Fundamenta Informaticae XX (2012) 1–15
DOI 10.3233/FI-2012-620
IOS Press

Size Constrained Distance Clustering: Separation Properties and Some Complexity Results

Alberto Bertoni, Massimiliano Goldwurm, Jianyi Lin*†, Francesco Saccà
Department of Computer Science
University of Milan
Via Comelico 39/41, 20135 Milan, Italy
{*bertoni,goldwurm*}*@dsi.unimi.it;* {*jianyi.lin,francesco.sacca*}*@unimi.it*

Abstract. In this paper we study the complexity of some size constrained clustering problems with norm L_p. We obtain the following results:

(*i*) A separation property for the constrained 2-clustering problem. This implies that the optimal solutions in the 1-dimensional case verify the so-called "String Property";

(*ii*) The NP-hardness of the constrained 2-clustering problem for every norm L_p ($p > 1$);

(*iii*) A polynomial time algorithm for the constrained 2-clustering problem in dimension 1 for every norm L_p with integer p. We also give evidence that this result cannot be extended to norm L_p with rational non-integer p;

(*iv*) The NP-hardness of the constrained clustering problem in dimension 1 for every norm L_p ($p \geq 1$).

Keywords: clustering, size constraints, NP-hardness
Mathematics Subject Classification (2000): 68Q25, 68T10

1. Introduction

Clustering or cluster analysis [9] is a method in unsupervised learning and one of the most used techniques in statistical data analysis. Clustering has a wide range of applications in many areas like pattern

*Also works: Department of Mathematics, University of Milan, Milan, Italy
†Address for correspondence: Department of Computer Science, University of Milan, Via Comelico 39/41, 20135 Milan, Italy

recognition, medical diagnostics, data mining, biology, market research and image analysis among others. A cluster is a set of data points that in some sense are similar to each other, and clustering is a process of partitioning a data set into disjoint clusters. In *distance clustering*, the similarity among data points is obtained by means of a *distance* function.

Distance clustering is a difficult problem. For an arbitrary dimension d the problem is NP-hard even if the number k of clusters equals 2 [2]; the same occurs if $d = 2$ and k is arbitrary [13]. For the Euclidean distance, a well-known heuristic is Lloyd's algorithm [11, 12], also known as the k-Means Algorithm; since this is a heuristic procedure, there is no guarantee that it converges to the global optimum. This algorithm is usually very fast, but it can require exponential time in the worst case [16].

In real-world problems, often people have some information on the clusters: incorporating this information into traditional clustering algorithms can increase the clustering performance. Problems that include background information are called *constrained clustering* problems and are divided in two classes. On the one hand, clustering problems with instance-based constraints typically comprise a set of must-link constraints or cannot-link constraints [18], defining pairs of elements that must be included, respectively, in the same cluster or in different clusters. On the other hand, clustering problems with cluster-based constraints [3, 15] incorporate constraints concerning the size of the possible clusters. Recently, in [19] cluster size constraints are used for improving clustering accuracy; this approach, for instance, allows one to avoid extremely small or large clusters in standard cluster analysis.

In this paper we study distance clustering with cluster size constraints (constrained clustering for short), mainly in the 1-dimensional case. Once $p \geq 1$ is fixed, given a set X of n reals, an integer $k > 1$ and k integers $m_1, m_2, ..., m_k$ of size constraints, the problem consists in finding a k-partition $\{A_1, A_2, ..., A_k\}$ of X with $|A_1| = m_1, ..., |A_k| = m_k$, that minimizes the objective function

$$W(A_1, A_2, ..., A_k) = \sum_{i=1}^{k} \sum_{x \in A_i} |x - c_i|^p$$

where c_i is the centroid of A_i, i.e. $c_i = \operatorname{argmin}_\mu \sum_{x \in A_i} |x - \mu|^p$.
We prove that an optimal partition $\{A_1, A_2, ..., A_k\}$ for this problem verifies the so-called *String Property*, i.e. each A_i is a set of consecutive reals of the instance set X. The String Property has previously been proved in the particular case of 1-dimensional clustering with Euclidean distance $(p = 2)$ in [5] and extended to 1-dimensional clustering with $p > 1$ in [14]. We obtain this result as a particular case of a more general separation property of the optimal solutions in the multidimensional case.

We use the string property for obtaining a polynomial time algorithm of the constrained 2-clustering in dimension 1 when p is integer. On the contrary, we show that this problem in the multidimensional case is NP-hard. Also we briefly discuss the case of non-integer rational p, showing that for $p = 3/2$ even the simpler problem of centroid localization is related to the open problem of determining the complexity of SQRT-Sum, i.e. deciding whether $\sqrt{a_1} + ... + \sqrt{a_q} > \sqrt{b_1} + ... + \sqrt{b_r}$ for positive integers $a_1, ..., a_q, b_1, ..., b_r$ [6, 1].

Finally we show that, even in the 1-dimensional case, the size constrained clustering problem is NP-hard for every p. It should be observed that in dimension 1 the clustering problem (without constraints) is solvable in polynomial time at least for $p = 2$.

This paper is organized as follows: in Section 2 we set the formal statement of the problem, in Section 3 we show some properties about the separation of the clusters in the optimal solutions, and in Section 4 we demonstrate the String Property for size constrained clustering. In Section 5 we give a hardness

result concerning the size constrained 2-clustering problem, in Section 6 we discuss the complexity of the constrained 2-clustering in the 1-dimensional case, while in Section 7 we prove the NP-hardness of the size constrained clustering in dimension 1.

2. Definitions and preliminaries

We now introduce some basic notions and preliminary results. Hereafter, for a positive integer d, we consider the space $\mathbb{R}^d$ equipped with the p-norm denoted by $\|\cdot\|_p$, with fixed $p \geq 1$, where $\|(\alpha_1, \alpha_2, ..., \alpha_d)\|_p = (\sum |\alpha_i|^p)^{\frac{1}{p}}$.

