Controlling 2D Artificial Data Mixtures Overlap

Mohammed Ouali¹,², Walid Mahdi¹, Radhwane Gharbaoui³, Seyyid Ahmed Medjahed³

¹ College of Computers and Information Technology, Taif University, Arabia Saudi
² Thales Canada Inc., Canada
³ Université des Sciences et Technologie, Département d’Informatique, Algérie

{mouali, wmahdi}@tu.edu.sa, {radouane.gharbaoui, sa.medjahed}@univ-usto.dz

Abstract. Clustering methods are used for identifying groups of similar objects considered as homogenous set. Unfortunately, analytic performance evaluation of clustering methods is a difficult task because of their ad-hoc nature. In this paper, we propose a new test case generator of artificial data for 2 dimensional Gaussian mixtures. The proposed generator has two interesting advantages: the first one is its ability to produce simulated mixture for any number of components, while the second one resides in the fact that it formally quantifies the overlap rate which allows us to add some complexity to the data. Clustering algorithms and validity indices behavior is also analyzed by changing the overlap rate between clusters.

Keywords. Clustering algorithms, unsupervised learning, Gaussian mixture, Gaussian components overlap.

1 Introduction

Clustering methods are defined as unsupervised learning processes used to divide a set of observations into clusters [36, 38, 31, 28]. Clusters are groups of similar observations which are sufficiently far from each other. Several clustering methods are defined in the literature and all share a common nature - the difficulty in their analytical evaluation [39, 3, 9, 23, 14].

The discrete frequencies of observations and the measures of similarities between entities and clusters produce many local minima which disturb the process of clustering to converge to the correct results. Therefore, one avenue is to evaluate clustering algorithms using artificially constructed data. Many authors have categorized unsupervised classification based on criteria such as similarity or dissimilarity measures, the nature of data, and the function to be optimized [18]. Based on evaluation, the two principal categories are hierarchical methods and mobile centers methods.

The ultra-metric inequality is one of the most often used methods to generate artificial data to evaluate hierarchical algorithms [10, 22]. A large number of popular methods are based on employing the mixture model and particularly the Gaussian mixtures [15, 2, 25, 26]. The mixture models must satisfy some properties that conform with the clustering methods [24]. These patterns are summarized in two criteria: the internal cohesion and the external isolation.

Internal cohesion ensures observations within the same cluster have similar properties. External isolation ensures observation from different clusters are very dissimilar. Several works in the literature considered the Gaussian distribution as design block for clustering algorithms due its...
well-known properties [15, 2]. On the other hand, several approaches have been proposed to generate artificial data. Salem and Nandy [30] proposed different structures for producing artificial observations in 2D spaces. The main rule to preserve the internal cohesion of the components mixture is to introduce empty space between the clusters despite the fact that empty space is not a sufficient condition to guarantee external isolation. In the case where the mixture components have close enough centers, no clustering method has the ability to identify the components [30]. In [34, 35], well separated data is generated for 2D and 3D cases.

The two criteria characterizing the cluster structure are strongly respected. Milligan [23] developed an algorithm for generating artificial data but was only able to avoid the total overlap for the first dimension. The claim is that avoiding overlap for the first dimension allows by transitivity to avoid total overlap for the rest of the dimensions [8, 29]. Milligan’s algorithm is verified by visual inspection. Kuiper and Fisher [20] and Bayne et al. [5] directly manipulated a variable which measured certain parameters as separability for a simple covariance matrix of normal clusters. Blashfield [6] and Edelbrock [13] used unconstrained multivariate Gaussian mixture with fairly complex covariance structures.

This allows to obtain cluster structure and the clusters are well separated [14]. Other authors have inserted noise in well separated data to add some complexity to the obtained simulated data [30, 34, 35]. Baudry et al. [4] proposed a verification method to estimate the Gaussian mixture model. This work clearly distinguishes cluster structures of the mixture where the components are well separated from the Gaussian mixtures in case of total overlap. In [17, 37], the authors proposed a new artificial data generator that embeds the notion of the rate of overlap for uncorrelated 2D artificial Gaussian data.

In this paper, we propose a new automatic method for generating artificial data by controlling mixture components overlap. This work tackles two main problems: the design of an artificial data generator for correlated 2D data, and the study of the behavior of clustering algorithms and their respective validity indices by varying the rate of overlap between the mixture components. We are interested in correlated data because of the growing number of applications in computer vision and image processing, as clustering is used as the core solution to solve problems such as segmentation and image matching. In these applications, correlated data that revealed useful when combined and could be used in the clustering process are pixel gray-level, local window gray-level, and local variance.

In this paper, we will show how the overlap rate is quantified and its use as the basis block in the artificial data generator. The rest of this paper is organized as follows: section 2 briefly presents the Gaussian mixture; section 3 deals with components separation; section 4 presents the quantification of component overlap; in section 5, the control of overlap is developed; the generation algorithm and the experimental results are shown in sections 6 and 7 respectively. Finally, the conclusion is drawn with some perspectives.

2 Bivariate Correlated Gaussian Mixture Model

Mixture models are widely used in many applications because many real and natural phenomena as well as sets of data in many disciplines are based on such distributions [14, 1, 3, 25, 30]. A mixture of M Gaussian 2D components is given by:

$$P(x, y) = \sum_{j=1}^{M} \kappa_j G_j(x, y, \theta_j),$$

where \(\sum_{j=1}^{M} \kappa_j = 1\) and \(\theta_j = (\mu_{xj}, \mu_{yj}, \sigma_{xj}, \sigma_{yj}, \rho_j)\) denotes the parameters of the \(j\)th distribution \(G_j\). \(G_j\) is given by:

$$G_j(x, y) = A \exp\left(-\frac{1}{2(1-\rho_j^2)} \left[ \frac{t_1^2}{\sigma_{xj}^2} + \frac{t_2^2}{\sigma_{yj}^2} - 2\rho_{jt_1t_2} \right] \right),$$

where \(A = \frac{1}{2\pi\sigma_{xj}\sigma_{yj}\sqrt{1-\rho_j^2}}\) is real and strictly positive. \(t_1 = (x - \mu_{xj})\) and \(t_2 = (y - \mu_{yj})\). \(\mu_{xj}, \mu_{yj}\) are the component center coordinates. \(\sigma_{xj}\) and \(\sigma_{yj}\) are the standard deviations of the...
first and second dimension respectively. \( \rho \) is the correlation coefficient between the two dimensions \( X \) and \( Y \).

