Supplementary materials for this article are available online.
Please click the JCGS link at http://pubs.amstat.org.

Simulating Data to Study Performance of Finite Mixture Modeling and Clustering Algorithms

Ranjan MAITRA and Volodymyr MELNYKOV

A new method is proposed to generate sample Gaussian mixture distributions according to prespecified overlap characteristics. Such methodology is useful in the context of evaluating performance of clustering algorithms. Our suggested approach involves derivation of and calculation of the exact overlap between every cluster pair, measured in terms of their total probability of misclassification, and then guided simulation of Gaussian components satisfying prespecified overlap characteristics. The algorithm is illustrated in two and five dimensions using contour plots and parallel distribution plots, respectively, which we introduce and develop to display mixture distributions in higher dimensions. We also study properties of the algorithm and variability in the simulated mixtures. The utility of the suggested algorithm is demonstrated via a study of initialization strategies in Gaussian clustering. This article has supplementary material online.

Key Words: Cluster overlap; Eccentricity of ellipsoid; Mclust; *MixSim*; Mixture distribution; Parallel distribution plots.

1. INTRODUCTION

There is a abundance of statistical literature on clustering datasets (Hartigan 1985; Murtagh 1985; Ramey 1985; McLachlan and Basford 1988; Kaufman and Rousseuw 1990; Everitt, Landau, and Leesem 2001; Fraley and Raftery 2002; Kettenring 2006). With no uniformly best method, it is important to understand the strengths and weaknesses of different algorithms. Many researchers evaluate performance by applying suggested methodologies to select classification datasets such as Fisher's Iris (Anderson 1935), Ruspini (1970), crabs (Campbell and Mahon 1974), textures (Brodatz 1966), etc., but this approach, while undoubtedly helpful, does not provide for an extensive investigation into the properties of

Ranjan Maitra is Associate Professor, Department of Statistics and Statistical Laboratory, Iowa State University, Ames, IA 50011. Volodymyr Melnykov is Assistant Professor, Department of Statistics, North Dakota State University, Fargo, ND 58102 (E-mail: *Volodymyr.Melnykov@ndsu.edu*).

Journal of Computational and Graphical Statistics, Volume 19, Number 2, Pages 354–376
DOI: 10.1198/jcgs.2009.08054

the algorithm. For one, the adage (see p. 74) of Box and Draper (1987) that "All models are wrong. Some models are useful." means that performance can not be calibrated in terms of models with known properties. Also, the relatively easier task of classification is often not possible to perfect on these datasets, raising further misgivings on using them to judge clustering ability. But the biggest drawback to relying exclusively on them to evaluate clustering algorithms is that detailed and systematic assessment in a wide variety of scenarios is not possible.

Dasgupta (1999) defined c-separation in the context of learning Gaussian mixtures as follows: two p-variate Gaussian densities $N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ and $N_p(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ are c-separated if $\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\| \geq c\sqrt{p \max(d_{\max}(\boldsymbol{\Sigma}_i), d_{\max}(\boldsymbol{\Sigma}_j))}$, where $d_{\max}(\boldsymbol{\Sigma})$ is the largest eigenvalue of $\boldsymbol{\Sigma}$. He mentioned that there is significant to moderate to scant overlap between at least two clusters for $c = 0.5$, 1.0, and 2.0. Likas, Vlassis, and Verbeek (2003), Verbeek, Vlassis, and Krose (2003), and Verbeek, Vlassis, and Nunnink (2003) used these values to evaluate performance of their clustering algorithms on different simulation datasets. Maitra (2009) modified the above to require equality for at least one pair (i, j), calling it "exact-c-separation" between at least two clusters, and used it to study different initialization strategies vis-a-vis different values of c. He, however, pointed out that separation between clusters as defined above depends only on the means and the largest eigenvalues of the cluster dispersions, regardless of their orientation or mixing proportion. Thus, the degree of difficulty of clustering is, at best, only partially captured by exact-c-separation.

Other suggestions have been made in the clustering and applied literature. Most are built on the common-sense assumption that difficulty in clustering can be indexed in some way by the degree of overlap (or separation) between clusters. We refer to the work of Steinley and Henson (2005) for a detailed description of many of these methods, presenting only a brief summary here. Milligan (1985) developed a widely used algorithm that generates well-separated clusters from truncated multivariate normal distributions. But the algorithm's statements on degree of separation may be unrealistic (Atlas and Overall 1994) and thus clustering methods can not be fully evaluated under wide ranges of conditions (Steinley and Henson 2005). Similar shortcomings are also characteristic of methods proposed by Blashfield (1976), Kuiper and Fisher (1975), Gold and Hoffman (1976), McIntyre and Blashfield (1980), and Price (1993). Atlas and Overall (1994) manipulated intraclass correlation to control cluster overlap, but they mentioned that their description is not "perceptually meaningful" (p. 583). Waller, Underhill, and Kaiser (1999) provided a qualitative approach to controlling cluster overlap which lacks quantitation and can not be extended to high dimensions.

Recent years have also seen development of the "OCLUS" (Steinley and Henson 2005) and "GenClus" (Qiu and Joe 2006a) algorithms. In "OCLUS," marginally independent clusters are generated with known (asymptotic) overlap between two clusters, with the proviso that no more than three clusters overlap at the same time. This automatically rules out many possible configurations. The algorithm is also limited in its ability to generate clusters with differing correlation structures. The R package "GenClus" (Qiu and Joe 2006a) uses Qiu and Joe (2006b)'s separation index which is defined in an univariate framework as the difference of the biggest lower and the smallest upper quantiles divided by the difference of

the biggest upper and the smallest lower quantiles. The ratio is thus close to unity when the gap between two clusters is substantial, and negative with a lower bound of -1 when they overlap. This index is not directly extended to multiple dimensions, so Qiu and Joe (2006a) proposed finding the one-dimensional projection with approximate highest separation index. The attempt to characterize separation between several multidimensional clusters by means of the best single univariate projection clearly loses substantial information, and thus resulting statements on cluster overlap are very partial and can be misleading.

In this article, we define overlap between two Gaussian clusters as the sum of their misclassification probabilities (Section 2). Computing these probabilities is straightforward in spherical and homogeneous clustering scenarios but involves evaluating the cumulative distribution function (cdf) of the distribution of linear combinations of independent noncentral chi-squared and normal random variables in the general case. This is accomplished using the algorithm AS 155 proposed by Davies (1980). We compute exact overlap and develop an iterative algorithm to generate random clusters with prespecified average or/and maximum overlap. Our algorithm applies to all dimensions and to Gaussian mixture models, and is illustrated and analyzed for different overlap characteristics in many settings in Section 3 and in the supplement. We also introduce a *parallel distribution plot* to display multivariate mixture distributions. Section 4 calibrates four different initialization strategies for the expectation-maximization (EM) algorithm for Gaussian mixtures as an example of how our algorithm may be utilized. We conclude with some discussion.

2. METHODOLOGY

Let $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n$ be independent, identically distributed (iid) p-variate observations from the mixture density $g(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \phi(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, where π_k is the probability that $\mathbf{X}_i$ belongs to the kth group with mean $\boldsymbol{\mu}_k$ and dispersion matrix $\boldsymbol{\Sigma}_k$ and p-dimensional multivariate normal density $\phi(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = (2\pi)^{-p/2}|\boldsymbol{\Sigma}_k|^{-1/2} \exp\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k)\}$. Our goal is to devise ways to specify $\{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$'s, such that generated realizations $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n$ satisfy some prespecified characteristic measure summarizing clustering complexity, for which we use a surrogate measure in the form of overlap between clusters. Consider two clusters indexed by $\phi(\mathbf{x}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ and $\phi(\boldsymbol{x}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ and with probabilities of occurrence π_i and π_j. We define overlap ω_{ij} between these two clusters in terms of the sum of the two misclassification probabilities $\omega_{j|i}$ and $\omega_{i|j}$, where

$$\begin{aligned}\omega_{j|i} &= \Pr[\pi_i \phi(\mathbf{X}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) < \pi_j \phi(\mathbf{X}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) | \mathbf{X} \sim N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)] \\ &= \Pr_{N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}\Bigg[(\mathbf{X} - \boldsymbol{\mu}_j)'\boldsymbol{\Sigma}_j^{-1}(\mathbf{X} - \boldsymbol{\mu}_j) - (\mathbf{X} - \boldsymbol{\mu}_i)'\boldsymbol{\Sigma}_i^{-1}(\mathbf{X} - \boldsymbol{\mu}_i) \\ &\qquad < \log \frac{\pi_j^2 |\boldsymbol{\Sigma}_i|}{\pi_i^2 |\boldsymbol{\Sigma}_j|}\Bigg] \qquad (2.1)\end{aligned}$$

and similarly, $\omega_{i|j} = \Pr_{N_p(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}[(\mathbf{X} - \boldsymbol{\mu}_i)'\boldsymbol{\Sigma}_i^{-1}(\mathbf{X} - \boldsymbol{\mu}_i) - (\mathbf{X} - \boldsymbol{\mu}_j)'\boldsymbol{\Sigma}_j^{-1}(\mathbf{X} - \boldsymbol{\mu}_j) < \log \frac{\pi_i^2 |\boldsymbol{\Sigma}_j|}{\pi_j^2 |\boldsymbol{\Sigma}_i|}]$.

2.1 OVERLAP BETWEEN TWO CLUSTERS

When both Gaussian clusters have the same covariance structure $\mathbf{\Sigma}_i = \mathbf{\Sigma}_j \equiv \mathbf{\Sigma}$, derivation of overlap is relatively straightforward. For then $\omega_{j|i} = \Phi(-\frac{1}{2}((\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)'\mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i))^{-1/2} + \log\frac{\pi_j}{\pi_i}[(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)'\mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)]^{-1/2})$, where $\Phi(x)$ is the standard normal cdf at x. $\omega_{i|j}$ is essentially the same, with the only difference that π_i is interchanged with π_j. It follows that if $\pi_i = \pi_j$, then $\omega_{j|i} = \omega_{i|j}$, resulting in $\omega_{ij} = 2\Phi(-\frac{1}{2}\sqrt{(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)'\mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)})$. For spherical clusters with $\pi_i \neq \pi_j$, $\omega_{j|i} = \Phi(-\|\boldsymbol{\mu}_j - \boldsymbol{\mu}_i\|/2\sigma + \sigma\log\frac{\pi_j}{\pi_i}/\|\boldsymbol{\mu}_j - \boldsymbol{\mu}_i\|)$, with, once again, a similar expression for $\omega_{i|j}$. For equal mixing proportions $\pi_i = \pi_j$, $\omega_{ij} = 2\Phi(-\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|/2\sigma)$.

