Statistical analysis of a hierarchical clustering algorithm with outliers

Nicolas Klutchnikoff, Audrey Poterie, Laurent Rouviere

Statistical analysis of a hierarchical clustering algorithm with outliers

Nicolas Klutchnikoff* Audrey Poterie† Laurent Rouvière‡

Abstract

It is well known that the classical single linkage algorithm usually fails to identify clusters in the presence of outliers. In this paper, we propose a new version of this algorithm, and we study its mathematical performances. In particular, we establish an oracle type inequality which ensures that our procedure allows to recover the clusters with large probability under minimal assumptions on the distribution of the outliers. We deduce from this inequality the consistency and some rates of convergence of our algorithm for various situations. Performances of our approach is also assessed through simulation studies and a comparison with classical clustering algorithms on simulated data is also presented.

Keywords: Clustering, Outliers contamination, Single linkage.
AMS Subject Classification: 62G20, 62H30

1 Introduction

In unsupervised learning, clustering refers to a very broad set of tools which aim at finding a partition of the data into dissimilar groups so that the observations within each group are quite similar to each other. Considered as one of the most important questions in unsupervised learning, there is a vast literature on this paradigm. We refer the reader to Hartigan (1975), Jain and Dubes (1988), Duda et al. (2012) and references therein for a broad overview. Moreover, many clustering methods have been developed and studied, such as the k-means algorithm (MacQueen, 1967), the hierarchical clustering methods (Johnson, 1967), the spectral clustering algorithms (Ng et al., 2002), the model-based clustering approaches (McLachlan and Basford, 1988) or density based methods as DBSCAN (Ester et al., 1996). Clustering plays an important role in explanatory data analysis and has been used in many fields including pattern recognition (Satish and Sekhar, 2006), image analysis (Filipovych et al., 2011), document retrieval, bioinformatics (Yamanishi et al., 2004, Zeng et al., 2012) and data compression (Gersho and Gray, 2012). Overall, clustering tools are often used to help users understand the data structure. Furthermore, with the massive increase in the amount of collected and stored data, clustering methods can also be used as dimensionality reduction techniques (Yengo et al., 2014).

In this paper, we consider a mathematical framework close to the one used in Maier et al. (2009), Arias-Castro et al. (2011), Auray et al. (2015). The data are generated according to a mixture of several distributions whose supports are assumed to be disjoint in order to identify the groups. Furthermore, we assume that the data are contaminated by outliers, that

*Univ Rennes, CNRS, IRMAR (Institut de Recherche Mathématique de Rennes) - UMR 6625, F-35000 Rennes, France

†Univ Bretagne Sud, CNRS, LMBA (Laboratoire de Mathématiques Bretagne Atlantique) - UMR 6205, F-56000 Vannes, France

‡Univ Rennes, CNRS, IRMAR (Institut de Recherche Mathématique de Rennes) - UMR 6625, F-35000 Rennes, France

is observations that do not belong to any of the supports. Many authors have studied theoretical properties of nearest neighbor graphs, hierarchical and spectral clustering algorithms in a similar context. For instance, Maier et al. (2009) provides an in-depth analysis of k-nearest neighbor graphs while Arias-Castro (2011) studies the performance of spectral clustering algorithms and single linkage algorithms under assumptions on both the distances between supports and the number of outliers.

Single linkage algorithm is a hierarchical method which consists of recursively merging the two closest clusters in terms of minimal distance. Although this procedure has many interesting properties, it is well known that its performance is much lower in the presence of outliers. This issue comes from two different phenomena. On the one hand, the procedure may wrongly detect small clusters among the outliers. On the other hand, during the recursive clustering procedure, a chain of outliers may lead to merging two groups which contain observations that belong to two different supports. To overcome these problems, we propose an automatic procedure based on a new analysis of the dendrogram produced by the hierarchical agglomerative clustering in terms of minimal distance. This new procedure can be viewed as a simple modification of the classical single linkage algorithm adapted to the presence of outliers. Moreover, the proposed method allows the detection of clusters with low-dimensional geometrical structures. This last property, shared with spectral clustering, is of primary interest in several modern applications, see Arias-Castro (2011) for more details. Like spectral clustering, our data-driven procedure only requires knowledge of the number of groups to identify the clusters with high probability (under mild assumptions on the size of the clusters). Furthermore, our approach offers a decisive advantage over spectral clustering in terms of time complexity.

The paper is organized as follows. In Section 2, we introduce the mathematical framework, the model assumptions, and we define a criterion, called *clustering risk*, to measure performance of clustering algorithms. Section 3 describes the single linkage clustering algorithm and shows, through simple examples, that this algorithm often fails to recover true clusters in the presence of outliers. We then build a new variant of this algorithm which takes into account the possible presence of outliers in the data. This new procedure does not require, besides the number of groups desired, the calibration of any additional parameter. In section 4, we prove the efficiency of our procedure by exhibiting an oracle-type inequality. Some consistency results and rates of convergence are then deduced from this inequality. Section 5 is devoted to compare the performance of our method with other classical clustering algorithms through several synthetic datasets. The proofs are gathered at the end of the paper, in Section 6. The proposed clustering method has been implemented in R and the source code is available at `https://github.com/klutchnikoff/outliersSL`.

2 Mathematical framework

In this section, we define a general probabilistic model to generate data which locally belong to low-dimensional structures $S_1, \dots, S_M$ and which possibly contain outliers collected in a set S_0. We also specify what we expect from a clustering procedure in this framework, and we define a risk to quantify the performance of such a procedure.

2.1 Generative model

We specify in this section how the data are generated in $S_1, \dots, S_M$ and how the outliers are sampled outside these structures. We are given n independent $[0,1]^D$-valued random variables $X_1, \dots, X_n$, and we assume that their common distribution $\mathbb{P}$ can be written as a mixture of $M+1$ distributions $\mathbb{P}_0, \dots, \mathbb{P}_M$. More precisely, for $0 \leq \varepsilon < 1$ and a vector of convex weights

$(\gamma_1, \ldots, \gamma_M)$, $\mathbb{P}$ can be decomposed as follows:

$$\mathbb{P} = \varepsilon \mathbb{P}_0 + (1 - \varepsilon) \sum_{i=1}^{M} \gamma_i \mathbb{P}_i. \tag{1}$$

The value ε represents the proportion of outliers contained in the data while $\mathbb{P}_0$ stands for the distribution of these outliers. The second term of the right-hand side of this equation is, up to the factor $1 - \varepsilon$, the distribution of the *actual data* or *non-outlier data*. These actual data are distributed into M disjoint groups: γ_i represents the weight of the i-th group and $\mathbb{P}_i$ denotes its distribution. For any $i \in \{1, \ldots, M\}$, let $S_i = \text{supp}(\mathbb{P}_i)$ be the compact set of all points $x \in [0,1]^D$ for which any neighborhood A of x satisfies $\mathbb{P}_i(A) > 0$. We also define the set

$$S_0 = [0,1]^D \setminus \left(\bigcup_{i=1}^{M} S_i \right),$$

and we assume that $\text{supp}(\mathbb{P}_0) \subseteq \text{adh}(S_0)$ and $\mathbb{P}_0(\text{adh}(S_0) \cap S_i) = 0$ for $i \in \{1, \ldots, M\}$. Here $\text{adh}(A)$ stands for the adherence of a set A. These assumptions on $\mathbb{P}_0$ mean in particular that outliers are defined as observations that do not belong to the supports of the M groups.

2.2 Assumptions

Clusters are usually identified by high-density regions separated by low-density regions. For instance, Hartigan (1975) defines clusters as connected components of the level sets of the density of the observations. Moreover, the geometry of a cluster often corresponds to low-dimensional structures such as submanifolds of $[0,1]^D$ (see for example Arias-Castro, 2011, Arias-Castro et al., 2011). We consider a similar framework and detail the assumptions of our model below.

(A1) For each $i \in \{1, \ldots, M\}$, the set S_i is connected. Moreover,

$$\delta = \min_{1 \leq i < j \leq M} \min\{\|x - y\| : (x, y) \in S_i \times S_j\} > 0,$$

where $\|\cdot\|$ stands for the Euclidean norm.

Assumption **(A1)** ensures that the supports are disjoint and well separated. This implies that the model is identifiable since the decomposition (1) of $\mathbb{P}$ is then unique, up to any permutation of the indexes $\{1, \ldots, M\}$. Note also that, under this assumption, the dataset $\mathbb{X}_n = \{X_1, \ldots, X_n\}$ is split into $M + 1$ well-defined *groups*: $\mathbb{X}_n \cap S_0$ corresponds to the outliers whereas for $i \geq 1$, $\mathbb{X}_n \cap S_i$ correspond to the *true clusters* we want to recover.

Throughout the paper, for any $0 \leq s \leq D$, we denote by $\mathcal{H}^s$ the s-dimensional Hausdorff outer measure. We recall that, if s is an integer, this measure agrees with ordinary "s–dimensional surface area" on regular sets. In particular, $\mathcal{H}^D$ is the standard Lebesgue measure on the ambient space $\mathbb{R}^D$. We refer the reader to Evans and Gariepy (2015) for more details on this topic. We also define:

$$s_i = \dim_H(S_i) \quad \text{and} \quad d = \max\{s_i : i \in \{1, \ldots, M\}\},$$

where $\dim_H(S_i)$ denotes the Hausdorff dimension of the set S_i, that is the unique real number $s \in [0, D]$ such that $\mathcal{H}^t(S_i) = \infty$ if $t < s$ and $\mathcal{H}^t(S_i) = 0$ if $t > s$. Notice that if S_i is a submanifold of $\mathbb{R}^D$, its Hausdorff dimension s_i corresponds to its classical dimension.

(A2) There exists $\kappa_0 > 0$ such that, for any $A \subseteq S_0$, we have $\mathbb{P}_0(A) \leq \kappa_0 \mathcal{H}^D(A)$.

This assumption relates to the distribution of the outliers and can be reformulated as follows: $\mathbb{P}_0$ is absolutely continuous with respect to $\mathcal{H}^D$, with bounded Radon-Nikodym derivative. This implies, in some sense, that the outliers are nowhere dense in S_0 and thus prevents having clusters that correspond to groups of outliers.

(A3) For any $i \in \{1, \dots, M\}$, there exists $\kappa_i > 0$ such that, for any $A \subseteq S_i$, we have $\mathbb{P}_i(A) \geq \kappa_i^{-1} \mathcal{H}^{s_i}(A)$.

Assumption **(A3)** relates to the distribution of *actual data* and ensures that each $\mathbb{P}_i$ is quite dense on S_i. Note in particular that $\mathbb{P}_i$ can be singular with respect to $\mathcal{H}^{s_i}$ ($\mathbb{P}_i$ is singular with respect to $\mathcal{H}^D$ as soon as $s_i < D$).

Assumptions **(A1)**, **(A2)** and **(A3)** are classical in the clustering setting. The first one guarantees identifiability of the model while the other two highlight differences between outliers and actual data: the former are diffused while the latter are densely distributed into their supports.

Geometric assumptions on the supports S_i are also needed. To state them, we denote by $B(x, r)$ the Euclidean ball centered at $x \in \mathbb{R}^D$ with radius $r > 0$ and by Γ the usual gamma function. Recall that, for any $s > 0$, the function $\eta(s) = \pi^{s/2}\Gamma^{-1}(1 + s/2)$ generalizes to non-integer parameters the volume of the unit ball in dimension s.

(A4) There exists $\kappa_c \geq 1$ such that, for any $i \in \{1, \dots, M\}$, $x \in S_i$ and $0 < r \leq \Delta_i = \operatorname{diam}(S_i)$,

$$\kappa_c^{-1} \leq \frac{\mathcal{H}^{s_i}(S_i \cap B(x, r))}{\eta(s_i) r^{s_i}} \leq \kappa_c.$$

Here $\operatorname{diam}(S_i) = \max\{\|x - y\| : x \in S_i, y \in S_i\}$ denotes the diameter of the support S_i.

Assumption **(A4)** prevents the sets S_i from being "too narrow" in some places. A similar assumption is made in Arias-Castro (2011). Note also that, if S_i is a submanifold that satisfies a *reach* condition, then **(A4)** is automatically fulfilled (see Biau et al., 2007, and references therein).

(A5) For any $i \in \{1, \dots, M\}$, the Hausdorff dimension s_i of S_i agrees with its Minkowski-Bouligand dimension, that is:

$$s_i = \lim_{r \to 0} \frac{\log(N_r(S_i))}{\log(1/r)},$$

where $N_r(S_i)$ denotes the minimal number of open balls of radius r required to cover S_i.

