DE GRUYTER OPEN

Int. J. Appl. Math. Comput. Sci., 2014, Vol. 24, No. 1, 151–163
DOI: 10.2478/amcs-2014-0012

amcs

CENTER–BASED L_1–CLUSTERING METHOD

KRISTIAN SABO

Department of Mathematics
University of Osijek, Trg Lj. Gaja 6, HR 31 000 Osijek, Croatia
e-mail: ksabo@mathos.hr

In this paper, we consider the l_1-clustering problem for a finite data-point set which should be partitioned into k disjoint nonempty subsets. In that case, the objective function does not have to be either convex or differentiable, and generally it may have many local or global minima. Therefore, it becomes a complex global optimization problem. A method of searching for a locally optimal solution is proposed in the paper, the convergence of the corresponding iterative process is proved and the corresponding algorithm is given. The method is illustrated by and compared with some other clustering methods, especially with the l_2-clustering method, which is also known in the literature as a smooth k-means method, on a few typical situations, such as the presence of outliers among the data and the clustering of incomplete data. Numerical experiments show in this case that the proposed l_1-clustering algorithm is faster and gives significantly better results than the l_2-clustering algorithm.

Keywords: l_1-clustering, data mining, optimization, weighted median problem.

1. Introduction

Clustering or grouping a data set into conceptually meaningful clusters is a well-studied problem in recent literature (Äyrämö, 2006; Frąckiewicz and Palus, 2011; Gan *et al*., 2007; Iyigun, 2007; Kogan, 2007; Teboulle, 2007), and it has practical importance in a wide variety of applications such as computer vision, signal image video analysis, multimedia, networks, biology, medicine, geology, psychology, business, politics and other social sciences. The classification and ranking of objects are also becoming more and more interesting topics for researchers, decision makers and state administrations.

Generally speaking, clustering algorithms can be divided into two main groups (Jain, 2010), i.e., hierarchical and partitional. The former (Gan *et al*., 2007) recursively find nested clusters either in agglomerative mode (starting with each data point in its own cluster and merging the most similar pair of clusters successively to form a cluster hierarchy) or in divisive mode (starting with all the data points in one cluster and recursively dividing each cluster into smaller clusters). The latter, on the other hand, find all the clusters simultaneously as a partition of the data and do not impose a hierarchical structure. The most well-known hierarchical algorithms are single-link, complete-link, average-link and Ward algorithms; the most popular and the simplest partitional algorithm is the k-means algorithm.

Partitional clustering algorithms can be divided into two classes, i.e., hard clustering, where each data belongs to only one cluster, and soft clustering, where every data point belongs to every cluster to a certain degree. Well-known soft clustering methods include fuzzy k-means (Bezdek, 1981), the expectation maximization algorithm (see, e.g., Duda *et al*., 2001), the smooth k-means algorithm based on the Euclidean l_2-norm (Kogan, 2007; Teboulle, 2007), etc.

Motivated by the smooth k-means method (l_2-method) (Kogan, 2007; Teboulle, 2007), in this paper we consider a soft clustering method that is based on the l_1-norm. The method is a generalization of the one-dimensional center-based l_1-clustering method proposed by Sabo *et al*. (2012). It is well known that the n-dimensional clustering problem can be reduced to one-dimensional ones by projection of data-points onto the line that corresponds to the main principal axes associated with the data-point set (Kogan, 2007), or by some nonlinear multidimensional scaling method (Gan *et al*., 2007). Since numerical experiments show that, in comparison with the corresponding one-dimensional method, the n-dimensional l_1-clustering method generally gives better results, the main aim of this paper is to provide formal theoretical background for the n-dimensional case.

In this paper, by $\mathbb{R}^n$ we will denote the n-dimensional Euclidean space, whose elements are n-tuples of real numbers, which are called points. The space $\mathbb{R}^n$ is equipped with a structure of a real vector space, with usual addition and multiplication with scalars. Analogously, the set of all points $\theta = (c^1, \ldots, c^k)$, whereby $c^s \in \mathbb{R}^n$, $s = 1, \ldots, k$, will be denoted by $\mathbb{R}^{kn}$. Finally, the set of nonnegative real numbers will be denoted by $\mathbb{R}_+$.