For the case of general covariance matrices, we are led to the following

Theorem 1. *Consider two p-variate Gaussian clusters indexed by $N_p(\boldsymbol{\mu}_i, \mathbf{\Sigma}_i)$ and $N_p(\boldsymbol{\mu}_j, \mathbf{\Sigma}_j)$ and mixing proportions π_i and π_j, respectively. Define $\mathbf{\Sigma}_{j|i} \equiv \mathbf{\Sigma}_i^{1/2}\mathbf{\Sigma}_j^{-1}\mathbf{\Sigma}_i^{1/2}$, with spectral decomposition given by $\mathbf{\Sigma}_{j|i} = \mathbf{\Gamma}_{j|i}\mathbf{\Lambda}_{j|i}\mathbf{\Gamma}'_{j|i}$, where $\mathbf{\Lambda}_{j|i}$ is a diagonal matrix of eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_p$ of $\mathbf{\Sigma}_{j|i}$, and $\mathbf{\Gamma}_{j|i}$ is the corresponding matrix of eigenvectors $\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_p$ of $\mathbf{\Sigma}_{j|i}$. Then*

$$\omega_{j|i} = \Pr_{N_p(\boldsymbol{\mu}_i, \mathbf{\Sigma}_i)}\Bigg[\sum_{\substack{l=1 \\ l:\lambda_l \neq 1}}^{p}(\lambda_l - 1)U_l + 2\sum_{\substack{l=1 \\ l:\lambda_l = 1}}^{p}\delta_l W_l \leq \sum_{\substack{l=1 \\ l:\lambda_l \neq 1}}^{p}\frac{\lambda_l\delta_l^2}{\lambda_l - 1} - \sum_{\substack{l=1 \\ l:\lambda_l = 1}}^{p}\delta_l^2 + \log\frac{\pi_j^2|\mathbf{\Sigma}_i|}{\pi_i^2|\mathbf{\Sigma}_j|}\Bigg], \tag{2.2}$$

where U_l's are independent noncentral-χ^2-distributed random variables with one degree of freedom and noncentrality parameter given by $\lambda_l^2\delta_l^2/(\lambda_l - 1)^2$ with $\delta_l = \gamma_l'\mathbf{\Sigma}_i^{-1/2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)$ for $l \in \{1, 2, \ldots, p\} \cap \{l : \lambda_l \neq 1\}$, independent of the W_l's, which are independent $N(0, 1)$ random variables, for $l \in \{1, 2, \ldots, p\} \cap \{l : \lambda_l = 1\}$.

Proof: When $\mathbf{X} \sim N_p(\boldsymbol{\mu}_i, \mathbf{\Sigma}_i)$, it is well known that $(\mathbf{X} - \boldsymbol{\mu}_i)'\mathbf{\Sigma}_i^{-1}(\mathbf{X} - \boldsymbol{\mu}_i) \sim \chi_p^2$. Let $\mathbf{Z} \sim N_p(\mathbf{0}, \mathbf{I})$. Then $(\mathbf{X} - \boldsymbol{\mu}_j)'\mathbf{\Sigma}_j^{-1}(\mathbf{X} - \boldsymbol{\mu}_j) \stackrel{d}{=} (\mathbf{\Sigma}_i^{1/2}\mathbf{Z} + \boldsymbol{\mu}_i - \boldsymbol{\mu}_j)'\mathbf{\Sigma}_j^{-1}(\mathbf{\Sigma}_i^{1/2}\mathbf{Z} + \boldsymbol{\mu}_i - \boldsymbol{\mu}_j) = [\mathbf{Z} + \mathbf{\Sigma}_i^{-1/2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)]'\mathbf{\Sigma}_i^{1/2}\mathbf{\Sigma}_j^{-1}\mathbf{\Sigma}_i^{1/2}[\mathbf{Z} + \mathbf{\Sigma}_i^{-1/2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)] = [\mathbf{W} + \mathbf{\Gamma}'_{j|i}\mathbf{\Sigma}_i^{-1/2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)]'\mathbf{\Lambda}_{j|i}[\mathbf{W} + \mathbf{\Gamma}'_{j|i}\mathbf{\Sigma}_i^{-1/2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)] = \sum_{l=1}^{p}\lambda_l(W_l + \delta_l)^2$, where $\mathbf{W} = \mathbf{\Gamma}'_{j|i}\mathbf{Z}$. Note also that $(\mathbf{X} - \boldsymbol{\mu}_i)'\mathbf{\Sigma}_i^{-1}(\mathbf{X} - \boldsymbol{\mu}_i) \stackrel{d}{=} \mathbf{Z}'\mathbf{Z} = \mathbf{W}'\mathbf{W} = \sum_{l=1}^{p}W_l^2 \sim \chi_p^2$. Thus $\omega_{j|i}$ reduces to $\Pr[\sum_{l=1}^{p}\lambda_l(W_l + \delta_l)^2 - \sum_{l=1}^{p}W_l^2 < \log\frac{\pi_j^2|\mathbf{\Sigma}_i|}{\pi_i^2|\mathbf{\Sigma}_j|}] = \Pr[\sum_{l=1}^{p}\{(\lambda_l - 1)W_l^2 + 2\lambda_l\delta_l W_l + \lambda_l\delta_l^2\} < \log\frac{\pi_j^2|\mathbf{\Sigma}_i|}{\pi_i^2|\mathbf{\Sigma}_j|}]$. Further reduction proceeds with regard to whether λ_l is greater than, less than, or equal to 1, which we address separately.

(a) $\lambda_l > 1$: In this case, $(\lambda_l - 1)W_l^2 + 2\lambda_l\delta_l W_l + \lambda_l\delta_l^2 = (\sqrt{\lambda_l - 1}W_l + \lambda_l\delta_l/\sqrt{\lambda_l - 1})^2 - \lambda_l\delta_l^2/(\lambda_l - 1)$. Note that $\sqrt{\lambda_l - 1}W_l + \lambda_l\delta_l/\sqrt{\lambda_l - 1} \sim N(\lambda_l\delta_l/\sqrt{\lambda_l - 1}, \lambda_l - 1)$ so that $(\sqrt{\lambda_l - 1}W_l + \lambda_l\delta_l/\sqrt{\lambda_l - 1})^2 \sim (\lambda_l - 1)\chi^2_{1,\lambda_l^2\delta_l^2/(\lambda_l - 1)^2}$.

(b) $\lambda_l < 1$: Here $(\lambda_l - 1)W_l^2 + 2\lambda_l\delta_l W_l + \lambda_l\delta_l^2 = -(\sqrt{1-\lambda_l}W_l - \lambda_l\delta_l/\sqrt{1-\lambda_l})^2 - \lambda_l\delta_l^2/(\lambda_l - 1)$ where, using similar arguments as before, $(\sqrt{1-\lambda_l}W_l - \lambda_l\delta_l/\sqrt{1-\lambda_l})^2 \sim (1-\lambda_l)\chi^2_{1,\lambda_l^2\delta_l^2/(\lambda_l-1)^2}$.

(c) $\lambda_l = 1$: In this case, $(\lambda_l - 1)W_l^2 + 2\lambda_l\delta_l W_l + \lambda_l\delta_l^2 = 2\delta_l W_l + \delta_l^2$.

Combining (a), (b), and (c) and moving terms around yields (2.2) in the statement of the theorem. □

Analytic calculation of $\omega_{j|i}$ is impractical, but numerical computation is readily done using Algorithm AS 155 (Davies 1980). Thus we can calculate ω_{ij} between any pair of Gaussian clusters. Our goal now is to provide an algorithm to generate mean and dispersion parameters for clusters with specified overlap characteristics, where overlap is calculated using the above.

2.2 Simulating Gaussian Cluster Parameters

The main idea underlying our algorithm for generating clusters with prespecified overlap characteristic is to simulate random cluster mean vectors and dispersion matrices and to scale the latter iteratively such that the distribution of calculated overlaps between clusters essentially matches the desired overlap properties. Because overlap between clusters may be specified in several ways, we fix ideas by assuming that this specification is in the form of the maximum or average (but at this point, not both) overlap between all cluster pairs. We present our algorithm next.

2.2.1 Clusters With Prespecified Average or Maximum Pairwise Overlap

The specific steps of our iterative algorithm are as follows:

1. *Generating initial cluster parameters.* Obtain K random p-variate cluster centers $\{\boldsymbol{\mu}_k; k = 1, 2, \ldots, K\}$. To do so, take a random sample of size K from some user-specified distribution (such as p-dimensional uniform) over some hypercube. Generate initial random dispersion matrices $\{\boldsymbol{\Sigma}_k; k = 1, \ldots, K\}$. Although there are many approaches to generating $\boldsymbol{\Sigma}_k$'s, we propose using realizations from the standard Wishart distribution with degrees of freedom given by $p + 1$. This is speedily done using the Bartlett (1939) decomposition of the Wishart distribution. While the low choice of degrees of freedom allows for great flexibility in orientation and shape of the realized matrix, it also has the potential to provide us with dispersion matrices that are near-singular. This may not be desirable, so we allow prespecification of a *maximum eccentricity* $e_{\max}$ for all $\boldsymbol{\Sigma}_k$'s. Similar to the case with two-dimensional ellipses, we define eccentricity of $\boldsymbol{\Sigma}$ in p-dimensions as $e = \sqrt{1 - d_{(p)}/d_{(1)}}$, where $d_{(1)} \geq d_{(2)} \geq \cdots \geq d_{(p)}$ are the eigenvalues of $\boldsymbol{\Sigma}$. Thus, given current realizations $\boldsymbol{\Sigma}_k$, we get corresponding spectral decompositions $\boldsymbol{\Sigma}_k = \mathbf{V}_k\mathbf{D}_k\mathbf{V}_k'$, with $\mathbf{D}_k$ as the diagonal matrix of eigenvalues (in decreasing order) of $\boldsymbol{\Sigma}_k$ and $\mathbf{V}_k$ as the matrix of corresponding eigenvectors, and calculate the eccentricity e_k's. For those $\boldsymbol{\Sigma}_k$'s (say,

$\boldsymbol{\Sigma}_l$) for which $e_k > e_{\max}$, we shrink eigenvalues toward $d_{(1)}^{(l)}$ such that $e_{\text{new}}^{(l)} = e_{\max}$. To do this, we obtain new eigenvalues $d_{(i)}^{(l*)} = d_{(1)}^{(l)}(1 - e_{\max}^2(d_{(1)}^{(l)} - d_{(i)}^{(l)})/(d_{(1)}^{(l)} - d_{(p)}^{(l)}))$. Note that $d_{(1)}^{(l*)} \geq d_{(2)}^{(l*)} \geq \cdots \geq d_{(p)}^{(l*)}$, $d_{(1)}^{(l*)} = d_{(1)}^{(l)}$, and $e_{\text{new}}^{(l)} = e_{\max}$. Reconstitute $\boldsymbol{\Sigma}_l^* = \mathbf{V}_l \mathbf{D}_l^* \mathbf{V}_l'$, where $\mathbf{V}_l$ is as above, and $\mathbf{D}_l^*$ is the diagonal matrix of the new eigenvalues $d_{(1)}^{(l*)} \geq d_{(2)}^{(l*)} \geq \cdots \geq d_{(p)}^{(l*)}$. To simplify notation, we continue referring to the new matrix $\boldsymbol{\Sigma}_l^*$ as $\boldsymbol{\Sigma}_l$.