Assumption **(A5)** is necessary to obtain sharp bounds on the covering numbers $N_r(S_i), r > 0$ for any $i \in \{1, \dots, M\}$. Indeed, in general, we only have

$$s_i \leq \liminf_{r \to 0} \frac{\log(N_r(S_i))}{\log(1/r)} \leq \limsup_{r \to 0} \frac{\log(N_r(S_i))}{\log(1/r)} \leq D.$$

Here we assume that the limit inferior matches with the limit superior and that these limits equal s_i. This technical assumption is not too restrictive. We offer two simple generic examples. First, if S_i is a submanifold of $\mathbb{R}^D$ then **(A5)** holds. Indeed, in this case, the Hausdorff dimension and the Minkowski-Bouligand dimension both match with the usual dimension of S_i. Next **(A5)** is also satisfied if S_i is a self-similar set. Indeed, using Assumption **(A4)** with $r = \Delta_i$, we obtain that $0 < \mathcal{H}^{s_i}(S_i) < +\infty$. This implies that S_i satisfies the *open set condition* which allows us to conclude that both the Hausdorff and the Minkowski-Boulingand dimensions match with the affinity dimension of the self-similar set S_i (see Falconer, 2014, chapter 9).

(A6) Let $\gamma_* = \min\{\gamma_i : i \in \{1, \dots, M\}\}$, $\gamma^* = \max\{\gamma_i : i \in \{1, \dots, M\}\}$ and $\not\gamma = \gamma_* - \gamma^*/2$. We assume that:

$$\gamma^* < 2\gamma_* \quad \text{and} \quad 0 \leq \varepsilon < \not\gamma/(1 + \not\gamma).$$

This assumption allows differentiating actual clusters from the set of outliers. It implies that the sizes of the actual clusters should be of the same order since the largest cluster cannot be twice as large as the smallest one. The number of allowed outliers is constrained by the difference in size of the groups. The more homogeneous the sizes of the groups are, the higher the proportion of outliers can be. For instance, ε must be equal to 0 when the largest cluster is twice as large as the smallest one, *i.e.* when $\gamma^* = 2\gamma_*$. This proportion could increase when the gap between cluster sizes reduces. In particular, it could reach $1/(2M + 1)$ when $\gamma^* = \gamma_* = 1/M$.

2.3 Clustering risk

We aim at finding a clustering procedure that groups together the data that lie within the same set S_i, for each $i \in \{1, \dots, M\}$. Regarding the outliers, they can be assigned to any other group or collected into a specific group by the procedure. A clustering procedure consists of splitting the data $\mathbb{X}_n$ into M disjoint clusters. In other words, a clustering algorithm provides a family of clusters $\mathcal{X} = \{\mathcal{X}_1, \dots, \mathcal{X}_M\}$ such that, for any $1 \leq i \neq j \leq M$,

$$\mathcal{X}_i \neq \emptyset, \qquad \mathcal{X}_i \cap \mathcal{X}_j = \emptyset, \qquad \text{and} \qquad \bigcup_{i=1}^{M} \mathcal{X}_i \subseteq \mathbb{X}_n. \tag{2}$$

Observe that the family $\mathcal{X}$ may not cover the whole set $\mathbb{X}_n$. It could be the case if the algorithm reveals some outliers that are not assigned to any cluster. In our context, a clustering procedure is efficient if each cluster contains all the observations from (only) one of the supports S_i, $i \in \{1, \dots, M\}$. It means that there exists a unique permutation $\pi \in \Pi_m$ from the set of all permutations of $\{1, \dots, M\}$, such that, for any $i \in \{1, \dots, M\}$, the data $\mathbb{X}_n \cap S_i$ are included into $\mathcal{X}_{\pi(i)}$. In this context, we measure the performance of a clustering procedure by the *clustering risk* defined as

$$\mathcal{R}_n(\mathcal{X}) = \mathbb{P}(\forall \pi \in \Pi_M, \exists i \in \{1, \dots, M\},\ \mathbb{X}_n \cap S_i \not\subseteq \mathcal{X}_{\pi(i)}), \tag{3}$$

where $\mathcal{X} = \{\mathcal{X}_1, \dots, \mathcal{X}_M\}$ is the clustering family selected by the clustering procedure. This quantity is the probability that the clustering procedure does not correctly recover one subset of observations from at least one of the S_i's. The smaller the risk, the better the clustering procedure.

3 Single linkage algorithm for outliers

Many clustering algorithms have been studied in a context similar to our setting. For instance, Maier et al. (2009), Arias-Castro (2011) and Arias-Castro et al. (2011) prove that algorithms based on pairwise distances (k-nearest neighbor graph, spectral clustering...) are efficient as soon as the supports $S_i, i = 1, \dots, M$ are sufficiently separated. The single linkage hierarchical clustering algorithm has also been investigated by Arias-Castro (2011) and Auray et al. (2015). However, it is well known that this algorithm is sensitive to outliers. We propose here to address the weaknesses of this algorithm in the presence of outliers.

3.1 Agglomerative clustering with single linkage

Many hierarchical clustering algorithms rely on the notion of r–connected set of points in $[0,1]^D$, where r is a nonnegative real number. A subset $A \subseteq \mathbb{R}[0,1]^D$ is said to be r–connected, if

$$B(A, r/2) = \bigcup_{a \in A} B(a, r/2)$$

is a connected set, from a topological point of view. In particular $A = \{x, y\}$ is r–connected if $\|x - y\| \leq r$. The single linkage algorithm may be defined with this notion of connected set of points. For any $r \geq 0$, the set $B(\mathbb{X}_n, r/2)$ can be expanded into $M(r) \in \{1, \ldots, n\}$ connected components denoted by $B_m(r)$ for $m \in \{1, \ldots, M(r)\}$. These connected components provide a partition of $\mathbb{X}_n$ into $M(r)$ clusters defined, for any $m \in \{1, \ldots, M(r)\}$, by

$$\mathcal{Y}_m(r) = B_m(r) \cap \mathbb{X}_n.$$

The family $\mathcal{Y}(r) = \{\mathcal{Y}_m(r) \colon m \in \{1, \ldots, M(r)\}\}$ provides clusters of the single linkage algorithm with radius r.

We can observe that the number of possible families $\mathcal{Y}(r)$ is finite when we let r move in $\mathbb{R}^+$. Indeed, as r increases the clustering process consists of recursively merging the clusters. To see that, consider the single linkage distance between two r-connected components $\mathcal{Y}_m(r)$ and $\mathcal{Y}_{m'}(r)$. It is defined as the distance between the two closest members between these components

$$\mathrm{dist}(\mathcal{Y}_m(r), \mathcal{Y}_{m'}(r)) = \inf\{\|X_k - X_l\| \colon X_k \in \mathcal{Y}_m(r),\, X_l \in \mathcal{Y}_{m'}(r)\}.$$

At the beginning, for $r = \rho_0 = 0$ we have a first family

$$\mathcal{Y}(\rho_0) = \{\mathcal{Y}_m(\rho_0),\, m \in \{1, \ldots, M(\rho_0)\}\}.$$

Observe that if $X_i \neq X_j$ for all $1 \leq i \neq j \leq n$ then $M(\rho_0) = n$ and each cluster $\mathcal{Y}_m(\rho_0)$ corresponds with one observation. Next the two closest clusters are merged according to the (smallest) distance $\mathrm{dist}(\cdot)$. Denote by $\rho_1 > 0$ the distance between the two closest clusters in $\mathcal{Y}(\rho_0)$, we obtain the second family

$$\mathcal{Y}(\rho_1) = \{\mathcal{Y}_m(\rho_1), m \in \{1, \ldots, M(\rho_1)\}\}.$$

This process is then recursively repeated until all (distinct) observations belong to a single cluster. We denote by K the (random) number of iterations, observe that $K \leq n - 1$ almost surely.

Remark 3.1. *Let us make some general remarks about this procedure. At every step k with $1 \leq k \leq K$, the new selected radius ρ_k is larger than the previous one: $\rho_k > \rho_{k-1}$. This radius corresponds to the distance between the two closest clusters belonging to $\mathcal{Y}(\rho_{k-1})$. Moreover, for any $\rho \in [\rho_k, \rho_{k+1}[$ with $0 \leq k \leq K - 1$ and $\rho_K = \infty$, we have*

$$\mathcal{Y}(\rho) = \mathcal{Y}(\rho_k).$$

At the end of the process, we obtain a sequence $\mathcal{Y}(\rho_0), \ldots, \mathcal{Y}(\rho_{K-1})$ of partitions of the data. The aim is to determine how to choose one partition in this sequence. In other words, we have to select a radius in the sequence $\rho_0, \ldots, \rho_{K-1}$. Since the number of clusters is known, a natural way is to choose the radius such that the associated number of clusters is close to M. More precisely, it is usually chosen such that

$$\widehat{\rho}_{n,SL} \in \operatorname*{argmax}_{\rho \in \{\rho_k \colon k \in \{0, \ldots, K-1\}\}} \{M(\rho) \geq M\}.$$

Observe that $\widehat{\rho}_{n,SL}$ exists as soon as each support S_i contains at least one observation. This algorithm is known to be consistent without outliers (i.e. if $\varepsilon = 0$) and under assumptions close to ours (Arias-Castro, 2011, Auray et al., 2015).

3.2 Dealing with outliers

In the presence of outliers, clusters in $\mathcal{Y}(\widehat{\rho}_{n,SL})$ may fail to recover supports S_i with high probability. We provide two toy examples to show that.

Example 1 Figure 1 displays clusters obtained by the classical single linkage algorithm on 2 datasets. The first one (data1) contains two groups and these two groups are perfectly identified by the algorithm. For the second one (data2), one outlier has been added. We observe that this single outlier defines one group while all the other observations are in the second group. Here, the performance of the classical single linkage is dramatically affected by this outlier.

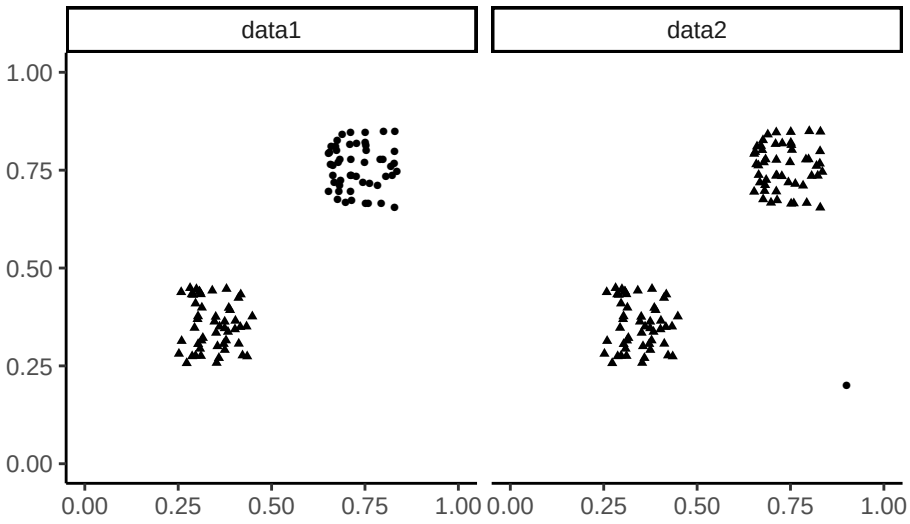


Fig. 1: Results of classical single linkage algorithm performed on 2 datasets.

Example 2 We consider data generated according to the following univariate distribution :

$$\mathbb{P} = \frac{1-\varepsilon}{2}(\mathbb{P}_1 + \mathbb{P}_2) + \varepsilon \mathbb{P}_0,$$

where $\mathbb{P}_1 = \delta_{-1}$, $\mathbb{P}_2 = \delta_1$, $\mathbb{P}_0 = \mathcal{U}([-3,3])$ and $\varepsilon > 0$. For ε small enough, it is easily seen that assumptions presented in section 2.2 are satisfied. However, simple calculations show that the single linkage procedure fails. Indeed, for any $n > 2$,

$$\mathcal{R}_n(\mathcal{Y}(\widehat{\rho}_{n,SL})) \geq \frac{2}{3} - \frac{8}{3}\frac{1}{(n+1)\varepsilon}.$$

As n increases, the clustering risk tends to 2/3.

3.3 OSL algorithm

Results in the previous section show that the single linkage procedure is generally not efficient in the presence of outliers. To address this issue, Arias-Castro (2011) considers a modified version of the procedure that requires the knowledge of the minimal separation distance δ defined in Assumption **(A1)**. Moreover, to prove some consistency results, it is also assumed that the minimal distance between the outliers and the actual data is bounded below by δ. Here, we adopt a different strategy that, from our point of view, seems both more realistic and reasonnable: we let the user choose the number of groups rather than the parameter δ. With

this in mind, we propose to take the cardinality of the M-th largest clusters into account in our procedure. More precisely, our method consists of selecting the radius of the single linkage algorithm which maximizes the size of the M-th largest cluster. In the following, we describe the procedure.