A partition of the data-points set $\mathcal{A} = \{a^i \in \mathbb{R}^n \colon i = 1, \ldots, m\} \subset I^n \subset \mathbb{R}^n$, where $I^n = \{(x_1, \ldots, x_n) \in \mathbb{R}^n \colon \alpha_i \le x_i \le \beta_i,\ \alpha_i, \beta_i \in \mathbb{R}\}$, into k disjoint nonempty subsets $\pi_1, \ldots, \pi_k$, $1 \le k \le m$, such that

$$\begin{aligned} \bigcup_{i=1}^{k} \pi_i &= \mathcal{A}, \\ \pi_i \cap \pi_j &= \emptyset, \quad i \ne j, \\ |\pi_j| &\ge 1, \quad j = 1, \ldots, k, \end{aligned} \tag{1}$$

will be further denoted by $\Pi(\mathcal{A}) = \{\pi_1, \ldots, \pi_k\}$, and the elements $\pi_1, \ldots, \pi_k$ of such a partition are called *clusters in* $\mathbb{R}^n$.

If $d \colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_+$ is some distance-like function (see, e.g., Kogan, 2007; Teboulle, 2007), then with each cluster $\pi_j \in \Pi$ we can associate its center c^j defined by

$$c^j = \arg \min_{x \in \operatorname{conv}(\pi_j)} \sum_{a^i \in \pi_j} d(x, a^i), \tag{2}$$

where $\operatorname{conv}(\pi_j)$ is a convex hull of the set π_j.

If we define an objective function $\mathcal{F} \colon \mathcal{P}(\mathcal{A}, k) \to \mathbb{R}_+$ on the set of all partitions $\mathcal{P}(\mathcal{A}, k)$ of the set $\mathcal{A}$ containing k clusters by

$$\mathcal{F}(\Pi) = \sum_{j=1}^{k} \sum_{a^i \in \pi_j} d(c^j, a^i), \tag{3}$$

then we can define an optimal partition $\Pi^\star$, such that

$$\mathcal{F}(\Pi^\star) = \min_{\Pi \in \mathcal{P}(\mathcal{A}, k)} \mathcal{F}(\Pi).$$

Conversely, for a given set of centers $c^1, \ldots, c^k \in I^n$, applying the *minimal distance principle* (see, e.g., Kogan, 2007; Teboulle, 2007), we can define the partition $\Pi = \{\pi_1, \ldots, \pi_k\}$ of the set $\mathcal{A}$. Therefore, the problem of finding an optimal partition of the set $\mathcal{A}$ can be reduced to the following global optimization problem:

$$\min_{c^1, \ldots, c^k \in I^n} F(c^1, \ldots, c^k), \tag{4}$$

$$F(c^1, \ldots, c^k) = \sum_{i=1}^{m} \min_{j=1, \ldots, k} d(c^j, a^i),$$

where $F \colon I^{kn} \to \mathbb{R}_+$ and $I^{kn} = \{(x^1, \ldots, x^k) \colon x^s \in I^n,\ s = 1, \ldots, k\} \subset \mathbb{R}^{kn}$. In general, the functional F is not differentiable and it may have many local or global minima. The optimization problem (4) can also be found in the literature as a *center-based clustering problem* or a *k-median problem* (Iyigun, 2007; Leisch, 2006; Teboulle, 2007).

If $d(x, y) = \|x - y\|_2^2$, we deal with the l_2 or *Least Squares* (LS) clustering problem, and if $d(x, y) = \|x - y\|_1$, it is the l_1 or *Least Absolute Deviations* (LAD) clustering problem. The l_1-clustering problem can be reduced to the following nonconvex and nonsmooth optimization one:

$$\min_{c^1, \ldots, c^k \in I^n} \Phi(c^1, \ldots, c^k), \tag{5}$$

$$\Phi(c^1, \ldots, c^k) = \sum_{i=1}^{m} \min_{j=1, \ldots, k} \|c^j - a^i\|_1,$$

where $\Phi \colon I^{kn} \to \mathbb{R}_+$, is a continuous function.