2. *Calculate overlap between clusters.* For each cluster pair $\{(i, j); 1 \leq i < j \leq K\}$ indexed by corresponding parameters $\{(\pi_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), (\pi_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\}$, obtain $\omega_{j|i}$, $\omega_{i|j}$, $\boldsymbol{\Sigma}_{j|i}$, $\boldsymbol{\Sigma}_{i|j}$, $\boldsymbol{\delta}_{i|j}$, and $\boldsymbol{\delta}_{j|i}$ where $\boldsymbol{\delta}_{i|j} = \boldsymbol{\Gamma}_{i|j}' \boldsymbol{\Sigma}_j^{-1/2}(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)$. Calculate ω_{ij} for the cluster pair using the Davies (1980) AS 155 algorithm on (2.2) in Theorem 1. Compute $\hat{\breve{\omega}}$ or $\hat{\bar{\omega}}$ depending on whether the desired controlling characteristic is $\breve{\omega}$ or $\bar{\omega}$, respectively. If the difference between the $\hat{\breve{\omega}}$ (or $\hat{\bar{\omega}}$) and $\breve{\omega}$ (correspondingly, $\bar{\omega}$) is negligible (i.e., within some prespecified tolerance level ϵ), then the Gaussian cluster parameters $\{(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) : k = 1, 2, \ldots, K\}$ provide parameters for a simulated dataset that correspond to the desired overlap characteristic.

3. *Scaling clusters.* If Step 2 is not successful, replace each $\boldsymbol{\Sigma}_k$ with its scaled version $c\boldsymbol{\Sigma}_k$, where c is chosen as follows. For each pair of clusters (i, j), calculate $\omega_{j|i}$ (and thus $\hat{\bar{\omega}}(c)$ or $\hat{\breve{\omega}}(c)$) as a function of c, by applying Theorem 1 to $c\boldsymbol{\Sigma}_k$'s. Use root-finding techniques to find a c satisfying $\hat{\breve{\omega}}(c) = \breve{\omega}$ (or $\hat{\bar{\omega}}(c) = \bar{\omega}$). For this c, the Gaussian cluster parameters $\{(\pi_k, \boldsymbol{\mu}_k, c\boldsymbol{\Sigma}_k) : k = 1, 2, \ldots, K\}$ provide parameters for the simulated dataset that correspond to our desired overlap characteristic.

Some additional comments are in order. Note that, but for the minor adjustment $\boldsymbol{\delta}_{i|j}^{\text{new}} = c^{-1/2}\boldsymbol{\delta}_{i|j}$, Step 3 does not require recomputation of the quantities already calculated in Step 2 and involved in (2.2). This speeds up computation, making root-finding in Step 3 practical.

Step 3 is successful only when Step 1 yields valid candidate $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k)$'s, that is, those capable of attaining the target $\bar{\omega}$ (or $\breve{\omega}$). As $c \to \infty$, $\boldsymbol{\delta}_{i|j}^{\text{new}} \to 0$ and $\omega_{j|i} \to \omega_{j|i}^{\infty} = \Pr_{N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}[\sum_{l=1}^{p}{}_{:\lambda_l \neq 1}(\lambda_l - 1)U_l \leq \log \frac{\pi_j^2|\boldsymbol{\Sigma}_i|}{\pi_i^2|\boldsymbol{\Sigma}_j|}]$ where $U_l \overset{\text{iid}}{\sim}$ central-χ^2 with one degree of freedom. Thus for every candidate pair, a limiting overlap $(\omega_{ij}^{\infty} = \omega_{j|i}^{\infty} + \omega_{i|j}^{\infty})$ can be obtained, with corresponding limiting average $(\hat{\bar{\omega}}^{\infty})$ or maximum $(\hat{\breve{\omega}}^{\infty})$ overlaps. If $\hat{\bar{\omega}}^{\infty} < \bar{\omega}$ (or $\hat{\breve{\omega}}^{\infty} < \breve{\omega}$), then the desired overlap characteristic is possibly unattainable using the (consequently invalid) candidate $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k)$'s, and new candidates are needed from Step 1. By continuity arguments, and $\hat{\bar{\omega}}(c) \to 0 (\hat{\breve{\omega}}(c) \to 0)$ as $c \to 0$, Step 3 is always successful for valid candidate $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k)$'s. Also, the asymptotic overlap is completely specified by $(\boldsymbol{\Sigma}_k, \pi_k)$'s; candidate $\boldsymbol{\mu}_k$'s play no role whatsoever. Finally, note that a conceptually more efficient approach would be to compare the maximum of $\hat{\bar{\omega}}(c)$—instead of $\hat{\bar{\omega}}^{\infty}$ (or $\hat{\breve{\omega}}(c)$—instead of $\hat{\breve{\omega}}^{\infty}$)—with the target $\bar{\omega}$ (or $\breve{\omega}$), thereby retaining possible candidate $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k)$'s otherwise discarded because $\hat{\bar{\omega}}^{\infty} < \bar{\omega}$ (or $\hat{\breve{\omega}}^{\infty} < \breve{\omega}$). However, maximizing $\hat{\bar{\omega}}(c)$ by taking derivatives produces functions as in Theorem 1 of the article by Press (1966), which require expensive calculations of the confluent hypergeometric function ${}_1\mathrm{F}_1$ (Slater 1960). Potential gains would likely be lost, especially considering our experience

that the maximum of $\omega_{ij}(c)$ never exceeded ω_{ij}^{∞} for any pair of components (i, j) in thousands of simulation experiments.

Our objective of avoiding computationally expensive calculations involving ${}_1\mathrm{F}_1$ is also why we choose not to use a derivative-based Newton–Raphson method for finding a root in Step 3. We instead first hone in on bounds on the target c by restricting attention to (positive or negative) powers of 2, and then find a root using the method of Forsythe, Malcolm, and Moler (1980).

The material presented so far details a computationally practical approach to generating Gaussian clustered data satisfying some overlap characteristic such as average or maximal overlap. However, a single characteristic is unlikely to comprehensively capture overlap in a realization. For instance, the average overlap may come about from few cluster pairs with substantial overlap, or where many cluster pairs have overlap measures close to each other (and the average). At the other end, the maximal overlap is driven entirely by one cluster pair (the one with largest overlap, which amount we control). Consequently, we may obtain scenarios with very varying clustering difficulty, yet summarized by the same characteristic. Thus, we need strategies which can control at least two overlap characteristics. We address a way to generate Gaussian cluster parameters satisfying two overlap characteristics—the average and maximal overlap—next.

2.2.2 Clusters With Prespecified Average and Maximum Pairwise Overlap

The basic philosophy here is to first use Section 2.2.1 to generate $\{(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k) : k = 1, 2, \ldots, K\}$ satisfying $\check{\omega}$. The component pair satisfying $\check{\omega}$ is held fixed while the remaining clusters are scaled to also achieve the targeted $\bar{\omega}$. The algorithm is iterative in spirit with specifics as follow:

1. *Initial generation.* Use Section 2.2.1 to obtain $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k)$'s satisfying $\check{\omega}$. Find $(i^*, j^*) \ni \omega_{i^*j^*} = \check{\omega}$. Unless the realization is discarded in Step 2 below, $(\boldsymbol{\mu}_{i^*}, \boldsymbol{\Sigma}_{i^*}, \pi_{i^*})$ and $(\boldsymbol{\mu}_{j^*}, \boldsymbol{\Sigma}_{j^*}, \pi_{j^*})$ are kept unchanged all the way through to termination of the algorithm.

2. *Validity check.* Find c (call it $c_\vee$) such that $\omega_{ij}(c_\vee) \le \check{\omega}\ \forall (i, j) \ne (i^*, j^*)$. If $\hat{\bar{\omega}}(c_\vee) < \bar{\omega}$, Step 3 may not terminate, so discard the realization and redo Step 1 again.

3. *Limited scaling.* Redo Step 3 of the algorithm in Section 2.2.1 to obtain the targeted $\bar{\omega}$, with $c \in (0, c_\vee)$. Note that the pair (i^*, j^*) does not participate, and thus $\boldsymbol{\lambda}_{i^*|j}^{\text{new}} = c\boldsymbol{\lambda}_{i^*|j}$, $\boldsymbol{\lambda}_{i|j^*}^{\text{new}} = \boldsymbol{\lambda}_{i|j^*}/c$, and $\boldsymbol{\delta}_{i|j^*}^{\text{new}} = \boldsymbol{\delta}_{i|j^*}$ in the calculations for (2.2) of Theorem 1.

4. *Final check.* If $\omega_{ij}(c) > \omega_{i^*j^*}$ for some $(i, j) \ne (i^*, j^*)$, discard realization and redo Step 1.

A $c_\vee$ in Step 2 is guaranteed to exist, because for every pair $(i, j) \ne (i^*, j^*)$, $\omega_{ij}(c) \to 0$ as $c \to 0$. We find $c_\vee$ by first considering the pair (i', j') with largest asymptotic overlap $\omega_{i'j'}^{\infty}$. If $\omega_{i'j'}^{\infty} \le \check{\omega}$, then the desired $c_\vee$ is obtained: let $c_\vee \equiv \infty$. Otherwise, find c for which $\omega_{i'j'}(c) = \check{\omega}$. This is our candidate $c_\vee^0$: we evaluate $\omega_{ij}(c_\vee^0)$ for all other pairs (i, j) (including pairs with one component i^* or j^*). If $\omega_{i''j''}(c_\vee^0) > \check{\omega}$ for some pair (i'', j''), we find

an updated candidate $c_\vee^1 \in (0, c_\vee^0)$ satisfying $\omega_{i''j''}(c_\vee^1) = \breve{\omega}$. The process continues until a global $c_\vee$ satisfying all pairs is found.

Step 4 is a final bulwark against the possibility that any configuration at this stage does not satisfy both $\bar{\omega}$ and $\breve{\omega}$. This last may be a rare possibility, however, given that none of our realizations were discarded at this stage in any of the thousands of simulation experiments reported in this article.

Controlling overlap characteristics through both $\bar{\omega}$ and $\breve{\omega}$ provides greater definition to the complexity of the clustering problem, while keeping implementation of the algorithm practical. However, for $K > 2$ and $\bar{\omega}$ very close to $\breve{\omega}$, it may still not be possible to obtain a realization in a reasonable amount of computer time. For most practical scenarios, however, $\breve{\omega}$ is unlikely to be very close to $\bar{\omega}$ so this may not be that much of an issue.

