that the value 0.159 for this model is independent of p, while for finite n the classification problem is obviously more difficult for larger p including "noise" variables. This is properly reflected in the fact that the Bhattacharyya bound increases on average with p (for $n = 50$ there is more variation in the simulated values for larger p, so that the 5%-quantile for $p = 1$ is larger than for larger p), which gives it some justification apart from bounding the misclassification probability.

The values in the table (probably rather the more pessimistic ones from the uniform model) only make sense as cutoff values if the major aim is to have some confidence that estimated components for data from such borderline distributions will be merged in most cases. n in the table may be chosen as the number of points $\tilde{n}$ in the dataset to be analyzed, but if the statistician wants to safeguard against merging smaller data subsets generated from unimodal underlying components modelling only part of the data, alternatively a cutoff value could be determined by looking up the table for $\tilde{n}/k^*$ with some suitable $k^* > 1$.

For many if not most applications, however, it is rather of interest to estimate the number of clusters, accepting that some unimodal distributions may be split up in more than 5% of the cases, than to "always merge if in doubt". In such a situation the index values of the different methods may be interpreted directly in terms of their definition, and the researcher may want to have cutoff values independent of n and perhaps even of p. In such situations, the tabulated values at least can help with "translating" the cutoff values of the different methods in order to make the methods comparable. Analysing the ratios of these values, in the following sections the following cutoff values are used for ridgeline ratio, dip test, Bhattacharyya, DEMP and prediction strength, which may at least serve as a guideline at which ratios they can be interpreted as "merging about equally strongly" (note that some more experiments with the simulated setups were also taken into account fixing these, so that they can be interpreted to some extent as "default recommendations"):

$$r^* = 0.2, \quad p^* = 0.05, \quad d^* = 0.1, \quad q^* = 0.025, \quad c^* = 0.75. \tag{9}$$

For determining the dip test cutoff value p^* the results for the uniform model have been ignored because they would not lead to a value that could be reasonably interpreted. It is a general weakness of the dip test method that it cannot be prevented that there is significant multimodality for some sub-mixtures occurring in the fit of some unimodal distributions.

Table 1 also shows that not all methods behave in the same way with respect to changes in n and p. The ridgeline ratio method requires clusters to be much closer to estimated unimodality for larger p. If the "visible" separation between clusters is required to be fixed regardless of n and p, then it may make sense to increase the cutoff value for the ridgeline ratio with increasing p (though not necessarily with increasing n, because the increase of the entries in Table 1 with increasing n are mainly due to more variation in the estimated ridgeline ratio for smaller n). The DEMP estimate of the misclassification probability becomes less optimistic with increasing n and p. Keeping its cutoff value fixed over n means that for larger n clusters with the same misclassification probability may be merged slightly more easily, which may make some sense in some applications because gaps between clusters can be expected to be "filled" with more points. The behaviour of Bhattacharyya over p differs surprisingly strongly between the Gaussian mixture and the uniform setup, which is mainly due to higher variability in the uniform setup for larger p. The prediction strength approach needs larger cutoff values for smaller p. It is interesting that larger cutoff values are required for larger n for the unimodal Gaussian mixture, but for smaller n for the uniform distribution. This is probably due to the fact that even asymptotically there is no unique optimal approximation for a uniform distribution by a fixed number of Gaussians, whereas the partition of the unimodal Gaussian mixture is asymptotically stable.

If it is desired to change the values above but to keep their ratios fixed, keep in mind that c^* is derived from an upper quantile of maximum values, so $1 - c^*$, not c^* itself, is comparable to the other values (though even the ratio of $1 - c^*$ and the other cutoff values varies quite a bit in Table 1).

6.2 Some benchmark situations

Some benchmark situations were defined in order to compare the performance of the methods with cutoff values given in (9) and particularly in order to show how the performance depends on the data and the cluster concept.

Situations were designed to be challenging but simple, so that it is transparent how to interpret the results, that they could discriminate between the methods, and that one may imagine such situations to occur at least in subsets of real datasets.

The results given here only concern the number of clusters before and after merging, because in most situations it was clear and consistent throughout the simulations which clusters were found as long as the number was among the "correct ones" (in some situations more than one cluster number could be regarded as "correct" depending on the cluster concept).