For example, Cominetti and Michelot (1997) present a sufficient condition for clustering in l_1-location problems, based on the concept of an attraction cluster, and Zhang *et al*. (2012) propose a cluster-dependent multi-metric clustering approach by using the l_p-norm with special stress placed on robust clustering and outlier detection methods. Various clustering methods based on the l_1-norm can be found in the works of Jajuga (1987; 1991) and Späth (1976; 1987).

Let us mention several interesting applications of l_1-optimality clustering that often occur in the literature. For example, Angulo and Serra (2007) propose a new polar representation for quantitative image processing by using the l_1-norm, and Jörnsten (2004) considers a classifier based on the l_1 data depth for the analysis of microarray gene expression data. Li *et al*. (2010) propose a novel rotational invariant l_1-norm based discriminant analysis in the presence of outliers. Choulakian (2001) and Meng *et al*. (2012) consider a principal component analysis of a data set based on the l_1-norm. In the work of Grbić *et al*. (2013), the problem of global data approximation on the basis of data containing outliers is considered and a new method named the moving least absolute deviations method is proposed.

The optimization problem (5) can be transformed in the following way. Since the nondifferentiable function $f \colon \mathbb{R}^k \to \mathbb{R}$, $f(z) = \max_{j=1, \ldots, k} z_j$ can be approximated by a differentiable function

$$f_\epsilon(z) = \epsilon \ln \sum_{j=1}^{k} \exp\left(z_j / \epsilon\right)$$

(see, e.g., Boyd and Vandenberghe, 2004; Malinen and Fränti, 2012), instead of solving the problem (5), we can solve the following optimization problem (Kogan, 2007; Teboulle, 2007):

$$\min_{c^1, \ldots, c^k \in I^n} \Phi_\epsilon(c^1, \ldots, c^k), \tag{6}$$

$$\Phi_\epsilon(c^1,\dots,c^k) = -\epsilon \sum_{i=1}^{m} \ln \sum_{j=1}^{k} \exp\left(-\tfrac{1}{\epsilon}\|c^j - a^i\|_1\right),$$

where $\Phi_\epsilon\colon I^{kn} \to \mathbb{R}$. This is a continuous optimization problem, where the objective function does not have to be either convex or differentiable, and generally it may have many local or global minima and consequentially several stationary points in the sense of Clarke (1990).

Inspired by the method given by Kogan (2007) and Teboulle (2007), in our paper (Sabo *et al*., 2012) we propose an iterative procedure for determining stationary points of the function Φ_ϵ given by (6) for a special case $n = 1$. Now, we give a natural generalization of that iterative procedure for an arbitrary dimension of data $n \in \mathbb{N}$.

The paper is organized as follows. Section 2 gives some properties of the function Φ_ϵ. In Section 3, a *weighted median of the data-point set* $\mathcal{A} \subset \mathbb{R}^n$ is defined, by means of which in Section 4 an iterative procedure is constructed, which always converges to some stationary point of the function Φ_ϵ. Typical situations, such as the presence of outliers among the data and clustering of incomplete data, are illustrated by means of one example on synthetic data and three examples on empirical data.

2. Properties of the function Φ_ϵ

In this section we are going to analyze some properties of the function Φ_ϵ. To simplify the notation, we write $\theta := (c^1,\dots,c^k) \in \mathbb{R}^{kn}$, $c^s \in \mathbb{R}^n$, $s = 1,\dots,k$. Analogously as in the work of Sabo *et al*. (2012), the relationship between the function Φ given by (5) and Φ_ϵ given by (6) (Lemma 1) and the Lipschitz property of the function Φ_ϵ (Lemma 2) can be shown.