Recall that for any fixed radius $r > 0$, the agglomerative clustering, presented at the beginning of Section 3, provides $M(r)$ clusters

$$\mathcal{Y}(r) = \{\mathcal{Y}_m(r), m \in \{1, \dots, M(r)\}\}.$$

With no loss of generality, we reorder the indices of the r-connected components in $\mathcal{Y}(r)$ such that

$$|\mathcal{Y}_1(r)| > |\mathcal{Y}_2(r)| > \dots > |\mathcal{Y}_{M(r)}(r)|.$$

In the event of a tie, *i.e.* if $|\mathcal{Y}_m(r)| = |\mathcal{Y}_{m'}(r)|$, several rules can be applied to break them. We use the following convention: $\mathcal{Y}_m(r)$ is declared "bigger" than $\mathcal{Y}_{m'}(r)$ if

$$\min\{k \in \{1, \dots, n\} : X_k \in \mathcal{Y}_m(r)\} < \min\{k \in \{1, \dots, n\} : X_k \in \mathcal{Y}_{m'}(r)\}.$$

This means that tie breaking is done by the smallest index in the cluster.

Our robust single linkage clustering procedure proposes to consider only the M (which is assumed to be known) largest clusters and to merge the other clusters together. Formally, for a fixed value of $r > 0$ we consider the M clusters

$$\mathcal{X}_1(r) = \mathcal{Y}_1(r), \dots, \mathcal{X}_M(r) = \mathcal{Y}_M(r). \tag{4}$$

If $M(r) < M$, we define $\mathcal{X}_m(r) = \emptyset$, for any $m \in \{M(r) + 1, \dots, M\}$. Otherwise, when $M(r) > M$, the observations that belong to

$$\mathcal{X}_0(r) = \bigcup_{m=M+1}^{M(r)} \mathcal{Y}_m(r)$$

are not assigned to any group $\mathcal{X}_m(r)$, $m \in \{1, \dots, M\}$. For a fixed value of $r > 0$, this procedure provides the family of clusters $\mathcal{X}(r) = \{\mathcal{X}_1(r), \dots, \mathcal{X}_M(r)\}$.

To propose a data-dependent choice of the radius r, a few remarks are necessary. Too small values of r may result in large values of $M(r)$. In this case, the M largest clusters could be too small and the clustering procedure may fail to recover all the supports $S_1, \dots, S_M$. On the opposite, too large values of r may increase both the risk to gather observations from different supports in the same cluster, and the risk to obtain clusters defined by outliers. The algorithm's performance then depends greatly on the choice of the radius r. With this in mind, we select the radius in $\{\rho_k \colon k \in \{0, \dots, K-1\}\}$ which maximizes the size of the M-th cluster:

$$\widehat{r}_n = \max \operatorname*{argmax}_{\rho \in \{\rho_k \colon k \in \{0, \dots, K-1\}\}} |\mathcal{X}_M(\rho)|. \tag{5}$$

Remark 3.2. *The main difference compared to the single linkage clustering is that this algorithm selects the partition which maximizes the size of the M-th cluster. Observations that belong to $\mathcal{X}_0(\widehat{r}_n)$ are not assigned to any group and might be considered as outliers.*

4 Main results

The selection procedure (5) defines the family of clusters $\mathcal{X}(\widehat{r}_n) = \{\mathcal{X}_m(\widehat{r}_n), m \in \{1, \dots, M\}\}$ whose clustering risk is given by:

$$\mathcal{R}_n(\mathcal{X}(\widehat{r}_n)) = \mathbb{P}(\forall \pi \in \Pi_M, \exists i \in \{1, \dots, M\},\ \mathbb{X}_n \cap S_i \nsubseteq \mathcal{X}_{\pi(i)}(\widehat{r}_n)).$$

The following theorem provides an oracle-type inequality which ensures that this clustering risk is close to the optimal clustering risk, *i.e.*, the one achieved with the best value of r.

Theorem 4.1. *Assume* **(A1)** *and* **(A6)** *hold. Let* $\eta_0 > 0$ *and* $\eta_1 > 0$ *be such that*

$$\eta_0 = 1 - [(1-\varepsilon)(1+\not\gamma)]^{-1} \quad \text{and} \quad \frac{4\eta_1}{1-\eta_1} = \frac{\gamma_*}{\gamma^*} - \frac{1}{2}.$$

Then for all $0 < \eta \leq \min(\eta_0, \eta_1)$ *and all* $n \geq M$*, the clustering risk for clusters* $\mathcal{X}(\widehat{r}_n)$ *satisfies*

$$\mathcal{R}_n(\mathcal{X}(\widehat{r}_n)) \leq \inf_{r>0} \mathcal{R}_n(\mathcal{X}(r)) + 2M \exp(-\psi(\eta)(1-\varepsilon)\not\gamma n) \tag{6}$$

where for $\eta > 0$ $\psi(\eta) = (1+\eta)(\log(1+\eta) - 1) + 1 > 0$.

This theorem ensures that the data-driven selection of r proposed in (5) is efficient for n large enough. Indeed, since $\psi(\eta)(1-\varepsilon)\not\gamma > 0$, inequality (6) guarantees that the performance of our procedure is optimal, up to a remainder term which tends to zero at an exponential rate. In particular, if there exists a specific value r_n of r such that the clustering risk of $\mathcal{X}(r_n)$ tends to 0 as n increases, Theorem 4.1 implies that the risk of $\mathcal{X}(\widehat{r}_n)$ also tends to 0.

To study the clustering risk of $\mathcal{X}(r)$ for a given value of $r > 0$, we define the parameters

$$\mathfrak{a} = \frac{\gamma_*(\kappa^*\kappa_c)^{-1}\eta_*(d)}{1+\not\gamma} \quad \text{and} \quad \mathfrak{b} = \eta(D)\kappa_0, \tag{7}$$

where

$$\kappa^* = \max_{i\in\{1,\dots,M\}} \kappa_i \quad \text{and} \quad \eta_*(d) = \min_{0\leq s\leq d} \eta(s) = \min(1, \eta(d)). \tag{8}$$

These parameters measure to some extent the complexity of the problem. Indeed, $\mathfrak{b}$ essentially depends on the density of the outliers through the parameter κ_0. Problems with sparse outliers will correspond to a small value of $\mathfrak{b}$. The second parameter $\mathfrak{a}$ is related to the distribution of the actual data in their supports $S_i, i \in \{1,\dots,M\}$ and on the regularity of these supports. Regular supports (κ_c small) with a large density of observations (κ^* small) lead to large values of $\mathfrak{a}$. To summarize, difficult problems correspond to small $\mathfrak{a}$ and/or large $\mathfrak{b}$.

The following theorem controls the clustering risk of $\mathcal{X}(r)$ in terms of the parameters of the model.

Theorem 4.2. *Under assumptions* **(A1)**–**(A6)** *we have, for any* $0 < r < \min(\min_i \Delta_i, \delta)$ *and for any* η *such that* $0 < \eta < \eta_0$

$$\mathcal{R}_n(\mathcal{X}(r)) \leq \Lambda r^{-d}\exp(-\mathfrak{a}nr^d) + n\varepsilon(\mathfrak{b}\varepsilon n r^D)^{\lfloor \frac{\delta}{r} \rfloor} + 2M\exp(-\psi(\eta)(1-\varepsilon)\not\gamma n), \tag{9}$$

where Λ *is a positive constant specified in the proof of the Theorem.*

The upper bound in (9) is governed by the first two terms since the last term generally tends to zero much faster. Recall that the cluster family $\mathcal{X}(r)$ may fail to recover the true clusters if one of these two conditions is satisfied:

1. Observations in a same support are not r-connected: there exists $i \in \{1,\dots,M\}$ such that $\mathbb{X}_n \cap S_i$ is not r-connected;

2. Some observations that belong to different supports are r-connected: there is a r-connected path between S_i and S_j for $(i,j) \in \{1,\dots,M\}^2$ with $i \neq j$.

The first term on the right-hand side of (9) corresponds to the first condition. Unsurprisingly, this term is small for large values of r and/or $\mathfrak{a}$. The second term is related to the second condition and, unlike the first term, it tends to decrease when r decreases. This second term also depends on the distribution of the outliers. We observe that it is equal to zero when there

is no outlier, and it increases as the outlier parameter $\mathfrak{b}$ and/or the proportion of outliers ε grows.

Observe also that the minimal distance between supports δ occurs through the exponent $\lfloor \delta/r \rfloor$. If $\mathfrak{b}\varepsilon n r^D < 1$, the second error term decreases as $\lfloor \delta/r \rfloor$ increases. Moreover, we can remark that the upper bound involves two dimensions: the (maximal) Hausdorff dimension d of the support S_i and the dimension D of the outlier space S_0. For fixed values of D, we could obtain slower rates as d increases because it is more difficult to connect observations for large values of d. On the opposite, keeping d constant, rates could be faster when D grows because the probability to connect observations in S_0 decreases. Combining Theorems 4.1 and 4.2 we obtain:

Theorem 4.3. *Under assumptions* **(A1)**–**(A6)** *we have, for all* $n \geq M$*, for any* $0 < r < \min(\min_i \Delta_i, \delta)$ *and all* $\eta \leq \min(\eta_0, \eta_1)$*:*

$$R_n(\mathcal{X}(\widehat{r}_n)) \leq \inf_{r>0} \left\{ \Delta r^{-d} \exp(-\mathfrak{a} n r^d) + n\varepsilon(\mathfrak{b}\varepsilon n r^D)^{\lfloor \frac{\delta}{r} \rfloor} \right\} + 4M \exp(-\psi(\eta)(1-\varepsilon) \curlywedge n).$$

If we intend to prove any consistency results regarding $\mathcal{R}_n(\mathcal{X}(\widehat{r}_n))$, we have to exhibit at least one value of r such that the first terms in this upper bound tends to zero. The following corollary provides sufficient conditions for the consistency of the clustering procedure, *i.e.*, conditions for which we have

$$\lim_{n \to +\infty} \mathcal{R}_n(\mathcal{X}(\widehat{r}_n)) = 0. \tag{10}$$

Except for D and d, all parameters (δ, κ^*, ε...) may vary with n. For simplicity, we only let ε vary with n and keep all other parameters fixed in the conditions.

Corollary 1. *Under the assumptions of Theorem 4.3, consistency* (10) *holds if either* $d < D$ *or* $D = d$ *and* $\varepsilon < (\mathfrak{b} \log n)^{-1}$*.*

We obtain this result by taking $r^d = D \log(n)/(\mathfrak{a} d n)$. This corollary ensures that consistency holds as soon as the Hausdorff dimensions of the supports $S_1, \ldots, S_M$ are smaller than the dimension D of the ambient space. When these dimensions match, the proportion of outliers should tend to 0 much faster than $1/\log n$. Observe also that without outliers ($\varepsilon = 0$), convergence occurs for all $d \leq D$. Using similar tools, we can obtain many rates of convergence with respect to the proportion ε of outliers. Some examples are gathered in the following corollary.

Corollary 2. *Under the assumptions of Theorem 4.3, there exists two universal constants* C_1 *and* C_2 *such that the following propositions hold:*

1. *Few outliers: if* $\varepsilon = \exp(-n)$ *then:*

$$R_n(\mathcal{X}(\widehat{r}_n)) \leq C_1 n \exp(-C_2 n).$$

2. *Small dimensions of the supports: if* $D > d + 1$ *then*

$$R_n(\mathcal{X}(\widehat{r}_n)) \leq C_1 n \exp(-C_2 n^{1/(d+1)}).$$

3. *Large dimensions of the supports with few outliers: if* $d \leq D \leq d + 1$ *and* $\varepsilon = n^{-\beta}$ *with* $\beta \geq 1 - D/(d+1)$*, then*

$$R_n(\mathcal{X}(\widehat{r}_n)) \leq C_1 n^{d/(d+1)} \exp(-C_2 n^{1/(d+1)}).$$

4. *Large dimensions of the supports with many outliers: if* $d \leq D \leq d + 1$ *and* $\varepsilon = n^{-\beta}$ *with* $0 < \beta < 1 - D/(d+1)$*, then*

$$R_n(\mathcal{X}(\widehat{r}_n)) \leq C_1 n^{(1-\beta)d/D} \exp(-C_2 n^{1+(\beta-1)d/D}).$$

To summarize, in each of the above situations, we obtain an upper bound of the form

$$R_n(\mathcal{X}(\widehat{r}_n)) \leq C_1 n^A \exp\left(-C_2 n^B\right)$$

where A, B are given positive constants that depend on the complexity of the problem. We would like to highlight some key points. The fastest rates of convergence are reached in case 1, when the proportion of outliers is at its lowest level. In case 2, the data lie into sets whose dimension is much smaller than the dimension of the ambient space and the rates of convergence only depend on the parameter d. In the last two cases, d is close to D. Rates of convergence mainly depend on the proportion of outliers.