For every setup there were 200 simulation runs. Every merging method was run on the MCLUST output for the simulated data. With some exceptions (see below), EM/BIC was used, with selection of the covariance matrix model by the BIC.

Here are the setups:

Simple uniform. 1,000 points ($p = 2$) were generated from a uniform distribution on $[0, 1]^2$. In this setup there should be one cluster pretty much regardless of the cluster concept. Results are given in Table 2. EM/BIC was applied with fully flexible covariance matrices (i.e., the covariance matrix model was not chosen by BIC to limit the computing time). It can be seen that the ridgeline unimodal method does not reduce the number of clusters from EM/BIC at all. Generally, this method very rarely merges any components. The ridgeline ratio, DEMP and predictive strength method always merge all components here and the Bhattacharyya method does this in the vast majority of cases. The dip test method does not achieve this in a reliable way. Note that the simulation version of the dip test was tried out as well (results not shown). Here it only merged all components in 17 out of 200 cases.

Table 2 Numbers of clusters (n.o.c.) found by the merging methods for the simple uniform setup

Number of clusters	1	2	3	4	5	6	7	8	9
EM/BIC	0	0	0	8	79	72	25	12	4
ridgeuni	0	0	0	8	79	72	25	12	4
ridgerat	200	0	0	0	0	0	0	0	0
dip	111	18	17	26	16	9	2	1	0
bhat	195	5	0	0	0	0	0	0	0
DEMP	200	0	0	0	0	0	0	0	0
predstr	200	0	0	0	0	0	0	0	0

Desired n.o.c. is 1. Additionally to Table 1, the ridgeline unimodal method (ridgeuni) has been used

Table 3 Numbers of clusters found by the merging methods for the double rectangle uniform setup

Number of clusters	1	2	3	4	5	6
EM/BIC	0	6	73	88	27	6
ridgeuni	0	6	75	86	27	6
ridgerat	163	37	0	0	0	0
dip	137	26	27	10	0	0
bhat	139	58	3	0	0	0
DEMP	169	30	1	0	0	0
predstr	196	4	0	0	0	0

Desired n.o.c. is 1

Table 4 Numbers of clusters found by the merging methods for the uniform mixture setup

Number of clusters	1	2	3	4	5	6
EM/BIC	0	13	90	83	11	3
ridgeuni	0	13	90	84	10	3
ridgerat	102	97	1	0	0	0
dip	132	41	23	4	0	0
bhat	3	196	1	0	0	0
DEMP	2	196	2	0	0	0
predstr	8	152	40	0	0	0

Desired n.o.c. is 1 or 2, depending on the cluster concept

Double rectangle uniform. 213 points ($p = 2$) were generated from a uniform distribution on the union of the rectangles $[-8, 8] \times [0, 4]$ and $[6, 8] \times [4, 7]$ as shown on the left side of Fig. 3. In this setup most people would see a single cluster as well; the situation is unimodal (though there are bimodal one-dimensional projections). Results are given in Table 3. Apart from the typically unsatisfactory performance of ridgeline unimodal, the dip test and Bhattacharyya method have some difficulties to come up with a single cluster consistently. The ridgeline ratio and DEMP method do fairly well,and the prediction strength delivers by far the best performance.

Uniform mixture. 200 points ($p = 2$) were generated from a uniform distribution on the rectangles $[-8, 8] \times [0, 4]$ and 130 points were generated from a uniform distribution on $[6, 8] \times [4, 7]$ (note that the "mixture setups" in this simulation study were not really mixtures because the numbers of points in the "mixture components" were fixed). This is shown on the right side of Fig. 3. Results are given in Table 4. In this setup different "true" numbers of clusters can be seen. The distribution is unimodal, so from a modality based point of view there should be a single cluster, but from a pattern based perspective there should be two of them. Consequently, ridgeline ratio and the dip test method rather favour a single cluster (this is somewhat instable, which is appropriate because it is in fact a borderline situation) while Bhattacharyya, the DEMP and the prediction strength method clearly point to two clusters, though the latter too often comes up with three clusters.