Lemma 1. *Let $\mathcal{A} = \{a^i \in \mathbb{R}^n\colon i = 1,\dots,m\} \subset I^n \subset \mathbb{R}^n$, $I^n = \{(x_1,\dots,x_n) \in \mathbb{R}^n\colon \alpha_i \le x_i \le \beta_i,\ \alpha_i,\beta_i \in \mathbb{R}\}$, be a given set of data-points, and let Φ and Φ_ϵ, $\epsilon > 0$, be functions given by* (5) *and* (6)*, respectively. Then, for all $\theta \in I^{kn}$, the following inequalities hold:*

$$0 < \Phi(\theta) - \Phi_\epsilon(\theta) \le \epsilon\, m \ln k. \tag{7}$$

Lemma 2. *For all $\theta_1, \theta_2 \in I^{kn}$, there holds*

$$|\Phi_\epsilon(\theta_2) - \Phi_\epsilon(\theta_1)| \le mn\, \|\theta_2 - \theta_1\|_\infty.$$

The function Φ_ϵ is continuous, and according to Lemma 1, it is bounded below,

$$\Phi_\epsilon(\theta) \ge \Phi(\theta) - \epsilon\, m \ln k \ge -\epsilon\, m \ln k.$$

Therefore, since $I^{kn} \subset \mathbb{R}^{kn}$ is compact, Φ_ϵ attains its global minimum.

Since the function $\Phi_\epsilon\colon I^{kn} \to \mathbb{R}_+$ is Lipschitz-continuous, we have a well-defined Clarke generalized subdifferential (see, e.g., Ruszczynski, 2006), which can be written as

$$\begin{aligned}\partial\Phi_\epsilon(\theta) = \{&(u^1,\dots,u^k) \in \mathbb{R}^{kn}\colon \\ &u^s = \sum_{i=1}^{m} w_i^s(\theta)\left(\sigma_{\lambda_1}(c_1^s, a_1^i),\dots,\sigma_{\lambda_2}(c_n^s, a_n^i)\right) \\ &\lambda_j \in [-1,1]\},\end{aligned} \tag{8}$$

where

$$\sigma_\zeta(c,a) = \begin{cases} \operatorname{sign}(c-a) & \text{if } c \ne a, \\ \zeta & \text{if } c = a, \end{cases} \tag{9}$$

$$w_i^s(\theta) = \frac{\exp(-\frac{1}{\epsilon}\|c^s - a^i\|_1)}{\sum_{j=1}^{k} \exp(-\frac{1}{\epsilon}\|c^j - a^i\|_1)},$$

$$\theta = (c^1,\dots,c^k), \quad s = 1,\dots,k, \quad i = 1,\dots,m.$$

If $\theta^\star \in I^{kn}$ is a local minimum of the Lipschitz continuous function $\Phi_\epsilon\colon I^{kn} \to \mathbb{R}_+$, then $0 \in \partial\Phi_\epsilon(\theta^*)$. Conversely, every point $\hat\theta \in I^{kn}$ for which $0 \in \partial\Phi_\epsilon(\hat\theta)$ is a *stationary point* of the function Φ_ϵ.

3. Weighted median of the data-point set $\mathcal{A} \subset \mathbb{R}^n$

In this section we define a *weighted median of the data-point set* $\mathcal{A} \subset \mathbb{R}^n$ (see also Sabo and Scitovski, 2008; Vardi and Zhang, 2000; Vazler *et al*., 2012), which will be used for construction of an iterative procedure for searching for stationary points of the function Φ_ϵ defined by (6).