Let $X = \{x_1, x_2, ..., x_n\} \subset \mathbb{R}^d$. A *$k$-clustering* is a k-partition of X, i.e. a family $\{A_1, A_2, ..., A_k\}$ of k nonempty subsets of X such that $\cup_{i=1}^k A_i = X$ and $A_i \cap A_j = \emptyset$, for $i \neq j$. Every A_i is called a *cluster*. The *p-centroid* (or simply *centroid* when p is clearly understood) C_A of a cluster $A \subseteq X$ is

$$C_A = \operatorname*{argmin}_{\mu \in \mathbb{R}^d} \sum_{x \in A} \|x - \mu\|_p^p$$

If $p > 1$ it is well-known that the centroid is unique; in particular when $p = 2$ the centroid is the mean $C_A = (\sum_{x \in A} x)/|A|$. In the case $p = 1$ we can have different centroids; one of them is the componentwise median. The *cost* $W(A)$ of a cluster A is

$$W(A) = \sum_{x \in A} \|x - C_A\|_p^p \tag{1}$$

while the *cost* of a k-clustering $\{A_1, A_2, ..., A_k\}$ is $W(A_1, A_2, \cdots, A_k) = \sum_1^k W(A_i)$. The classical *Clustering Problem* is formulated as follows.

Definition 1. (Clustering Problem)
Given a point set $X = \{x_1, x_2, ..., x_n\} \subset \mathbb{R}^d$ and an integer $k > 1$, find a k-clustering $\{A_1, A_2, ..., A_k\}$ that minimizes the cost

$$W(A_1, A_2, \cdots, A_k) = \sum_1^k W(A_i).$$

In this paper we are interested in a version of clustering problem, where the cardinalities of the clusters are constrained. Formally, the problem can be stated as follows:

Definition 2. (Size Constrained Clustering Problem (SCC))
Given a point set $X = \{x_1, x_2, ..., x_n\} \subset \mathbb{R}^d$, an integer $k > 1$ and k positive integers $m_1, m_2, ..., m_k$ such that $\sum_1^k m_i = n$, find a k-clustering $\{A_1, A_2, ..., A_k\}$ with

$$|A_i| = m_i \quad \text{for } i = 1, ..., k$$

that minimizes the cost

$$W(A_1, A_2, \cdots, A_k) = \sum_1^k W(A_i).$$

We stress that in the SCC problem the integers n, k and d are part of the instance. On the contrary, if d is fixed the problem is called SCC-d; if k is fixed the problem is called k-SCC; furthermore, if both d and k are fixed the problem is called k-SCC-d.

We will classify the problems of kind SCC-1 or k-SCC-1 by means of the classical complexity classes [8]. In this regard, we suppose that $X = \{x_1, ..., x_n\}$ is composed by positive integers $x_1 < x_2 < ... < x_n$ represented in binary notation, whose size is $\sum |x_k|_b$, where $|x_k|_b$ is the number of bits of x_k. Observe that this is equivalent to considering $\{x_, ..., x_n\} \subset \mathbb{Q}$, the set of rational numbers, since the solution of the problems is invariant to translating and scaling. In fact, the instances $X_1 = \{x_1, ..., x_n\}$, $X_2 = \{x_1 + c, ..., x_n + c\}$, $X_3 = \{cx_1, ..., cx_n\}$ do admit the same optimal solution.

3. Separation results

In this section we prove a separation property for the optimal solution of 2-SCC. We first need a simple lemma stating that if $p > 1$ then the centroid of a set of points moves whenever one of the points changes. The property is not true in the case $p = 1$.

Lemma 1. Given $n + 1$ reals $x_1, x_2, ..., x_n, \bar{x}_1$ and $p > 1$, let $C(x_1, x_2, ..., x_n)$ be the centroid of $\{x_1, x_2, ..., x_n\}$ and $C(\bar{x}_1, x_2, ..., x_n)$ be the centroid of $\{\bar{x}_1, x_2, ..., x_n\}$.
If $\bar{x}_1 \neq x_1$, then $C(x_1, x_2, ..., x_n) \neq C(\bar{x}_1, x_2, x_3, ..., x_n)$.

Proof:
Let's suppose that $C(x_1, x_2, ..., x_n) = C(\bar{x}_1, x_2, x_3, ..., x_n) = C$. Setting $F(\mu) = \sum_i |x_i - \mu|^p$, since $F(\mu)$ is strictly convex [14], it follows that $F'(C) = 0 = \sum \text{sgn}(x_i - C)|x_i - C|^{p-1}$. Analogously, we have $0 = \text{sgn}(\bar{x}_1 - C)|\bar{x}_1 - C|^{p-1} + \sum_2^n \text{sgn}(x_i - C)|x_i - C|^{p-1}$. This implies that $\text{sgn}(\bar{x}_1 - C)|\bar{x}_1 - C|^{p-1} = \text{sgn}(x_1 - C)|x_1 - C|^{p-1}$, that is $x_1 = \bar{x}_1$. □

Corollary 2. Fixed $p > 1$, let C be the centroid of $\{x_1, x_2, ..., x_n\} \subset \mathbb{R}^d$ and $\bar{C}$ the centroid of $\{\bar{x}_1, x_2, x_3, ..., x_n\} \subset \mathbb{R}^d$, where $\bar{x}_1 \neq x_1$. Then:

$$\sum_{i=1}^{n} \|x_i - C\|_p^p < \sum_{i=1}^{n} \|x_i - \bar{C}\|_p^p$$

Proof:
Since $\bar{x}_1 \neq x_1$ there is a component (say l, with $1 \leq l \leq d$) of x_1 different from the corresponding component of $\bar{x}_1$. By Lemma 1, the l-component of C is different from the l-component of $\bar{C}$, hence $C \neq \bar{C}$. Since C is the unique minimum point of the function $\sum_i \|x_i - \mu\|_p^p$, the thesis follows. □

Proposition 3. Fixed $p > 1$, let $\{A, B\}$ be an optimal solution of a 2-SCC problem on the instance $\{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^d$ with $|A| = k$. If $x_i \in A$ and $x_j \in B$, it holds:

$$\|x_i - C_A\|_p^p + \|x_j - C_B\|_p^p < \|x_i - C_B\|_p^p + \|x_j - C_A\|_p^p$$

Proof:
Since $\{A, B\}$ is a partition, then $x_i \neq x_j$. Suppose by contradiction that:

$$\|x_i - C_A\|_p^p + \|x_j - C_B\|_p^p \geq \|x_i - C_B\|_p^p + \|x_j - C_A\|_p^p \tag{2}$$