### 3 Well Separated Components

Initially, the clustering methods and the validity indices were evaluated by using well separated data before using any other simulated data. Most works are not based on a formal way to generate artificial data and the main technique to construct isolated mixture components is visual inspection [35, 34, 11, 4].

The objective is to propose a definition that helps qualify and quantify well separated components by involving all the mixture parameters. Mixture components are considered well separated if they exhibit a minimum overlap between clusters [25, 26]; we define:

\[
\begin{align*}
    x_{int} &= \mu_1 + 4\sigma_1, \\
    y_{int} &= \mu_2 - 4\sigma_2,
\end{align*}
\]

where \( x_{int} \) is not really the intersection point, but for a value sufficiently far from the centers of the two components, \( x_{int} \) is approximated to be the intersection point. To be more precise in our description, \( x_{int} \) is the unique intersection point between \( C_1 \) and \( C_2 \) where \( C_1 \) (respectively \( C_2 \)) is the projection of the intersection point between component \( \Gamma_1 \) (respectively \( \Gamma_2 \)) and the line \( x_{int} \) : \( y = \frac{\mu_2 - \mu_1}{2\sigma_1}e^{-8} \) (\( \Delta_2 : y = \frac{\mu_2 - \mu_1}{2\sigma_2}e^{-8} \)). But, for \( x_{int} \) value sufficiently far from the center of the two components, \( x_{int} \) is approximated to take the form of the equation 1. In a Gaussian cluster, 99.7% of the observations belong to the interval \( [\mu - 3\sigma, \mu + 3\sigma] \), which indicates that the above minimum definition implies the presence of empty spaces between data.

In [17], we presented in 2D the minimum overlap between two components \( \Gamma_1 \) and \( \Gamma_2 \) so that the intersection point after the projection satisfies:

\[
\begin{align*}
    \Gamma_1(x_{int}, y_{int}) &= \frac{\kappa_1}{2\pi\sigma_1\sigma_1\sqrt{1-\rho_1^2}}e^{-8}, \\
    \Gamma_2(x_{int}, y_{int}) &= \frac{\kappa_2}{2\pi\sigma_2\sigma_2\sqrt{1-\rho_2^2}}e^{-8}.
\end{align*}
\]

As in the 1D case, more than 99.7% of the observations are located inside the ellipse defined by the intersection of the plane defined by \( z_1 = \frac{\kappa_1}{2\pi\sigma_1\sigma_1\sqrt{1-\rho_1^2}}e^{-8} \) (respectively \( z_2 = \frac{\kappa_2}{2\pi\sigma_2\sigma_2\sqrt{1-\rho_2^2}}e^{-8} \)) and \( \Gamma_1 \) (respectively \( \Gamma_2 \)), where the condition of minimum overlap for the 2D data guarantees the presence of empty space between the data.

The probability density function of the generated data (pdf) is constrained to have the same configuration for the well separated components so that:

**Definition 1:** Two adjacent Gaussian components \( \Gamma_1(\kappa_1, \mu_x, \mu_y, \sigma_x, \sigma_y, \rho_1) \) and \( \Gamma_2(\kappa_2, \mu_x, \mu_y, \sigma_x, \sigma_y, \rho_2) \) are well separated if the intersection point between \( C_1 \) and \( C_2 \) is a unique point, where \( C_1 \) is the projection of the intersection points between \( \Gamma_1 \) and the plane \( T_1 : z = \frac{\kappa_1}{2\pi\sigma_1\sigma_1\sqrt{1-\rho_1^2}}e^{-8} \), and \( C_2 \) is the projection of the intersection points between \( \Gamma_2 \) and the plane \( T_2 : z = \frac{\kappa_2}{2\pi\sigma_2\sigma_2\sqrt{1-\rho_2^2}}e^{-8} \).

Formally, \( \Gamma_1 \) and \( \Gamma_2 \) are well separated if:

\[
\begin{align*}
    \Gamma_1(x_{int}, y_{int}) &= \frac{\kappa_1}{2\pi\sigma_1\sigma_1\sqrt{1-\rho_1^2}}e^{-8}, \\
    \Gamma_2(x_{int}, y_{int}) &= \frac{\kappa_2}{2\pi\sigma_2\sigma_2\sqrt{1-\rho_2^2}}e^{-8},
\end{align*}
\]

where \( (x_{int}, y_{int}) \) is the coordinate of the highest intersection point from among the infinity of intersection points between the two components. Figure 1 shows a mixture of two well separated components.
3.1 Components Overlap

During the generation of a large set of data, it is important to ensure that the generated mixture components are not in a case of total overlap. Components in a case of total overlap violate the two criteria of internal cohesion and external isolation.

To better explain the meaning of total overlap, let us examine the example of figure 2. In figure 2 (a), the mixture is composed of three components; however, only two are visible; the total overlap can only be detected by visual inspection. A case of maximum overlap is shown in figure 2 (b). We can still distinguish that there are three components. In figure 2 (c), there is a partial overlap between the three components of the mixture. It is clear that the mixture is composed of three components:

It is meaningless to evaluate clustering algorithms on total overlapped structures.

3.2 Overlap Between Two Equivalent Bivariate Gaussian Components

In order to control components overlap, a formal quantification is needed.

In a bivariate space, let us consider two components $\Gamma_1(\mu_x, \mu_y, \sigma_x, \sigma_y, \rho_1)$ and $\Gamma_2(\mu_x, \mu_y, \sigma_x, \sigma_y, \rho_2)$, where $(\mu_x, \mu_y)$ and $(\mu_x, \mu_y)$ represent the centers of the first and the second components. $\sigma_x$ and $\sigma_y$ represent the standard deviations along each axis; $\rho_1$ and $\rho_2$ are the correlation coefficients and satisfy the equality $|\rho_1| = |\rho_2|$.

