Foundations of Perturbation Robust Clustering [1]

Jarrod Moore* and Margareta Ackerman †
* Florida State University, Tallahassee, Fl.; † San José State University, San Jose, California
Email: jdm10c@my.fsu.edu; margareta.ackerman@sjsu.edu

***Abstract*—Clustering is a fundamental data mining tool that aims to divide data into groups of similar items. Intuition about clustering reflects the ideal case – exact data sets endowed with flawless dissimilarity between individual instances. In practice however, these cases are in the minority, and clustering applications are typically characterized by noisy data sets with approximate pairwise dissimilarities. As such, the efficacy of clustering methods necessitates robustness to perturbations. In this paper, we address foundational questions on perturbation robustness, studying to what extent can clustering techniques exhibit this desirable characteristic. Our results also demonstrate the type of cluster structures required for robustness of popular clustering paradigms.**

I. Introduction

Clustering is a popular data mining tool, due in no small part to its general and intuitive goal of dividing data into groups of similar items. Yet in spite of the seeming simplicity of this task, successful application of clustering techniques in practice is oftentimes challenging. In particular, there are inherent difficulties in the data collection process and design of pairwise dissimilarity measures, both of which may significantly impact the behavior of clustering algorithms.

Intuition about clustering often reflects the ideal case – flawless data sets with well-suited dissimilarity between individual instances. In practice, however, these cases are rare. Errors are introduced into a data set for a wide variety of reasons, from precision of instruments (a student's ruler to the Large Hadron Collider alike have a set precision) to human error when data is user-reported (common in the social sciences). Additionally, the dissimilarity between pairwise instances is often based on heuristic measures, particularly when non-numeric attributes are present. Furthermore, the dynamic nature of prominent clustering applications (such as personalization for recommendation systems) implies that by the time the data has been clustered, it has already changed.

The ubiquity of flawed input poses a serious challenge. If clustering is to operate strictly under the assumption of ideal data, its applicability would be reduced to fairly rare applications where such data can be attained. As such, it would be desirable for clustering algorithms to provide some qualitative guarantees about their output when partitioning noisy data. This leads us to explore whether there are any algorithms for which such guarantees can be provided.

Although data can be faulty in a variety of ways, our focus here is on inaccuracies of pairwise distances. At a minimum, small perturbation to data should not radically affect the output of an algorithm. It would be natural to expect that some clustering techniques are more robust than others, allowing users to rely on perturbation robust techniques when pairwise distances are inexact.

In this paper, we investigate foundational questions concerning perturbation robustness, starting by asking which algorithms possess this property. However, our investigation reveals that no reasonable clustering algorithm exhibits this desirable characteristic. In fact, both additive and multiplicative perturbation robustness are unrealistic requirements. We show that no clustering algorithm can satisfy robustness to perturbation without violating even more fundamental requirements. Not only do existing methods lack this desirable characteristic, but our findings also preclude the possibility of designing novel perturbation robust clustering methods.

Perhaps it is already surprising that no reasonable clustering algorithm can be perfectly perturbation robust, but our results go further. Instead of requiring that the clustering remain unchanged following a perturbation, we allow up to *four-ninths* of all pairwise cluster relationships to change (from in-cluster to between-cluster, or vice-versa). It turns out that this substantial relaxation doesn't overcome our impossibility theorem.

Luckily, further exploration paints a more optimistic picture. A careful examination of this issue requires a look back to the underlying goal of clustering, which is to discover clustering structure in data *when such structure is present*. Our investigation suggests that sensitivity to small perturbations is inevitable only on unclusterable instances, for which clustering is inherently ill-suited. As such, it can be argued that whether an algorithm exhibits robustness on such data is inconsequential.

On the other hand, we show that when data is endowed with inherent structure, existing methods can often successfully reveal that structure even on faulty (perturbed) data. We investigate the type of cluster structures required for the success of popular clustering techniques, showing that the robustness of k-means and related methods is directly proportional to the degree of inherent cluster structure. Similarly, we show that popular linkage-based techniques are robust when clusters are well-separated. Furthermore, different cluster structures are necessary for different algorithms to exhibit robustness to perturbations.