Then, denoting with $F_X(\mu) = \sum_{x \in X} \|x - \mu\|_p^p$, we have:

$$\begin{aligned} W(A,B) &= F_A(C_A) + F_B(C_B) \\ &= F_{A \setminus \{x_i\}}(C_A) + \|x_i - C_A\|_p^p + F_{B \setminus x_j}(C_B) + \|x_j - C_B\|_p^p \\ &\geq F_{A \setminus \{x_i\}}(C_A) + \|x_j - C_A\|_p^p + F_{B \setminus \{x_j\}}(C_B) + \|x_i - C_B\|_p^p \quad \text{(by (2))} \\ &= F_{A \setminus \{x_i\} \cup \{x_j\}}(C_A) + F_{B \setminus \{x_j\} \cup \{x_i\}}(C_B) \\ &> F_{A \setminus \{x_i\} \cup \{x_j\}}(C_{A \setminus \{x_i\} \cup \{x_j\}}) + F_{B \setminus \{x_j\} \cup \{x_i\}}(C_{B \setminus \{x_j\} \cup \{x_i\}}) \text{ (by Cor. 2)} \\ &= W(A \setminus \{x_i\} \cup \{x_j\}, B \setminus \{x_j\} \cup \{x_i\}) \end{aligned}$$

This is a contradiction, since $A \neq A \setminus \{x_i\} \cup \{x_j\}$, but $|A| = |A \setminus \{x_i\} \cup \{x_j\}| = k$. This would imply that $\{A, B\}$ is not optimal. □

Theorem 4. (Separation Result)
Fixed $p > 1$, let $\{A, B\}$ be an optimal solution of a 2-SCC on the instance $\{x_1, x_2, ..., x_n\} \subset \mathbb{R}^d$ with size constraint $|A| = k$. Then we have that:

1. $C_A \neq C_B$
2. there exists $c \in \mathbb{R}$ such that:

$$\begin{aligned} x \in A \text{ implies } \|x - C_A\|_p^p - \|x - C_B\|_p^p < c \\ x \in B \text{ implies } \|x - C_A\|_p^p - \|x - C_B\|_p^p > c \end{aligned}$$

Proof:
We notice that, by Proposition 3, if $x_i \in A$ and $x_j \in B$ it holds:

$$\|x_i - C_A\|_p^p - \|x_i - C_B\|_p^p < \|x_j - C_A\|_p^p - \|x_j - C_B\|_p^p \tag{3}$$

Since $x_i \neq x_j$ it follows that $C_A \neq C_B$, otherwise (3) yields $0 < 0$. Let $\alpha = \max_{x \in A} \|x - C_A\|_p^p - \|x - C_B\|_p^p$ and $\beta = \min_{x \in B} \|x - C_A\|_p^p - \|x - C_B\|_p^p$. By (3) we obtain $\alpha < \beta$. Setting $c = \frac{\alpha + \beta}{2}$, it holds $\alpha < c < \beta$, hence:

$$\begin{aligned} x \in A \text{ implies } \|x - C_A\|_p^p - \|x - C_B\|_p^p \leq \alpha < c \\ x \in B \text{ implies } \|x - C_A\|_p^p - \|x - C_B\|_p^p \geq \beta > c \end{aligned}$$

□

The previous theorem states that, in $\mathbb{R}^d$ the hypersurface of equation

$$\|x - C_A\|_p^p - \|x - C_B\|_p^p = c \tag{4}$$

is well-defined and strictly separates the sets A and B of an optimal solution. In the particular case $p = 2$, the hypersurface becomes a hyperplane; in fact we have that (4) reduces to

$$\langle x, (C_B - C_A) \rangle = \frac{c + \|C_B\|_2^2 - \|C_A\|_2^2}{2}$$

which is the equation of a hyperplane in $\mathbb{R}^d$ (here $\langle \cdot, \cdot \rangle$ denotes the scalar product).

4. One-dimensional case: String Property

In this section we consider the case $d = 1$, i.e. $X = \{x_1, x_2, ..., x_n\}$ where $x_i \in \mathbb{R}$ for each i, and we show a structural property (String Property) of the optimal size constrained k-clustering. In this way we extend to the constrained clustering a property observed in the clustering problem by Fisher [5] in the case $p = 2$, and Novick [14] in the case $p > 1$.

Definition 3. A k-clustering $\{A_1, A_2, ..., A_k\}$ of $X = \{x_1, x_2, ..., x_n\}$ is said to have the *String Property* iff for all x_i, x_j and x_l, and for all A_s, if $x_i, x_j \in A_s$ and $x_i < x_l < x_j$ then $x_l \in A_s$.

In the case of 1-dimensional clustering with euclidean norm ($p = 2$), it is proved that any optimal solution has the String Property [5]. In [14] this result is extended to every norm $\|\cdot\|_p$ with $p > 1$.

In this section we further extend this result to the 1-dimensional size constrained clustering problem. First of all, we treat the case of 1-dimensional 2-SCC for any $p > 1$.

Proposition 5. Let $\{A, B\}$ be an optimal 2-clustering for the 2-SCC problem on instance $\{x_1, x_2, ..., x_n\}$ with $|A| = k$. Then $\{A, B\}$ has the String Property.

Proof:
Consider the function $f(x) = |x - C_A|^p - |x - C_B|^p$, where C_A, C_B are the centroids of A, B respectively. By Theorem 4 there exists c such that $x \in A$ implies $f(x) < c$, while $x \in B$ implies $f(x) > c$. Now, suppose $C_A < C_B$. We have that:

$$\begin{aligned} \text{if } x > C_B \text{ then } f'(x) &= p((x - C_A)^{p-1} - (x - C_B)^{p-1}) > 0 \\ \text{if } C_B \geq x > C_A \text{ then } f'(x) &= p((x - C_A)^{p-1} + (C_B - x)^{p-1}) > 0 \\ \text{if } C_A \geq x \text{ then } f'(x) &= p(-(C_A - x)^{p-1} + (C_B - x)^{p-1}) > 0 \end{aligned}$$

Therefore $f(x)$ is increasing; moreover it can be easily observed that $\lim_{x\to+\infty} f(x) = +\infty$ and $\lim_{x\to-\infty} f(x) = -\infty$. Since $f(x)$ is continuous, we conclude that there is a unique x^* such that $f(x^*) = c$; moreover: $x \in A$ implies $x < x^*$, $x \in B$ implies $x > x^*$. This means that, under the assumption $C_A < C_B$, $\{A, B\}$ has the String Property. Analogous reasoning applies when $C_A > C_B$, thus yielding the String Property again. □

We notice that the two half-lines $H = \{x | f(x) < c\}$ and $\bar{H} = \{x | f(x) > c\}$ are disjoint sets; furthermore A is contained in one half-line, while B is contained in the other one. We now extend the previous result to the k-SCC.