For two equivalent components, we propose the following condition for the maximum overlap:
\[ \Gamma_1(x_{int}, y_{int}) = \Gamma_2(\ldots) = \frac{0.5}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} e^{-1/2}, \]

(4)

where \((x_{int}, y_{int})\) represents the coordinate of the highest intersection point that has the highest value. Figure 3 illustrates the three situations related to our condition. In figure 3 (a), the value at the intersection point is higher than the value of the condition given in equation (4), which is indicative of a case of total overlap; it is impossible to visually distinguish between the two mixture components. In figure 3 (b), the intersection point obeys the condition (4). We consider this situation as the case of maximum overlap or a limit case between the total and the partial overlap. In figure 3 (c), it is clear the mixture consists of two components. The two components are in partial overlap.

The value at the intersection point is lower than the value given in condition (4). This results in the relationship between the visual inspection and the formal quantification. We will propose in the next section a definition characterizing the overlap cases. In the rest of this paper, we will use the notation \(\Gamma_i(\kappa_i, \mu_{xi}, \mu_{yi}, \sigma_{xi}, \sigma_{yi}, \rho_i)\) to describe the parameters of the \(i^{th}\) component \(\Gamma_i\) where \(\kappa_i\) denotes the mixture coefficient, \((\mu_{xi}, \mu_{yi})\) are the coordinates of the components' centers, \(\sigma_{xi}\) and \(\sigma_{yi}\) are the standard deviations, \(\rho_i\) is the coefficient of correlation between the two component dimensions and \(S_i = \kappa_i\frac{\sigma_{xi}}{\sigma_{yi}\sqrt{1-\rho^2}} e^{-1/2}\).

4 Formal Quantification of the Overlap

We propose the definition of the maximum overlap. Later we formalize the degree of overlap by the notion of rate. This definition is similar to that proposed in [17] except that in our case, the definitions are more general in order to support correlated and uncorrelated data. The overlap between components must be controlled to avoid the case of total overlap. We consider the overlap only between the two adjacent components. We will exploit the results of the previous section to propose the definitions.

4.1 Maximum Overlap

The maximum overlap is considered as a limit between the undesirable case of total overlap and the case of partial overlap. The condition of equation (4) is extended to support non-equivalent components and we set the following definition.

**Definition 2:** Two adjacent Gaussian bivariate components \(\Gamma_1\) and \(\Gamma_2\) are in case of maximum overlap if the value at the highest intersection point \(\Gamma_1(x_{int}, y_{int}) = \min(S_1, S_2)\). Figure 4 illustrates an example of maximum overlap between two Gaussian components.

4.2 Rate of Overlap

In the literature, the notion of overlap is not quantified in a way that an artificial data can be constructed. On the other hand, there are many indices proposed to measure the shared observations or resemblance between clusters. For the most popular model, the Gaussian model, an interesting description of the frequently used indices for computing the overlap rate between clusters is presented in [12, 32]. The Mahalanobis distance, \(D_{Mah} = ((\mu_1 - \mu_2)^T\Sigma^{-1}(\mu_1 - \mu_2))^{1/2}\), assumes that the two clusters have the same covariance matrix and the same mixture coefficients [16]. The Bhattacharyya distance is an extension of the Mahalanobis distance, \(D_{Bhatt} = \frac{1}{2}(\mu_1 - \mu_2)^T[\Sigma_{12}^{1/2} - \Sigma_{22}^{1/2}]^{-1}(\mu_1 - \mu_2)^T[\Sigma_{21}^{1/2} - \Sigma_{11}^{1/2}]^{-1}(\mu_2 - \mu_1)^T\).
It is difficult to use such an index because of its computing complexity, so it replaced by its upper bound \(B_{Bhatt} = \sqrt{\alpha_1 \alpha_2} e^{-D_{Bhatt}}\) in practical applications [16]. Other measures use the PDF to extract a measure for the overlap and similarity between clusters, for example the Kullback-Leibler distance \(D_{KL} = p_1(x) \ln \left( \frac{p_1(x)}{p_2(x)} \right)\) [21]. The major inconvenience of this kind of index is that it is not symmetric. The proximity measures presented are relative and assume some conditions on the data which are in most cases simply not verified (like the equality of the components’ coefficients or matrix covariance).

We propose the definition of overlap rate \(\lambda\) by modeling the partial overlap. This concept is based on the following points:

— The rate of overlap takes values between 0 and 1, so that the value of 1 implies the presence of maximum overlap and the value of 0 implies that the two components are “well separated”.

— The overlap rate must include all the parameters of the two components: the mixture coefficients, the centers, the standard deviations and the coefficients of correlation.

**Definition 3:** The rate of overlap between two adjacent bivariate Gaussian components is defined as the ratio of the value at the highest intersection point to the value at the highest intersection point in the case of maximum overlap. Formally, the rate of overlap can be written as:

\[
\lambda = \frac{\min(S_1, S_2)}{\min(S_{1\text{max}}, S_{2\text{max}})}
\]

These three definitions are very interesting because they employ visual inspection as a basis for the generation and verification of artificial data. Additionally, the rate of overlap definition involves symmetrically all the parameters of the two adjacent Gaussian components. We propose an algorithm for generating artificial data in order to avoid the case of total overlap and control the overlap rate \(\lambda\). The parameters of the initial component are randomly generated and the parameters of the second component are computed in accordance with one of the three definitions depending on which case is to be reproduced.

**5 Controlling Mixture Overlap**

As mentioned above, we randomly generate the parameters of the first component - the mixture coefficients, the standard deviations and the coefficients of correlation with the other components. We also introduce the angles of intersection between the components randomly. The angles of intersection are used to measure the deviation of the intersection points from the \(x\) axis. After that, we fix the centers of the components, one at a time, according to the rate of overlap.

**5.1 Fixing Partial Overlap Rate**

For two components \(\Gamma_1(\kappa_1, \mu_{x1}, \mu_{y1}, \sigma_{x1}, \sigma_{y1}, \rho_1)\) and \(\Gamma_2(\kappa_2, \mu_{x2}, \mu_{y2}, \sigma_{x2}, \sigma_{y2}, \rho_2)\), we know all the parameters of \(\Gamma_1\) and \(\Gamma_2\) and we compute the center of the second component \((\mu_{x2}, \mu_{y2})\) according to the rate of overlap \(\lambda\). We apply the definition of the overlap rate on the two components. There are two cases: \(S_1 \geq S_2\) and \(S_1 < S_2\).