Weakly separated Gaussian mixture. 300 points ($p = 5$) were generated from a 5-dimensional $\mathcal{N}(\mathbf{0}, \mathbf{I}_5)$-distribution and 50 points from a $\mathcal{N}((3.2, 0, 0, 0, 0)^t, \mathbf{D})$-distribution, where $\mathbf{D}$ is a diagonal matrix with diagonal $(0.1, 1, 1, 1, 1)$, i.e., all

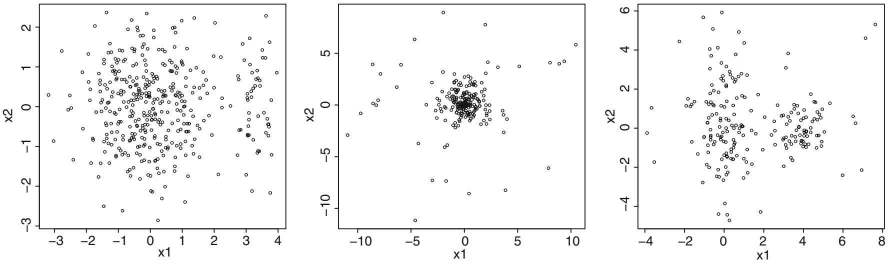

Fig. 6 Data generated from the simulation setups weakly separated Gaussian mixture, scale mixture and Gaussian mixture with noise

Table 5 Numbers of clusters found by the merging methods for the weakly separated Gaussian mixture setup

Number of clusters	1	2
EM/2	0	200
ridgeuni	3	197
ridgerat	26	174
dip	195	5
bhat	8	192
DEMP	51	149
predstr	0	200

Desired n.o.c. is 2

information separating clusters is in the first dimension. Some data generated from this setup (first two dimensions only) can be seen on the left side of Fig. 6. EM/2 with the covariance matrix model fixed to be fully flexible was used for this setup, because EM/BIC often estimated $\hat{s} = 1$ or $\hat{s} = 2$ with a wrong covariance matrix model affecting the estimated separation between components, which is counterproductive for measuring the quality of component merging methods. Results are given in Table 5.

The distribution is clearly bimodal, so for most applications (except if clusters are required to be more strongly separated) one would not want to merge the two components. However, the setup is somewhat difficult because the separation is not very strong and there is only clustering information in a single dimension. The true ridgeline ratio for this distribution is 0.145, for the Bhattacharyya distance $\exp(-d) = 0.074$ and the true Bayesian misclassifcation probability is 0.015 ($p_{12} \approx p_{21}$). All these values are not very far away from the cutoff values (9).

Ridgeline unimodal surprisingly merged components in three cases, whereas prediction strength never merged the components. Bhattacharyya did reasonably well, followed by ridgeline ratio. DEMP and ended up with two clusters in a clear majority of cases (which is at least better than what EM/BIC achieved without merging; not shown) while the dip test merged the components in almost all cases; obviously its p-value derived from a uniform null model is much too conservative here. Note that all methods can be expected to perform better in setups with more clearly separated clusters.

Table 6 Numbers of clusters found by the merging methods for the Gaussian scale mixture setup

Number of clusters	1	2	3	4	5
EM/BIC	0	0	173	26	1
ridgeuni	199	1	0	0	0
ridgerat	200	0	0	0	0
dip	198	2	0	0	0
bhat	3	196	1	0	0
DEMP	195	5	0	0	0
predstr	0	44	156	0	0

Desired n.o.c. is 1, 2 or 3, depending on the cluster concept

Gaussian scale mixture. Data were generated from three Gaussian components ($p = 3$) all with the same mean vector 0. In the third dimension, the components were all $\mathcal{N}(0, 1)$-distributed, independent of the first two dimensions. In the first two dimensions, the covariance matrix was $c\mathbf{I}_2$ with $c = 0.01$ for the first component (150 points), $c = 1$ for the second component (150 points) and $c = 20$ for the third component (50 points). A dataset (first two dimensions) is shown in the middle of Fig. 6. Results are given in Table 6. For these data it depends on the cluster concept how many clusters there are. With a modality based cluster concept, this should be a single cluster. With a pattern based cluster concept it depends on the separation required between clusters whether there are two or three clusters (or even a single one, if strong separation is required).