Theorem 6. Let $\{A_1, A_2, ...A_k\}$ be an optimal k-clustering for SCC on instance $X = \{x_1, x_2, ..., x_n\}$ with constraints $\{m_1, m_2, ..., m_k\}$. Then $\{A_1, A_2, ...A_k\}$ has the String Property.

Proof:
Let us reason by induction on $k \geq 2$. The case $k = 2$ is clearly solved by Proposition 5. For $k > 2$, given an optimal k-clustering $\{A_1, A_2, ..., A_k\}$, for any j we denote $v_j = \min A_j$, $V_j = \max A_j$, and set $c = \min v_j = v_\ell$. Let us consider any index $i \neq \ell$; obviously $v_i > v_\ell$. We want to show that also $v_i > V_\ell$ holds. In fact, consider the 2-SCC problem on instance $A_\ell \cup A_i$ with constraints $\{m_\ell, m_i\}$; its

optimal solution $\{A_\ell, A_i\}$ verifies the String Property because of Proposition 5, and hence $V_\ell \leq v_i$. As a consequence, every $A_i (i \neq \ell)$ is contained in the half-line $H = \{x | x > V_\ell\}$, while A_ℓ is contained in the complementary half-line $H^C = \{x | x \leq V_\ell\}$.
Let's now consider the optimal solution $\{A_1, ..., A_{\ell-1}, A_{\ell+1}, ..., A_k\}$ to the $(k-1)$-SCC problem on instance $X \setminus A_\ell$ with constraints $\{m_1, ..., m_{\ell-1}, m_{\ell+1}, ..., .m_k\}$. By induction hypothesis, $\{A_1, ..., A_{\ell-1}, A_{\ell+1}, ..., A_k\}$ verifies the String Property, and hence by the discussion above also $\{A_1, ..., A_\ell, ..., A_k\}$ does. □

5. NP-hardness of constrained 2-clustering problem

In order to highlight the usefulness of the String Property in the design of algorithms for the 1-dimensional constrained clustering, let's consider the following problem for a fixed $p > 1$:

Definition 4. (Half-Partition (HP))
Given d and $X = \{x_1, ..., x_{2n}\} \subset \mathbb{N}^d$, find the optimal 2-clustering $\{A, B\}$ of X with $|A| = |B| = n$.

When $d = 1$, we call the problem HP-1. HP-1 is solvable in polynomial time for any $p > 1$. Indeed, given the reals $x_1, x_2, ..., x_{2n}$, the unique partition $\{A, B\}$ that verifies the String Property with $|A| = |B| = n$ is $\{\{x_1, ..., x_n\}, \{x_{n+1}, ..., x_{2n}\}\}$, which hence turns out to be the optimal solution. It follows that, for any $p > 1$:

Fact 7. HP-1 is solvable in polynomial time (for any $p > 1$).

On the contrary, we show that the HP problem is NP-hard. This implies that also 2-SCC is NP-hard.

Theorem 8. HP is NP-hard (for any $p > 1$).

Proof:
We prove the result by a reduction from the Minimum Bisection Problem, which is known to be NP-hard [7]. This problem consists of determining, for an undirected graph $G = (V, E)$ with $|V| = 2n$, a subset $A \subset V$ of cardinality $|A| = n$ such that the value

$$cut(A) = |\{\ell \in E \mid \ell = \{x, y\}, x \in A, y \notin A\}|$$

is minimum.

In order to construct the reduction, let $G = (V, E)$ be an undirected graph with $V = \{1, 2, \ldots, 2n\}$ and define, for every $v \in V$, the array $X_v \in \mathbb{R}^E$ with indices in E, such that

$$X_v[\ell] = \begin{cases} 1 & \text{if } v \in \ell \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Thus, the family of arrays $\{X_1, X_2, \ldots, X_{2n}\}$ forms an instance of the Half-Partition problem for an arbitrary $p > 1$.

Given $A \subset V$ with $|A| = n$, let us compute its centroid C_A. For every $\ell \in E$, we have the following cases:

1. If both vertices of ℓ are in A then
$$C_A[\ell] = \operatorname*{argmin}_x \{2(1-x)^p + (n-2)x^p\} = \frac{1}{1+\left(\frac{n-2}{2}\right)^{\frac{1}{p-1}}} = \alpha_n$$
2. If only one vertex of ℓ is in A then
$$C_A[\ell] = \operatorname*{argmin}_x \{(1-x)^p + (n-1)x^p\} = \frac{1}{1+(n-1)^{\frac{1}{p-1}}} = \beta_n$$
3. If no vertices of ℓ is in A then $C_A[\ell] = 0$.