It may also be desirable to specify distribution of the overlaps in terms of other characteristics such as $\bar{\omega}$ and standard deviation ω_σ. We propose rewriting such characteristics (which may be harder to implement for cluster generation using the methods above) in approximate terms of $\bar{\omega}$ and $\breve{\omega}$. We assume that the pairwise overlaps are $K(K+1)/2$ random draws from a $\beta(\gamma, \nu)$ distribution, and use the relationships $\mathbb{E}(\omega) = \gamma/(\gamma+\nu)$ and $\mathrm{Var}(\omega) = \gamma\nu/[(\gamma+\nu)^2(\gamma+\nu+1)]$. Equating the above with $\bar{\omega}$ and ω_σ^2, respectively, provides the following estimates: $\gamma = \frac{\bar{\omega}}{2\bar{\omega}+1}(\frac{\bar{\omega}(1+\bar{\omega})}{\omega_\sigma^2(2\bar{\omega}+1)^2} - 1)$ and $\nu = \frac{1+\bar{\omega}}{2\bar{\omega}+1}(\frac{\bar{\omega}(1+\bar{\omega})}{\omega_\sigma^2(2\bar{\omega}+1)^2} - 1)$. Note that there are constraints on the set of possible values for $\bar{\omega}$ and ω_σ related to the fact that $\gamma, \nu > 0$. Given the distribution of overlaps, we are thus able to use order statistic theory to find the density of $\breve{\omega}$. The mode of this density is calculated numerically and can be used in conjunction with $\bar{\omega}$ to obtain clusters with desired overlap characteristics. We note that we have found this relationship to hold in all our empirical experiments with K larger than 5 and values of $\bar{\omega} \leq 0.05$. In the clustering context, the last is a reasonable restriction. Thus, we propose using this empirical relationship for $K > 5$ and $\bar{\omega} \leq 0.05$. For $K \leq 5$ or $\bar{\omega} > 0.05$, we continue with specifying overlap in terms of $\bar{\omega}$ and $\breve{\omega}$ directly. If some realization does not satisfy some desired additional characteristic, we discard the sample and regenerate a new proposal.

2.2.3 Incorporating Scatter in Clustered Datasets

There has of late been great interest in the development of algorithms addressing clustering in the presence of scatter (Tseng and Wong 2005; Maitra and Ramler 2009). Experimental cases allowing for algorithms to be calibrated are fairly straightforward given our algorithm: we generate p-dimensional uniform realizations on the desired hypercube containing the clusters, but outside the $100(1-\alpha_s)\%$ regions of concentration for the mixture density $\sum_{k=1}^K \pi_k \phi(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. The proportion of scatter (s) as well as α_s are parameters in the cluster generation, impacting clustering performance, and can be preset as desired.

2.2.4 Comparison of Overlap Probability and c-Separation

We end this section by comparing our overlap measure with exact-c-separation (Dasgupta 1999; Maitra 2009) for homogeneous spherical clusters. In this case, $\breve{\omega} =$

$2\Phi\{-\frac{c\sqrt{p}}{2}\}$, so that for homogeneous spherical clusters, using $\check{\omega}$ as the sole characteristic to describe cluster overlap is equivalent to using exact-c-separation.

3. ILLUSTRATION AND ANALYSIS OF ALGORITHM

In this section, we examine and illustrate different aspects of the algorithm presented in Section 2. We examine overlap characteristics of some commonly used classification datasets and present realized mixture densities with substantial and low $\bar{\omega}$ and low and moderate variation ω_σ in these overlaps. We chose $\epsilon = 10^{-6}$ and $e_{\max} = 0.9$ in all experiments. Further, we set π_k's to all be at least $\pi_\wedge$, the smallest value for which there is, with high probability, at least $(p+1)$ observations from each component in a dataset containing n observations from the mixture. We also study possible geometries of generated mixture densities, and present studies on convergence of Section 2.2.1 Step 3 and Section 2.2.2 Step 2. For brevity, we summarize our results here, with details on many of these issues relegated to the supplementary materials. In what follows, figures and tables labeled with the prefix "S-" refer to figures and tables in the supplement.

3.1 ILLUSTRATION ON CLASSIFICATION DATASETS

To understand possible values of $\bar{\omega}$ and $\check{\omega}$, we first calculate overlap characteristics of some commonly used classification datasets. These are the Iris (Anderson 1935), Ruspini (1970), crabs (Campbell and Mahon 1974), textures (Brodatz 1966), wine (Forina 1991), image (Newman et al. 1998), and Ecoli (Nakai and Kinehasa 1991) datasets. We summarize in Table 1 our calculated $\bar{\omega}$ and $\check{\omega}$, misclassification rates τ, and adjusted Rand measures $\mathcal{R}$ (Hubert and Arabie 1985) using quadratic discriminant analysis (QDA) and model-based clustering using EM. Both τ and $\mathcal{R}$ are calculated between the true classifications on one hand and the QDA and EM groupings each on the other. Note that $\mathcal{R} \leq 1$ with equality attained for a perfectly matched grouping. Further, to minimize impact of initialization, the EM was started using the true group means, dispersions, and mixing proportions. Thus, the attained $\mathcal{R}$'s can be regarded as the best-case values when using EM. Table 1 also illustrates the challenges of relying on such datasets. All calculations are made assuming a Gaussian distribution, with each dataset as one realization from this presumed distribution. Thus,

Table 1. Misclassification rates (τ_Q, τ_C) and adjusted Rand indices $(\mathcal{R}_Q, \mathcal{R}_C)$ obtained on some standard classification datasets using quadratic discriminant analysis (QDA) and EM-clustering.

Dataset	n	p	K	$\bar{\omega}$	$\check{\omega}$	τ_Q	$\mathcal{R}_Q$	τ_C	$\mathcal{R}_C$
Ruspini	75	2	4	0.000	0.001	0	1	0	1
Texture	5500	37	11	0.000	0.000	0	1	0	1
Wine	178	13	3	0.002	0.004	0.006	0.982	0.006	0.982
Iris	150	4	3	0.016	0.049	0.02	0.941	0.033	0.904
Crabs	200	5	4	0.020	0.087	0.04	0.897	0.07	0.828
Image	2310	11	7	0.001	0.007	0.099	0.820	0.222	0.683
Ecoli	327	5	5	0.044	0.238	0.101	0.822	0.128	0.783

restricting attention to classification datasets means that comprehensive understanding of clustering performance may be difficult. The results on the image dataset perhaps highlight this dilemma the best: we are unclear whether its relatively poorer performance is driven by K, p, model inadequacy, imprecise estimation of parameters, whether it is just because the dataset is one less probable realization from the mixture, etc. We note that for similar (K, p, n), low values of $\bar{\omega}$ and $\breve{\omega}$ generally provide lower τ's and higher $\mathcal{R}$'s, while higher values correspond to worse performance (higher τ's and lower $\mathcal{R}$'s).

3.2 SIMULATION EXPERIMENTS

3.2.1 Validity of Overlap as Surrogate for Clustering Complexity

We first investigated the validity of using our defined overlap as a surrogate measure for clustering complexity. We simulated 2000 datasets, each with 100 observations, from a two-dimensional two-component Gaussian mixture satisfying $\pi_{\wedge} = 0.2$ and a given overlap measure (because $K = 2$, $\bar{\omega} = \breve{\omega} \equiv \omega$). For each dataset, we applied the EM algorithm (initialized, with the parameter values that generated the dataset, to eliminate possible ill effects of improper initialization), obtained parameter estimates, and computed $\mathcal{R}$ on the derived classification. This was done for $\omega \in \{0.001, 0.01, 0.02, 0.03, \ldots, 1.0\}$. Figure 1 displays the mean and standard deviations of $\mathcal{R}$ for the different values of ω. Clearly, ω tracks $\mathcal{R}$ very well (inversely), providing support for using our overlap measure as a reasonable surrogate for clustering complexity.

Figures S-1 and S-2 display sample two-component two-dimensional finite mixture models, for $\omega \in [0.001, 0.75]$. From these figures, it appears that distinctions between the two components are sometimes unclear for $\omega = 0.15$. The trend is accentuated at higher