**Case 1:** For \(\Gamma_1\), after applying the overlap rate definition, we find:

\[
\Gamma_1(x_{int}, y_{int}) = S_2.
\]
which means that:

$$
\Gamma_i(x_{int}, y_{int}) = A_i \exp \left( -\frac{1}{2(1-\rho_i^2)} \left[ \frac{t_{i1}^2}{\sigma_{i1}^2} + \frac{t_{i2}^2}{\sigma_{i2}^2} - 2\rho_{i1}t_{i1}t_{i2} \right] \right)
= A_2 \exp(-1/2),
$$

where: $A_i = \frac{\kappa_i}{2\sigma_{i1}\sigma_{i2}\sqrt{1-\rho_i^2}}$ and $i \in \{1, 2\}$.

We have:

$$
a_1(x_{int} - \mu_{x1})^2 + b_1(y_{int} - \mu_{y1})^2 + c_1(x_{int} - \mu_{x1})(y_{int} - \mu_{y1}) - 1 = 0,
\tag{5}
$$

where:

$$
e_1 = 1 - 2 \ln \left( \frac{\lambda_{\kappa_1\sigma_{x1}\sigma_{y1}\sqrt{1-\rho_1^2}}}{\kappa_1\sigma_{x1}\sigma_{y1}\sqrt{1-\rho_1^2}} \right),
\quad a_1 = \frac{1}{1 - \rho_1^2}\sigma_{x1}^2 e_1,
\quad b_1 = \frac{1}{1 - \rho_1^2}\sigma_{y1}^2 e_1,
\quad c_1 = -\frac{1}{\sigma_{x1}\sigma_{y1}(1-\rho_1^2)} e_1.
\tag{6}
$$

From the inequality $S_1 \geq S_2$, we conclude that

$$
\frac{\kappa_1}{\sigma_{x1}\sigma_{y1}\sqrt{1-\rho_1}} \geq \frac{\kappa_2}{\sigma_{x2}\sigma_{y2}\sqrt{1-\rho_2}}.
$$

This means that $0 < \frac{\kappa_2}{\kappa_1} \frac{\sigma_{y1}}{\sigma_{y2}} \frac{\sqrt{1-\rho_1}}{\sqrt{1-\rho_2}} \geq 1$. With $0 < \lambda \geq 1$, it is clear that $e_1 > 0$. So, we deduce that $a_1$ and $b_1$ are also strictly positive.

For the second component, by applying the same reasoning, we find:

$$
a_2(x_{int} - \mu_{x2})^2 + b_2(y_{int} - \mu_{y2})^2 + c_2(x_{int} - \mu_{x2})(y_{int} - \mu_{y2}) - 1 = 0,
\tag{7}
$$

where:

$$
e_2 = 1 - 2 \ln \left(\frac{\lambda_{\kappa_2\sigma_{x2}\sigma_{y2}\sqrt{1-\rho_2^2}}}{\kappa_2\sigma_{x2}\sigma_{y2}\sqrt{1-\rho_2^2}}\right),
\quad a_2 = \frac{1}{1 - \rho_2^2}\sigma_{x2}^2 e_2,
\quad b_2 = \frac{1}{1 - \rho_2^2}\sigma_{y2}^2 e_2,
\quad c_2 = -\frac{1}{\sigma_{x2}\sigma_{y2}(1-\rho_2^2)} e_2.
\tag{8}
$$
e_2, a_2, b_2$ are also real and strictly positive.

**Case 2:** In this case, we find the same equations (5,7), but with these parameters for the first component:

$$
e_1 = 1 - 2 \ln(\lambda),
\quad a_1 = \frac{1}{(1-\rho_1^2)\sigma_{x1}^2 e_1},
\quad b_1 = \frac{1}{(1-\rho_1^2)\sigma_{y1}^2 e_1},
\quad c_1 = -\frac{1}{\sigma_{x1}\sigma_{y1}(1-\rho_1^2)} e_1.
\tag{9}
$$

and these parameters for the second component:

$$
e_2 = 1 - 2 \ln \left( \frac{\lambda_{\kappa_2\sigma_{x2}\sigma_{y2}\sqrt{1-\rho_2^2}}}{\kappa_2\sigma_{x2}\sigma_{y2}\sqrt{1-\rho_2^2}} \right),
\quad a_2 = \frac{1}{(1-\rho_2^2)\sigma_{x2}^2 e_2},
\quad b_2 = \frac{1}{(1-\rho_2^2)\sigma_{y2}^2 e_2},
\quad c_2 = -\frac{1}{\sigma_{x2}\sigma_{y2}(1-\rho_2^2)} e_2.
\tag{10}
$$

In this case, $a_1$, $b_1$, $e_1$, $a_2$, $b_2$ and $e_2$ are all real and strictly positive.

In the plane defined by the equation $(T)$: $z = \min(S_1, S_2)$, the two equations 5 and 7 are characteristic equations of two ellipses with centers respectively at $(\mu_{x1}, \mu_{y1})$ and $(\mu_{x2}, \mu_{y2})$. This means that fixing the center of the second component fixes the center of the second ellipse. First, we compute the value of the intersection point after we compute the center of the second component. We proceed to some transformations in the referential $(R)$, we will translate the referential after we rotate it so that the major axis of the ellipse will be parallel to the $X$ axis of the new referential.

We proceed to translate the referential $(R)$ by the vector $\bar{m}(\mu_{x1}, \mu_{y1})$. Equation (5) becomes:

$$
a_1 x_{int}^2 + b_1 y_{int}^2 + c_1 x_{int} y_{int} - 1 = 0.
\tag{11}
$$

We obtain an ellipse which center is the center of referential. After translating the referential, we proceed to the rotation in which the major axis of the ellipse will be parallel to the $X$ axis - let us call this new referential $R_1$. Figure 6 illustrates the referential and the angles used for the rotation. We consider the rotation angle $\phi_1$. Rotation in a bivariate space is given by:

$$
\left\{ \begin{array}{l}
x' = x \cos(\phi_1) - y \sin(\phi_1), \\
y' = x \sin(\phi_1) + y \cos(\phi_1),
\end{array} \right.
\tag{12}
$$
where $x', y'$ are the coordinates in the new referential. To have a referential in which the major axis of the ellipse is parallel to the $x$ axis, the characteristic equation of the ellipse must have the following form in the new referential:

$$a_1'x' + b_1'y' - 1 = 0,$$

where $a_1'$ and $b_1'$ are real and strictly positive because the rotation function is isometric: it preserves the distance which means that the ellipse stays an ellipse after rotation.