A. Previous work

This work follows a line of research on theoretical foundations of clustering. Efforts in the field began as early as the 1970s with the pioneering work of Wright [25] on axioms of clustering, as well analysis of clustering properties by Fisher et al [17] and Jardine et al [20], among others. This field saw a renewed surge of activity following Kleinberg's [21] famous impossibility theorem, when he showed that no clustering function can simultaneously satisfy three simple properties. Also related to our work is a framework for selecting clustering methods based on differences in their input-output behavior [4, 2, 20, 26, 3, 5] as well as research on clusterability, which aims to quantify the degree of inherent cluster structure in data [13, 1, 12, 9, 22].

Previous work on perturbation robustness studies it from a computational perspective by identifying new efficient algorithms for robust instances [15, 1, 8, 11]. Ben-David and Reyzin [14] recently studied corresponding NP-hardness lower bounds. Our analysis of established methods is an essential complement to efforts in algorithmic development, as the need for understanding established methods is amplified by the fact that most clustering users rely on a small number of well-known techniques. Further, we consider a generalized notion of perturbation robustness as well as prove when this property fails to hold.

Another line of research considers how clustering algorithms behave in the presence of outliers and noise [10, 5, 16, 18, 19]. Please note that robustness to a small number of additional points is a distinct characteristic from perturbation robustness, which we consider here.

II. Definitions and notation

Clustering is a wide and heterogeneous domain. For most of this paper, we focus on a basic sub-domain where the input to a clustering function is a finite set of points endowed with a between-points dissimilarity function and the number of clusters (k), and the output is a partition of that domain.

A *dissimilarity function* is a symmetric function $d : X \times X \to R^+$, such that $d(x, x) = 0$ for all $x \in X$. The data sets that we consider are pairs (X, d), where X is some finite domain set and d is a dissimilarity function over X. A *k-clustering* $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$ of a data set X is a partition of X into k disjoint subsets (or, clusters) of X (so, $\bigcup_i C_i = X$). A *clustering* of X is a k-clustering of X for some $1 \leq k \leq |X|$.

For a clustering $\mathcal{C}$, let $|\mathcal{C}|$ denote the number of clusters in $\mathcal{C}$ and $|C_i|$ denote the number of points in a cluster C_i. For a domain X, $|X|$ denotes the number of points in X, which we denote by n when the domain is clear from context. We write $x \sim_{\mathcal{C}} y$ if x and y are both in some cluster C_j; and $x \not\sim_{\mathcal{C}} y$ otherwise. The relationship $x \sim_c y$ is an equivalence relation.

The *Hamming distance* between clusterings $\mathcal{C}$ and $\mathcal{C}'$ of the same domain set X is defined by

$$\Delta(\mathcal{C}, \mathcal{C}') = \frac{|\{\{x, y\} \subset X \mid (x \sim_{\mathcal{C}} y) \oplus (x \sim_{\mathcal{C}'} y)\}|}{\binom{|X|}{2}},$$

where $\oplus$ denotes the logical XOR operation. That is, the difference is the number of edges that disagree, being in-cluster in one of the clusterings and between-cluster in the other. The maximum distance between clusterings is when the Hamming distance is 1. Lastly, we formally define clustering functions.

Definition 1 (Clustering function)**.** *A* clustering function *is a function $\mathcal{F}$ that takes as input a pair (X, d) and a parameter $1 \leq k \leq |X|$, and outputs a k-clustering of the domain X.*

III. Robustness as a property of an algorithm

Whenever a user is faced with the task of clustering faulty data, it would be natural to select an algorithm that is robust to perturbations of pairwise dissimilarities. As such, we begin our study of perturbation robustness by casting it as a property of an algorithm. If we could classify algorithms based on whether or not (or to what degree) they are perturbation robust, then clustering users could incorporate this information when making decisions regarding which algorithms to apply on their data. First, we define what it means to perturb a dissimilarity function.

Definition 2 (ϵ-additive perturbation of a dissimilarity function)**.** *Given a pair of dissimilarity functions d and d' over a domain X, d' is an* ϵ-additive perturbation *of d, for $\epsilon > 0$, if for all $x, y \in X$, $d(x, y) - \epsilon \leq d'(x, y) \leq d(x, y) + \epsilon$.*

Multiplicative perturbation of a dissimilarity function is defined analogously. It is important to note that all of our results hold for both additive and multiplicative perturbation robustness. Perturbation robust algorithms should be invariant to data perturbations; that is, if data is perturbed, then the output of the algorithm shouldn't change. This view of perturbation robustness is not only intuitive, but is also based on previous formulations [23, 15, 8] (This can be formalized as a property of clustering functions by setting $\delta = 0$ in Definition 3 below).