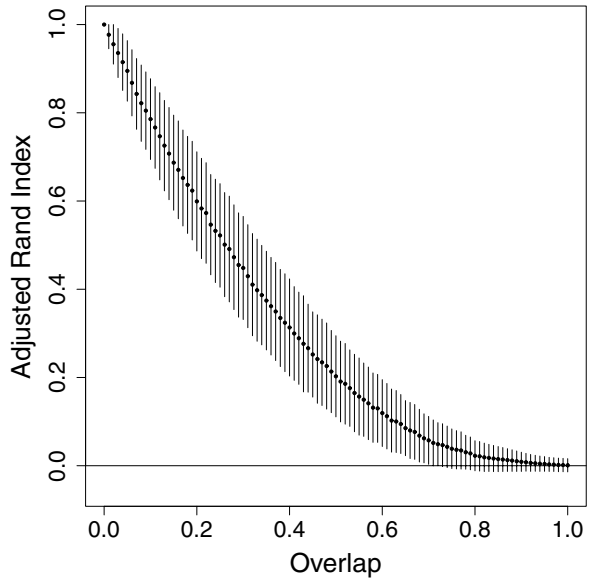


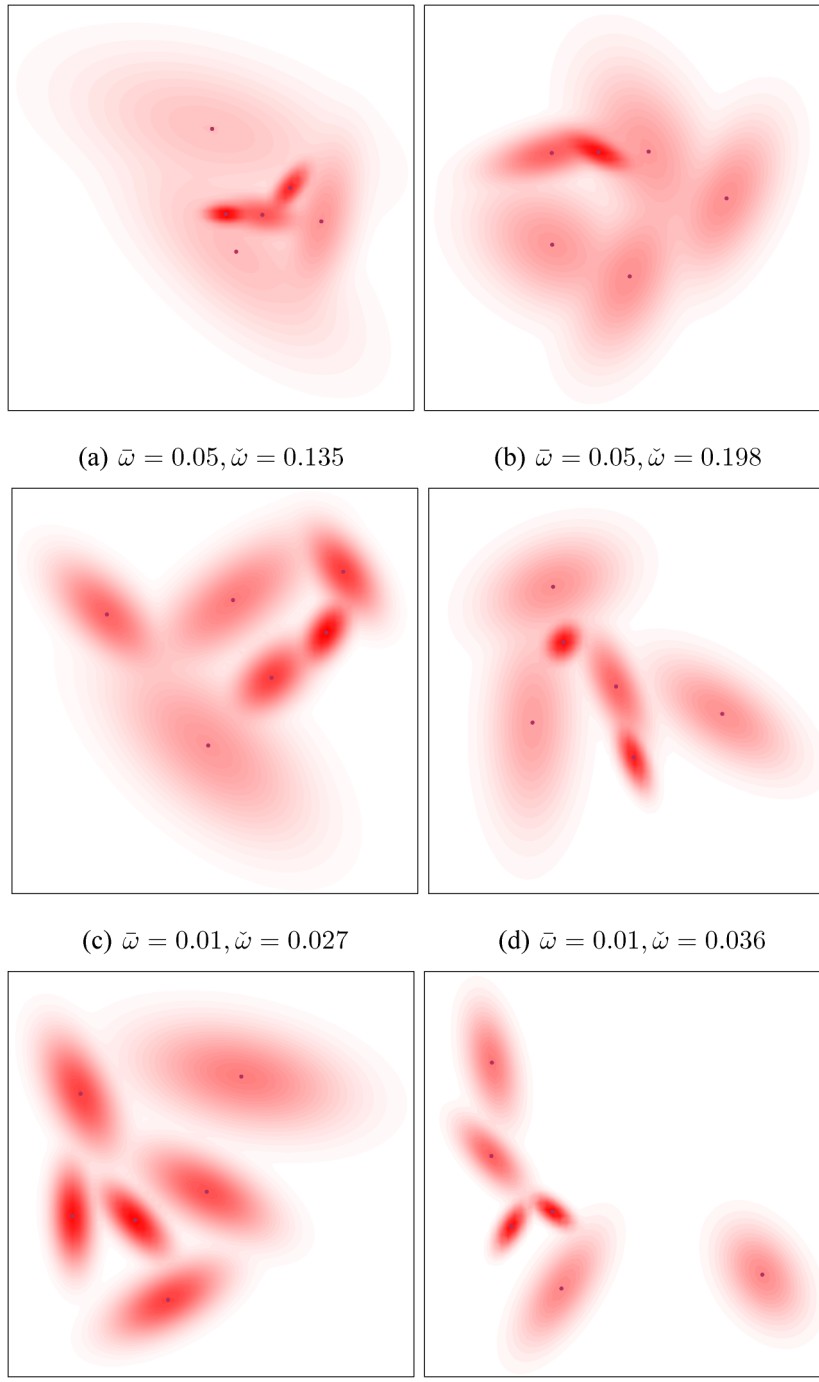
Figure 1. Mean ± SD's of $\mathcal{R}$'s to evaluate clustering in two-component two-dimensional Gaussian mixtures with different levels of overlap.

values: the two components are visually virtually indistinguishable for $\omega > 0.5$ or so and only the knowledge of their being two Gaussian components perhaps provides us with some visual cue into their individual structures. This matches the trend in Figure 1, because as components become indistinguishable, we essentially move toward a random class assignment, for which case $\mathcal{R}$ is constructed to have zero expectation (Hubert and Arabie 1985). Based on Figure 1 and the realized mixtures in Figures S-1 and S-2, we conclude that pairwise overlaps of below 0.05 indicate well-separated components, while those between around 0.05 and 0.1 indicate moderate separation. Pairwise overlaps above 0.15 in general produce poorly separated components. We use this in determining possible choices for $\breve{\omega}$ and $\bar{\omega}$. Figures S-1 and S-2 also provide some inkling into possible geometries induced by our simulation algorithm for different ω.

3.2.2 Two-Dimensional Experiments

Figure 2 presents contour plots of sample two-dimensional mixture distributions generated for $K = 6$ and different values $(\bar{\omega}, \breve{\omega})$. The choice of $\breve{\omega}$ was dictated by $\omega_\sigma = \bar{\omega}$ and $2\bar{\omega}$ and using the empirical Beta distribution for cluster overlaps discussed in Section 2.2.2. We set $\pi_\wedge = 0.14$. Different choices of $\bar{\omega}$ and $\breve{\omega}$ provide us with realizations with widely varying mixture (and cluster) characteristics. To see this, note that Figure 2, (a) and (b), has high average overlap ($\bar{\omega} = 0.05$) between cluster pairs. In Figure 2(a), $\breve{\omega}$ is comparatively low, which means that quite a few pairs of clusters have substantial overlap between them. In Figure 2(b), however, the clusters are better-separated except for the top left cluster pair which is quite poorly separated, satisfying $\breve{\omega}$ and contributing substantially to the high $\bar{\omega}$ value of 0.05. Thus, in the first case, many pairs of clusters have considerable overlap between them, but in the second case, a few cluster pairs have substantial overlap while the rest overlap moderately. The same trends are repeated for Figure 2, (c) and (d) and Figure 2, (e) and (f), even though the cluster pairs are increasingly better-separated, satisfying $\bar{\omega} = 0.01$ and 0.001, respectively. Thus in Figure 2(e), there is at best modest overlap between any two clusters, while in Figure 2(f), there is even less overlap, save for the two clusters with smallest dispersions, which have far more overlap than the clusters in Figure 2(e).

Impact on Performance of Mclust. We also clustered a sample dataset obtained from each mixture model in Figure 2. We generated 150 realizations from each distribution and classified each observation to its most likely group, based on the true parameter values. The resulting classification, displayed in Figure 3, (a)–(f) via plotting character, represents the idealized grouping that can perhaps ever be achieved at each point. (Note that even supervised learning in the form of QDA may not always be expected to achieve this result, because the parameter values are estimated from the training data.) For each dataset, we used model-based clustering via the Mclust function in the R package MCLUST (Fraley and Raftery 2006). The function uses BIC to obtain the best-fitting model over a range of mixing components, set to be between 1 and 12 here. We invoked Mclust to choose the best model with unstructured dispersion matrices. Figure 3, (a)–(f), uses color to display the resulting groupings. We also report $\mathcal{R}$ between the Mclust and the idealized classifications.

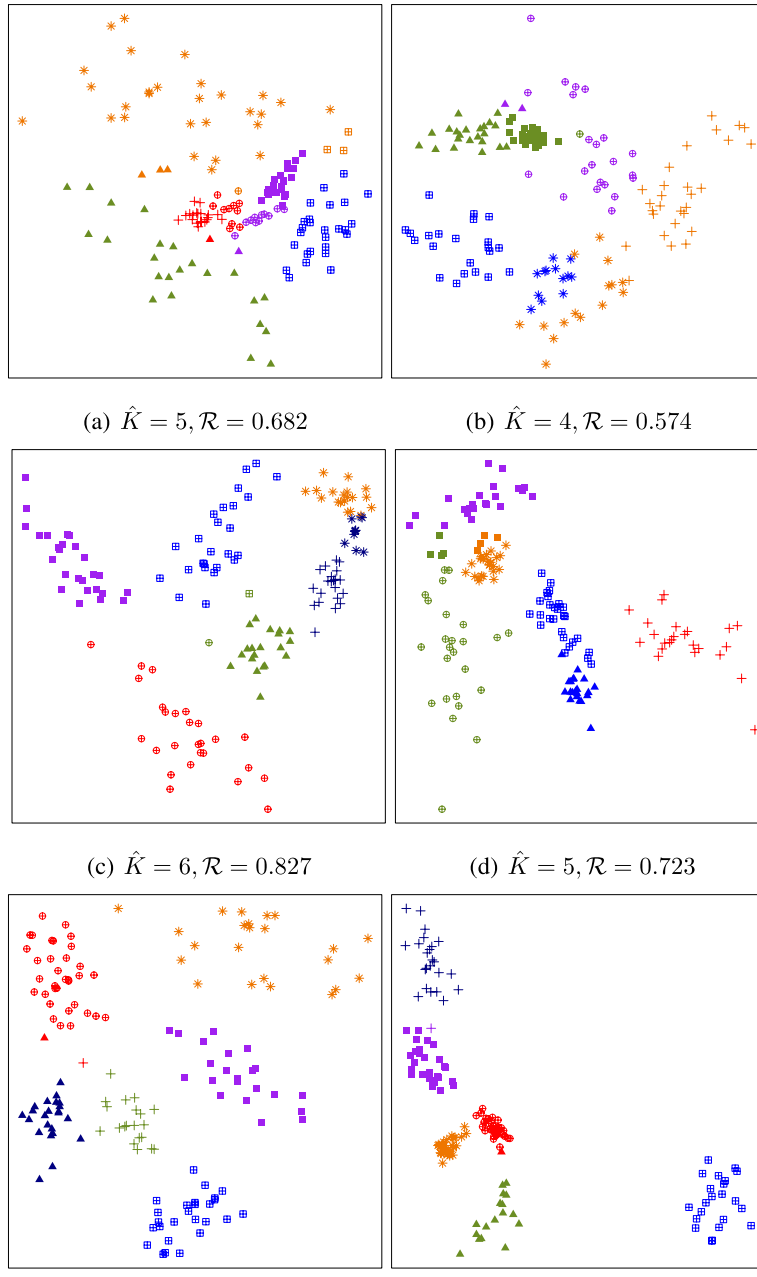


(a) $\bar{\omega} = 0.05, \check{\omega} = 0.135$ (b) $\bar{\omega} = 0.05, \check{\omega} = 0.198$

(c) $\bar{\omega} = 0.01, \check{\omega} = 0.027$ (d) $\bar{\omega} = 0.01, \check{\omega} = 0.036$

(e) $\bar{\omega} = 0.001, \check{\omega} = 0.0027$ (f) $\bar{\omega} = 0.001, \check{\omega} = 0.0036$

Figure 2. Contour plots of sample six-component mixture distributions in two dimensions obtained using our algorithm and different settings of $\bar{\omega}$ and $\check{\omega}$. A color version of this figure is available in the electronic version of this article.



(a) $\hat{K} = 5, \mathcal{R} = 0.682$ (b) $\hat{K} = 4, \mathcal{R} = 0.574$

(c) $\hat{K} = 6, \mathcal{R} = 0.827$ (d) $\hat{K} = 5, \mathcal{R} = 0.723$

(e) $\hat{K} = 6, \mathcal{R} = 0.964$ (f) $\hat{K} = 6, \mathcal{R} = 0.968$

Figure 3. Groupings obtained using Mclust on a realization from the corresponding mixture distributions of Figure 2. Color indicates the Mclust grouping and character the best possible classification with known parameter values. Optimal numbers of clusters as determined by BIC are also noted in the captions for each subfigure. A color version of this figure is available in the electronic version of this article.