Definition 1. A *weighted median of the data-point set* $\mathcal{A} = \{a^i = (a_1^i,\dots,a_n^i) \in \mathbb{R}^n\colon i = 1,\dots,m\} \subset \mathbb{R}^n$ with the corresponding weights $w = (w_1,\dots,w_m)$, $w_i > 0$, is any point from the set

$$\begin{aligned}\operatorname{Med}(w,\mathcal{A}) := \Big\{&(u_1,\dots,u_n) \in \mathbb{R}^n\colon \\ &u_l \in \operatorname*{Med}_{i=1,\dots,m}(w_i, a_l^i),\ l = 1,\dots,n\Big\},\end{aligned} \tag{10}$$

where

$$\begin{aligned}\operatorname*{Med}_{i=1,\dots,m}(w_i, a_l^i) = \Big\{&x_l^* \in \mathbb{R}\colon \sum_{i=1}^{m} w_i |x_l^* - a_l^i| \\ &\le \sum_{i=1}^{m} w_i |x - a_l^i|,\ \forall x \in \mathbb{R}\Big\},\end{aligned} \tag{11}$$

is the set of all weighted medians of real numbers $(a_l^i,\ i = 1,\dots,m)$ with the corresponding weights $w_i > 0$.

Remark 1. For every $l = 1, \ldots, n$, the set $\mathrm{Med}_{i=1,\ldots,m}(w_i, a_l^i)$ given by (11) is obtained as a solution of a weighted median problem, and it can be a singleton $\{a_l^i\}$ for some $i \in \{1, \ldots, m\}$ or an interval of real numbers $[a_l^i, a_l^j]$ for some $i, j \in \{1, \ldots, m\}$. The elements of the set $\mathrm{Med}_{i=1,\ldots,m}(w_i, a_l^i)$ will be denoted by $\mathrm{med}_{i=1,\ldots,m}(w_i, a_l^i)$. Thus, for every $l = 1, \ldots, n$ there exists $\mathrm{med}_{i=1,\ldots,m}(w_i, a_l^i) \in \mathrm{Med}_{i=1,\ldots,m}(w_i, a_l^i)$, which coincides with some of the numbers $a_l^1, \ldots, a_l^m$.

The set $\mathrm{Med}(w, \mathcal{A})$ given by (10) belongs to the convex hull of the set $\mathcal{A}$, and it can have only one point $(u_1, \ldots, u_n) \in \mathrm{Med}(w, \mathcal{A})$, where $u_l \in \{a_l^1, \ldots, a_l^m\}$ for every $l = 1, \ldots, n$, or it can be a hyperrectangle, with the vertex of the form $(v_1, \ldots, v_n)$, where $v_l \in \{a_l^1, \ldots, a_l^m\}$ for every $l = 1, \ldots, n$. The elements of the set $\mathrm{Med}(w, \mathcal{A})$ are denoted by $\mathrm{med}(w, \mathcal{A})$.

Note that, if we write

$$\mathfrak{A} = \{(u_1, \ldots, u_n) \in \mathbb{R}^n \colon u_l \in \{a_l^1, \ldots, a_l^m\}, \quad l = 1, \ldots, n\},$$

then we can conclude that there exists $\mathrm{med}(w, \mathcal{A}) \in \mathrm{Med}(w, \mathcal{A})$, such that $\mathrm{med}(w, \mathcal{A}) \in \mathfrak{A}$. In practical situations, such as the aforementioned iterative procedure for determining stationary points of the function Φ_ϵ, it will not be necessary to know the whole set $\mathrm{Med}(w, \mathcal{A})$, but it would suffice to determine just one of its representatives belonging to the set $\mathfrak{A}$. Vardi and Zhang (2000) propose a new, simple, fast, monotonically converging algorithm for deriving the weighted median of the data-point set in $\mathbb{R}^n$.

Example 1. Let $\mathcal{A} = \{(1,1), (2,1), (5,2), (6,3), (4,5), (2,4)\}$ and $w = (1,1,1,1,1,1)$ be the corresponding weights. The points of the set $\mathfrak{A}$ are placed in nodes of the network shown in Fig. 1. Since

$$\operatorname*{Med}_{i=1,\ldots,6}(w_i, a_1^i) = [2, 4],$$

and

$$\operatorname*{Med}_{i=1,\ldots,6}(w_i, a_2^i) = [2, 3],$$

it follows that $\mathrm{Med}(w, \mathcal{A}) = [2, 4] \times [2, 3]$. Note that in this case the set $\mathrm{Med}(w, \mathcal{A})$ does not contain any point from the data-points set $\mathcal{A}$, but it contains four points from the set $\mathfrak{A}$ (see Fig. 1(a)).