From a practical point of view, it is likely that a user who has a possible perturbation of the true data set is likely to be satisfied with an approximately correct solution. This notion is similar to that used in [11]. As such, we introduce a relaxation that allows some error in the output of the algorithm on perturbed data. Multiplicative perturbation robustness is defined analogously.

Definition 3. *A clustering function $\mathcal{F}$ is* (ϵ, δ)-additive perturbation robust *if, given any data set (X, d) and $1 \leq k \leq |X|$, whenever d' is an ϵ-additive perturbation of d, $\Delta(\mathcal{F}(X, d, k), \mathcal{F}(X, d', k)) \leq \delta$.*

A. Impossibility theorem for clustering functions

We now proceed to show that perturbation robustness is too strong a requirement for clustering algorithms, and as such neither existing nor novel techniques can have this desirable characteristic.

Particularly notable is that the impossibility results persist when δ *is as high as* $4/9$, meaning that a perturbation is allowed to change up to four-ninths of all pairwise distances from in-cluster to between-cluster, or vise-versa. As such, we show that no reasonable clustering algorithm can preserve five-ninths or more of its pairwise distances after a perturbation.

The following impossibility result derives from the pioneering work of Wright [25] on axioms of clustering. Wright originally proposed his axioms in Euclidean space, here we generalize them for arbitrary pairwise dissimilarities. The first axiom we discuss follows from Wright's 11th axiom. Considering an elementary scenario, it requires that given exactly three points, an algorithm asked for two clusters should group the two closest elements.

Definition 4 (Three-body rule)**.** *Given a data set $X = \{a, b, c\}$, if $d(a, b) > d(b, c)$ and $d(a, c) > d(b, c)$, then $\mathcal{F}(X, d, 2) = \{\{a\}, \{b, c\}\}$.*

Wright's 6th axiom requires that replicating all data points by the same number should not change the clustering output. We *replicate* a point x by adding a new element x' and setting $d(x', y) = d(x, y)$, $\forall y \in X$.

Definition 5 (Replication invariance)**.** *Given any positive integer r, if all points are replicated r times, then the partitioning of the original data is unchanged and all replicas lie in the same cluster as their original element.*

Not only are these two axioms natural, as violating them leads to counterintuitive behavior, but they also hold for common techniques. It is easy to show that they are satisfied by common clustering paradigms, including cost-based methods such as k-means, k-median, and k-medoids, as well as linkage-based techniques, such as single-linkage, average-linkage and complete-linkage.

We now prove that no clustering function that satisfies the three-body rule and replication invariance can be perturbation robust. Furthermore, our result holds for all values of $\delta \leq 4/9$. Note that the following result applies to arbitrarily large data sets, for both multiplicative and additive perturbations.

Theorem 1. *For any $\delta \leq 4/9$ and $\epsilon > 0$, there is no clustering function that satisfies (ϵ, δ)-additive perturbation robustness, replication invariance, and the three-body rule. Further, the result holds for arbitrarily large data.*

Proof. We proceed by contradiction, assuming that there exists a clustering function $\mathcal{F}$ that is replication invariant, adheres to the three-body rule, and is (ϵ, δ)-additive perturbation robust for some $\delta \leq 4/9$.

Consider a data set $X = \{a, b, c\}$ with a distance function d such that $d(b, c) < d(a, b) < d(a, c)$ and $d(a, b) = d(b, c) + 0.5\epsilon$. By the three-body rule, $\mathcal{F}(X, d, 2) = \{\{b, c\}, \{a\}\}$. We now replicate each point an arbitrary number of times, r, creating three sets A, B, C such that all points that are replicas of the point a and a itself belong to A and similarly for B and C, referring to the new distance function as d_r. By replication invariance, $\mathcal{F}(A \cup B \cup C, d_r, 2) = \{B \cup C, A\}$.