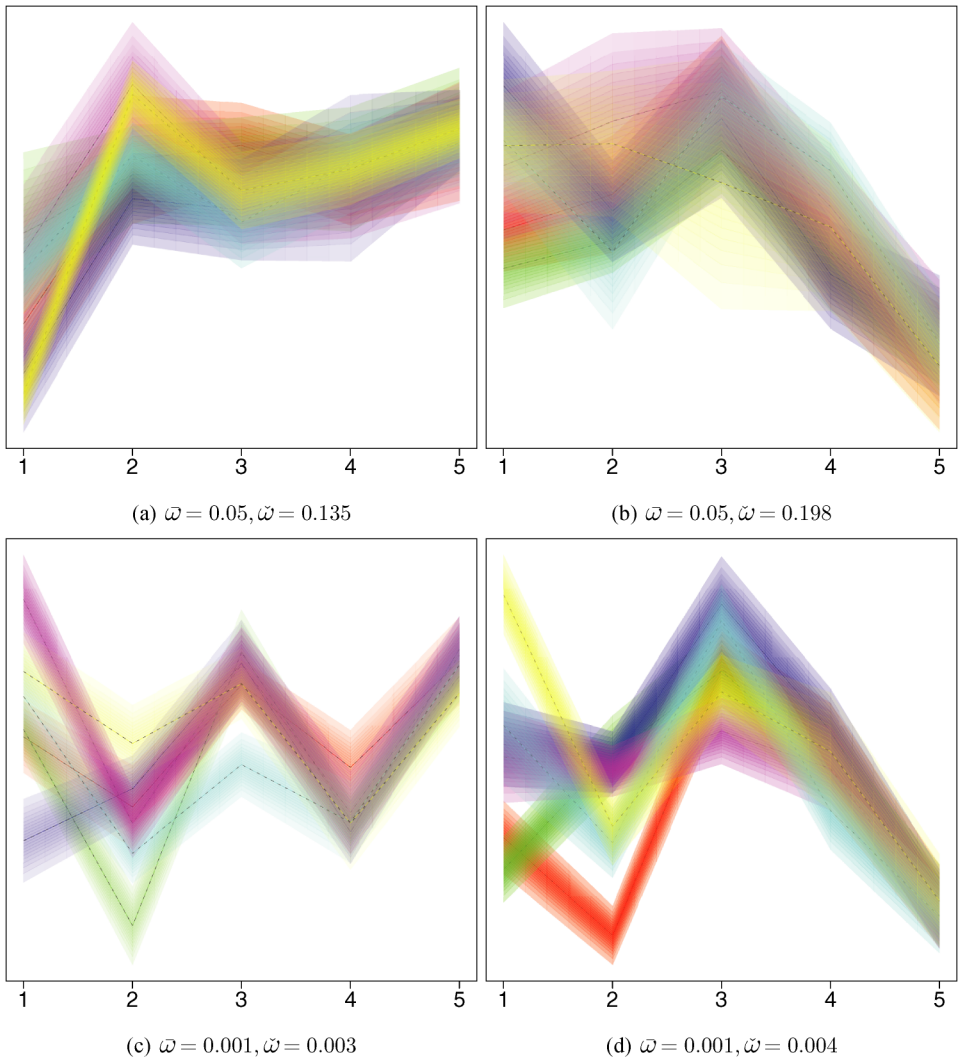
Figure 3 illustrates the impact on the performance of Mclust of varying the amount and nature of overlap between clusters. Thus Mclust's performance was worse for larger $\bar{\omega}$, but the magnitude of poor performance depended on $\check{\omega}$. For instance, consider the two cases for which $\bar{\omega} = 0.05$. In Figure 3(a) where there is considerable overlap between many clusters, Mclust identified five groups, misclassifying portions of almost every class and distributing observations from the least-separated cluster into two other classes. In Figure 3(b), however, $\mathcal{R}$ was lower with the major part of the disagreement due to the merger of the two most-overlapping groups, and the near-merger of the next most-overlapping clusters—thus BIC optimally identified only four groups. When $\bar{\omega} = 0.01$, performance was better with $\check{\omega} = 0.027$ (Figure 3(c); six optimal clusters) because all cluster pairs at best modestly overlapped with each other. On the other hand, with $\bar{\omega} = 0.01$ and $\check{\omega} = 0.036$, there were expectedly more misclassifications between the few cluster pairs with moderate overlap while the remaining groups were well identified: BIC found only five clusters to be optimal. With $\bar{\omega} = 0.001$ both scenarios performed well, even though the case with $\check{\omega} = 0.004$ had a slightly higher $\mathcal{R}$ value than the one with $\check{\omega} = 0.003$. The results of these illustrative experiments indicate that the nature and degree of overlap between groups, as summarized by $\bar{\omega}$ and $\check{\omega}$, have the potential to impact performance of clustering algorithms, in this case Mclust. We note that our statements on clustering performance are inferred here based only on one sample realization from each setting; obtaining a comprehensive understanding will entail generating several datasets with given overlap characteristics and evaluating performance on each. Indeed, this last is the objective of our cluster generation algorithm, which we demonstrate in Section 4.

We conclude this section by referring to Figures S-3 and S-4 (in the supplement) which provide four sample mixture model realizations, for each of the six pairs of settings in Figure 2. These figures display very well the range of six-component mixture models that can be obtained using our simulation algorithm. Additionally, as with the two-component examples, it seems that many different kinds of geometries can be induced by our algorithm, for different values of $(\bar{\omega}, \check{\omega})$.

3.2.3 Higher Dimensional Examples

We have also successfully used our algorithm to simulate mixture distributions of up to 50 components and for dimensions of up to 100. Note that in terms of computational effort, our algorithm is quadratic in the number of clusters because we calculate all pairwise overlaps between components. There is no ready technique for displaying higher dimensional distributions, so we introduce the *parallel distribution plot*, and use this in Figure 4 to illustrate some sample mixture distributions realized using our algorithm.

Parallel Distribution Plot. The objective of the parallel distribution plot is to display a multivariate mixture distribution in a way that contrasts and highlights the distinctiveness of each mixture component. We note that the dispersion matrix of a mixture density of the kind considered in this article, that is, $g(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \phi(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, is given by $\boldsymbol{\Sigma}_\bullet = \sum_{k=1}^{K} \pi_k \boldsymbol{\Sigma}_k + \sum_{k=1}^{K} \pi_k \boldsymbol{\mu}_k \boldsymbol{\mu}_k' - \sum_{l=1}^{K} \sum_{k=1}^{K} \pi_l \pi_k \boldsymbol{\mu}_l \boldsymbol{\mu}_k'$. Let $\mathbf{V}_\bullet$ be the matrix of orthonormal eigenvectors $[\mathbf{v}_1 \vdots \mathbf{v}_2 \vdots \ldots \vdots \mathbf{v}_p]$, corresponding to the eigenvalues $d_1 \geq d_2 \geq$



(a) $\varpi = 0.05, \breve{\omega} = 0.135$ (b) $\varpi = 0.05, \breve{\omega} = 0.198$

(c) $\varpi = 0.001, \breve{\omega} = 0.003$ (d) $\varpi = 0.001, \breve{\omega} = 0.004$

Figure 4. Parallel distribution plots of six-component mixture distributions in five dimensions and different settings of $\bar{\omega}$ and $\breve{\omega}$. A color version of this figure is available in the electronic version of this article.

$\cdots \geq d_p$ of $\mathbf{\Sigma}_\bullet$. Applying the rotation $\mathbf{V}_\bullet'$ to the mixture provides us with the principal components (PC's) of the mixture. These PC's are uncorrelated with decreasing variance given by $d_1 \geq d_2 \geq \cdots \geq d_p$. Also, the distribution of these PC's is still a mixture, of (rotated) Gaussians, with the kth rotated component having mixing proportion π_k and multivariate Gaussian distribution with mean vector $\mathbf{V}_\bullet'\boldsymbol{\mu}_k$ and dispersion $\mathbf{V}_\bullet'\mathbf{\Sigma}_k\mathbf{V}_\bullet$. We display the distribution of each mixture component, borrowing ideas from parallel coordinate plots (Inselberg 1985; Wegman 1990). Note that each component k only contributes a total mass of probability π_k. For each component k and jth marginal distribution (in projection space), locate the quantiles $\{q_{kj1}, q_{kj2}, \ldots, q_{kjm}\}$ corresponding to the $m-1$ equal increments in probabilities in $(0, \pi_k)$. The quantile q_{kji} of the jth marginal distrib-

ution is connected to the quantile $q_{k(j+1)i}$ of the $(j+1)$th marginal distribution by means of a parallel coordinate plot. The polygon formed by the two successive quantiles and the lines joining each is shaded with a given color, unique to every cluster component, and with varying opacity. The opacity at the vertical edges of each polygon is proportional to the density at the midpoint of the interval $(q_{kji}, q_{kj(i+1)})$. Inside the polygon, the opacity varies smoothly in the horizontal direction (only) as a convex combination of the edge opacities. Thus, we get a parallel distribution plot. Finally, we contend that though we develop and use parallel distribution plots here solely to display mixture distributions, they can also be used to display grouped data, by replacing each theoretical quantile by its empirical cousin.

Figure 4 displays parallel distribution plots of sample six-component five-dimensional mixture distributions obtained using our algorithm at four different settings. As expected, the total spread of the mixture distribution decreases with increase in PC dimension. The mixture components are also more separable in the first few PC's than in the later ones. This is because the dominant source of variability in the mixture distribution is between-cluster variability and that is better separated in the first few components. Toward the end, the PC's are dominated by random noise arising from the within-cluster variability over the between-cluster variability. Figure 4(a) illustrates the case for when $\bar{\omega} = 0.05$ and $\check{\omega} = 0.135$, while Figure 4(b) illustrates the case for when $\bar{\omega} = 0.05$ and $\check{\omega} = 0.198$. In both cases, there is overlap between several clusters, but in the second case, the overlap between the green and red components dominates. In both cases, between-cluster variability is dominated by its within-cluster cousin fairly soon among the PC's. On the other hand, Figure 4, (c) and (d), indicates that between-cluster variability continues to dominate within-cluster variability even for higher PC's. Also, the mixture components are much better separated on the whole for Figure 4, (c) and (d), than for Figure 4, (a) and (b), but the blue and cyan clusters dominate the average overlap in Figure 4(d).

3.2.4 Other Properties of Simulation Algorithm

We also explored variability in simulated mixture distributions in higher dimensions. We generated 25 datasets each, for different combinations of $(\bar{\omega}, \check{\omega})$ from 7-, 9-, and 11-component multivariate normal mixtures in 5-, 7-, and 10-dimensional spaces, respectively. For every p, we set $\bar{\omega}$ to be 0.05, 0.01 and $\check{\omega}$ to be such that the empirical coefficient of variation $(\omega_\sigma/\bar{\omega})$ in overlap between components was 1 and 2. $\pi_\wedge$'s were stipulated to be 0.06, 0.04, and 0.03 for the 5-, 7-, and 10-dimensional experiments, respectively, so that there would be at least $p+1$ observations from each of the 7, 9, and 11 components with very high probability, when we drew samples of size $n = 500$, 1000, and 1500, respectively, in further experiments in Section 4. Expected Kullback–Leibler divergences calculated for each pair of mixture model densities obtained for each setting, and detailed in Table S-1 (supplement), show substantial variability in simulated models.