5 Numerical experiments

This section is dedicated to the evaluation of our clustering procedure through two different simulation studies that are described below. First, we simulate data according to the design introduced in Section 2.1. This part can be viewed as an illustration of Theorem 4.3. More precisely, we examine and compare the performance of both our procedure (OSL) and the single linkage algorithm (SL) in the presence of outliers. We also consider the spectral clustering algorithm (SC, see Von Luxburg, 2007, for a brief presentation of this method). Indeed, SC shares several properties with our method such as the ability to detect clusters with low-dimensional geometrical structures and the fact that the number of groups is the main tuning parameter of the algorithm. Next, we use several labelled clustering problems that can be found in the literature. Note that the corresponding data do not necessarily follow our model or satisfy the assumptions introduced in Section 2.1. For these datasets, OSL is compared with SL, SC and also other common clustering algorithms such as the k-means algorithm (KMeans, see MacQueen, 1967), the trimmed k-means (TKMeans), an extension of KMeans introduced in Cuesta-Albertos et al. (1997), the density-based spatial clustering of applications with noise algorithm (DBSCAN, see Ester et al., 1996) and its hierarchical version (HDBSCAN, see Campello et al., 2013). In all experiments, the number of groups M is assumed to be known. Simulation studies have been performed in `R`. A `R` implementation of OSL is available at `https://github.com/klutchnikoff/outliersSL`.

5.1 Sensitivity to the parameters of the model

Here, we examine the performance of OSL in the mathematical framework described in Section 2.1. As stated in Section 4, the efficiency of the proposed clustering algorithm depends on the complexity of the clustering problem, measured through the parameters $(\mathfrak{a}, \mathfrak{b}, \delta, \epsilon)$. Among them, the most sensitive are the intergroup distance δ and the proportion of outliers ε. Of course, the larger ε and the smaller δ, the more difficult the problem. The following subsection describes the considered scenarios.

5.1.1 Description of the models

To simulate our data, we use three different models with different values for the intergroup distance δ, the proportion of outliers ε and the sample size n. More precisely, for each model, we consider two values of δ (one "small value" which corresponds to an *easy* case and one "large value" for a *tricky* case), five values of ε (equally spaced between 0 and 0.2 with a step of 0.05), and two different sample sizes ($n = 200$ and $n = 500$). For each model, both groups and outliers are uniformly sampled over their supports $S_i, i = 1, \ldots, M$ and S_0. We now describe the three models.

Squares model Data are grouped in three distinct squares with similar areas. We use the same weights for each group. Easy and tricky cases correspond to intergroup distances 0.35 and 0.07 respectively, see Figure 2.

Concentric circles This model consists of two nested rings with weight 0.4 for the smallest ring and 0.6 for the largest ring. Intergroup distances are fixed to 2.6 (easy) and 1.6 (tricky), see Figure 3. Outliers are generated only between the two rings.

Sine model This model includes 3 groups with various shapes. The first group is tight and represents the sine curve while the two others are two compact squares. We use the same weights for each group, they are separated from 1.18 (easy) and 0.76 (tricky), see Figure 4.

Simple calculations show that these models satisfy assumptions **(A1)-(A6)** when $\varepsilon < 1/11$ for the "concentric circles" model and $\varepsilon < 1/7$ for the two others. We can also remark that the Hausdorff dimension for all supports equals 2, except for the sine group where it equals 1.

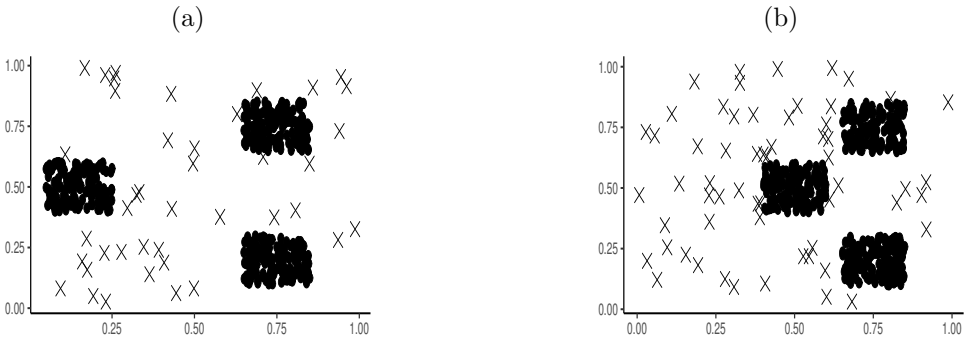
5.1.2 Performance according to the proportion of outliers and the sample size

Through various numerical experiments based on the scenarios described in the previous section, the performance of OSL is evaluated and compared with the one about SL and SC. Regarding the implementation and the calibration of the algorithms, OSL and SL have been implemented by using the functions `hclust` (package `fastcluster`) and `cutree` (package `stats`). SC has been implemented following Ng et al. (2002) and using the function `specc` (package `kernlab`). The scaling parameter is set to the optimal value provided by `specc`, and we consider 20 different random starts for the k-means step of the algorithm. The three algorithms require the knowledge of the number of groups M which is assumed to be known. Observe that for our proposed data-driven approach OSL, M is the only parameter that needs to be tuned. In each scenario, the clustering risk (3) of each clustering algorithm is approximated by its empirical estimator computed over $B = 1000$ Monte Carlo replications

$$\frac{1}{B}\sum_{b=1}^{B} \mathbf{1}_{\left\{\forall\, \pi\in\Pi_M,\, \exists i=1,\dots,M,\, \mathbb{X}_n^b\cap S_i\not\subseteq \mathcal{X}_{\pi(i)}^b\right\}}, \tag{11}$$

where $\mathcal{X}^b = \{\mathcal{X}_1^b, \dots, \mathcal{X}_M^b\}$ denotes the clusters obtained by the procedure on the b-th Monte Carlo sample of data $\mathbb{X}_n^b = \{X_1^b, \dots, X_n^b\}$. Estimator (11) corresponds to the proportion of Monte Carlo replications in which the clustering procedure does not correctly recover one subset of observations from at least one of the S_i's. Figures 2-4 display the three models and the empirical estimate (11) of the clustering risk according to ε, n and δ for all algorithms and each model.

First, as expected, the estimated clustering risk behaves as an increasing function of ε in all experiments and for all clustering algorithms. Moreover, for a fixed value of ε, the clustering risk of all methods is higher when both the number n of observations and the intergroup distance δ take small values. In all experiments, when there is no outlier (i.e. $\varepsilon = 0$), the clustering risk of both SL and OSL is roughly zero. This result is consistent with Theorem 4.1, as well as with the results stated in Arias-Castro (2011) and Auray et al. (2015). In these papers, the authors prove that, under assumptions close to **(A1)-(A6)** and when $\varepsilon = 0$, SL is consistent and its clustering risk tends quickly to zero. As discussed in Section 3.2, in presence of outliers SL often fails to recover the true clusters and its clustering risk increases quickly as ε grows. On the contrary, OSL seems less sensitive to the outlier proportion ε. In the numerical experiments, OSL always works better than SL when there are some outliers.



	$n = 200$		$n = 500$	
	$\varepsilon = 0$	$\varepsilon = 0.2$	$\varepsilon = 0$	$\varepsilon = 0.2$
	Case (a): δ large			
OSL	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
SL	0.000 (0.000)	0.958 (0.006)	0.000 (0.000)	0.997 (0.002)
SC	0.000 (0.000)	0.162 (0.012)	0.000 (0.000)	0.145 (0.011)
	Case (b): δ small			
OSL	0.000 (0.000)	0.014 (0.004)	0.000 (0.000)	0.002 (0.001)
SL	0.000 (0.000)	1.000 (0.000)	0.000 (0.000)	1.000 (0.000)
SC	0.000 (0.000)	0.604 (0.015)	0.000 (0.000)	0.469 (0.016)

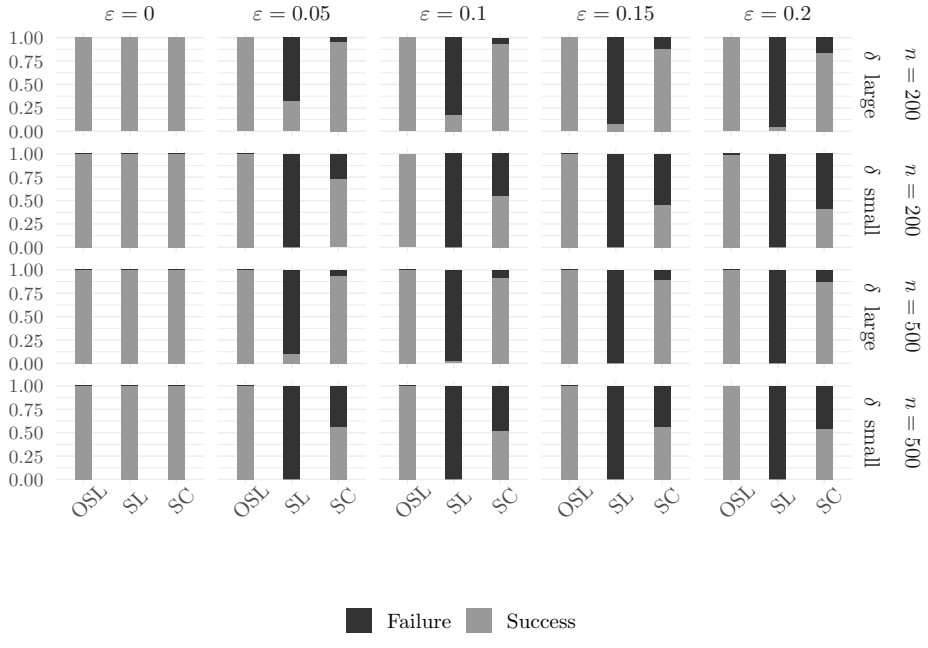
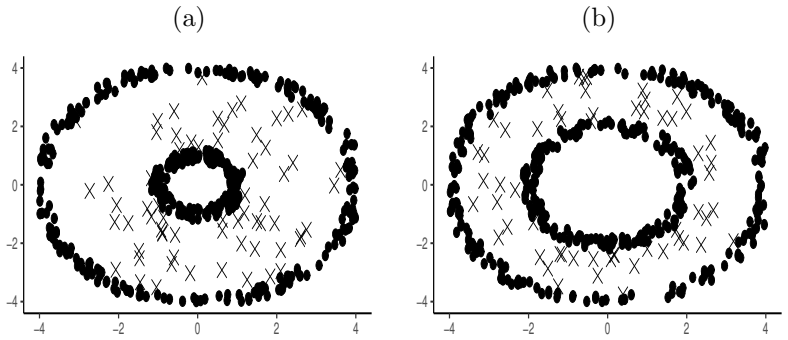


Fig. 2: Results in *squares* model. From top to bottom : a sample of $n = 500$ observations with $\varepsilon = 0.1$ and (a) $\delta = 0.35$ and (b) $\delta = 0.07$; a table and a barplot displaying the empirical estimate of the clustering risk according to ε, n and δ.



	$n = 200$		$n = 500$	
	$\varepsilon = 0$	$\varepsilon = 0.2$	$\varepsilon = 0$	$\varepsilon = 0.2$
	Case (a): δ large			
OSL	0.001 (0.001)	0.863 (0.011)	0.000 (0.000)	0.342 (0.015)
SL	0.000 (0.000)	0.973 (0.005)	0.000 (0.000)	0.993 (0.003)
SC	0.000 (0.000)	0.744 (0.014)	0.000 (0.000)	0.192 (0.012)
	Case (b): δ small			
OSL	0.001 (0.001)	0.997 (0.002)	0.000 (0.000)	0.909 (0.009)
SL	0.000 (0.000)	0.999 (0.001)	0.000 (0.000)	1.000 (0.000)
SC	0.001 (0.001)	0.996 (0.002)	0.000 (0.000)	0.863 (0.011)