Next we apply an ϵ-additive perturbation to create a distance function d'_r such that $d'_r(a, b) < d'_r(b, c) < d'_r(a, c)$ and $d'_r(c, b) = d'(b, a) + 0.5\epsilon$ $\forall a \in A, b \in B, c \in C$. By the three-body rule, $\mathcal{F}(A \cup B \cup C, d'_r, 2) = \{B \cup A, C\}$, and yet $(\epsilon, 4/9)$-additive perturbation robustness requires that the Hamming distance between $\mathcal{F}(A \cup B \cup C, d_r, 2)$ and $\mathcal{F}(A \cup B \cup C, d'_r, 2)$ be less than or equal to $4/9$. The number of in/out cluster relationships that change is $2(\frac{n}{3})^2$. The Hamming distance between the two clusterings will be $\frac{2(\frac{n}{3})^2}{\binom{n}{2}} = \frac{4n}{9(n-1)}$. Therefore for any $n \in \mathcal{Z}^+$, the Hamming distance will be greater than $4/9$.

□

IV. Robustness as a property of data

The above section demonstrates an inherent limitation of perturbation robustness as a property of clustering algorithms, showing that no reasonable clustering algorithm can exhibit this desirable characteristic. However, it turns out that perturbation robustness is possible to achieve when we restrict our attention to data endowed with inherent structure.

As such, perturbation robustness becomes a property of both an algorithm and a specific data set. We introduce a definition of perturbation robustness that directly addresses the underlying data.

Definition 6 ((ϵ, δ)-additive perturbation robustness of data)**.** *A data set (X, d) satisfies (ϵ, δ)*-additive perturbation robustness *with respect to clustering function $\mathcal{F}$ and $1 \leq k \leq |X|$, if for any d' that is an ϵ-additive perturbation of d, $\Delta(\mathcal{F}(X, d, k), \mathcal{F}(X, d', k)) < \delta$.*

Multiplicative perturbation robustness of data is defined analogously. This perspective at perturbation robustness raises a natural question: On what types of data are algorithms perturbation robust? Next, we explore the type of structures that allow popular cost-based paradigms and linkage-based methods to uncover meaningful clusters even when data is faulty.

A. Robustness of k-means and similar methods

We begin our study of data-dependent perturbation robustness by considering cluster structures required for perturbation robustness of one of the most popular clustering functions, k-means. Recall that k-means [24] finds the clustering $\mathcal{C} = \{C_1, \ldots, C_k\}$ that minimizes $\sum_{i=1}^{k} \sum_{x \in C_i} d(x, c_i)^2$, where c_i is the center of mass of cluster C_i.

Many different notions of clusterability have been proposed in prior work [1, 13]. Although they all aim to quantify the same tendency, it has been proven that notions of clusterability are often pairwise inconsistent [1]. As such, care must be taken when selecting amongst them.

In order to analyze k-means and related functions, we turn our attention to an intuitive cost-based notion, which requires that clusterings of near-optimal cost be structurally similar to the optimal solution. That is, this notion characterizes clusterable data as that which has a unique optimal solution in a strong sense, by excluding the possibility of having radically different clusterings of similar cost. See Figure 1 for an illustration.

This property, called "uniqueness of optimum" [1] and closely related variations were investigated by [12], [22], [7] and [5], among others. See [12] for a detailed exposition.

[1]This notion of clusterability appeared under several different names. The term "uniqueness of optimum" was coined by Ben-David [13].

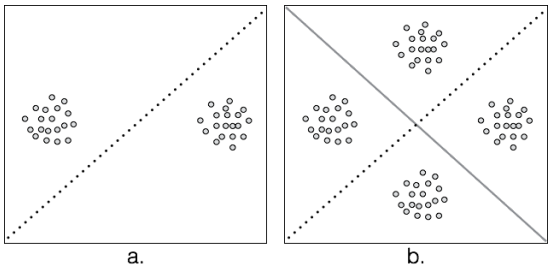

Fig. 1. An illustration of the uniqueness of optimum notion of clusterability for two clusters. Consider k-means, k-medoids, or min-sum. The highly-clusterable data depicted in (a) has a unique optimal solution, with no structurally different clusterings of near-optimal cost. In contrast, (b) displays data with two radically different clusterings of near-optimal cost, making this data poorly-clusterable for $k = 2$.