Now, given a 2-clustering $\{A, B\}$ with $|A| = |B| = n$, the value of objective function $W(A, B)$ can be written in the form
$$W(A,B) = \sum_{\ell \in E} \left(\sum_{i \in A} |X_i[\ell] - C_A[\ell]|^p + \sum_{j \in B} |X_j[\ell] - C_B[\ell]|^p \right)$$
If $\ell = \{i, j\}$ with $i \in A$ and $j \in B$ then we have
$$\sum_{i \in A} |X_i[\ell] - C_A[\ell]|^p + \sum_{j \in B} |X_j[\ell] - C_B[\ell]|^p = 2[(1-\beta_n)^p + (n-1)\beta_n^p]$$
On the contrary, if $\ell = \{i, j\}$ with either $\{i, j\} \subset A$ or $\{i, j\} \subset B$, then
$$\sum_{i \in A} |X_i[\ell] - C_A[\ell]|^p + \sum_{j \in B} |X_j[\ell] - C_B[\ell]|^p = 2(1-\alpha_n)^p + (n-2)\alpha_n^p$$
As a consequence, recalling that $cut(A)$ is the number of edges with a vertex in A and a vertex in B, we obtain
$$\begin{aligned} W(A,B) &= cut(A)2[(1-\beta_n)^p + (n-1)\beta_n^p] + (|E| - cut(A))[2(1-\alpha_n)^p + (n-2)\alpha_n^p] \\ &= |E| \cdot g(n,p) \; + \; cut(A) \cdot s(n,p) \end{aligned} \tag{6}$$
where $g(n, p)$ does not depend on $\{A, B\}$ and
$$s(n,p) = 2[(1-\beta_n)^p + (n-1)\beta_n^p] - 2(1-\alpha_n)^p - (n-2)\alpha_n^p$$
Now, for any fixed $p > 1$, as n tends to $+\infty$ we have
$$\alpha_n \sim \left(\frac{2}{n}\right)^{\frac{1}{p-1}}, \;\; \beta_n \sim n^{-\frac{1}{p-1}}$$
and hence
$$s(n,p) \sim (p-1)\left(2^{\frac{p}{p-1}} - 2\right) \cdot n^{-\frac{1}{p-1}} \; > \; 0$$
Therefore, from equation (6), if n is sufficiently large we obtain
$$\operatorname*{argmin}_{|A|=n} W(A,B) = \operatorname*{argmin}_{|A|=n} cut(A)$$
□

6. Complexity of constrained 2-clustering problem in dimension 1

In the previous section we have shown that 2-SCC is NP-hard. Here we prove that 2-SCC-1 is solvable in polynomial time if $p \geq 1$ is an integer. 2-SCC-1 is an extension of HP-1: while any instance of HP-1 admits a unique 2-clustering verifying the String Property, every instance of 2-SCC-1 admits two 2-clusterings π_1, π_2 verifying the String Property. For finding the solution, we have to compare $W(\pi_1)$ and $W(\pi_2)$. The comparison is immediate if $p = 1$ or $p = 2$, as in these cases the centroids can be easily computed. If p is an integer greater than 2, the problem seems to be more difficult.

In this regard, we consider a slight extension of the 2-SCC-1 problem obtained by considering the integer p, given in unary, as part of the instance. More precisely, we consider the problem *Uniform 2-SCC-1*: given a set X of positive integers $x_1 < x_2 < ... < x_n$, the size constraints $\{s, n-s\}$ and an integer $p > 2$ in unary notation, find a 2-clustering $\{A_1, A_2\}$ of X with $|A_1| = s, |A_2| = n - s$ that minimizes the cost

$$\sum_{x_i \in A_1} \|x_i - C_1\|_p^p + \sum_{x_j \in A_2} \|x_j - C_2\|_p^p$$

where C_1 and C_2 are the p-centroids of A_1 and A_2 respectively.

We solve this problem by using approximation techniques; in this regard, we recall a useful result about the solutions of a square algebraic system appeared in [4].

Theorem 9. (Canny's Gap)
Let $(x_1, x_2, ..., x_N)$ be a solution of an algebraic system of N equations in N unknowns having a finite number of solutions, with maximum degree d and with coefficients in $\mathbb{Z}$ smaller or equal to M in absolute value. Then, for each $i = 1, ..., N$, either $x_i = 0$ or $|x_i| > (3Md)^{-Nd^N}$.

Given a set Y of integers $y_1 < y_2 < ... < y_m$, let j be the index such that the p-centroid C of Y verifies $y_j \leq C < y_{j+1}$. Fixed ε ($0 < \varepsilon < \frac{1}{2}$), we call *$\varepsilon$-approximation* of C a number $\bar{C}$ with $C \leq \bar{C} \leq C + \varepsilon$ if $y_{j+1} - C > C - y_j$, and $C - \varepsilon \leq \bar{C} \leq C$ otherwise. In any case, it holds $|\bar{C} - C| \leq \varepsilon$.

Lemma 10. Given an integer $p > 2$ and m integers $1 \leq y_1 < y_2 < ... < y_m$, let C be the p-centroid of $Y = \{y_1, ..., y_m\}$ and $W(Y)$ be the cost function defined in (1). Then, in polynomial time with respect to $p + \ln y_m$, one can compute polynomials $A(x) = \sum_0^{p-1} a_i x^i$ and $B(x) = \sum_0^p b_i x^i$ such that:

1. C is a root of $A(x)$;
2. $W(Y) = B(C)$;
3. $|a_i|, |b_i| \leq m \cdot (y_m + 1)^p$ for all $i = 1, ..., m$

Moreover, for any $\varepsilon > 0$, it is possible to compute in polynomial time with respect to $p + \log \frac{y_m}{\varepsilon}$ an $\varepsilon-$approximation $\bar{C}$ of C such that

4. $|B(C) - B(\bar{C})| \leq \varepsilon \cdot y_m^{p-1} \cdot p \cdot m$.

Proof:
We know that there is j such that $y_j \leq C < y_{j+1}$. To compute such a j note that, since $p > 2$, function $D(x) = \sum_{i=1}^m |x - y_i|^p$ admits first derivative in all $\mathbb{R}$ and every value $D'(y_i)$ is easily computable for each i. Since $D(x)$ is strictly convex, the required coefficient is the unique j such that $D'(y_j) \leq 0$ and $D'(y_{j+1}) > 0$. The computation time is clearly polynomial with respect to $p + \log y_m$. As a consequence

one can also compute the polynomials:

$$B(x) = \sum_{i=1}^{j}(x-y_i)^p + \sum_{i=j+1}^{m}(y_i-x)^p = \sum_{0}^{p} b_i x^i$$

$$A(x) = \frac{1}{p}B'(x) = \sum_{i=1}^{j}(x-y_i)^{p-1} - \sum_{i=j+1}^{m}(y_i-x)^{p-1} = \sum_{0}^{p-1} a_i x^i$$

The centroid C satisfies $A(C) = 0$; moreover $W(Y) = B(C)$. Observe now that, denoting with $[x^i]B(x)$ the coefficient of x^i in $B(x)$, we have:

$$|b_i| = |[x^i]B(x)| \leq [x^i]\sum_{h=1}^{m}(y_h+x)^p \leq \sum_{h=1}^{m}(y_h+1)^p \leq m(y_m+1)^p$$

$$|a_i| \leq [x^i]\sum_{h=1}^{m}(y_h+x)^{p-1} \leq \sum_{h=1}^{m}(y_h+1)^{p-1} \leq m(y_m+1)^{p-1} \leq m(y_m+1)^p$$