From equations 11, 12, and 13 and after some transformations, we have:

$$\begin{align*}
\phi_1 &= 0.5 \arctan \left( \frac{c_1}{b_1 - c_1} \right), \\
b_1' &= b_1 \cos^2(\phi_1) - a_1 \sin^2(\phi_1), \\
a_1' &= a_1 \cos^2(\phi_1) - b_1 \sin^2(\phi_1),
\end{align*}$$

where the angle $\theta$, chosen by the user, represents the deviation of the intersection point $P_{\text{int}}$ from the $X$ axis. The intersection angle $\theta_{\text{int}}$ is used for two reasons. The first one is to fix one solution for computing the center of the second component as there are an infinity of solutions in which the intersection point satisfies the condition imposed by the overlap rate; the second reason is to avoid the ternary overlap between three components. The intersection angle in the new referential ($R_1$), after the rotation, is given by:

$$\begin{align*}
\theta_{\text{int}} &= \phi_1 + \theta, \\
\text{if } \theta_{\text{int}} \geq \pi/2 &\text{ then } \theta_{\text{int}} = 4\pi/10, \\
\text{if } \theta_{\text{int}} \leq -\pi/2 &\text{ then } \theta_{\text{int}} = -4\pi/10.
\end{align*}$$

To avoid the ternary overlap, $P_{\text{int}}$ is treated in the interval $[-\pi/2, \pi/2]$. For this reason, we add the two conditions cited in the equation (15) and we choose the interval $[-4\pi/10, 4\pi/10]$ as a limit. In [17], there are no conditions on the intersection point interval because, for uncorrelated data, the mixture components have by definition their axes parallel to the referential axis and there is no need for rotation; the intersection angle $\theta_{\text{int}}$ is always within the interval $[-\pi/2, \pi/2]$. From the parametrical equation of the ellipse, $P_{\text{int}}$ coordinates $x_1$ and $y_1$ in the referential ($R_1$) are given by:

$$\begin{align*}
t &= \arctan \left( \frac{\sqrt{a_1}}{\cos(t)} \tan(\theta_{\text{int}}) \right), \\
x_1 &= \cos(t) \left( \frac{\sqrt{a_1}}{\cos(t)} \tan(\theta_{\text{int}}) \right), \\
y_1 &= \sin(t) \left( \frac{\sqrt{a_1}}{\cos(t)} \tan(\theta_{\text{int}}) \right).
\end{align*}$$

In order to compute the second component’s center, we need the value of the obliqueness at the intersection point. The obliqueness tangent is the tangent of the angle between the line tangent at the intersection point and the $X$ axis. We have three cases; case 1: $\theta_{\text{int}} \in [0, \pi/2]$; case 2: $\theta_{\text{int}} \in [-\pi/2, 0]$ and case 3: $\theta_{\text{int}} = 0$. For the first case, the function representing this ellipse is given by:

$$f(x) = \sqrt{1 - \frac{a_1'x^2}{b_1'}}.$$

The value of the tangent obliqueness $\delta_1$ in $P_{\text{int}}$ is:

$$\delta_1 = -\frac{a_1'x_1}{b_1\sqrt{1 - a_1'x_1^2}}.$$

For the second case, we find that the function presenting this part of ellipse is:

$$f(x) = -\sqrt{1 - \frac{a_1'x^2}{b_1'}}.$$
and the obliqueness of the line tangent on $P_{int}$ is:

$$
\delta_1 = \frac{a'_1 x_1}{b'_1 \sqrt{1 - a'_1 x_1^2}}.
$$  

(18)

For the third case, where $\theta = 0$, the value of the tangent obliqueness $\delta = \infty$. The direction vector of the tangent is parallel to the $Y$ axis. In this situation, there is no need to compute the obliqueness tangent.

$P_{int}(x_0, y_0)$ coordinates and the obliqueness of the tangent $\delta_0$ in $(R)$ are computed as:

$$
\begin{align*}
    x_0 &= x_1 \cos(-\phi_1) - y_1 \sin(-\phi_1) + \mu_{x1}, \\
    y_0 &= x_1 \sin(-\phi_1) + y_1 \cos(-\phi_1) + \mu_{y1}, \\
    \delta_0 &= \frac{\sin(-\phi_1) + \delta_1 \cos(-\phi_1)}{\cos(-\phi_1) - \delta_1 \sin(-\phi_1)}.
\end{align*}
$$  

(19)

In $(R_1)$, the direction vector of line tangent is $\vec{v}(1, \delta_1)$. We apply the rotation function to $\vec{v}$ to obtain:

$$
\begin{align*}
    v_x &= \cos(-\phi_1) - \delta_1 \sin(-\phi_1), \\
    v_y &= \sin(-\phi_1) + \delta_1 \cos(-\phi_1),
\end{align*}
$$

where $v_x$ and $v_y$ are the coordinates of the direction vector after the rotation. The obliqueness is $\delta_0 = \frac{v_y}{v_x}$.

We proceed to compute the intersection point $P_{int}$ and the tangent. After some transformation to the tangent line at the intersection point, we extract the coordinate of $P_{int}$ in a new referential $(R_2)$. The new referential $(R_2)$ has as origin the center of the second ellipse, and its axes are parallel to the axes of this ellipse. Next, the second mixture component center $(\mu_{x2}, \mu_{y2})$ is derived in the referential $(R)$.