Definition 7 (Uniqueness of optimum)**.** *Given a clustering function $\mathcal{F}$, a data set (X, d) is* (δ, c, c_0, k)-uniquely optimal *if for every k-clustering $\mathcal{C}$ of X where $cost(\mathcal{C}) \leq c \cdot cost(\mathcal{F}(X, d, k)) + c_0$, $\Delta(\mathcal{F}(X, d, k), \mathcal{C}) < \delta$.*

We show that whenever data satisfies the uniqueness of optimum notion of clusterability, k-means is perturbation robust. Furthermore, the degree of robustness depends on the extent to which the data is clusterable.

For the following proofs we will use $cost_d(\mathcal{C})$ to denote the cost of clustering $\mathcal{C}$ with the distance function d. We now show the relationships between uniqueness of optimum and perturbation robustness for k-means. The following theorem shows that if data is clusterable, then it is also perturbation robust.

Theorem 2. *Consider the k-means clustering function and a data set (X, d). If (X, d) is (δ, c, c_0, k)*-uniquely optimal*, then it is also (ϵ, δ, k)-additive perturbation robust for all $\epsilon < min(\frac{c-1}{2}, \frac{-M+\sqrt{M^2+4Mc_0}}{2M})$, where $M = \binom{n}{2}$.*

Proof. Consider a data set (X, d, k), and let d' be any ϵ-additive perturbation of d. We let $\mathcal{C} = \mathcal{F}(X, d, k)$, and let $\mathcal{C}' = \mathcal{F}(X, d', k)$. These corresponds to the optimal clustering of (X, d, k) and (X, d', k). We then calculate the k-means cost of $\mathcal{C}$ given distance function d'. The k-means objective function is equivalent to $\sum_{i=1}^{k} \frac{1}{|C_i|} \sum_{x,y \in C_i} d(x, y)^2$[22]. After an additive perturbation, any pairwise distance, $d(x, y)$, is bounded by

$d(x,y)+\epsilon$. It therefore follows that:

$$
\begin{aligned}
&cost_{d'}(\mathcal{C}') \leq cost_{d'}(\mathcal{C}) \\
&\leq \sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|}\left[d(x,y)^2 + 2d(x,y)\epsilon + \epsilon^2\right] \\
&= \sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} d(x,y)^2 \\
&+ \sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} 2d(x,y)\epsilon + \sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|}\epsilon^2.
\end{aligned}
$$

The first term, is equivalent to $cost_d(C)$. We deal with the second term in by defining two sets S_1 and S_2. To define S_1, we first define S_{1i}. $S_{1i} = \{\{x,y\} \subseteq C_i | d(x,y) > 1\}$. Then $S_1 = \{S_{1i} | 1 \leq i \leq k\}$. Similarly $S_{2i} = \{\{x,y\} \subseteq C_i | d(x,y) \leq 1\}$, and $S_2 = \{S_{2i} | 1 \leq i \leq k\}$. Therefore, $\sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} 2d(x,y)\epsilon$

$$
\leq \sum_{i=1}^{k} \sum_{\{x,y\}\in S_{1i}} \frac{1}{|C_i|} 2d(x,y)\epsilon + \sum_{i=1}^{k} \sum_{\{x,y\}\in S_{2i}} \frac{1}{|C_i|} 2d(x,y)\epsilon.
$$

Because for all $\{x,y\} \in S_{1i}$ for all $1 \leq i \leq k$, $d(x,y) > 1$, we can square the $d(x,y)$ value in the first term while only increasing the total value. Likewise, we can replace the $d(x,y)$ value in the second term with 1 while only increasing the total value. This produces: $\sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} 2d(x,y)\epsilon \leq$

$$
\sum_{i=1}^{k} \sum_{\{x,y\}\in S_{1i}} \frac{1}{|C_i|} 2d(x,y)^2\epsilon + \sum_{i=1}^{k} \sum_{\{x,y\}\in S_{2i}} \frac{1}{|C_i|} 2\epsilon.
$$

Since S_{1i} and S_{2i} both consist of point pairs in C_i and we are looking for an upper bound: $\sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} 2d(x,y)\epsilon \leq$

$$
\sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} 2d(x,y)^2\epsilon + \sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} 2\epsilon.
$$