Consequently, ridgeline unimodal, ridgeline ratio and the dip test method favour a single cluster in a quite stable manner, as well as somewhat surprisingly the DEMP method. Bhattacharyya went for the two-cluster solution here (the second and third component were not separated enough; different tuning constants would be required to find three clusters). They were very stable as well. The prediction strength method came up with three clusters in 156 simulation runs and is therefore most sensible for finding these patterns that are not separated in terms of location.

Gaussian mixture with noise. For this setup $p = 4$, but again all the clustering information is in the first two dimensions, with independent standard Gaussians for all data in dimensions 3 and 4. In dimensions 1 and 2, 120 points were generated from $\mathcal{N}((0, 0)^t, \mathbf{D})$ with $\mathbf{D}$ a diagonal matrix with diagonal (0.7, 3.5). 70 points were generated from $\mathcal{N}((4, 0)^t, 0.7\mathbf{I}_2)$. 20 points were generated from a uniform distribution on $[-4, 8] \times [-6, 6]$. The uniform noise was designed in order to blur the distinction between the two Gaussian components, see the right side of Fig. 6. The Gaussian mixture here was fitted with an additional uniform component (using the facility of the mclustBIC function in the MCLUST package, Fraley and Raftery 2002), which was ignored in the merging process, in order to limit the influence of outliers. Results are given in Table 7. Two clusters (not including the noise component) should be found, unless the noise is regarded as additional cluster (which in most applications probably would not be sensible at least as long as the number of "noise points" is small).

All methods found two clusters in the vast majority of simulation runs. The ridgeline methods are a bit less stable than the others with DEMP being the "winner" here (though not by a significant margin).

Table 7 Numbers of clusters found by the merging methods for the Gaussian mixture with noise setup

Desired n.o.c. is 2

Number of clusters	1	2	3	4
EM/BIC	0	169	30	1
ridgeuni	0	171	29	0
ridgerat	28	172	0	0
dip	13	185	2	0
bhat	14	186	0	0
DEMP	11	189	0	0
predstr	0	182	18	0

Table 8 Numbers of clusters found by the merging methods for the exponential setup

Desired n.o.c. is 1

Number of clusters	1	2	3	4	5	6
EM/BIC	0	2	47	111	30	10
ridgeuni	6	25	60	80	24	5
ridgerat	195	5	0	0	0	0
dip	164	23	7	5	1	0
bhat	184	15	1	0	0	0
DEMP	188	11	1	0	0	0
predstr	185	15	0	0	0	0

Exponential. 200 points ($p = 2$) were generated from two independent Exponential(1)-distributions on order to produce a homogeneous but skew setup. Results are given in Table 8. It would be difficult to argue in favour of any number of clusters different from 1.

Again, all methods (except of ridgeline unimodal) do the correct job in the vast majority of cases. This time the ridgeline ratio performed best (using pairwise tests for equal proportions, the ridgeline ratio with 195 successes is not significantly better than DEMP with 188, but it is significantly better than the prediction strength with 185, $p = 0.038$) and the dip test was a little less stable than the others.

Overall, the results confirm that it depends on the cluster concept which method should be chosen. However, the results indicate that the dip test method and, more clearly, the ridgeline unimodal method, are not to be recommended (except perhaps in some special situations in which what they do is precisely what is desired), particularly because of their inability to merge some non-Gaussian but unimodal populations reliably.

All other methods emerged as "winners" at least somewhere and no clear ranking can be given. So the ridgeline ratio method can be recommended for the modality based cluster concept whereas Bhattacharyya, prediction strength and DEMP are better for the pattern based cluster concept, though DEMP did not work as required for this concept in the Gaussian Scale Mixture. The prediction strength showed a generally good performance, but needs by far the most computational effort and depends on random numbers for subsetting.

A general result is that the merging methods (except of ridgeline unimodal, which was included rather for illustrative purposes) almost consistently deliver more stable numbers of clusters (in terms of the number of simulation runs in their mode, whatever it is) than EM/BIC. Of course this is partly caused by the fact that the merging methods can only produce smaller or equal, but never larger numbers of clusters than EM/BIC. Nevertheless, It looks as if at least in situations where the Gaussian mixture model assumptions are violated (all setups except of Weakly Separated Gaussian Mixture and Gaussian Scale Mixture), the merging methods produce more reliable results.