We also analyzed the number of times initial realizations had to be discarded and regenerated to guarantee Step 3 of the algorithm in Section 2.2.1 or Step 2 of the algorithm in Section 2.2.2 in the generation of these sets of mixture models. The results are reported in

Table S-2 in the supplement. As expected, there is a large number of regenerations needed for larger numbers of clusters and when the average overlap is closer to the maximum overlap. We note, however, that these regenerations are at the trial stage of each algorithm, with evaluations and regenerations done in each case before the iterative stage is entered into.

4. UTILITY OF PROPOSED ALGORITHM

Here we demonstrate utility of the proposed algorithm in evaluating some initialization methods for Gaussian finite mixture modeling and model-based clustering algorithms. Initialization is crucially important in the performance of many iterative optimization-partitioning algorithms with a number of largely heuristic approaches in the literature. We evaluate four initialization methods that have previously shown promise in model-based clustering. Our objective is to illustrate the utility of our algorithm in comparing and contrasting performance under different scenarios and see if recommendations, if any, can be made with regard to each method.

Our first initialization approach was the *em-EM* algorithm of Biernacki, Celeux, and Govaert (2003), so named because it uses several *short runs* of the EM, each initialized with a *valid* (in terms of existence of likelihood) random start as parameter estimates. Each short run stops the EM algorithm, initialized with the above random start, according to a lax convergence criterion. The procedure is repeated until an overall prespecified number of total short run iterations is exhausted. At this point, the solution with the highest log-likelihood value is declared to be the initializer for the long EM, which then proceeds to termination using the desired stringent convergence criterion. We used p^2 total short run iterations and a lax convergence criterion of no more than one percent relative change in log-likelihood for our experiments. Our second approach, the *Rnd-EM*, used Maitra (2009)'s proposed modification that eliminates each short EM step by just evaluating the likelihood at the initial valid random start and choosing the parameters with highest likelihood. With the short *em* run eliminated, the best initializer is thus obtained from a number of candidate points equivalent to the total number (p^2) of short run iterations. Our third approach used Mclust which integrates model-based hierarchical clustering with a likelihood gain merge criterion to determine an initial grouping (Fraley and Raftery 2006) from where initializing parameter estimates are fed into the EM algorithm. Finally, we used the multistaged approach proposed by Maitra (2009) in providing initial values for the EM algorithm. This is a general strategy proposed by him to find a large number of local modes, and to choose representatives from the K most widely separated ones. In our studies here, we have used the algorithm specifically as implemented in Maitra (2009).

Our demonstration suite used simulated datasets generated from the mixture models in Section 3.2.4. In this utility demonstrator, we assumed that K was known in all cases. We calculated $\mathcal{R}$ for each derived classification relative to the grouping obtained by classifying the datasets with EM initialized with the true parameter values. In each case, we also evaluated estimation performance of these converged EM results by calculating the

expected Kullback–Leibler divergence ($\mathcal{KL}$) of the estimated model relative to the true mixture model.

Table 2 summarizes results for the 5- and 10-dimensional experiments. Results for the 7-dimensional experiments are provided in Table S-3. It is clear that Mclust outperforms the other algorithms for cases with small average overlap. In general, it also does better than *emEM* or *Rnd-EM* when the variation in overlap between clusters is higher. Further, there is some degradation in Mclust's performance with increasing dimensionality. On the other hand, *emEM* and *Rnd-EM* do not appear to be very different from each other: the latter seems to perform better with higher dimensions. This may be because higher dimensions mean a larger number of short run iterations (for *emEM*) and random starts (for *Rnd-EM*) in our setup. This suggests that using computing power to evaluate more potential starting points may be more profitable than using it to run short run iterations. It is significant to note, however, that both *emEM* and *Rnd-EM* are outclassed by Mclust in all cases when cluster pairs have low average overlap. We note that the multistaged approach of Maitra (2009) very rarely outperforms the others. This is a surprising finding in that it contradicts the findings of Maitra (2009) where performance was calibrated on simulation experiments indexed using exact-c-separation. Finally, we note that while there is not much difference between performance in clustering and maximum likelihood parameter estimation in mixture models, with both $\mathcal{R}$ and $\mathcal{KL}$ having very similar trends, they are not completely identical. This is a pointer to the fact that what may be the perfect sauce for the goose (parameter estimation in finite mixture models) may not be perfect for the gander (model-based clustering) and vice versa.

The above is a demonstration of the benchmarking that can be made possible using our cluster simulation algorithm. We note that Mclust is the best performer when clusters are well separated and *Rnd-EM* and *emEM* are better performers when clusters are less well separated. Note that *Rnd-EM* and *emEM* perform similarly and often split honors in many cases. There is thus not much distinction between these two methods, even though *Rnd-EM* seems to be better at more efficient use of computing resources. Thus, Mclust may be used when clusters are a priori known to be well separated. When clusters are known to be poorly separated, *Rnd-EM* may be used. Otherwise, if separation between clusters is not known, a better option may be to try out *Rnd-EM* and Mclust and choose the one with the highest observed log-likelihood.

5. DISCUSSION

In this article, we develop methodology and provide an algorithm for generating clustered data according to desired cluster overlap characteristics. Such characteristics serve as a surrogate for clustering difficulty and can lead to better understanding and calibration of the performance of clustering algorithms. Our algorithm generates clusters according to exactly prespecified average and maximal pairwise overlap. We illustrate our algorithm in the context of mixture distributions and in sample two- and five-dimensional experiments with six components. We also introduce the parallel distribution plot to display mixture distributions in high dimensions. An R package implementing the algorithm has been de-

Table 2. Adjusted Rand ($\mathcal{R}$) similarity measures of EM-cluster groupings and expected Kullback–Leibler divergences ($\mathcal{KL}$) of estimated mixture model densities obtained, using different initialization strategies (*Starts*), over 25 replications for different overlap characteristics. Summary statistics represented are the median $\mathcal{R}$ ($\mathcal{R}_{1/2}$) and median $\mathcal{KL}$ ($\mathcal{KL}_{1/2}$) and corresponding interquartile ranges ($\mathcal{I}^{\mathcal{R}}_{1/2}$ and $\mathcal{I}^{\mathcal{KL}}_{1/2}$). Finally, $\mathcal{R}_{\#1}$ and $\mathcal{KL}_{\#1}$ represent the number of replications (out of 25) for which the given initialization strategy did as well (in terms of higher $\mathcal{R}$ and lower $\mathcal{KL}$) as the best strategy.

Starts		$p=5, k=7, n=500$								$p=10, k=11, n=1,500$							
	$\bar{\omega}$	0.15	0.10	0.05		0.01		0.001		0.15	0.10	0.05		0.01		0.001	
	$\check{\omega}$	0.49	0.319	0.15	0.27	0.03	0.05	0.003	0.005	0.62	0.41	0.20	0.44	0.04	0.08	0.004	0.008
emEM	$\mathcal{R}_{1/2}$	0.35	0.45	0.69	0.67	0.79	0.81	0.89	0.88	0.25	0.34	0.57	0.55	0.84	0.86	0.87	0.92
	$\mathcal{I}^{\mathcal{R}}_{1/2}$	0.13	0.15	0.15	0.14	0.19	0.14	0.16	0.20	0.18	0.15	0.08	0.08	0.07	0.10	0.12	0.07
	$\mathcal{R}_{\#1}$	6	7	10	7	5	2	6	3	13	11	13	9	8	10	2	0
	$\mathcal{KL}_{1/2}$	0.30	0.30	0.32	0.28	0.29	0.29	0.34	0.30	0.46	0.49	0.48	0.49	0.52	0.47	0.47	0.50
	$\mathcal{I}^{\mathcal{KL}}_{1/2}$	0.05	0.08	0.05	0.07	0.17	0.11	0.09	0.14	0.05	0.07	0.05	0.07	0.09	0.08	0.14	0.12
	$\mathcal{KL}_{\#1}$	10	9	11	10	6	5	1	1	10	10	14	11	5	7	0	1
Rnd-EM	$\mathcal{R}_{1/2}$	0.37	0.46	0.64	0.61	0.84	0.83	0.89	0.91	0.23	0.31	0.58	0.58	0.83	0.78	0.91	0.92
	$\mathcal{I}^{\mathcal{R}}_{1/2}$	0.18	0.12	0.12	0.18	0.16	0.13	0.17	0.13	0.11	0.19	0.15	0.11	0.11	0.11	0.14	0.12
	$\mathcal{R}_{\#1}$	13	8	6	10	5	2	3	6	6	11	11	12	8	0	1	0
	$\mathcal{KL}_{1/2}$	0.30	0.29	0.32	0.30	0.28	0.29	0.33	0.34	0.47	0.50	0.50	0.49	0.47	0.46	0.48	0.46
	$\mathcal{I}^{\mathcal{KL}}_{1/2}$	0.07	0.04	0.07	0.09	0.10	0.05	0.19	0.07	0.05	0.06	0.05	0.06	0.09	0.07	0.07	0.14
	$\mathcal{KL}_{\#1}$	14	9	9	7	3	3	5	4	7	12	8	10	11	4	2	1

Table 2. (Continued.)

Starts		$p=5, k=7, n=500$								$p=10, k=11, n=1,500$							
	$\bar{\omega}$	0.15	0.10	0.05		0.01		0.001		0.15	0.10	0.05		0.01		0.001	
	$\check{\omega}$	0.49	0.319	0.15	0.27	0.03	0.05	0.003	0.005	0.62	0.41	0.20	0.44	0.04	0.08	0.004	0.008
Mclust	$\mathcal{R}_{1/2}$	0.28	0.40	0.65	0.62	0.95	0.96	1.0	1.0	0.16	0.25	0.45	0.48	0.83	0.84	1.0	1.0
	$\mathcal{I}^{\mathcal{R}}_{1/2}$	0.16	0.13	0.19	0.14	0.14	0.15	0.00	0.01	0.16	0.21	0.16	0.20	0.09	0.10	0.01	0.00
	$\mathcal{R}_{\#1}$	5	6	9	6	15	19	23	20	5	2	0	3	9	12	22	25
	$\mathcal{KL}_{1/2}$	0.35	0.33	0.35	0.32	0.22	0.24	0.18	0.19	0.50	0.58	0.55	0.55	0.47	0.42	0.33	0.31
	$\mathcal{I}^{\mathcal{KL}}_{1/2}$	0.08	0.08	0.05	0.07	0.10	0.07	0.04	0.05	0.09	0.09	0.12	0.06	0.07	0.10	0.08	0.05
	$\mathcal{KL}_{\#1}$	0	2	4	6	15	17	17	13	5	2	2	3	8	13	23	23
Multi-staged	$\mathcal{R}_{1/2}$	0.22	0.40	0.50	0.52	0.68	0.70	0.86	0.91	0.05	0.12	0.36	0.42	0.64	0.70	0.78	0.78
	$\mathcal{I}^{\mathcal{R}}_{1/2}$	0.17	0.24	0.21	0.16	0.24	0.17	0.22	0.20	0.12	0.18	0.22	0.33	0.16	0.20	0.14	0.17
	$\mathcal{R}_{\#1}$	1	4	0	2	0	2	1	5	1	1	1	1	0	3	0	0
	$\mathcal{KL}_{1/2}$	0.35	0.32	0.42	0.36	0.42	0.41	0.35	0.30	0.56	0.58	0.65	0.58	0.63	0.62	0.77	0.73
	$\mathcal{I}^{\mathcal{KL}}_{1/2}$	0.16	0.09	0.23	0.12	0.20	0.17	0.34	0.22	0.85	0.08	0.17	0.21	0.12	0.20	0.43	0.28
	$\mathcal{KL}_{\#1}$	1	5	1	2	1	0	2	7	3	1	1	1	1	1	0	0

veloped and will be publicly released soon. Finally, we demonstrate potential utility of the algorithm in a test scenario where we evaluate performance of four initialization strategies in the context of EM-clustering in a range of clustering settings over several dimensions and numbers of true groups. This ability to study properties of different clustering algorithms and related strategies is the main goal for devising our suggested methodology.