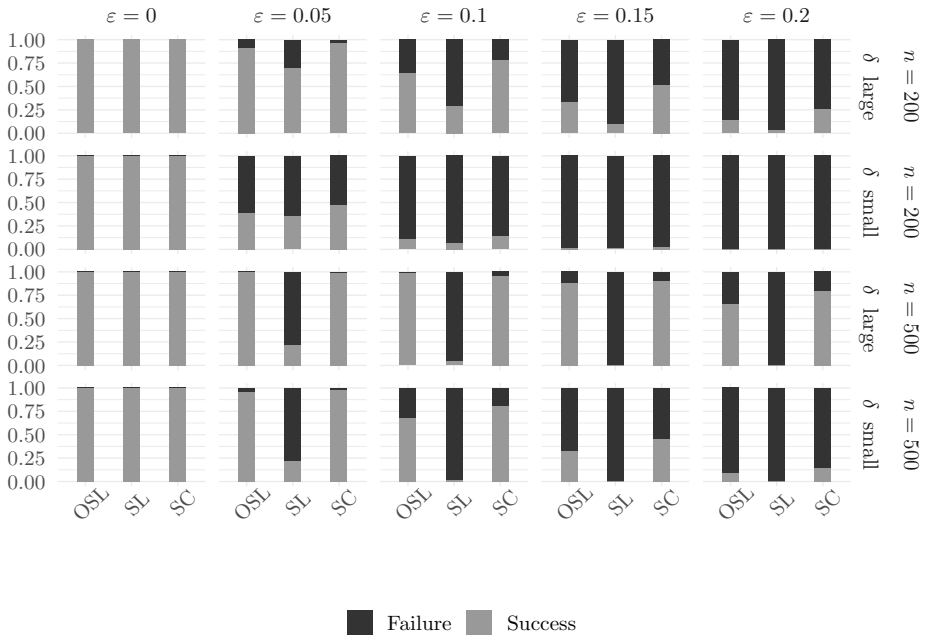
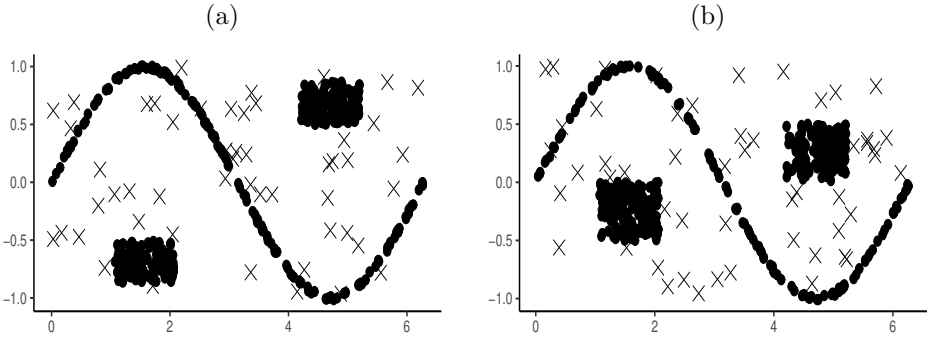


Fig. 3: Results in *concentric circles* model. From top to bottom : a sample of $n = 500$ observations with $\varepsilon = 0.1$ and (a) $\delta = 2.6$ and (b) $\delta = 1.6$; a table and a barplot displaying the empirical estimate of the clustering risk according to ε, n and δ.



	$n = 200$		$n = 500$	
	$\varepsilon = 0$	$\varepsilon = 0.2$	$\varepsilon = 0$	$\varepsilon = 0.2$
	Case (a): δ large			
OSL	0.001 (0.001)	0.796 (0.013)	0.000 (0.000)	0.307 (0.015)
SL	0.001 (0.001)	0.979 (0.005)	0.000 (0.000)	1.000 (0.000)
SC	0.009 (0.003)	0.621 (0.015)	0.000 (0.000)	0.208 (0.013)
	Case (b): δ small			
OSL	0.058 (0.007)	0.939 (0.008)	0.000 (0.000)	0.548 (0.016)
SL	0.058 (0.007)	0.994 (0.002)	0.000 (0.000)	1.000 (0.000)
SC	0.077 (0.008)	0.899 (0.010)	0.002 (0.001)	0.502 (0.016)

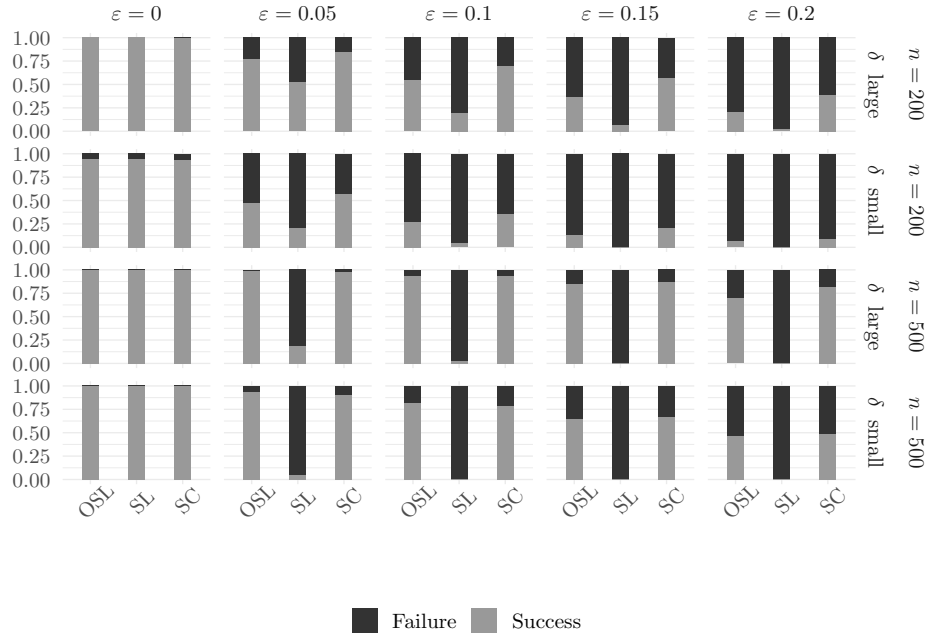


Fig. 4: Results in *sine* model. From top to bottom : a sample of $n = 500$ observations with $\varepsilon = 0.1$ and (a) $\delta = 1.18$ and (b) $\delta = 0.76$; a table and a barplot displaying the empirical estimate of the clustering risk according to ε, n and δ.

As expected, SC is well adapted to non-linearly separable groups but can work quite badly with compact groups (see Nadler and Galun, 2007). Contrary to SC, OSL is competitive in each experiment, and so it seems to perform well whatever the shape of the different groups. Moreover, observe that compared to SC, OSL and SL are exact in the sense that they do not require any random process (for instance the random starts used in SC).

Finally, Table 1 displays the computation time required by OSL and SC in the *squares* model for various values of n. The computations have been carried out on a MacBook Pro, 2,4 GHz Intel Core i5 and 16Gb of RAM memory. Table 1 shows that OLS is substantially faster than SC as n increases.

n	OSL	SC
100	1.41	0.10
200	1.56	0.38
500	2.09	3.86
1000	3.37	27.80
2000	3.27	2×10^3
5000	3.94	4×10^4
10000	6.68	3×10^5

Table 1: Time complexity in seconds to perform OSL and SC as a function of n in the *squares* model.

5.1.3 Performance according to the distribution of the outliers

In the previous simulations, we only consider situations where $D = d$ and the outliers are uniformly distributed over their support. In this subsection, we will consider scenarios for which one or both conditions are not satisfied. We first address situations where $D \geq d$. To do so, we consider the tricky case of the sine model for many values of D and n. For each couple of values (D, n), outliers are uniformy generated in

$$[0, 2\pi]^D \setminus \bigcup_{i=1}^{M} S_i,$$

while the supports of the three groups remain unchanged (the dimension of the supports of the two compact groups is 2 while that of the support of the sine group equals 1). Table 2 displays the clustering risk of OSL according to D and n.

	D								
n	2	3	4	5	6	7	8	9	10
100	0.984	0.928	0.876	0.848	0.808	0.786	0.784	0.784	0.791
200	0.929	0.672	0.405	0.253	0.198	0.160	0.161	0.163	0.161
500	0.558	0.080	0.013	0.002	0.000	0.000	0.000	0.000	0.000
1000	0.153	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 2: Clustering risk (averaged over 1000 replications) of OSL as a function of D (columns) and n (rows) for the sine model (tricky case with $\varepsilon = 0.20$).

As proved in Theorem 4.3, we observe a fast decrease of the risk when D or n increases. Moreover, for small values of n the risk reaches a minimum value which can be viewed as an "optimal risk", i.e. the lowest possible risk for these (small) numbers of observations. In this case, difficulties do not come from the outliers, but it is rather explained by the fact that we do not have enough observations to recover correctly the clusters.

We then address situations where the outliers are not uniformly distributed over their support S_0. We consider again the tricky case of the sine model, but we assume that the outliers are densely distributed between the two squares. More precisely, $\mathbb{P}_0$ is a Gaussian law with mean $(\pi, 0)$ and covariance matrix with variances $2\sigma^2$ on the x-axis, σ^2 on the y-axis and correlation coefficient ρ. Many values for σ^2 and ρ are considered. Figure 5 displays several examples. Note that $\mathbb{P}_0$ is truncated in order to avoid that outliers fall into the supports of the clusters. Performances of OSL are given in Table 3 for $n = 500$ and $\varepsilon = 0.1$.

	σ^2				
ρ	0.01	0.25	0.5	0.75	1
0	0.010	0.618	0.348	0.227	0.160
0.25	0.015	0.683	0.419	0.257	0.169
0.5	0.026	0.747	0.485	0.298	0.201
0.75	0.029	0.847	0.562	0.408	0.311
1	0.037	0.982	0.964	0.920	0.869

Table 3: Clustering risk (averaged over 1000 Monte Carlo replications) of OSL as a function of ρ (rows) and σ^2 (columns) for the sine model (tricky case with $n = 500$ and $\varepsilon = 0.10$).

Some comments can be made about Table 3. First, we notice that the procedure is efficient for large values of σ^2 (except for the particular value $\rho = 1$). This is simply because large values of σ^2 provide sparse outliers, so that the procedure identifies correctly the three clusters. This is no longer the case when σ^2 decreases since the error term is then increasing. In particular, the algorithm is not efficient when $\sigma^2 = 0.25$. For this value, the distribution of the outliers is so dense between the square clusters that a path appears between these clusters and the procedure fails to correctly identify the groups. Next, when σ^2 becomes much smaller, the error term becomes very small. Indeed, when $\sigma^2 = 0.01$, the outliers are gathered around the sine curve and prevent the two squares from being connected. In this case, the outliers are assigned to the group formed by the sine curve. Since the clustering risk measures the ability of a statistical procedure to correctly group observations that belong to true clusters only (and ignores how outliers are assigned), this quantity is not affected by assigning outliers to a group. That's why, the error term remains small in this case. Lastly, we can remark that the risk increases as ρ becomes larger. Indeed, for large values of ρ, the outliers are gathered around a segment that connects the two squares. These two groups are thus connected, and the procedure fails with high probability.

Remark 5.1. *The last scenario with $\sigma^2 = 0.01$ needs to be discussed a bit further. Indeed, when $\sigma^2 = 0.01$, outliers are densely grouped around $(\pi, 0)$ (see the first column in Figure 5). So, it seems not easy to differentiate outliers from groups and one could consider that outliers define a group while the sine curve represents the outliers. However, based on the definition of our model, this is not the case. Indeed, let us recall that our model identifies groups in terms of both density in the supports (assumption **A3**) and cluster size (assumption **A6**). As the sine wave satisfies these two properties, it must define a group. So OLS procedure is correct when it identifies the sine curve as a true cluster.*

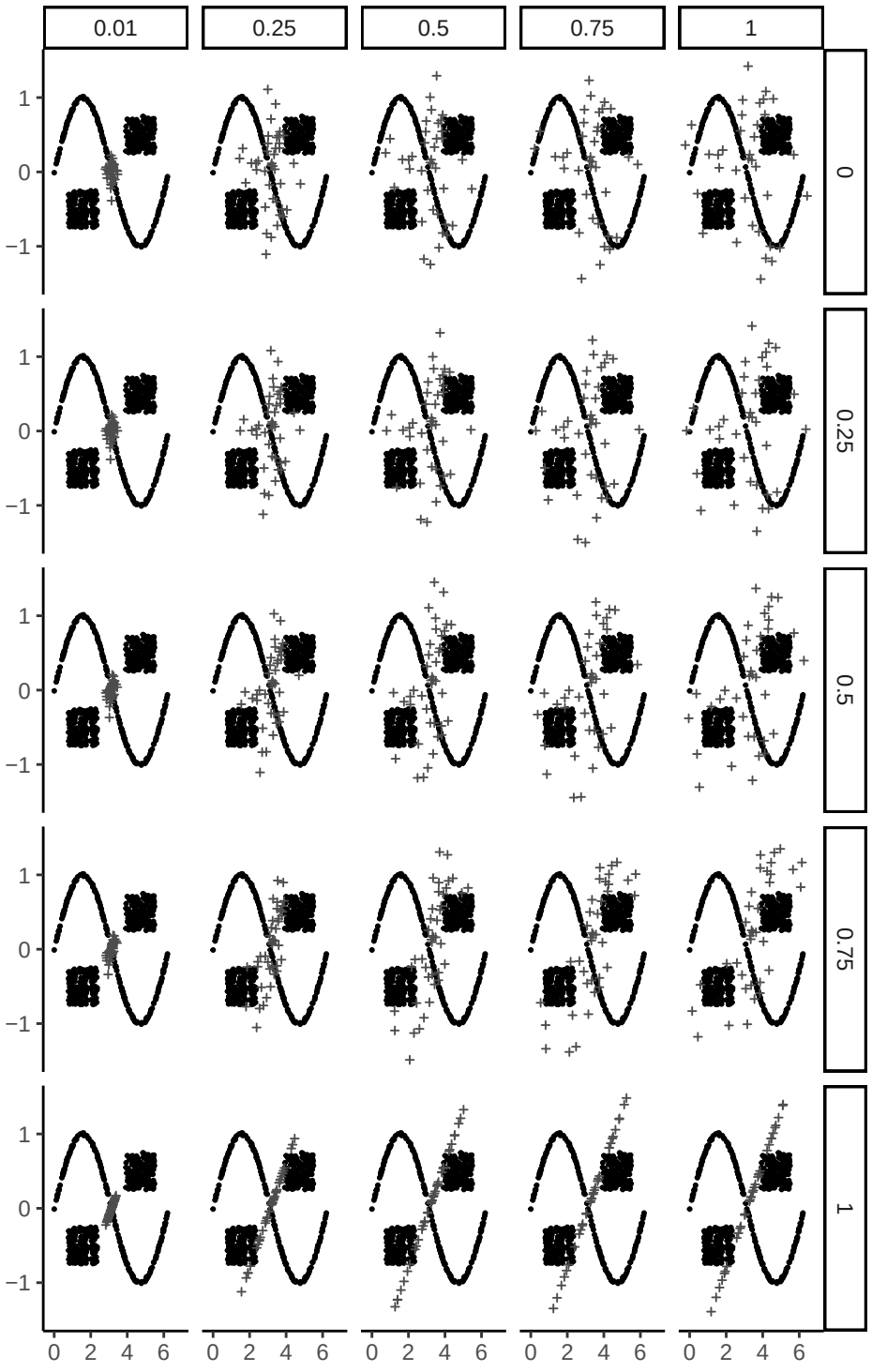


Fig. 5: Samples with Gaussian noises in terms of σ^2 (colums) and ρ (rows).