The treatment of the second ellipse is identical to that of the first ellipse (result of the projection of the component onto the $xy$ plane). The referential $(R)$ is translated by the direction vector $\vec{v}(\mu_{x2}, \mu_{y2})$. We compute the angle $\phi_2$ so that the resultant referential $(R_2)$ has an axis $X$ parallel to the major axis of the second ellipse. After these transformations, we have:

$$
\begin{align*}
    \phi_2 &= 0.5 \arctan\left(\frac{2 \delta_2}{b_2 - a_2}\right), \\
    b'_2 &= \frac{b_2 \cos^2(\phi_2) - a_2 \sin^2(\phi_2)}{\cos(2\phi_2)}, \\
    a'_2 &= \frac{a_2 \cos^2(\phi_2) - b_2 \sin^2(\phi_2)}{\cos(2\phi_2)},
\end{align*}
$$  

(20)

where the strictly positive real numbers $b'_2$ and $a'_2$ verify that the resultant characteristic equation of the second ellipse after the rotation is:

$$
    b'_2 y^2 + a'_2 x^2 = 1.
$$

The value of the line obliqueness $\delta_2$ in $(R_2)$ is given by:

$$
\delta_2 = \begin{cases} 
    -\frac{\cos(\phi_2)}{\sin(\phi_2)}, & \text{if } (\theta_{int} = 0), \\
    \frac{\cos(\phi_2) - \delta_0 \sin(\phi_2)}{\cos(\phi_2)}, & \text{otherwise}
\end{cases}
$$  

(21)

For the special case where $\theta_{int} = 0$, the obliqueness tangent $\delta_0 = \infty$ (the direction vector is parallel to the referential $Y$ axis). It is easy to compute $\delta_2$ by rotating the direction vector $\vec{v}(1, 0)$. The intersection point is finally given by: $(x_2, y_2)$ in $(R_2)$:

$$
\begin{align*}
    x_2 &= -\sqrt{\frac{a'_2 b_2}{a'_2 x_2^2 + b_2}}, \\
    y_2 &= \sqrt{\frac{1 - a'_2 x_2^2}{b_2}} \text{ if } \delta_2 < 0, \\
    y_2 &= -\sqrt{\frac{1 - a'_2 x_2^2}{b_2}} \text{ if } \delta_2 > 0.
\end{align*}
$$  

(22)

$x_2$ can take positive values but in order to ensure that there is no total overlap between adjacent components, we choose the negative values.

We compute the coordinates $x_2$ and $y_2$ in $(R)$ on function of $\mu_{x2}$ and $\mu_{y2}$ by applying the inverse rotation with $\phi_2$ and by translating with $(\mu_{x2}, \mu_{y2})$ afterwards. Finally, $\mu_{x2}$ and $\mu_{y2}$ are given by:

$$
\begin{align*}
    \mu_{x2} &= x_2 \cos(-\phi_2) - y_2 \sin(-\phi_2) + \mu_{x0}, \\
    \mu_{y2} &= x_2 \sin(-\phi_2) + y_2 \cos(-\phi_2) + \mu_{y0}.
\end{align*}
$$  

(23)

It is possible to substitute $\lambda = 1$ in the previous development to get the maximum overlap, which is a particular case of the partial overlap. Figure 7 shows five mixture components in case of maximum overlap.

By applying the definition of well separated components to the two components $\Gamma_1$ and $\Gamma_2$, we have:

$$
\begin{align*}
    a_1 (x_{int} - \mu_{x1})^2 + b_1 (y_{int} - \mu_{y1})^2 + \\
    c_1 (x_{int} - \mu_{x1})(y_{int} - \mu_{y1}) - 1 &= 0,
\end{align*}
$$
where:
\[
\begin{align*}
    c_1 &= 16, \\
    a_1 &= \frac{1}{1 - \rho_1^2 \sigma_1^2} \sigma_1^2 \varepsilon_1, \\
    b_1 &= \frac{1}{1 - \rho_1^2 \sigma_1^2} \sigma_1^2 \varepsilon_1, \\
    c_1 &= -\frac{\sigma_1 \sigma_2 (1 - \rho_1^2) \varepsilon_1}. \\
\end{align*}
\]

For the second component, we find that:
\[
\begin{align*}
    a_2(x_{int} - \mu_{x2})^2 + b_2(y_{int} - \mu_{y2})^2 + c_2(x_{int} - \mu_{x2})(y_{int} - \mu_{y2}) - 1 &= 0, \\
\end{align*}
\]
where:
\[
\begin{align*}
    c_2 &= 1 - 0.5 \ln(\lambda), \\
    a_2 &= \frac{1}{(1 - \rho_2^2) \sigma_2^2}, \\
    b_2 &= \frac{1}{(1 - \rho_2^2) \sigma_2^2} \sigma_2 \varepsilon_2, \\
    c_2 &= -\frac{\sigma_2 \sigma_3 (1 - \rho_2^2) \varepsilon_2}. \\
\end{align*}
\]

The two equations 24 and 26 are characteristic equations of two ellipses in the plane \(T : z = 0\). We follow the same equations to find the coordinates of the second component center \((\mu_{x2}, \mu_{y2})\) that satisfies components well separatedness.

6 Generation Algorithm for Gaussian Bivariate Artificial Correlated Data

In this section, the algorithm of generation of the artificial data is summarized. The general algorithm starts by introducing random values to the parameters of the first component.

We also introduce the mixture coefficients, the standard deviations of the components, the coefficient of correlation and the deviation angles of the other components. The components’ centers are derived afterwards.

In order to avoid the overlap between three components, we suggest to introduce the deviation angles \(\theta\) in the interval \([-\pi/3, \pi/3]\). We suggest also an interval of generation \([1, \sigma_{max}]\) for the standard deviations.

Figures 8 and 9 show a mixture of four components for each rate of overlap. If we exclude the centers of the components, the mixture components have the same parameters. Table 1 shows the generator initializations. The columns represent the mixture coefficients \(n_i\), the first dimension standard deviation, the second dimension standard deviation, the correlation coefficient and the intersection angle \(\theta_{int}\). \(\theta_{int}\) in the table represents the angle between the current component and the next one. These parameters are obtained by varying the rate of overlap to take the values of 0, 0.5, 0.75 and 1.

We choose to give the same first component center \(\mu_1 = (60, 40)\) for each of the experiments.