For computing an $\varepsilon-$approximation $\bar{C}$, remark that for all x such that $y_j \leq x < y_{j+1}$, it holds $A(x) < 0$ if $x < C$, while $A(x) > 0$ whenever $x > C$. Therefore, we use $\lceil \log \frac{y_{j+1}-y_j}{\varepsilon} \rceil \leq \lceil \log \frac{y_m}{\varepsilon} \rceil$ many steps of a binary search for localizing C, i.e. computing an interval $[\alpha, \beta]$ such that $\alpha \leq C \leq \beta$ and $\beta - \alpha \leq \varepsilon$. If $A\left(\frac{y_j+y_{j+1}}{2}\right) < 0$ then we set $\bar{C} = \alpha$, otherwise $\bar{C} = \beta$. The computation time at every step is bounded by a polynomial in $p + \log \frac{y_m}{\varepsilon}$.
To prove the last point we can write $|B(C) - B(\bar{C})|$ as:

$$\left|\left[\sum_{1}^{j}(C-y_i)^p + \sum_{j+1}^{m}(y_i-C)^p\right] - \left[\sum_{1}^{j}(\bar{C}-y_i)^p + \sum_{j+1}^{m}(y_i-\bar{C})^p\right]\right| =$$

$$= \left|\left[\sum_{1}^{j}(\underbrace{C-y_i}_{u})^p - (\underbrace{\bar{C}-y_i}_{\bar{u}})^p\right] + \left[\sum_{j+1}^{m}(\underbrace{y_i-C}_{-u})^p - (\underbrace{y_i-\bar{C}}_{-\bar{u}})^p\right]\right|$$

Fixed the index i in the first summation, denote $u = C - y_i$ and $\bar{u} = \bar{C} - y_i$. Since $|u|, |\bar{u}| \leq y_m$, it holds: $|u^p - \bar{u}^p| = |(u-\bar{u})(u^{p-1} + u^{p-2}\bar{u} + ... + \bar{u}^{p-1})| \leq \varepsilon \cdot p \cdot y_m^{p-1}$. Observe that every single term in the last parenthesis is $u^h \bar{u}^{p-1-h} = (C-y_i)^h(\bar{C}-y_i)^{p-1-h} \leq y_m^{p-1}$, thus yielding $|u^p - \bar{u}^p| \leq \varepsilon p y_m^{p-1}$. On the other hand, when fixing the index i in the second summation, the same upper bound is obtainable. We can conclude that $|B(C) - B(\bar{C})| \leq m\varepsilon p y_m^{p-1}$. □

We are ready now to state the main result of this section.

Theorem 11. The Uniform 2-SCC-1 problem is solvable in polynomial time.

Proof:
Given a set X of positive integers $x_1 < x_2 < ... < x_n$ and constraints $\{s, n-s\}$, because of the String Property the optimal solution of the problem must be chosen between the two partitions

$$\pi_1 = \{A, B\} \text{ with } A = \{x_1, ..., x_s\}, B = \{x_{s+1}, ..., x_n\}$$
$$\pi_2 = \{D, E\} \text{ with } D = \{x_1, ..., x_{n-s}\}, E = \{x_{n-s+1}, ..., x_n\}$$

Thus, it is sufficient to calculate the costs of the clusters A, B, D, E and check whether

$$W = W(\pi_1) - W(\pi_2) = W(A) + W(B) - W(D) - W(E)$$

is positive, null or negative.

To this end, we can consider the system of equations:

$$\begin{cases} A_i(z_i) = 0 & \text{(for all } i = 1, ..., 4) \\ w = B_1(z_1) + B_2(z_2) - B_3(z_3) - B_4(z_4) \end{cases}$$

with polynomials A_i, B_i ($i = 1, ..., 4$) obtained according to Lemma 10 separately for each cluster A, B, D, E respectively. This is a system of 5 algebraic equations of degree at most p in 5 unknowns $z_1, ..., z_4, w$, which is satisfied by the solution (C_1, C_2, C_3, C_4, W), where C_i's ($i = 1, ..., 4$) are centroids of A, B, D, E respectively. By Lemma 10, the coefficients of the polynomials in the system are bounded by $M = n(x_n + 1)^p$. Hence, by applying Canny's Gap Theorem, either $W = 0$ or $|W| > \delta$ where

$$\delta = [3n(x_n + 1)^p p]^{-5p^5}$$

Thus, if we find an approximation $\bar{W}$ of W up to $\frac{\delta}{3}$ we can conclude:

if $\bar{W} < -\frac{\delta}{2}$ then $W < 0$ and π_1 is the optimal solution;

if $\bar{W} > \frac{\delta}{2}$ then $W > 0$ and π_2 is the optimal solution;

if $|\bar{W}| \leq \frac{\delta}{2}$ then $W = 0$ and both π_1 and π_2 are optimal solutions.

$\bar{W}$ can be obtained by computing $\bar{W} = B_1(\bar{C}_1) + B_2(\bar{C}_2) - B_3(\bar{C}_3) - B_4(\bar{C}_4)$, where $\bar{C}_i$ is an ε-approximation of C_i, with ε that guarantees $|W - \bar{W}| \leq \frac{\delta}{3}$. By the last point of Lemma 10, we know that:

$$|W - \bar{W}| \leq 4\varepsilon p \cdot n(x_n)^{p-1}$$

It is sufficient to choose ε such that

$$4\varepsilon p n(x_n)^{p-1} < \frac{\delta}{3} = \frac{1}{3}[3n(x_n + 1)^p p]^{-5p^5}$$

Then we can approximate C_i up to the s-th binary digit after the point, where $\varepsilon = 2^{-s}$. By the previous equation we have

$$s = O(p^6 \log x_n).$$

The approximate centroids $\bar{C}_i$ ($i = 1, ..., 4$) can be obtained in polynomial time as in Lemma 10, and the computation of $\bar{W}$ requires a polynomial number of arithmetic operations on numbers of polynomial size. □

The previous method cannot be extended to the case of rational non-integer p. To put in evidence the subtleties of this case, we briefly discuss the problem of localizing the p-centroid.