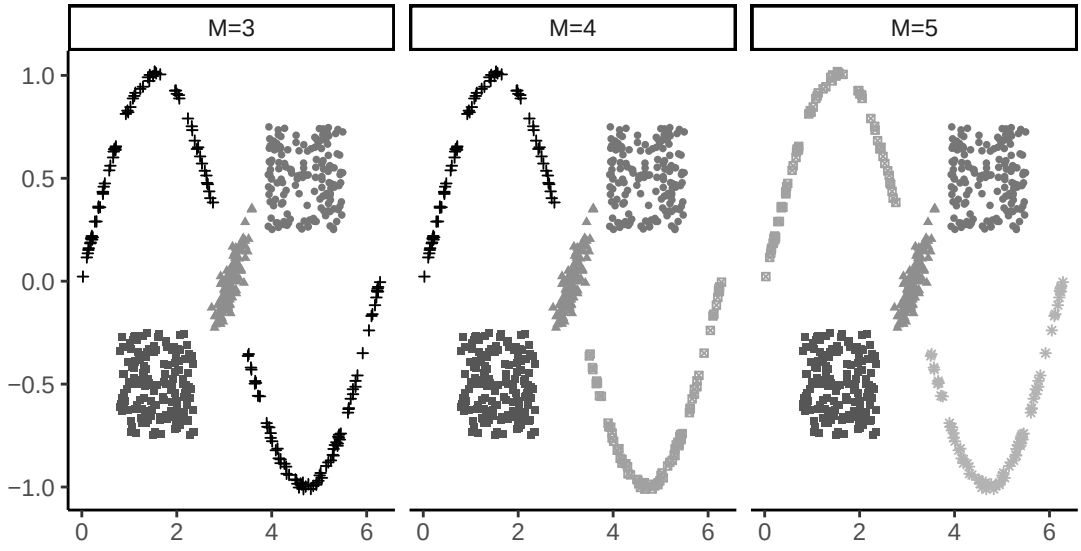


Fig. 6: Results for many values of M with $\varepsilon = 0.2$, $\sigma^2 = 0.01$ and $\rho = 0.85$. Outliers identified by OSL correspond to "+".

Remark 5.2. *Another issue is the robustness of the results depending on the choice of the number of clusters (see Coretto and Henning, 2017). Since outliers may be interpreted as a true group in the last scenario when* $\sigma^2 = 0.01$*, we can be interested in the behavior of the algorithm for* $M = 4$*. However, it is easy to see that the sampling design does not satisfy the model assumptions for* $M = 4$*. Indeed, the supports of the sine curve and the outliers intersect so that it is no longer possible to identify 4 groups. Running OSL with* $M = 4$ *thus identifies outliers as a true cluster while sine observations are considered as either observations of the fourth group or outliers. A correct scenario with* $M = 4$ *can be obtained by adding a small separation between the sine group and the outliers. Figure 6 displays results of OSL with such a small gap for many values of* M*. For* $M = 3$*, the algorithm actually identifies outliers as a true cluster while the two parts of the sine curve are considered as outliers. One part of the sine curve becomes the fourth group for* $M = 4$*. The other part corresponds to the fifth group for* $M = 5$*. Observe that no observations are identified as outliers for* $M = 5$*.*

5.2 Comparison with several common clustering algorithms

Here, OSL is compared with SL, SC and additional clustering algorithm namely the k-means algorithm (KMeans), the trimmed k-means (TKMeans), the density-based spatial clustering of applications with noise algorithm (DBSCAN) and the hierarchical density-based spatial clustering of applications with noise (HDBSCAN) based on three datasets from literature.

5.2.1 Description of the data

In the three datasets downloaded from https://github.com/deric/clustering-benchmark, data lie into subspace of $\mathbb{R}^2$ and groups have various shapes and various sizes. Moreover, the proportion and the distribution of outliers is not the same in the three datasets.

- In the first scenario titled *pathbased* and used in Chang and Yeung (2008), the 300 observations belong to three disjoint groups with various shapes. The first two groups are compact with similar areas. There are surrounded by a third group with a ring-shape. In this scenario there is no outlier.

- The second dataset named *cure-t2-4k* (see Van Craenendonck and Blockeel, 2017, Nerurkar et al., 2018) includes 4200 observations. 4.7% of observations are outliers and the remaining observations lie into 4 well-separated groups. The first three groups are circular with various areas: one is very large while the two others are small. The fourth group is defined by two similar circles joined by a segment. Outliers are distributed quite uniformly all around the groups.

- In the third dataset named *compound* (see Zahn, 1971) there are 399 observations including 50 outliers (12.5% of observations). Observations that are not outliers are distributed into 5 compact groups of various shapes. The three first groups are distinct and well separated while the last two groups with circular shape touch each other but do not overlap. Outliers are uniformly distributed only around a group and so lie into a subspace of the input space.

The three datasets are displayed in Figure 7. Observe that these three scenarios do not necesarily satisfy all the design conditions stated in Section 4. Indeed, in the first scenario, the distance between the ring-shaped group and the two compact groups is almost zero. In the second scenario, two clusters are also not well separated and assumption **(A6)** is not satisfied: the largest group includes 39.6% of observations while the smallest one includes only 4% of observations. In the third dataset, assumption **(A6)** is also not satisfied: about 42% of observations belong to the largest group while only 9.5% of observations are in the smallest one.

5.2.2 Comparison results

For each scenario, 1000 Monte Carlo replications are used by sampling each time without replacement 75% of the observations. The adjusted rand index (ARI) introduced by Hubert and Arabie (1985) and the time complexity (TC) are estimated for each clustering algorithm and each Monte Carlo replication. Here, the ability of each algorithm to identify the groups is measured using ARI. This performance criterion is less strict than the clustering risk defined in equation (3) in the sense that it is not a binary criterion that considers that a clustering procedure fails as soon as one observation that belongs to one true cluster has not been assigned to the right group. ARI is a measure of similarity between two partitions. It has a value between 0 and 1, with 0 indicating that the two partitions do not agree on any pair of points and 1 indicating that the two partitions match perfectly. Here, for each clustering algorithm and each Monte Carlo replication, we compare the resulted partition with the true partition. Then, ARI represents the proportion of agreements over all the possible pairs of points between the resulted partition and the true one. This less strict criterion seems more adapted to the three considered datasets in which, as on many real applications, some clusters could not be well separated. So the definition of the true partition is not straightforward and several configurations could be considered for a same dataset, see for instance the descriptions of the *pathbased* dataset and the *compound* dataset above.

Implementation and calibration of OSL, SL and SC are the same as those used in the first simulation studies, see subsection 5.1. The function `stats::kmeans` is used to perform KMeans with 20 distinct random starts (parameter `nstart`). TKMeans is performed using the function `tclust::tkmeans` with 50 distinct random starts (the default value of parameter `nstart`). In TKMeans, the proportion α of trimmed observations is tuned using the true proportion of outliers ε: at each step, before updating the centers, the algorithm removed the top $\lceil \alpha n \rceil = \lceil \varepsilon n \rceil$ observations with the largest distance from its closest center. DBSCAN and HDBSCAN are performed by using the functions `dbscan::dbscan` and `dbscan::hdbscan`. In DBSCAN, to calibrate the two main tunning parameters that are the radius of each neighborhood (`eps`) and

the number of minimum points required for each neighborhood (`minPts`), we use the approach explained by Ester et al. (1996): for each scenario, `minPts` is set to its default value 4 and `eps` is calibrated once on the entire dataset. For HDBSCAN, we fix the number of minimum points (`minPts`) at the minimal possible value, 2, and we use the knowledge of M to choose the final partition as we do with the algorithms OSL, SL, KMeans and SC. Note that in presence of outliers, this automatic calibration strategy might not be optimal for HDBSCAN and the selection of the final partition might be rather performed on each run in a non-automatic manner (for instance the study of the silhouette score). As in the previous simulation study, all computations have been carried out on a MacBook Pro, 2.4 GHz Intel Core i5 with 16 Gb of RAM.

Results are displayed in Figure 7. First, we can observe that OLS generally performs quite well to recover the true groups. The method is part of the top three algorithms in the two first scenarios. Note that in the two first scenarios, OLS outperforms SL even when there is no outlier.

In the *Pathbased* dataset and particularly in the *Compound* dataset, performance of OLS is affected by the fact that some clusters are not well separated and the distance between some clusters is almost zero. Indeed, for instance in the *Compound* dataset, clustering performance of OLS improves greatly when we consider the partition in which the two groups that intersect are grouped together, see Figure 8. Remark that in this case OLS outperforms all the other clustering algorithms.

Generally, our clustering method is thus competitive compared to the other algorithms. As previously noticed, OLS does not seem very sensitive to the group shape. As DSBCAN, OLS seems able to identify group completly surrounded by another group and seems quite robust toward outliers. Moreover, note that contrary to DBSCAN, OLS seems less sensitive to datasets with large density variations between groups (see for instance the *Compound* dataset). Indeed, DBSCAN is very sensitive to the choice of its two main tunning parameters (`MinPts` and `eps`) and these two parameters cannot be chosen appropriately for all clusters when the density varies a lot between clusters.

Finally, OLS is compared to the other algorithms in terms of time complexity. OLS appears a bit less fast than SL, KMeans, TKMeans and DBSCAN. Nonetheless, the time complexity of our apporach remains raisonnable when n is larger. Moreover, as peviously noticed (see Table 1), OLS is faster than SC, especially when n is large.