Note that $\sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} 2d(x,y)^2\epsilon$ is equivalent to $2\epsilon cost_d(C)$. We can now return to the original inequality: $cost_{d'}(\mathcal{C}') \leq cost_{d'}(\mathcal{C}) \leq (1+2\epsilon)cost_d(\mathcal{C}) + \sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} 2\epsilon + \sum_{i=1}^{k} \sum_{\{x,y\}\subseteq C_i} \frac{1}{|C_i|} \epsilon^2$. Considering the minimum and maximum cluster sizes it follows that $cost_{d'}(\mathcal{C}') \leq cost_{d'}(\mathcal{C}) \leq (1+2\epsilon)cost_d(\mathcal{C}) + \binom{n}{2}(2\epsilon+\epsilon^2)$. Then, $c \geq 1+2\epsilon$, so $\epsilon \leq \frac{c-1}{2}$. Similarly, $c_0 \geq M(\epsilon^2+2\epsilon)$, so $\epsilon \leq \frac{-M+\sqrt{M^2+4MC_0}}{2M}$ where $M = \binom{n}{2}$. So, $\epsilon < min(\frac{c-1}{2}, \frac{-M+\sqrt{M^2+4MC_0}}{2M})$. □

Similar results hold for multiplicative perturbations and other centroid based objective functions (specifically k-mediods and min-sum) in both the additive and multiplicative cases, but are omitted for brevity.

B. Perturbation robustness of Linkage-Based algorithms

We now move onto Linkage-Based algorithms, which in contrast to the methods studied in the previous section, do not seek to optimize an explicit objective function. Instead, they perform a series of merges, combining clusters according to their own measure of between-cluster distance. Given clusters $A, B \subseteq X$, the following are the between-cluster distances of some of the most popular Linkage-Based algorithms:

- **Single linkage:** $\min_{a\in A, b\in B} d(a,b)$
- **Average linkage:** $\sum_{a\in A, b\in B} \frac{d(a,b)}{(|A|\cdot|B|)}$
- **Complete linkage:** $\max_{a\in A, b\in B} d(a,b)$

We consider Linkage-Based algorithms with the k-stopping criterion, which terminate an algorithm when k clusters remain, and return the resulting partitioning. Because no explicit objective functions are used, we cannot rely on the uniqueness of optimum notion of clusterability. To define the type of cluster structure on which Linkage-Based algorithms exhibit perturbation robustness, we introduce a natural measure of clusterability based on a definition by Balcan et al [9]. The original notion required data to contain a clustering where every element is closer to all elements in its cluster than to all other points. This notion was also used in [2], [23], and [6].

Definition 8 ((ϵ,k)-Strictly Additive Separable)**.** *A data set (X,d) is* (ϵ,k)-Strictly Additive Separable *if there exists a unique clustering $\mathcal{C} = \{C_1,\ldots,C_k\}$ of X so that for all $i \neq j$ and all $x,y \in C_i$, $z \in C_j$, $d(x,y)+\epsilon \leq d(x,z)$.*

The definition for (α,k)-strictly multiplicative separable is analogous. We now show that whenever data is strictly separable, then it is also perturbation robust with respect to some of the most popular Linkage-Based algorithms. A similar result holds for multiplicative perturbation robustness. Proofs are omitted for brevity.

Theorem 3. *Single-Linkage, Average-Linkage, and Complete-Linkage are $(\epsilon,0)$-perturbation robust on all $(2\epsilon,k)$-strictly additive separable data sets.*

V. Conclusions

As a property of an algorithm, perturbation robustness fails in a strong sense, contradicting even more fundamental requirements of clustering functions. As such, no algorithm can exhibit this desirable characteristic on all data sets. Notably, this result persists even if we allow four-ninths of all pairwise distance to change following a perturbation.

However, a more optimistic picture emerges when considering clusterable data, and we show that popular paradigms are able to discover some cluster structures even on faulty data. Further, different clustering techniques are perturbation robust on different cluster structures. This has important implications for the

"user's dilemma," which is the problem of selecting a suitable clustering algorithm for a given task. Faced with the challenge of clustering data with imprecise dissimilarities between pairwise entities, a user cannot simply elect to apply a perturbation robust technique as no such methods exist, and as such the selection of suitable methods calls for some insight on the underlying structure of the data.