If we replace the point $(2, 4)$ by $(2, 3)$, the weighted median $\mathrm{Med}(w, \mathcal{A})$ does not change, but now $a^6 \in \mathrm{Med}(w, \mathcal{A})$ (Fig. 1(b)).

If we drop the point a^6 from the set $\mathcal{A}$, the weighted median of the data-points set $\mathcal{A}$ becomes the point which belongs to the set $\mathfrak{A} \setminus \mathcal{A}$ (Fig. 1(c)). ♦

An overview of useful properties of the weighted median of real numbers can be found in the work of Vazler *et al.* (2012). If the number of real numbers is large, calculation of the weighted median of the data may require a lot of computing time (Cupec *et al.*, 2009; Sabo *et al.*, 2011; Sabo and Scitovski, 2008). Several fast algorithms are given by Gurwitz (1990). The following proposition holds.

Proposition 1. *The set* $\mathrm{Med}(w, \mathcal{A})$ *is equal to the set* $\arg\min_{\xi \in \mathbb{R}^n} \phi(\xi)$ *of all global minimizers (i.e., points of global minima) of the convex function* $\phi \colon \mathbb{R}^n \to \mathbb{R}_+$ *given by the formula*

$$\phi(\xi) = \sum_{i=1}^{m} w_i \, \|a^i - \xi\|_1.$$

Proof. Let

$$\xi^* = \left(\operatorname*{med}_{j=1,\ldots,m}(w_j, a_1^j), \ldots, \operatorname*{med}_{j=1,\ldots,m}(w_j, a_n^j) \right) \in \mathrm{Med}(w, \mathcal{A}).$$

Let us show that

$$\xi^* \in \arg\min_{\xi \in \mathbb{R}^n} \phi(x).$$

There holds

$$\begin{aligned}
\phi(\xi^*) &= \sum_{i=1}^{m} w_i \|a^i - \xi^*\|_1 \\
&= \sum_{i=1}^{m} w_i \sum_{l=1}^{n} |a_l^i - \operatorname*{med}_{j=1,\ldots,m}(w_j, a_l^j)| \\
&= \sum_{l=1}^{n} \sum_{i=1}^{m} w_i |a_l^i - \operatorname*{med}_{j=1,\ldots,m}(w_j, a_l^j)| \\
&= \sum_{l=1}^{n} \min_{\xi_l \in \mathbb{R}} \sum_{i=1}^{m} w_i |a_l^i - \xi_l| \\
&= \min_{\xi = (\xi_1, \ldots, \xi_n) \in \mathbb{R}^n} \sum_{l=1}^{n} \sum_{i=1}^{m} w_i |a_l^i - \xi_l| \\
&= \min_{x \in \mathbb{R}^n} \sum_{i=1}^{m} w_i \|a^i - \xi\|_1 \\
&= \min_{\xi \in \mathbb{R}^n} \phi(\xi),
\end{aligned}$$

i.e., $\xi^* \in \arg\min_{\xi \in \mathbb{R}^n} \phi(\xi)$.