One reviewer has asked why we define pairwise overlap in terms of the unweighted sum $\omega_{i|j} + \omega_{j|i}$, rather than the weighted sum $(\pi_j\omega_{i|j} + \pi_i\omega_{j|i})/(\pi_i + \pi_j)$. We preferred the former because weighting would potentially damp out the effect of misclassification of components with small relative mixing proportions, even though they affect clustering performance (as indexed by $\mathcal{R}$). We have not investigated using the weighted sum as an overlap measure; it would be interesting to study its performance. We note, however, that Figure 1 indicates that $\mathcal{R}$ is very well tracked by our defined overlap measure. Also, our algorithm was developed in the context of soft clustering using Gaussian mixture models. However, the methodology developed here can be very easily applied to the case of hard clustering with a fixed partition Gaussian clustering model. A third issue not particularly emphasized in this article is that Theorem 1 can help summarize the distinctiveness between groups obtained by Gaussian clustering of a given dataset. Thus, clustered data can be analyzed to study how different one cluster is from another by measuring the overlap between them. As mentioned in Section 2.2.3, desired scatter can also be very easily incorporated in our algorithm. In this article, we characterize cluster overlap in terms of the average and maximum pairwise overlap. It would be interesting to see if summaries based on other properties of overlap could also be developed in a practical setting. Finally, our algorithm is specific to generating Gaussian mixtures with desired cluster overlap properties and can not readily handle heavy-tailed or skewed distributions. One possibility may be to use the Box–Cox transformation in each dimension to match desired skewness and (approximately) desired overlap characteristics. Thus, while the methodology suggested in this article can be regarded as an important contribution to evaluating clustering algorithms, quite a few issues meriting further attention remain.

SUPPLEMENTAL MATERIALS

Additional Investigations: Sections S-1–S-3 along with Figures S-1–S-4 and Tables S-1–S-3 are in the supplementary materials archive of the journal website. (appendix.pdf)

R-package: R-package "MixSim" implementing this algorithm is available on C-RAN. (MixSim_0.1-04.tar.gz)

ACKNOWLEDGMENTS

We thank the editor, an associate editor, and two referees whose helpful suggestions and insightful comments greatly improved the quality of this manuscript. This research was supported in part by the National Science Foundation CAREER grant DMS-0437555 and by the National Institutes of Health DC-0006740.

[Received May 2008. Revised September 2009.]

REFERENCES

Anderson, E. (1935), "The Irises of the Gaspe Peninsula," *Bulletin of the American Iris Society*, 59, 2–5. [354,362]

Atlas, R., and Overall, J. (1994), "Comparative Evaluation of Two Superior Stopping Rules for Hierarchical Cluster Analysis," *Psychometrika*, 59, 581–591. [355]

Bartlett, M. S. (1939), "A Note on Tests of Significance in Multivariate Analysis," *Proceedings of the Cambridge Philosophical Society*, 35, 180–185. [358]

Biernacki, C., Celeux, G., and Govaert, G. (2003), "Choosing Starting Values for the EM Algorithm for Getting the Highest Likelihood in Multivariate Gaussian Mixture Models," *Computational Statistics and Data Analysis*, 413, 561–575. [370]

Blashfield, R. K. (1976), "Mixture Model Tests of Cluster Analysis—Accuracy of 4 Agglomerative Hierarchical Methods," *Psychological Bulletin*, 83, 377–388. [355]

Box, G. E. P., and Draper, N. R. (1987), *Empirical Model-Building and Response Surfaces*, New York: Wiley. [355]

Brodatz, P. (1966), *A Photographic Album for Artists and Designers*, New York: Dover. [354,362]

Campbell, N. A., and Mahon, R. J. (1974), "A Multivariate Study of Variation in Two Species of Rock Crab of Genus *Leptograsus*," *Australian Journal of Zoology*, 22, 417–425. [354,362]

Dasgupta, S. (1999), "Learning Mixtures of Gaussians," in *Proc. IEEE Symposium on Foundations of Computer Science*, Washington: IEEE Computer Society, pp. 633–644. [355,361]

Davies, R. (1980), "The Distribution of a Linear Combination of χ^2 Random Variables," *Applied Statistics*, 29, 323–333. [356,358,359]

Everitt, B. S., Landau, S., and Leesem, M. (2001), *Cluster Analysis* (4th ed.), New York: Hodder Arnold. [354]

Forina, M. (1991), *PARVUS—An Extendible Package for Data Exploration, Classification and Correlation*, Genoa, Italy: Institute of Pharmaceutical and Food Analysis and Technologies. [362]

Forsythe, G., Malcolm, M., and Moler, C. (1980), *Computer Methods for Mathematical Computations*, Moscow: Mir. [360]

Fraley, C., and Raftery, A. E. (2002), "Model-Based Clustering, Discriminant Analysis, and Density Estimation," *Journal of the American Statistical Association*, 97, 611–631. [354]

——— (2006), "MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering," Technical Report 504, University of Washington, Dept. of Statistics, Seattle, WA. [364,370]

Gold, E. M., and Hoffman, P. J. (1976), "Flange Detection Cluster Analysis," *Multivariate Behavioral Research*, 11, 217–235. [355]

Hartigan, J. (1985), "Statistical Theory in Clustering," *Journal of Classification*, 2, 63–76. [354]

Hubert, L., and Arabie, P. (1985), "Comparing Partitions," *Journal of Classification*, 2, 193–218. [362,364]

Inselberg, A. (1985), "The Plane With Parallel Coordinates," *The Visual Computer*, 1, 69–91. [368]

Kaufman, L., and Rousseuw, P. J. (1990), *Finding Groups in Data*, New York: Wiley. [354]

Kettenring, J. R. (2006), "The Practice of Cluster Analysis," *Journal of Classification*, 23, 3–30. [354]

Kuiper, F. K., and Fisher, L. (1975), "A Monte Carlo Comparison of Six Clustering Procedures," *Biometrics*, 31, 777–783. [355]

Likas, A., Vlassis, N., and Verbeek, J. J. (2003), "The Global k-Means Clustering Algorithm," *Pattern Recognition*, 36, 451–461. [355]

Maitra, R. (2009), "Initializing Partition-Optimization Algorithms," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 6, 144–157. [355,361,370,371]

Maitra, R., and Ramler, I. P. (2009), "Clustering in the Presence of Scatter," *Biometrics*, 65, 341–352. [361]

McIntyre, R. M., and Blashfield, R. K. (1980), "A Nearest-Centroid Technique for Evaluating the Minimum-Variance Clustering Procedure," *Multivariate Behavioral Research*, 15, 225–238. [355]

McLachlan, G. J., and Basford, K. E. (1988), *Mixture Models: Inference and Applications to Clustering*, New York: Marcel Dekker. [354]

Milligan, G. W. (1985), "An Algorithm for Generating Artificial Test Clusters," *Psychometrika*, 50, 123–127. [355]

Murtagh, F. (1985), *Multi-Dimensional Clustering Algorithms*, Würzburg: Springer-Verlag. [354]

Nakai, K., and Kinehasa, M. (1991), "Expert System for Predicting Protein Localization Sites in Gram-Negative Bacteria," *PROTEINS: Structure, Function, and Genetics*, 11, 95–110. [362]

Newman, D. J., Hettich, S., Blake, C. L., and Merz, C. J. (1998), "UCI Repository of Machine Learning Databases," University of California, Irvine, Dept. of Information and Computer Sciences, available at *http://www.ics.uci.edu/~mlearn/MLRepository.html*. [362]

Press, S. J. (1966), "Linear Combinations of Non-Central Chi-Square Variates," *Annals of Mathematical Statistics*, 37, 480–487. [359]

Price, L. J. (1993), "Identifying Cluster Overlap With Normix Population Membership Probabilities," *Multivariate Behavioral Research*, 28, 235–262. [355]

Qiu, W., and Joe, H. (2006a), "Generation of Random Clusters With Specified Degree of Separation," *Journal of Classification*, 23, 315–334. [355,356]

——— (2006b), "Separation Index and Partial Membership for Clustering," *Computational Statistics and Data Analysis*, 50, 585–603. [355]

Ramey, D. B. (1985), "Nonparametric Clustering Techniques," in *Encyclopedia of Statistical Science*, Vol. 6, New York: Wiley, pp. 318–319. [354]

Ruspini, E. (1970), "Numerical Methods for Fuzzy Clustering," *Information Science*, 2, 319–350. [354,362]

Slater, L. J. (1960), *Confluent Hypergeometric Functions*, London: Cambridge University Press. [359]

Steinley, D., and Henson, R. (2005), "OCLUS: An Analytic Method for Generating Clusters With Known Overlap," *Journal of Classification*, 22, 221–250. [355]

Tseng, G. C., and Wong, W. H. (2005), "Tight Clustering: A Resampling-Based Approach for Identifying Stable and Tight Patterns in Data," *Biometrics*, 61, 10–16. [361]

Verbeek, J., Vlassis, N., and Krose, B. (2003), "Efficient Greedy Learning of Gaussian Mixture Models," *Neural Computation*, 15, 469–485. [355]

Verbeek, J., Vlassis, N., and Nunnink, J. (2003), "A Variational EM Algorithm for Large-Scale Mixture Modeling," in *Annual Conference of the Advanced School for Computing and Imaging*, The Netherlands: University of Amsterdam, pp. 1–7. [355]

Waller, N. G., Underhill, J. M., and Kaiser, H. A. (1999), "A Method for Generating Simulated Plasmodes and Artificial Test Clusters With User-Defined Shape, Size, and Orientation," *Multivariate Behavioral Research*, 34, 123–142. [355]

Wegman, E. (1990), "Hyperdimensional Data Analysis Using Parallel Coordinates," *Journal of the American Statistical Association*, 85, 664–675. [368]