REFERENCES

[1] M. Ackerman and S. Ben-David. Clusterability: A theoretical study. *Proceedings of AISTATS-09, JMLR: W&CP*, 5(1-8):53, 2009.

[2] M. Ackerman, S. Ben-David, S. Branzei, and D. Loker. Weighted clustering. *Proc. 26th AAAI Conference on Artificial Intelligence*, 2012.

[3] M. Ackerman, S. Ben-David, and D. Loker. Characterization of linkage-based clustering. COLT, 2010.

[4] M. Ackerman, S. Ben-David, and D. Loker. Towards property-based classification of clustering paradigms. NIPS, 2010.

[5] M. Ackerman, S. Ben-David, D. Loker, and S. Sabato. Clustering oligarchies. *Proceedings of AISTATS-12, JMLR: W&CP*, 31(6674), 2013.

[6] M. Ackerman and S. Dasgupta. Incremental clustering: The case for extra clusters. In *Advances in Neural Information Processing Systems*, pages 307–315, 2014.

[7] Manu Agarwal, Ragesh Jaiswal, and Arindam Pal. k-means++ under approximation stability. In *Theory and Applications of Models of Computation*, pages 84–95. Springer, 2013.

[8] Pranjal Awasthi, Avrim Blum, and Or Sheffet. Center-based clustering under perturbation stability. *Information Processing Letters*, 112(1):49–54, 2012.

[9] Maria-Florina Balcan, Avrim Blum, and Santosh Vempala. A discriminative framework for clustering via similarity functions. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 671–680. ACM, 2008.

[10] Maria Florina Balcan and Pramod Gupta. Robust hierarchical clustering. In *in Proceedings of the Conference on Learning Theory (COLT*. Citeseer, 2010.

[11] Maria Florina Balcan and Yingyu Liang. Clustering under perturbation resilience. In *Automata, Languages, and Programming*, pages 63–74. Springer, 2012.

[12] M.F. Balcan, A. Blum, and S. Vempala. A discriminative framework for clustering via similarity functions. In *Proceedings of the 40th annual ACM symposium on Theory of Computing*, pages 671–680. ACM, 2008.

[13] Shai Ben-David. Computational feasibility of clustering under clusterability assumptions. *arXiv preprint arXiv:1501.00437*, 2015.

[14] Shalev Ben-David and Lev Reyzin. Data stability in clustering: A closer look. *Theoretical Computer Science*, 558:51–61, 2014.

[15] Yonatan Bilu and Nathan Linial. Are stable instances easy. In *1st Symposium on Innovations in Computer Science (ICS*, 2010.

[16] Rajesh N Dave. Characterization and detection of noise in clustering. *Pattern Recognition Letters*, 12(11):657–664, 1991.

[17] L. Fisher and J.W. Van Ness. Admissible clustering procedures. *Biometrika*, 58(1):91–104, 1971.

[18] Luis Angel García-Escudero, Alfonso Gordaliza, Carlos Matrán, and Agustín Mayo-Iscar. A review of robust clustering methods. *Advances in Data Analysis and Classification*, 4(2-3):89–109, 2010.

[19] C. Hennig. Dissolution point and isolation robustness: robustness criteria for general cluster analysis methods. *Journal of Multivariate Analysis*, 99(6):1154–1176, 2008.

[20] N. Jardine and R. Sibson. Mathematical taxonomy. *London*, 1971.

[21] J. Kleinberg. An impossibility theorem for clustering. *Proceedings of International Conferences on Advances in Neural Information Processing Systems*, pages 463–470, 2003.

[22] R. Ostrovsky, Y. Rabani, L.J. Schulman, and C. Swamy. The effectiveness of Lloyd-type methods for the k-means problem. In *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*, pages 165–176, 2006.

[23] Lev Reyzin. Data stability in clustering: A closer look. In *Algorithmic Learning Theory*, pages 184–198. Springer, 2012.

[24] D. Steinley. K-means clustering: a half-century synthesis. *British Journal of Mathematical and Statistical Psychology*, 59(1):1–34, 2006.

[25] W.E. Wright. A formalization of cluster analysis. *Pattern Recognition*, 5(3):273–282, 1973.

[26] R.B. Zadeh and S. Ben-David. A uniqueness theorem for clustering. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 639–646. AUAI Press, 2009.