6 Proofs

6.1 Technical lemmas

Lemma 1. *Fix $i = 1, \dots, M$ and $0 < r < \Delta_i$. Under* **(A1)-(A3)-(A4)**, *there exists a positive constant Λ_i such that*

$$\psi_{n,i}(r) = \mathbb{P}(\mathbb{X}_n \cap S_i \text{ is not } r\text{-connected}) \leq \Lambda_i r^{-d} \exp(-\mathfrak{a} n r^d).$$

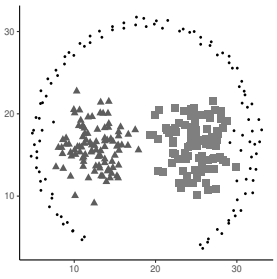
Proof: We denote by $N_r^*(S_i)$ the minimal number of balls of radius $r > 0$, centered at points of S_i, required to cover S_i. Note that, for any $r > 0$ we have by definition $N_r(S_i) \leq N_r^*(S_i)$. Moreover, using triangle inequality we also have $N_r^*(S_i) \leq N_{r/2}(S_i)$. Using Assumption **(A5)**, this implies that

$$s_i = \lim_{r \to 0} \frac{\log(N_r^*(S_i))}{\log(1/r)}.$$

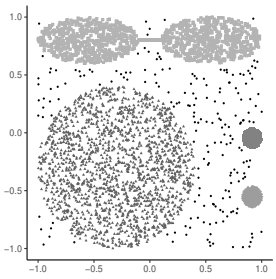
Thus, there exists a positive constant Λ_i (that depends on S_i) such that, for any $0 < r < \Delta_i$ we have:

$$N_r^*(S_i) \leq 4^{-d} \Lambda_i r^{-s_i} \leq 4^{-d} \Lambda_i r^{-d}.$$

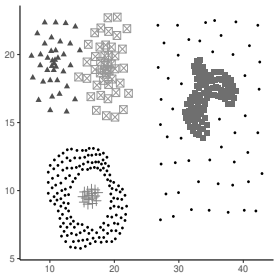
Pathbased dataset



Cure-t2-4k dataset

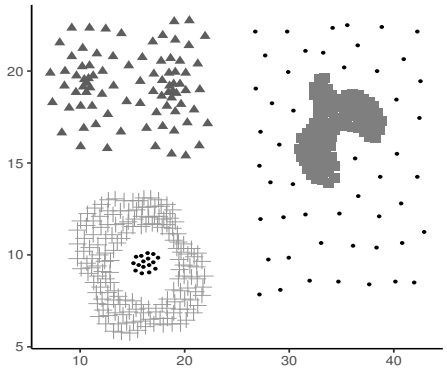


Compound dataset



	Adjusted rand index	Running time (in sec.)
	Pathbased dataset	
OSL	**0.58** (0.16)	0.02 (0.01)
SL	0.07 (0.14)	5.9e-4 (8.1e-4)
SC	0.39 (0.19)	0.05 (0.01)
KMeans	0.46 (0.03)	1.8e-3 (1.6e-3)
TKMeans	**0.9** (0.04)	7.1e-3 (1.2e-3)
DBSCAN	**0.6** (0.06)	3.8e-4 (8e-05)
HDBSCAN	0.07 (0.14)	7.4e-3 (1.8e-3)
	Cure-t2-4k dataset	
OSL	**0.9** (0.13)	1.43 (0.07)
SL	0.02 (0.08)	0.06 (0.01)
SC	**0.91** (0.09)	98.5 (3.6)
KMeans	0.54 (0.01)	0.03 (5.9e-3)
TKMEANS	0.53 (0.01)	0.27 (0.01)
DBSCAN	**0.96** (0.021)	3.0e-3 (5.9e-4)
HDBSCAN	0.02 (0.08)	0.82 (0.05)
	Compound dataset	
OSL	0.48 (0.33)	0.02 (5e-3)
SL	**0.69** (0.12)	9.5e-4 (4.1e-3)
SC	**0.72** (0.11)	0.10 (9.4e-3)
KMeans	0.58 (0.04)	2.3e-3 (6.4e-4)
TKMEANS	0.61 (0.02)	0.01 (1.3e-3)
DBSCAN	0.32 (0.08)	4e-04 (4e-05)
HDBSCAN	**0.69** (0.12)	8.3e-3 (1.2e-3)

Fig. 7: Results on the *Pathbased*, *Compound* and *Aggregation* datasets. For each parameter, the mean over the 1000 Monte Carlo replications is displayed with in brackets the standard error.



	Adjusted rand index	Running time (s)
OSL	**0.7** (0.22)	2.5e-3 (2.5e-3)
SL	0.66 (0.16)	7.7e-4 (4.5e-4)
SC	**0.75** (0.03)	0.10 (8.9e-3)
KMeans	0.65 (0.07)	2.2e-3 (4.3e-3)
TKMEANS	**0.67** (0.08)	0.01 (1e-3)
DBSCAN	0.35 (0.07)	4.1e-4 (4e-05)
HDBSCAN	0.66 (0.16)	8.4e-4 (1.4e-3)

Fig. 8: Results on the modified version of the *Compound* datasets. For each parameter, the mean is displayed with in brackets the standard error.

This implies that there exist both an index set $\mathcal{L}_i$, whose cardinality is bounded above by $\Lambda_i r^{-d}$, and a family of balls $(B_\ell)_{\ell\in\mathcal{L}_i}$ centered at points that belong to S_i, with radius $r/4$, which satisfy:

$$S_i \subset \bigcup_{\ell\in\mathcal{L}_i} B_\ell.$$

Since, for any $\ell \in \mathcal{L}_i$, we have $(\mathbb{X}_n \cap S_i) \cap B_\ell \neq \emptyset$, there exists $\alpha_\ell \in \{1,\dots,n\}$ such that $X_{\alpha_\ell} \in B_\ell \cap S_i$. Using triangle inequality, this implies that $B(X_{\alpha_\ell}, r/2) \supset B_\ell$. Thus,

$$\mathbb{X}_n \cap S_i \subset \bigcup_{\ell\in\mathcal{L}_i} B(X_{\alpha_\ell}, r/2) \quad \text{with} \quad X_{\alpha_\ell} \in \mathbb{X}_n \cap S_i.$$

We deduce that $\mathbb{X}_n \cap S_i$ is r-connected and

$$\begin{aligned}
\psi_{n,i}(r) &\leq \mathbb{P}\left(\exists \ell \in \mathcal{L}_i, B_\ell \cap (\mathbb{X}_n \cap S_i) = \emptyset\right) \\
&\leq \mathbb{P}\left(\exists \ell \in \mathcal{L}_i,\ \forall k \in \{1,\dots,n\}, X_k \notin S_i \text{ or } (X_k \in S_i, X_k \notin B_\ell)\right) \\
&\leq \sum_{\ell\in\mathcal{L}_i} \left(\mathbb{P}(X \notin S_i) + \mathbb{P}(X \notin B_\ell \mid X \in S_i)\mathbb{P}(X \in S_i)\right)^n \\
&\leq \sum_{\ell\in\mathcal{L}_i} \left(1 - \mathbb{P}(X \in B_\ell \mid X \in S_i)\mathbb{P}(X \in S_i)\right)^n \\
&\leq \sum_{\ell\in\mathcal{L}_i} \left(1 - (1-\varepsilon)\gamma_i \mathbb{P}_i(X \in B_\ell)\right)^n.
\end{aligned}$$

Moreover,

$$\begin{aligned}
\mathbb{P}_i(X \in B_\ell) &= \mathbb{P}_i(X \in B_\ell \cap S_i) && \text{from } \mathbf{(A1)} \\
&\geq \kappa_i^{-1} \mathcal{H}^{s_i}(B_\ell \cap S_i) && \text{from } \mathbf{(A3)} \\
&\geq (\kappa_i \kappa_c)^{-1} \eta(s_i) r^{s_i} && \text{from } \mathbf{(A4)} \\
&\geq (\kappa^* \kappa_c)^{-1} \eta_*(d) r^d,
\end{aligned}$$

where κ^* and $\eta_*(d)$ are defined by (8). Putting all pieces together we obtain

$$\begin{aligned}
\psi_{n,i}(r) &\leq |\mathcal{L}_i| \left(1 - (1-\varepsilon)\gamma_* (\kappa^* \kappa_c)^{-1} \eta_*(d) r^d\right)^n \\
&\leq \Lambda_i r^{-d} \exp(-\mathfrak{a} n r^d),
\end{aligned}$$

where $\mathfrak{a}$ is defined in (7). □

Lemma 2. *Let $r > 0$ and denote by $\varphi_n(m, r)$ the probability that there exists, in S_0, a path of at least m r-connected observations. If assumptions* **(A1)** *and* **(A2)** *hold, we have*

$$\varphi_n(m, r) \leq n\varepsilon(\mathfrak{b}n\varepsilon r^D)^{m-1},$$

where $\mathfrak{b}$ is defined in (7).

Proof: Fix $r > 0$. For any $I \subseteq \{1, \ldots, n\}$ we denote by $\mathcal{A}_I$ the following event: there exists a permutation $i_1 < \ldots < i_m$ of I such that $\|X_{i_j} - X_{i_{j+1}}\| \leq r$ for any $j = 1, \ldots, m-1$. We have:

$$\begin{aligned}\varphi_n(m, r) &\leq \sum_{\substack{I \subseteq \{1,\ldots,n\} \\ |I|=m}} \mathbb{P}(\mathcal{A}_I \cap \{X_I \subseteq S_0\}) \\ &\leq \sum_{\substack{I \subseteq \{1,\ldots,n\} \\ |I|=m}} \varepsilon^m \mathbb{P}_0(\mathcal{A}_I).\end{aligned}$$

Now remark that

$$\begin{aligned}\mathbb{P}_0(\mathcal{A}_I) &\leq m!\mathbb{E}_0\left(\prod_{j=1}^{m-1} \mathbf{1}_{[0,r]}(\|X_{i_j} - X_{i_{j+1}}\|)\right) \\ &= m! \int_{S_0} \cdots \int_{S_0} \mathbf{1}_{[0,r]}(\|x_1 - x_2\|) \ldots \mathbf{1}_{[0,r]}(\|x_{m-1} - x_m\|) \mathrm{d}\mathbb{P}_0(x_1, \ldots, x_m).\end{aligned}$$

Note also that, using **(A2)**:

$$\int_{S_0} \mathbf{1}_{[0,r]}(\|x - y\|) \mathrm{d}\mathbb{P}_0(y) \leq \mathbb{P}_0(B(x, r)) \leq \kappa_0 \mathcal{H}^D(B(x, r)) = \kappa_0 \eta(D) r^D.$$

This, combined with Fubini's theorem implies that:

$$\mathbb{P}_0(\mathcal{A}_I) \leq m!(\eta(D)\kappa_0 r^D)^{m-1}.$$

Finally, we obtain:

$$\begin{aligned}\varphi_n(m, r) &\leq \frac{n!}{(n-m)!}\varepsilon^m(\eta(D)\kappa_0 r^D)^{m-1} \\ &\leq n\varepsilon(\mathfrak{b}\varepsilon n r^D)^{m-1}.\end{aligned}$$

□

Lemma 3. *Assume that assumptions* **(A1)** *and* **(A6)** *hold. Define $N_i = |\mathbb{X}_n \cap S_i|, i \in \{0, \ldots, M\}$ and for $0 < \eta \leq \eta_0$ let*

$$\Omega_\eta = \bigcap_{i=1}^{M} \left\{(1-\eta)(1-\varepsilon)\gamma_i n < N_i < (1+\eta)(1-\varepsilon)\gamma_i n\right\}.$$

We have

(i) $\mathbb{P}(\overline{\Omega_\eta}) \leq 2M \exp(-\psi(\eta)(1-\varepsilon) \curlywedge n)$;

(ii) $N_0 < \frac{\curlywedge}{\gamma_*} \min_{i \in \{1,\ldots,M\}} N_i < \min_{i \in \{1,\ldots,M\}} N_i$ *under* Ω_η.

Proof: Since $N_i \sim B(n, (1-\varepsilon)\gamma_i)$, (i) is a direct consequence of Shorack and Wellner (1986, page 440). For (ii), observe that $(1-\varepsilon)(1-\eta_0) = 1/(1+\not{\gamma})$. Since $0 < \eta \leq \eta_0$, it follows that

$$1-(1-\varepsilon)(1-\eta) \leq (1-\varepsilon)(1-\eta)\not{\gamma}\,.$$

Thus, under Ω_η,

$$\begin{aligned} N_0 &\leq n - \sum_{i=1}^{M} N_i \leq n\left(1-(1-\varepsilon)(1-\eta)\sum_{i=1}^{M}\gamma_i\right) \\ &\leq n(1-(1-\varepsilon)(1-\eta)) \leq n(1-\varepsilon)(1-\eta)\not{\gamma} \\ &\leq \frac{\not{\gamma}}{\gamma_*} n(1-\varepsilon)(1-\eta)\gamma_i < \frac{\not{\gamma}}{\gamma_*} N_i \quad \forall i = 1,\ldots,M. \end{aligned}$$

□

Lemma 4. *Assume that assumption* **(A6)** *holds. For each* $\eta \leq \min(\eta_0, \eta_1)$ *we have*

$$\frac{1+\eta}{1-\eta}\frac{\gamma^*}{\gamma_*} + \frac{\not{\gamma}}{\gamma_*} \leq 2.$$

Proof: Let $\eta \leq \min(\eta_0, \eta_1)$, then

$$\begin{aligned} \frac{1+\eta}{1-\eta}\frac{\gamma^*}{\gamma_*} + \frac{\not{\gamma}}{\gamma_*} &= \frac{1-\eta+2\eta}{1-\eta}\frac{\gamma^*}{\gamma_*} + \frac{\not{\gamma}}{\gamma_*} \\ &= \frac{\gamma * + \not{\gamma}}{\gamma_*} + \frac{1}{2}\frac{4\eta}{1-\eta}\frac{\gamma^*}{\gamma_*} \\ &\leq \frac{\gamma^*/2 + \gamma_*}{\gamma_*} + \frac{1}{2}\left(\frac{\gamma_*}{\gamma^*} - \frac{1}{2}\right)\frac{\gamma^*}{\gamma_*} \\ &= \frac{3}{2} + \frac{\gamma^*}{4\gamma_*} \leq 2. \end{aligned}$$

□

Lemma 5. *Assume* **(A6)** *holds. For* $r > 0$*, let*

$$\mathcal{E}(r) = \left\{\exists \pi \in \Pi_M \, \forall i = 1,\ldots,M \;\; \mathbb{X}_n \cap S_i \subseteq \mathcal{X}_{\pi(i)}(r)\right\}.$$

Let $\eta \leq \min(\eta_0, \eta_1)$*, then under* $\mathcal{E}(r) \cap \Omega_\eta$ *we have*

1. $\widehat{r}_n \geq r$ *almost surely;*
2. *There exists* $\pi \in \Pi_M$ *such that,* $\forall i = 1,\ldots,M$ $\mathcal{X}_i(r) \subseteq \mathcal{X}_{\pi(i)}(\widehat{r}_n)$.