7 Real data examples

7.1 Crabs data

The crabs dataset has already been introduced in the Introduction. Using the tuning constants in (9), three methods find the four cluster-solution closest to the true grouping, namely Bhattacharyya, ridgeline ratio and DEMP (the latter merging the fifth cluster at the borderline value of $q = 0.0251$). Ridgeline unimodal only merges two of the nine components

The dip test method reduces the number of components to seven clusters, merging cluster 1 with 3 and 6 with 8, compare Fig. 1. Note that these results do not change even when reducing the tuning constant to 0.01. As has been mentioned before, it cannot be taken for granted that a solution with more clusters than known "true classes" is wrong and it may point to some unknown subspecies structure, for example, though applying the methods from Hennig (2005) does not reveal strong separation between the remaining seven clusters.

The prediction strength method merges all components into a single cluster here. The prediction strength for two clusters is 0.650, quite some distance from the cutoff 0.75, and for three clusters 0.444. This may be considered to be practically not useful as a clustering, though it illustrates correctly that any clustering on these data is somewhat uncertain.

In terms of the cluster concept, it is a bit difficult to decide what would be required for these data. One would normally expect species and subspecies to be separated by gaps, pointing toward the modality cluster concept and the ridgeline ratio, but rather not the ridgeline unimodal or dip test method, because smaller groups of individuals could potentially exist causing several weakly separated modes within species. For sexes within the same species, on the other hand, it could not really be argued before knowing the data that they should be separated by gaps, even though this apparently happened in the given dataset.

7.2 Wisconsin cancer data

The Wisconsin cancer dataset is taken from the UCI Machine Learning Repository http://archive.ics.uci.edu/ml/. It was analysed first by Street et al. (1993). Data are given about 569 patients, and there are the two "true" classes of benign (357 cases) and malignant (212 cases) tumors. There are ten quantitative features in the dataset, the documentation of which can be obtained from the UCI website (actually the features have been recorded for every cell nucleus in an image and three different statistics

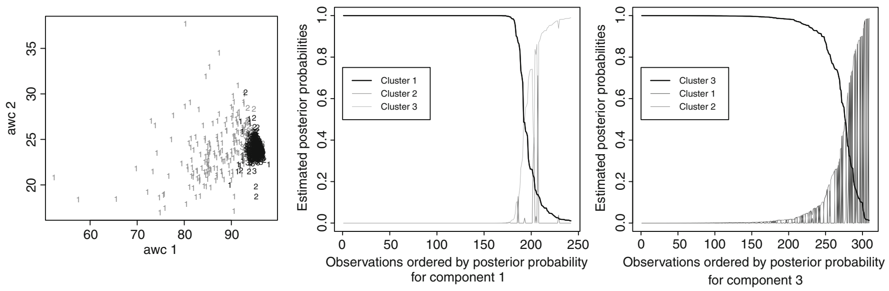

Fig. 7 Wisconsin cancer data. *Left*: asymmetric weighted coordinates (Hennig 2005) for benign patients (*bold*, numbers indicating EM/BIC components but separation between 2 and 3 is not clear from the given plot). *Middle*: ordered posterior plot for component 1 indicating some weak overlap with component 3. *Right*: ordered posterior plot for component 3 overlapping more strongly with both components 1 and 2

of these are in the original dataset so that the number of variables is 30, but only the first ten variables—the feature means—were used here). After transformation to square roots (which improves the classification accuracy of the EM/BIC solution a lot), EM/BIC fits three mixture components to the quantitative variables (ignoring the classification information).

Component 1 is strongly (178:13) dominated by malignant cases, while components 2 and 3 consist mostly of benign cases (93:7 and 251:27).

The relevant cluster concept in such data is often pattern based (not assuming too much prior knowledge what kind of classes and how many there are in the dataset but looking for “something malignant”), because there are many datasets in medicine in which patients affected by serious conditions deviate from the others basically by some more extreme measurements in more than one direction. So it may be that classes do not differ too strongly in terms of means or modes, but rather in terms of variance. (Note that pattern based methods can find mean/mode differences as well, if enough separated.)