Table 2 illustrates the centers computed by the generator according to the different overlap rate values. Figures 8 and 9 show both the probability density function pdf and the density scatter plots. We can clearly observe that the scatter approaches each other as \(\lambda\) moves towards 1. It is also shown in [17] an example
Fig. 8. Mixture of 4 components. Overlap rate of 1 and 0.75. Density and distributions

Fig. 9. Mixture of 4 components. Overlap rate of 0 and 0.5. Density and distributions
Algorithm 1 Bivariate correlated Gaussian mixture

1: Number of comp.: \( M \); overlap rate: \( \lambda \); 1st comp. parameters: \( \kappa_1, \mu_{x1}, \mu_{y1}, \sigma_{x1}, \sigma_{y1}, \rho_1 \).
2: for all \( i = 2, \ldots, M \) do
3: \( \) Generate \( \kappa_i, \sigma_{xi}, \sigma_{yi}, \rho_i \) and \( \theta_i / \sum_{i=1}^{M} \kappa_i = 1 \); \( \theta \in [-1\pi/3; 1\pi/3] \) to inhibit the third overlap.
4: end for
5: for all \( i = 2, \ldots, M \) do
6: if \( \lambda = 0 \) then
7: Compute params \( a_1, b_1, c_1 \) and \( e_1 \) (24).
8: Compute params \( a_2, b_2, c_2 \) and \( e_2 \) (26).
9: else
10: if \( (\eta = \frac{\kappa_i \sigma_{x1}^2 \sigma_{y1}^2 \sqrt{1-\rho_1^2}}{\kappa_i \sigma_{x1} \sigma_{y1} \sqrt{1-\rho_i^2}}) \geq 1 \) then
11: Parameters values \( a_1, b_1, c_1 \) as in (6).
12: Second comp. params as in (8).
13: else
14: Calculate the parameter of the first ellipse by using equation (9).
15: Compute the parameters of the second ellipse from (10).
16: end if
17: end if
18: Deduce the parameter of the second ellipse, resulting form translating and rotating the referential \( R \) as in (14).
19: Get the intersection angle \( \theta_{int} \) from (15).
20: Get the intersection point \( P_{int} \) from (16).
21: if \( \theta_{int} \in [0, \pi/2]\) then
22: Use (21) to compute \( \theta_1 \).
23: end if
24: if \( \theta_{int} \in [-\pi/2, 0]\) then
25: Use (22).
26: end if
27: \( P_{int} \) and \( \delta_1 \) in the referential \( (R) \) (19).
28: Compute the second comp. parameters in referential \( (R_2) \) (21).
29: Compute \( P^{(R_2)} \) from (22).
30: Compute the center of the second comp. \( (\mu_{xi}, \mu_{yi}) \) as in (23).
31: end for

of four component mixture by varying \( \lambda \) values; however, in the current work, the example is more general, as it is based on a general algorithm and includes unconstrained Gaussian bivariate artificial correlated data.

7 Experimental Results

In this section, we present the experimental protocol and results. We propose to use k-Means, Fuzzy C-Means (FCM) and FCM-based splitting Algorithm (FBSA) [27, 19]. For validity indices, we propose to use R-Square (RS), Partition Coefficient (PC), Davies-Bouldin (DB), Xie-Benie (XB), WSJ and Classification Entropy (CE) [40].

7.1 Determination of the Number of Components

As mentioned previously, the influence of the overlap rate on clustering results is discussed. The ability of the clustering methods and the validity indices to determine the number of components is also examined.

In order to give equal opportunity to all the clustering methods to reach the correct cluster structure, we use a unique configuration for choosing the initial centers. We arrange the set of observations according to the first dimension. The \( k^{th} \) element is assumed to be the center of the \( k^{th} \) cluster so that \( k = N/C + g \) where \( N \) represents the number of observations, \( C \) the number of clusters and \( g = N \mod C \). Tables 3 and 4 show the experimental results obtained by the proposed algorithm for 3 components and 5 components, respectively.

Tables 3 and 4 illustrate the results obtained by the proposed algorithm.

The first column represents the clustering methods used in this study: FCM, FBSA and K-Means. The second column contains the value of validity indices RS, PC, DB, XB, WSJ and CE. For each components’ number, we compute the result in group according to the rate of overlap \( \lambda \in \{0, 0.5, 0.75, 1\} \). Figures 8 and 9 represent the constructed data in form of clusters. As seen in Tables 3 and 4, we can easily observe that the indices RS, PC and WSJ give always the same results which means that these indices are not monotonous. For a number of observations which is sufficiently large relative to the number of components, these indices do not produce decent results.
In previous contributions [26, 17], we obtained the same result for the uni-variate and uncorrelated data. In [33], a study concerning the WSJ is presented. It is based on Bersdak suggestion, where the number of observations \( N = \sqrt{C_{\text{max}}} \). The proportion \( N/C_{\text{max}} \) in that study assures good results but in our case where the number of observations \( N = 3000 \), it’s clear that WSJ is not monotonous. For the same reason, PC and RS aren’t monotonous.

We also show that by increasing the number of components or the overlap rate, the quality of the results decrease. If we look at the experimental results in Table 3, we observe that the validity indices DB, XB, PC determine the component number to be 3, but with the same overlap rate in Table 4 the above validity indices do not have the ability to identify the true number of components.

A large number of components means that there are relatively a large number of global minima to locate, so that between these global minima a large set of local minima exist where the clustering methods could wrongly converge.
From Table 3, we can easily observe that the determination of the component number becomes less and less accurate as the overlap rate increases.

From the results illustrated in Table 3, we find that all the non-monotonous validity indices are able to find the number of components when the overlap rate \( \lambda = 0 \); but, none of them can find the exact number of components with an overlap rate \( \lambda = 1 \). These results are confirmed by examining Table 4.

Contrary to the 1D experiences presented in [26], the process of clustering, in this case, cannot converge towards the true models. The curse of dimensionality affects the process for two main reasons. The first one concerns the frequency of dispersion of the data.

For the same number of observations, in 1D space, the data is distributed only on one dimension which causes the data to be more compact and the PDF appears as a continuous function with fewer local minima. However, in 2D, the data is distributed on two axes. Analytically, these spaces are viewed as local minima, and in the PDF representation, they appear as noise. The second reason concerns the overlap between more than two adjacent components.

In [26], in 1D, it is confirmed that the worst situation in which clustering methods encounter difficulty in determining the exact components number is the one where there is a component with a small deviation between two components with large standard deviations. In these circumstances, the first component overlaps beyond the second adjacent component and reaches the third component.

In 2D, there are more chances of such ternary overlap. Suppose we have three 2D components such that the intersection angle between the first and the second components is \( \theta_{int} = 0.45\pi \), and the intersection angle between the second and the third components \( \theta_{int} = -0.45\pi \).

In this situation, the first component is so close to the third component that they are in case of total overlap. For this reason, we limited the intersection angles to be within \([-\pi/3, \pi/3]\), despite the fact that this makes it more difficult to control in 2D.