Proof: Let $\eta \leq \min(\eta_0, \eta_1)$ and assume that Ω_η is true. We first prove that $\widehat{r}_n \geq r$ with a reductio ad absurdum. Assume that $\widehat{r}_n < r$. Observe that

$$|\mathcal{Y}_M(\widehat{r}_n)| > |\mathcal{Y}_M(r)|\,,$$

by definition of $\widehat{r}_n$. It follows that

$$|\mathcal{Y}_1(\widehat{r}_n)| \geq \ldots \geq |\mathcal{Y}_M(\widehat{r}_n)| > |\mathcal{Y}_M(r)|\,.$$

Since $\widehat{r}_n < r$, we deduce that one of the $\mathcal{Y}_i(r), i = 1, \dots, M-1$ contains observations of at least two clusters among $\mathcal{Y}_i(\widehat{r}_n), i = 1, \dots, M$. It implies that

$$|\mathcal{Y}_1(r)| \geq 2\,|\mathcal{Y}_M(\widehat{r}_n)| > 2\,|\mathcal{Y}_M(r)|\,. \tag{12}$$

Moreover, under $\mathcal{E}(r)$ we have $N_{(1)} \leq |\mathcal{Y}_1(r)| \leq N_{(1)} + N_0$ where $N_i = |\mathbb{X}_n \cap S_i|$ and $N_{(i)}, i = 1, \dots, M$ are such that

$$N_{(1)} \geq \ldots \geq N_{(M)}.$$

Thus, under $\mathcal{E}(r) \cap \Omega_\eta$, we have from Lemma 3

$$|\mathcal{Y}_1(r)| \leq N_{(1)} + N_0 \leq N_{(1)} + \frac{\gamma}{\gamma_*} N_{(M)}.$$

Since $\left|\mathcal{Y}_{(M)}(r)\right| \geq N_M$, we obtain from Lemma 4

$$\frac{|\mathcal{Y}_1(r)|}{|\mathcal{Y}_M(r)|} \leq \frac{N_{(1)}}{N_{(M)}} + \frac{\gamma}{\gamma_*} \leq \frac{1+\eta}{1-\eta}\frac{\gamma^*}{\gamma_*} + \frac{\gamma}{\gamma_*} \leq 2,$$

which is a contradiction with (12). We deduce that $\widehat{r}_n \geq r$ almost surely.

For the second point, observe that since $\widehat{r}_n \geq r$, each $\mathcal{X}_i(\widehat{r}_n), i = 1, \dots, M$ may be written as a union of clusters in

$$\mathcal{X}_1(r), \dots, \mathcal{X}_M(r), \mathcal{Y}_{M+1}(r), \dots, \mathcal{Y}_{M(r)}(r).$$

Moreover, for each $i \in \{1, \dots, M\}$ there exists a unique $j \in \{1, \dots, M\}$ and a subset $\mathcal{T}(\widehat{r}_n)$ of $\{M+1, \dots, M(r)\}$ such that

$$\mathcal{X}_i(\widehat{r}_n) = \mathcal{X}_j(r) + \bigcup_{\ell \in \mathcal{T}(\widehat{r}_n)} \mathcal{Y}_\ell(r). \tag{13}$$

Indeed, if there exists $i \in \{1, \dots, M\}$ and $1 \leq j \neq j' \leq M$ such that

$$\mathcal{X}_j(r) \cup \mathcal{X}_{j'}(r) \subseteq \mathcal{X}_i(\widehat{r}_n),$$

then $\mathcal{X}_M(\widehat{r}_n)$ may be written as a union of clusters in

$$\{\mathcal{Y}_{M+1}(r), \dots, \mathcal{Y}_{M(r)}(r)\},$$

and thus $|\mathcal{X}_M(\widehat{r}_n)| \leq N_0$. This is not possible since, by definition of $\widehat{r}_n$ and by Lemma 3, we have

$$|\mathcal{X}_M(\widehat{r}_n)| \geq |\mathcal{X}_M(r)| \geq N_{(M)} > N_0.$$

We deduce that (13) is true. Therefore, there exists $\pi \in \Pi_M$ such that, $\forall i = 1, \dots, M$ $\mathcal{X}_i(r) \subseteq \mathcal{X}_{\pi(i)}(\widehat{r}_n)$. □

6.2 Proof of Theorems

Proof of Theorem 4.1: For $r > 0$, let

$$\mathcal{E}(r) = \left\{\exists \pi \in \Pi_M,\, \forall i = 1, \dots, M,\ \mathbb{X}_n \cap S_i \subseteq \mathcal{X}_{\pi(i)}(r)\right\} \tag{14}$$

Let $\eta \leq \min(\eta_0, \eta_1)$. Observe that

$$\begin{aligned} 1 - \mathcal{R}_n(\mathcal{X}(\widehat{r}_n)) &= \mathbb{P}\big(\mathcal{E}(\widehat{r}_n)\big) \\ &\geq \mathbb{P}\big(\mathcal{E}(\widehat{r}_n), \mathcal{E}(r), \Omega_\eta\big) \\ &= \mathbb{P}\big(\mathcal{E}(r), \Omega_\eta\big) \end{aligned}$$

where last line comes from Lemma 5. We deduce that

$$\begin{aligned}\mathcal{R}_n(\mathcal{X}(\widehat{r}_n)) &\leq 1 - \mathbb{P}(\Omega_\eta, \mathcal{E}(r)) \\ &\leq 1 - \mathbb{P}(\mathcal{E}(r)) + \mathbb{P}(\overline{\Omega_\eta}) \\ &\leq \mathcal{R}_n(\mathcal{X}(r)) + \mathbb{P}(\overline{\Omega_\eta})\end{aligned}$$

and the result follows from Lemma 3. □

Proof of Theorem 4.2: First observe that for $r > 0$,

$$\begin{aligned}1 - \mathcal{R}_n(\mathcal{X}(r)) &= \mathbb{P}\left(\exists \pi \in \Pi_M \, \forall i = 1, \ldots, M \; \mathbb{X}_n \cap S_i \subseteq \mathcal{X}_{\pi(i)}(r)\right) \\ &\geq \mathbb{P}\left(\exists \pi \in \Pi_M \, \forall i = 1, \ldots, M \; \mathbb{X}_n \cap S_i \subseteq \mathcal{X}_{\pi(i)}(r), \Omega_\eta\right) \qquad (15)\end{aligned}$$

where Ω_η is the event defined in Lemma 3. Since, under Ω_η, $N_0 < \min_{i \in \{1,\ldots,M\}} N_i$ the event in (15) equals

$$\begin{cases} \forall i = 1, \ldots, M, \; \mathbb{X}_n \cap S_i \text{ are } r\text{-connected} \\ \forall i \neq j, \text{ there is no } r\text{-connected path between } \mathbb{X}_n \cap S_i \text{ and } \mathbb{X}_n \cap S_j \\ \Omega_\eta, \end{cases}$$

which contains (since $0 < r < \delta$)

$$\begin{cases} \forall i = 1, \ldots, M, \; \mathbb{X}_n \cap S_i \text{ are } r\text{-connected} \\ \text{there is no } r\text{-connected path in } S_0 \text{ with at least } \lfloor \delta/r \rfloor + 1 \text{ observations} \\ \Omega_\eta. \end{cases}$$

We deduce from Lemmas 1 and 2 that

$$\begin{aligned}\mathcal{R}_n(\mathcal{X}(r)) \leq & \sum_{i=1}^{M} \psi_{n,i}(r) + \varphi_n\left(\left\lfloor \frac{\delta}{r} \right\rfloor + 1, r\right) + \mathbb{P}(\overline{\Omega_\eta}) \\ \leq & \Lambda r^{-d} \exp(-\mathfrak{a} n r^d) + n\varepsilon(\mathfrak{b}\varepsilon n r^D)^{\lfloor \frac{\delta}{r} \rfloor} + \mathbb{P}(\overline{\Omega_\eta}),\end{aligned}$$

where $\Lambda = \sum_{i=1}^{M} \Lambda_i$. Result follows from Lemma 3. □

References

E. Arias-Castro. Clustering based on pairwise distances when the data is of mixed dimensions. *IEEE Transaction on Information Theory*, 57(3):1692–1706, 2011.

E. Arias-Castro, G. Chen, and G. Lerman. Spectral clustering based on local linear approximations. *Electronic Journal of Statistics*, 5:1537–1587, 2011.

S. Auray, N. Klutchnikoff, and L. Rouvière. On clustering procedures and nonparametric mixture estimation. *Electronic Journal of Statistics*, 9:266–297, 2015.

G. Biau, B. Cadre, and B Pelletier. A graph-based estimator of the number of clusters. *ESAIM Probab. Stat.*, 11:272–280, 2007. ISSN 1292-8100. doi: 10.1051/ps:2007019. URL https://doi.org/10.1051/ps:2007019.

Ricardo JGB Campello, Davoud Moulavi, and Jörg Sander. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 160–172. Springer, 2013.

Hong Chang and Dit-Yan Yeung. Robust path-based spectral clustering. *Pattern Recognition*, 41(1):191–203, 2008.

P. Coretto and C. Henning. Consistency, breakdown robustness, and algorithms for robust improper maximum likelihood clustering. *Journal of Machine Learning Research*, 18:1–39, 2017.

Juan Antonio Cuesta-Albertos, Alfonso Gordaliza, and Carlos Matrán. Trimmed k-means: An attempt to robustify quantizers. *The Annals of Statistics*, 25(2):553–576, 1997.

R. Duda, P. Hart, and D. Stork. *Pattern classification*. John Wiley & Sons, 2012.

M. Ester, H. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.

L. C. Evans and R. F. Gariepy. *Measure theory and fine properties of functions*. CRC press, 2015.

Kenneth Falconer. *Fractal geometry*. John Wiley & Sons, Ltd., Chichester, third edition, 2014. ISBN 978-1-119-94239-9. Mathematical foundations and applications.

R. Filipovych, S. Resnick, and C. Davatzikos. Semi-supervised cluster analysis of imaging data. *NeuroImage*, 54(3):2185–2197, 2011.

Allen Gersho and Robert M Gray. *Vector quantization and signal compression*, volume 159. Springer Science & Business Media, 2012.

J.A. Hartigan. *Clustering Algorithms*. John Wiley, 1975.

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2(1): 193–218, 1985.

A. Jain and R. Dubes. Algorithms for clustering data. 1988.

S. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, 1967.

J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

M. Maier, M. Hein, and U. Von Luxburg. Optimal construction of k-nearest-neigbor graphs for identifying noisy clusters. *Theoritical Computer Science*, 410:1749–1764, 2009.

G. McLachlan and K. Basford. *Mixture models: Inference and applications to clustering*, volume 84. Marcel Dekker, 1988.

B. Nadler and M. Galun. Fundamental limitations of spectral clustering. In *Advances in neural information processing systems*, pages 1017–1024, 2007.

Pranav Nerurkar, Archana Shirke, Madhav Chandane, and Sunil Bhirud. Empirical analysis of data clustering algorithms. *Procedia Computer Science*, 125:770–779, 2018.

A. Y Ng, M. I Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.

D. Satish and C. Sekhar. Kernel based clustering and vector quantization for speech segmentation. In *Neural Networks, 2006. IJCNN'06. International Joint Conference on*, pages 1636–1641. IEEE, 2006.

R. Shorack and J. Wellner. *Empirical Processes with Applications to Statistics*. SIAM, 1986.

Toon Van Craenendonck and Hendrik Blockeel. Constraint-based clustering selection. *Machine Learning*, 106(9):1497–1521, 2017.

U. Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

Y. Yamanishi, J. Vert, and M. Kanehisa. Protein network inference from multiple genomic data: a supervised approach. *Bioinformatics*, 20(suppl_1):i363–i370, 2004.

L. Yengo, J. Jacques, and C. Biernacki. Variable clustering in high dimensional linear regression models. *Journal de la Societe Française de Statistique*, 155(2):19, 2014.

Charles T Zahn. Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Transactions on computers*, 100(1):68–86, 1971.

E. Zeng, C. Yang, T. Li, and G. Narasimhan. Clustering genes using heterogeneous data sources. In *Computational Knowledge Discovery for Bioinformatics Research*, pages 67–83. IGI Global, 2012.