This is confirmed by the results. Using the tuning constants in (9), Bhattacharyya and prediction strength method come up with the optimal possible two cluster-solution, merging components 2 and 3. The DEMP method, however, merges all three components. In the last step, the two remaining clusters (which are the same as above) are merged with $q = 0.032$, just above the cutoff value 0.025. The Bhattacharyya criterion to merge the last two clusters is 0.081, somewhat close to the cutoff. 0.809 is the value of the prediction strength at which they are merged, which is clearly above the cutoff 0.75 recommended here, but a borderline case according to the original recommendations of Tibshirani and Walther (2005).

The modality based methods, ridgeline ratio and dip test, merged all components, pointing indeed to something like a covariance mixture without clearly separated modes. This can be confirmed by the left side of Fig. 7, showing the more homogeneous group of benign cases against the higher variation among the malignant cases. Ordered posterior plots for components no. 1 and 3 are also given; component 3 is obviously not well separated from any of the other two components, while component 1 looks somewhat more “cluster-like”. Note that the information showed in the ordered posterior plots corresponds to the information used by the DEMP method.

Several other discriminant plots (not shown) have been produced to explore the separation between the three clusters of the EM/BIC solution, and they indicate that from the point of view of cluster analysis at least no evidence can be found to interpret component 2 and 3 as "separated clusters" (with component 1 the situation is somewhat more borderline at least with a pattern based cluster concept), so that merging makes sense.

8 Discussion

While it should be obvious that the merging problem is very relevant in practice, its solution may be regarded as somewhat arbitrary, due to the lack of identifiability. Instead of advertising a single method that may be "optimal" in some sense, several different methods have been proposed in the present paper, and the decision between them is always dependent on subjective decisions about the aim of clustering. The same can be said about the choice of a cutoff-value, which is needed for all of these methods. This should not be considered as a weakness of the methods, but rather as an inevitable feature of the merging problem (and more generally of cluster analysis, though this is often ignored particularly in papers advertising methods in a "one size fits it all"-style).

However, some differences in quality between the methods are perceivable. Quality can for example be measured in terms of stability in simulated setups with known truth and (depending on the cluster concept) known "desirable solutions". Based on the given results, the ridgeline unimodal and the dip test method seem to be problematic (note that the former approach is the only one that does not require a cutoff value). Even in truly unimodal setups, EM/BIC tends to produce mixture components that do not suggest unimodality, and therefore even under a modality based cluster concept the ridgeline ratio method with weaker conditions on merging seems to be preferable. In terms of interpretation the dip test is somewhat appealing and may be used in situations where it is acceptable to leave components showing any indication of multimodality along a single dimension unmerged (perhaps with an even larger cutoff value than 0.05 or using the simulation version to do a bit better in setups like the one in Table 5).

All the other methods performed strongly in some setups (again relative to the underlying cluster concept), were acceptable in most others, and were generally much more stable than EM/BIC.

Despite its good performance in the simulations and real data examples, a disadvantage with the Bhattacharyya method is that the direct interpretation of its tuning constant seems to be most difficult. Though it is linked to the misclassification probability by bounding it from above, in most cases this bound is much too high.

An interesting feature of the prediction strength method is that in the simulation studies it produced more clusters, on average, than DEMP, Bhattacharyya and ridgeline ratio. However, in the crabs dataset, which is probably less "nice" and less structured than simulated datasets, it was the strongest merger.

There are further conceivable versions of the hierarchical principle. Instead of the dip test, other homogeneity test statistics could be considered such as the gap statistic

(Tibshirani et al. 2001), which, however, implicitly assumes at least approximately equal spherical within-cluster covariance matrices. It has been tried in some of the simulation setups above and did not perform very well. Using p-values of a suitable test against a homogeneity alternative as a stopping rule generally formalizes the idea that clusters should be merged if their union does not deviate significantly from homogeneity (however defined), though this is affected by the problem of multiple data dependent testing. More alternative suggestions have recently been presented on conferences and are in preparation for publication by Baudry, Raftery, Celeux, Lo, and Gottardo, and Dean and Nugent.

Dissimilarity measures other than the Bhattacharyya distance could be used as well, based on the researcher's judgment how the idea of merging "similar" clusters should be formalized in a given situation. Standard distance measures between distributions such as the Kuiper metric (Davies 1995; most more well known candidates are highly problematic to compute in the situations of interest of the present paper) could be considered as well as measures more specifically designed for clustering with normal mixtures, see for example Qiu and Joe (2006) and further references given therein.