Definition 5. The problem of localizing the p-centroid (p-LC) consists of deciding, for a set X of integers $\{x_1, ..., x_n\}$ and an integer h, whether $C > h$, where C is the p-centroid of X.

It is easy to observe that the well-known problem SQRT-Sum is polynomially reducible to $\frac{3}{2}$-LC. SQRT-Sum requires to decide, given positive integers $a_1, ..., a_q$, $b_1, ..., b_r$, whether $\sqrt{a_1} + ... + \sqrt{a_q} > \sqrt{b_1} + ... + \sqrt{b_r}$.

Theorem 12. SQRT-Sum is polynomially reducible to $\frac{3}{2}$-LC.

Proof:
With the instance $a_1, ..., a_q, b_1, ..., b_r$ of SQRT-Sum we associate the instance $X = \{x_1, ..., x_{q+r}\}$ and h of $\frac{3}{2}$-LC where:

$$1) h = \max a_j \qquad 2) x_i = h - a_i \text{ for } i \leq q \qquad\qquad 3) x_{q+j} = h + b_j \text{ for } 1 \leq j \leq r$$

Setting $F(\mu) = \sum_{i=1}^{q+r} |x_i - \mu|^{\frac{3}{2}}$, since $F(\mu)$ is strictly convex, we have:
1) $F'(\mu)$ is increasing function;
2) if C is the $\frac{3}{2}$-centroid of X, then $F'(C) = 0$.
Observe now that

$$\frac{2}{3}F'(h) = \sum_{x_i \geq h} (x_i - h)^{\frac{1}{2}} - \sum_{x_i < h} (h - x_i)^{\frac{1}{2}} = \sqrt{b_1} + ... + \sqrt{b_r} - \sqrt{a_1} - ... - \sqrt{a_q}$$

We hence conclude that:

$$h < C \quad \text{iff} \quad F'(h) < F'(C) \quad \text{iff} \quad \sqrt{a_1} + ... + \sqrt{a_q} - \sqrt{b_1} - ... - \sqrt{b_r} > 0$$

This proves the reduction. □

The characterization of the computational complexity of SQRT-Sum was proposed as open problem in [6]; despite the efforts, the best-known result, due to Allender et al. [1], puts SQRT-Sum in CH, i.e. the Counting Hierarchy introduced in [17]. Theorem 12 implies that, if $\frac{3}{2}$-LC were solvable in polynomial time, then SQRT-Sum$\in$P would hold, despite still today a major open problem is to decide whether SQRT-Sum is solvable in NP.

7. NP-hardness of constrained clustering problem in dimension 1

Because of the String Property, the clustering problems in dimension 1 can be solved in polynomial time by a simple dynamic programming technique (in case of Euclidean norm). On the contrary, in this section we prove that the corresponding 1-dimensional constrained clustering (SCC-1) is NP-hard, for every $p \geq 1$.

First of all, we reformulate SCC-1 as a decision problem.

Definition 6. (SCC-1: decision version)
Given a set X of n integers $x_1 < x_2 < ... < x_n$, positive integers $m_1, ..., m_k$ such that $\sum m_i = n$, and a positive integer λ (called threshold), decide whether there exists a k-clustering $\{A_1, ..., A_k\}$ of X, with constraints $|A_i| = m_i$ ($i = 1, ..., k$), such that $W(A_1, ..., A_k) < \lambda$.

We first notice that the clustering problem without constraints is known to be solvable in polynomial time when $p = 2$. We prove that adding the constraints makes the problem hard. The proof is based on a reduction from the 3-Partition problem.

Definition 7. (3-Partition Problem)
Given a set $P = \{p_1, ..., p_{3m}\}$ of positive integers whose sum is mB, such that each p_i satisfies $B/4 < p_i < B/2$, decide whether there exists a partition $\{P_1, ..., P_m\}$ of P such that, for each $i = 1, ..., m$, $\sum_{x \in P_i} x = B$.

An equivalent version of this problem has been proved to be NP-complete in [10]; it remains NP-complete even if the numbers in P are all bounded by a polynomial in m. The problem was originally proved to be strongly NP-complete in [8] when P is a multiset.

Theorem 13. SCC-1 is NP-hard (for any $p \geq 1$).

Proof:
We want to reduce 3-Partition to the decision version of SCC-1. With the instance $P = \{p_1, ..., p_{3m}\}$ of 3-Partition we associate the instance of SCC-1 (decision version) given by $X = \cup_1^m X_j$ with constraints $\{p_1, ..., p_{3m}\}$ and threshold $\lambda = 3mB^{2p}$, where $X_j = \{Hj+h : h = 0, ..., B-1\}$, with $H = 6mB^2+B$ and $B = \sum_{i=1}^{3m} p_i/m$. Now let's show the correctness of this reduction.

A partition $\{A_1, ..., A_{3m}\}$ of X is said to be *fine* if for every A_i there is X_j with $A_i \subseteq X_j$: the main observation is that 3-Partition with instance P admits a solution if and only if there is a fine partition $\{A_1, ..., A_{3m}\}$ of X s.t. $|A_i| = p_i$ for all $i = 1, ..., 3m$. In fact, let $\{P_1, ..., P_m\}$ be a partition of P satisfying $\sum_{x \in P_i} x = B$; with every $P_i = \{p_{i1}, p_{i2}, p_{i3}\}$ we associate a partition $\mathcal{A}_i = \{A_{i1}, A_{i2}, A_{i3}\}$ of X_i s.t. $|A_{ij}| = p_{ij}$ $(j = 1, 2, 3)$, which is possible since $\sum_{x \in P_i} x = B = |X_i|$; thus $\cup_1^m \mathcal{A}_i$ is a fine partition of X satisfying the constraints $\{p_1, ..., p_{3m}\}$, since $\cup P_i = P$. Suppose now that the partition $\{A_1, ..., A_{3m}\}$ of X is fine and satisfies the constraints $|A_i| = p_i$, $i = 1, ..., 3m$. With every X_j we associate $P_j = \{|A_i| : A_i \subseteq X_j\}$: since $\sum_{x \in P_j} x = |X_j| = B$, $\{P_1, ..., P_m\}$ verifies the instance $\{p_1, ..., p_{3m}\}$ of 3-Partition.