Conversely, we show that, if

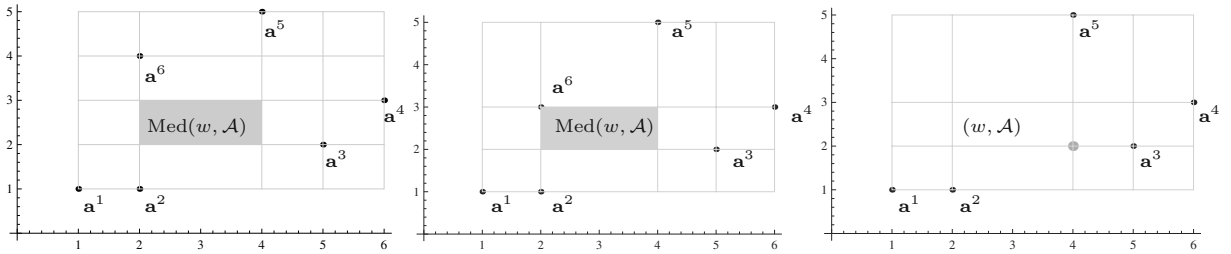
$$\xi^* = (\xi_1^*, \ldots, \xi_n^*) \in \arg\min_{\xi \in \mathbb{R}^n} \phi(\xi),$$

then

$$\xi_l^* \in \operatorname*{Med}_{j=1,\ldots,m}(w_i, a_l^j)$$

for every $l = 1, \ldots, n$. For that purpose, let us notice that for every $l = 1, \ldots, n$ the following holds:

$$\sum_{i=1}^{m} w_i |a_l^i - \xi_l^*| \geq \sum_{i=1}^{m} w_i |a_l^i - \operatorname*{med}_{j=1,\ldots,m}(w_j, a_l^j)|, \quad (12)$$



(a) (b) (c)

Fig. 1. Median of the data-point set $\mathcal{A}$: $\mathrm{Med}(w,\mathcal{A})$ is the set (a), $\mathrm{Med}(w,\mathcal{A})$ is the set (b), $\mathrm{Med}(w,\mathcal{A})$ is the point (c).

where

$$\operatorname*{med}_{j=1,\dots,m}(w_j,a_l^j)\in\operatorname*{Med}_{j=1,\dots,m}(w_j,a_l^j),$$

$\forall l=1,\dots,n$. Thereby, the equality in (12) holds if and only if

$$\xi_l^*\in\operatorname*{Med}_{j=1,\dots,m}(w_j,a_l^j),\quad l=1,\dots,n.$$

Adding up (12) for $l=1,\dots,n$, we obtain

$$\begin{aligned}\phi(\xi^*)&=\sum_{l=1}^{n}\sum_{i=1}^{m}w_i|a_l^i-\xi_l^*|\\&\geq\sum_{l=1}^{n}\sum_{i=1}^{m}w_i|a_l^i-\operatorname*{med}_{j=1,\dots,m}(w_j,a_l^j)|\\&=\phi\left(\operatorname*{med}_{j=1,\dots,m}(w_j,a_1^j),\dots,\operatorname*{med}_{j=1,\dots,m}(w_j,a_n^j)\right),\end{aligned}$$

whereby the equality holds if and only if

$$\xi_l^*\in\operatorname*{Med}_{j=1,\dots,m}(w_j,a_l^j)$$

for every $l=1,\dots,n$. Since under the assumption that vector ξ^* comes from the set of global minima of the function ϕ, there holds

$$\phi(\xi^*)\leq\phi\left(\operatorname*{med}_{j=1,\dots,m}(w_j,a_1^j),\dots,\operatorname*{med}_{j=1,\dots,m}(w_j,a_n^j)\right),$$

which together with (12) results in

$$\phi(\xi^*)=\phi\left(\operatorname*{med}_{j=1,\dots,m}(w_j,a_1^j),\dots,\operatorname*{med}_{j=1,\dots,m}(w_j,a_n^j)\right),$$

i.e.,

$$\xi_l^*\in\operatorname*{Med}_{j=1,\dots,m}(w_j,a_l^j)$$

for every $l=1,\dots,n$. ■

4. Method for finding stationary points of the function Φ_ϵ

Motivated by Cord *et al.* (2006), Kogan (2007) and Teboulle (2007), similarly to Sabo *et al.* (2012), in this section we construct an efficient iterative process for detecting stationary points of the function Φ_ϵ. Assuming that $\theta^{(t)}=\left(c^1(t),\dots,c^k(t)\right)\in\mathbb{R}^{kn}$, $c^s(t)\in\mathbb{R}^n$, $s=1,\dots,k$, is known, we are going to look for the next approximation $\theta^{(t+1)}=\left(c^1(t+1),\dots,c^k(t+1)\right)\in\mathbb{R}^{kn}$, $c^s(t+1)\in\mathbb{R}^n$, $s=1,\dots,k$, where $c^s(t+1)$ is the weighted median of the data-point set $\mathcal{A}$ with appropriate weights, i.e.,