Sometimes a uniform component is added to the mixture in order to catch outliers and data points not belonging to any cluster (Banfield and Raftery 1993; Fraley and Raftery 2002; Hennig and Coretto 2008). In such a case the methods introduced before can be applied to the Gaussian components only, ignoring the extent to which the data points are assigned to the uniform measured by the "posterior weight" computed by analogy to (2), as done in the Gaussian Mixture With Noise setup in the simulation study. This approach prevents small clusters of outliers and is therefore itself a kind of "merging" method.

Sometimes, particularly in situations with large p, sufficiently large n and flexible covariance matrix models, there is an issue with spurious components occasionally found by EM/BIC, i.e., components with very few (i.e., $\leq 1.2p$, say) points that lie closely together or on a lower dimensional hyperplane by chance, leading to estimated covariance matrix eigenvalues close to zero. These are difficult to merge because a covariance matrix cannot be reliably estimated with so few points, so at least the Bhattacharyya, ridgeline ratio and DEMP method can be expected to have problems merging them with anything. Even the dip test may reject unimodality significantly if such a component is involved, because of the data dependent choice of the dimension along which unimodality is tested, and the predictive strength method, while detecting their instability, does not merge them sooner than the underlying DEMP method. A straightforward solution for this could be to dissolve these components before starting the analysis by updating (2) (and the corresponding component parameters) for all other components ignoring the spurious ones. Some points (with too large Mahalanobis distance to any other component, say) may be declared as outliers. Note that generally the merging methods are not more affected by this than EM/BIC.

It is interesting to think about the asymptotic implications of the proposed methods, or, in other words, what they deliver when applied to theoretical distributions instead of datasets. For pairs of Gaussian distributions, there are properly defined ridgelines, Bhattacharyya dissimilarities and misclassification probabilities as estimated by the DEMP method (note that the upper bound on misclassification probabilities derived from Bhattacharyya dissimilarities is even asymptotically different from the misclassification probabilities). The dip test should eventually indicate whether a mixture of two involved Gaussians is unimodal along the discriminate coordinate separating their means. The prediction strength, however, will converge to 1 for $n \to \infty$ for any number of clusters if the involved clustering method is consistent for its own canonical parameters (which can be expected for EM/BIC if applied to a true Gaussian mixture, though there are some difficulties with proving it, see for example Redner and Walker 1984; Keribin 2000), because in such a case a clustering based on infinitely strong information will be stable. Therefore, there is a stronger case for letting the cutoff value depend on n for this method than for the others.

The considerations concerning pairs of Gaussians, however, do not tell the whole story. The indexes are applied to pairs of estimated Gaussians in situations where the underlying distribution is not a Gaussian mixture. The pairwise computations are embedded in a hierarchical scheme in which some of the methods reduce the information of an already merged mixture to its mean and covariance matrix. This is obviously much more difficult to analyse. For example it can be suspected that even for $n \to \infty$ merging of components estimated for uniform or other non-Gaussian unimodally distributed data requires merging of a non-unimodal mixture of estimated components at some time during the merging process. The prediction strength may still converge to 1 if a Gaussian mixture approximation for an underlying non-Gaussian distribution is consistent, but this is not always the case; for example, I presume that fitting a uniform distribution by a mixture of Gaussians is even asymptotically ambiguous.

A way around the identifiability problem explained in Sect. 2 would be a Bayesian approach that models the $(\pi_i, \mathbf{a}_i, \Sigma_i),\ i = 1, \ldots, s$, as random variables generated by k true clusters generated by hyperparameters. However, such an approach would require a model assumption about cluster generating parameter distributions. This means that the statistician would have to define first which patterns of Gaussian distributions should be considered as belonging to the same cluster. This is essentially the same kind of decision that is required by the frequentist as well, so that, while the identifiability problem could be technically resolved in the Bayesian way, the underlying issue of the need of deciding about a cluster concept of interest does not go away. Though it may be interesting to think about such a Bayesian approach, it is not considered in the present paper.