To prove the correctness of the reduction, it is sufficient to observe that $\{A_1, ..., A_{3m}\}$ is a clustering of X with constraints $\{p_1, ..., p_{3m}\}$ and cost $W(A_1, ..., A_{3m}) < \lambda$ iff $\{A_1, ..., A_{3m}\}$ is fine with constraints $\{p_1, ..., p_{3m}\}$. Suppose $\{A_1, ..., A_{3m}\}$ is fine, then $W(A_1, ..., A_{3m}) = \sum_{i=1}^{3m} W(A_i)$. For all A_i there is X_j s.t. $A_i \subseteq X_j$; therefore $W(A_i) \leq W(X_j) < B^{p+1}$. In conclusion: $W(A_1, ..., A_{3m}) = \sum W(A_i) < 3mB^{p+1} \leq 3mB^{2p} = \lambda$. Now suppose $\{A_1, ..., A_{3m}\}$ is not fine: there is A_i containing x, y with $x \in X_s$, $y \in X_t$ and $s \neq t$. Observe that $|x - y|$ is at least $H - B$; if μ is the p-centroid of A_i, then either $|x - \mu| \geq \frac{H-B}{2}$ or $|y - \mu| \geq \frac{H-B}{2}$. It follows that

$$\begin{aligned} W(A_1, ..., A_{3m}) \geq W(A_i) &\geq |x - \mu|^p + |y - \mu|^p \\ &\geq (\frac{H - B}{2})^p = (3mB^2)^p \geq 3mB^{2p} = \lambda \end{aligned}$$

□

8. Conclusions

In this paper we have studied some algorithmic problems on distance clustering with cluster size constraints. We have emphasized some analogies and differences between clustering and constrained clustering. In particular, we have obtained separation results for the optimal solutions that, in the 1-dimensional case, imply the so-called String Property. In this way a well-know result in clustering is extended to constrained clustering.

As in the case of classical clustering, the String Property allows us to obtain exact efficient algorithms for solving constrained 2-clustering in dimension 1, with norm L_p for integer p, while the problem turns out to be NP-hard when the dimension is not fixed. Moreover, we have given evidence that the method cannot be extended to the case of rational non-integer p. At last, we have shown that constrained clustering is NP-hard even in dimension 1, while the corresponding problem in classical clustering is solvable in polynomial time, at least with the Euclidean norm.

In this paper we have put the attention to exact algorithms. In this context, we leave open the problem of finding efficient algorithms for constrained k-clustering, for small $k > 1$; in fact, separation results seem to indicate that this kind of problems can be solved in polynomial time. Other open problems are those of determining efficient approximation algorithms for constrained 2-clustering in arbitrary dimension or for constrained clustering in dimension 1. Finally, an interesting issue is the development of "practical" heuristics, for the general constrained clustering, that play the same role of k-Means or k-Medoids in classical clustering.

References

[1] Allender, E., Bürgisser, P., Kjeldgaard-Pedersen, J., Miltersen, P. B.: On the Complexity of Numerical Analysis, *Proc. 21st Ann. IEEE Conf. on Computational Complexity (CCC'06)*, 2006.

[2] Aloise, D., Deshpande, A., Hansen, P., Popat, P.: NP-hardness of Euclidean sum-of-squares clustering, *Machine Learning*, **75**, 2009, 245–249.

[3] Bradley, P. S., Bennett, K. P., Demiriz, A.: *Constrained K-Means Clustering*, Technical Report MSR-TR-2000-65, Miscrosoft Research Publication, May 2000.

[4] Canny, J.: *The complexity of robot motion planning*, MIT Press, Cambridge, MA, USA, 1988.

[5] Fisher, W. D.: On Grouping for Maximum Homogeneity, *Journal of the American Statistical Association*, **53**(284), 1958, 789–798.

[6] Garey, M., Graham, R., Johnson, D.: Some NP-complete geometric problems, *Proceedings of the eighth annual ACM symposium on Theory of Computing*, ACM New York, NY, USA, 1976.

[7] Garey, M., Johnson, D., Stockmeyer., L.: Some simplified NP-complete graph problems, *Theor. Comput. Sci.*, **1**(3), 1976, 237–267.

[8] Garey, M. R., Johnson, D. S.: *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman & Co., New York, 1979, ISBN 0716710447.

[9] Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd edition, Springer-Verlag, 2009.

[10] Hulett, H., Will, T. G., Woeginger, G. J.: Multigraph realizations of degree sequences: Maximization is easy, minimization is hard, *Operations Research Letters*, **36**(5), 2008, 594 – 596.

[11] Lloyd, S.: Least squares quantization in PCM, *IEEE Transactions on Information Theory*, **28**(2), 1982, 129–137, ISSN 0018-9448.

[12] MacQueen, J. B.: Some method for the classification and analysis of multivariate observations, *Proceedings of the 5th Berkeley Symposium on Mathematical Structures*, 1967.

[13] Mahajan, M., Nimbhorkar, P., Varadarajan, K.: The Planar k-Means Problem is NP-Hard, in: *WALCOM: Algorithms and Computation* (S. Das, R. Uehara, Eds.), vol. 5431 of *Lecture Notes in Computer Science*, Springer Berlin/Heidelberg, 2009, 274–285.

[14] Novick, B.: Norm statistics and the complexity of clustering problems, *Discrete Applied Mathematics*, **157**, 2009, 1831–1839.

[15] Tung, A., Han, J., Lakshmanan, L., Ng, R.: Constraint-Based Clustering in Large Databases, in: *Database Theory ICDT 2001* (J. Van den Bussche, V. Vianu, Eds.), vol. 1973 of *Lecture Notes in Computer Science*, Springer Berlin/Heidelberg, 2001, 405–419.

[16] Vattani, A.: K-means requires exponentially many iterations even in the plane, *Proceedings of the 25th Symposium on Computational Geometry (SoCG)*, 2009.

[17] Wagner, K. W.: The complexity of combinatorial problems with succinct input representation, *Acta Informatica*, **23**(3), 1986, 325–356.

[18] Wagstaff, K., Cardie, C.: Clustering with Instance-level Constraints, *Proc. of the 17th Intl. Conf. on Machine Learning*, 2000.

[19] Zhu, S., Wang, D., Li., T.: Data clustering with size constraints, *Knowledge-Based Systems*, **23**(8), 2010, 883–889.