### 7.2 Determination of the Clusters’ Centers

In this section, we study the ability of clustering methods to determine the model parameters by knowing the number of components. The model parameters includes the mixture coefficients, the centers, the standard deviations and the correlation coefficients. The most important parameter is the centers because a small deviation from its real value has a significant influence on the other parameters.

Another point to take into consideration is that a given deviation of the components centers in the case of minimal overlap does not have the same influence in the case of maximum overlap. Because the data in maximum overlap is more compact, an error which appears negligible in minimal overlap case results in significant divergence in the mixture parameters in the maximum overlap case.

For these reasons, we have introduced a new measure \( ER_{avg} \) for computing the deviation of the mixture parameters form the real parameters. \( ER_{avg} \) is given by:

\[
ER_{avg} = \frac{\sum_{i=1}^{nc}\sqrt{(\mu_{xi} - C_{xi})^2 + (\mu_{yi} - C_{yi})^2}}{nc \cdot d_{max}},
\]

where \( nc \) represents the number of clusters; \( \mu_{xi} \) and \( \mu_{yi} \) are the coordinates of the \( i^{th} \) component; \( C_{xi} \) and \( C_{yi} \) symbolize the \( i^{th} \) cluster coordinates; and \( d_{max} \) is the maximum distance between two components centers.

Before computing \( ER_{avg} \), we must first associate each component center to a cluster center. There are two ways to do this. The first is to associate each component center to the nearest cluster center.

The second is to minimize the function defined as \( \min \sum_{i} d(\mu_i, C_i) \), where \( \mu_i \) is the \( i^{th} \) component center, \( \mu \) is the components centers set, and \( d(a, b) \) is the Euclidean distance between \( a \) and \( b \). Both methods produce similar results if the errors in centers are relatively small. But, in cases where the deviations in centers are large, the second method is more robust and provides better results. We use the same mixture that we used for the determination of the number of components. Table 5 illustrates the results.

---

**Table 5** illustrates the results.
Table 5. Results $ER_{avg}$ of clustering methods

<table>
<thead>
<tr>
<th></th>
<th>2 clusters</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\lambda$</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>0.75</td>
</tr>
<tr>
<td>k-means</td>
<td>$ER_{avg}$</td>
<td>0.03</td>
<td>0.15</td>
<td>0.18</td>
<td>0.13</td>
</tr>
<tr>
<td>fcm</td>
<td>$ER_{avg}$</td>
<td>0.0205</td>
<td>0.024</td>
<td>0.0678</td>
<td>0.11</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>3 clusters</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\lambda$</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>0.75</td>
</tr>
<tr>
<td>k-means</td>
<td>$ER_{avg}$</td>
<td>0.007</td>
<td>0.006</td>
<td>0.015</td>
<td>0.0178</td>
</tr>
<tr>
<td>fcm</td>
<td>$ER_{avg}$</td>
<td>0.0025</td>
<td>0.0040</td>
<td>0.0055</td>
<td>0.007</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>4 clusters</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\lambda$</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>0.75</td>
</tr>
<tr>
<td>k-means</td>
<td>$ER_{avg}$</td>
<td>0.12</td>
<td>0.11</td>
<td>0.0723</td>
<td>0.0705</td>
</tr>
<tr>
<td>fcm</td>
<td>$ER_{avg}$</td>
<td>0.0812</td>
<td>0.00421</td>
<td>0.0051</td>
<td>0.00822</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>5 clusters</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\lambda$</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>0.75</td>
</tr>
<tr>
<td>k-means</td>
<td>$ER_{avg}$</td>
<td>0.00052</td>
<td>0.00481</td>
<td>0.0052</td>
<td>0.041</td>
</tr>
<tr>
<td>fcm</td>
<td>$ER_{avg}$</td>
<td>0.0012</td>
<td>0.00551</td>
<td>0.015</td>
<td>0.0017</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>6 clusters</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\lambda$</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>0.75</td>
</tr>
<tr>
<td>k-means</td>
<td>$ER_{avg}$</td>
<td>0.054</td>
<td>0.0077</td>
<td>0.0033</td>
<td>0.049</td>
</tr>
<tr>
<td>fcm</td>
<td>$ER_{avg}$</td>
<td>0.044</td>
<td>0.0018</td>
<td>0.0241</td>
<td>0.035</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>7 clusters</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\lambda$</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>0.75</td>
</tr>
<tr>
<td>k-means</td>
<td>$ER_{avg}$</td>
<td>0.004</td>
<td>0.0087</td>
<td>0.01</td>
<td>0.013</td>
</tr>
<tr>
<td>fcm</td>
<td>$ER_{avg}$</td>
<td>0.004</td>
<td>0.00465</td>
<td>0.00612</td>
<td>0.0086</td>
</tr>
</tbody>
</table>

The results are proportional to the overlap rate. As the overlap rate increases, the $ER_{avg}$ increases. In Table 5, for 5 clusters we see that $ER_{avg} = 0.0012$ when $\lambda = 0$ and $ER_{avg} = 0.0183$ when $\lambda = 1$.

A large value of $\lambda$ means that there are many shared observations between data which makes the process of finding the true clusters more difficult. For the same reason, we find that increasing the number of components also increases the $ER_{avg}$.

8 Conclusion and Future Work

We have proposed an artificial data generator for evaluating the performance of clustering methods. The generator is used to produce artificial data for the mobile centers methods. It also benefits the hierarchical methods where the number of observations is relatively important.

Our approach is based on a formal definition and quantification of mixture components overlap. These definitions are extracted by a formal method in order to have a relationship between visual inspection of the overlap and its formal representation. We have selected three clustering algorithms to be benchmarked (FCM, FBSA and K-Means) and the validity indices RS, PC, DB, XB, WSJ and CE are used in this study.

The experiments are conducted under the same conditions including the initialization parameters and the artificial mixtures. The experimental results have shown the effectiveness and the accuracy.
of the produced observations especially when the overlap rate increases between components: some algorithms and validity indices outperform others and the monotonic nature of the validity indices is confirmed.

Acknowledgements

This work was funded by a Taif University Research Grant 1-437-5159. The authors are grateful to Taif University Deanship of Research for the institutional support.

References


Article received on 01/01/2020; accepted on 27/02/2020.

Corresponding author is Mohammed Ouali.