Giving a single practical recommendation is difficult, keeping in mind the varying performance qualities in the simulation study. The performances depend on the underlying cluster concept, so this has to be decided as a starting point in a practical situation. If the modality based cluster concept is of interest, the ridgeline ratio method can be recommended. For the pattern based cluster concept, the prediction strength method looks promising, though in some situations (and probably particularly with larger n) Gaussian mixture fits can be quite stable, and will therefore not be merged by the prediction strength method, even if the “patterns” fitted by them are not very distinctive. On the other hand, the prediction strength may merge too strongly in applications in which stability is not the most important concern. The Bhattacharyya method is a computationally cheap alternative, as well as the DEMP method, though the latter tends to merge scale mixtures. Choosing tuning constants more flexibly (dependent on n and p as well as the data structure) may be worthwhile as well. A tool for stability assessment of merged components is suggested in Hennig (2010). All methods introduced here will soon be incorporated in the R-package "fpc".

References

Banfield JD, Raftery AE (1993) Model-based Gaussian and non-Gaussian clustering. Biometrics 49:803–821

Baudry JP, Raftery AE, Celeux G, Lo K, Gottardo R (2008) Combining mixture components for clustering. Technical report 540, University of Washington, Seattle

Biernacki C, Celeux G, Govaert G (2000) Assessing a mixture model for clustering with the integrated completed likelihood. IEEE T Pattern Anal 22:719–725

Campbell NA, Mahon RJ (1974) A multivariate study of variation in two species of rock crab of genus Leptograpsus. Aust J Zool 22:417–425

Davies PL (1995) Data features. Stat Neerl 49:185–245

Fraley C, Raftery AE (2002) Model-based clustering, discriminant analysis, and density estimation. J Am Stat Assoc 97:611–631

Fraley C, Raftery AE (2003) Enhanced software for model-based clustering, density estimation and discriminant analysis. J Classif 20:263–286

Fukunaga K (1990) Introduction to statistical pattern recognition, 2nd edn. Academic Press, New York

Hartigan JA, Hartigan PM (1985) The dip test of unimodality. Ann Stat 13:70–84

Hastie T, Tibshirani R (1996) Discriminant analysis by Gaussian mixtures. J Roy Stat Soc B Met 58:155–176

Hennig C (2005) Asymmetric linear dimension reduction for classification. J Comput Graph Stat 13:930–945

Hennig C (2010) Ridgeline plot and clusterwise stability as tools for merging Gaussian mixture components. In: Locarek-Junge H, Weihs C (eds) Classification as a tool for research. Springer, Berlin, accepted for publication

Hennig C, Coretto P (2008) The noise component in model-based cluster analysis. In: Preisach C, Burkhard H, Schmidt-Thieme L, Decker R (eds) Data analysis, machine learning and applications. Springer, Berlin, pp 127–138

Keribin C (2000) Consistent estimation of the order of a mixture model. Sankhya Ser A 62:49–66

Li J (2004) Clustering based on a multilayer mixture model. J Comput Graph Stat 14:547–568

Matusita K (1971) Some properties of affinity and applications. Ann I Stat Math 23:137–155

McLachlan GJ, Peel D (2000) Finite mixture models. Wiley, New York

Qiu W, Joe H (2006) Generation of random clusters with specified degree of separation. J Classif 23:315–334

Ray S, Lindsay BG (2005) The topography of multivariate normal mixtures. Ann Stat 33:2042–2065

Redner RA, Walker HF (1984) Mixture densities, maximum likelihood and the EM algorithm. SIAM Rev 26:195–239

Schwarz G (1978) Estimating the dimension of a model. Ann Stat 6:461–464

Street WN, Wolberg WH, Mangasarian OL (1993) Nuclear feature extraction for breast tumor diagnosis. IS & T/SPIE 1993 international symposium on electronic imaging: science and technology, vol 1905, San Jose, CA, pp 861–870

Tantrum J, Murua A, Stuetzle W (2003) Assessment and pruning of hierarchical model based clustering. In: Proceedings of the 9th ACM SIGKDD international conference on knowledge discovery and data mining, Washington, DC, pp 197–205

Tibshirani R, Walther G (2005) Cluster validation by prediction strength. J Comput Graph Stat 14:511–528

Tibshirani R, Walther G, Hastie T (2001) Estimating the number of clusters in a dataset via the gap statistic. J Roy Stat Soc B Met 63:411–423

Ueda N, Nakano R, Ghahramani Z, Hinton GE (2000) SMEM algorithm for mixture models. Neural Comput 12:2109–2128