$$c^s(t+1)=\operatorname{med}\left(w^s(\theta^{(t)}),\mathcal{A}\right),\quad s=1,\dots,k,\quad(13)$$

where

$$w^s(\theta^{(t)})=(w_1^s(\theta^{(t)}),\dots,w_m^s(\theta^{(t)})),$$

$s=1,\dots,k$, and

$$w_i^s(\theta^{(t)})=\frac{\exp(-\frac{1}{\epsilon}\|c^s(t)-a^i\|_1)}{\sum\limits_{j=1}^{k}\exp(-\frac{1}{\epsilon}\|c^j(t)-a^i\|_1)},\quad i=1,\dots,m.$$

According to Definition 1, the weighted median (13) is some point from the convex hull $\operatorname{conv}(\mathcal{A})$ of the set $\mathcal{A}$, and its representative from the set $\mathfrak{A}\subset\operatorname{conv}(\mathcal{A})$ can always be chosen. Thus we further assume that a sequence $\left(\theta^{(t)}\right)$ is contained in the set $\mathfrak{A}$.

By Proposition 1 we can assume that each component $c^s(t+1)$ of the next approximation $\theta^{(t+1)}$ is obtained as a solution of the following optimization problem:

$$c^s(t+1)=\arg\min_{\zeta\in\mathbb{R}^n}g_s(\zeta;\theta^{(t)}),\quad(14)$$

where $g_s\colon\mathbb{R}^n\to\mathbb{R}_+$,

$$g_s(\zeta;\theta^{(t)})=\sum_{i=1}^{m}w_i^s(\theta^{(t)})\|\zeta-a^i\|_1.$$

Note that g_s are continuous, but non-differentiable convex functions. Let $g(\,\cdot\,;\theta^{(t)})\colon \mathbb{R}^{kn} \to \mathbb{R}_+$ be a convex function defined by

$$g(\theta;\theta^{(t)}) = \sum_{s=1}^{k} g_s(c_s;\theta^{(t)}), \quad \theta = (c^1,\ldots,c^k). \tag{15}$$

Because of the convexity of the function g, there exists (see, e.g., Boyd and Vandenberghe, 2004)

$$\theta^{(t+1)} = \arg\min_{\theta\in I^{kn}} g(\theta;\theta^{(t)}), \tag{16}$$

whereby

$$\begin{aligned} c^s(t+1) &= \arg\min_{\xi\in I^n} g_s(\xi;\theta^{(t)}) \\ &= \operatorname{med}\left(w^{(s)}(\theta^{(t)}),\mathcal{A}\right), \end{aligned} \tag{17}$$

$s = 1,\ldots,k$. In that way we defined the iterative process which associates the kn-tuple $\theta^{(t)}$ with the kn-tuple $\theta^{(t+1)}$.

Remark 2. Since we supposed that $\theta^{(t)} \in \mathfrak{A}^k$, i.e., $c^s(t) \in \mathfrak{A}$ for all $s = 1,\ldots,k$, the iterative process is defined in such a way that it searches for stationary points of Φ_ϵ among the points of the set $\mathfrak{A}^k$.

Because of symmetry properties of Φ and Φ_ϵ, if $\hat\theta = \left(\hat c^1,\ldots,\hat c^k\right)$ minimizes the functions Φ_ϵ and $\tilde\theta$ is an arbitrary componentwise permutation of $\hat\theta$, then also $\tilde\theta$ minimizes Φ_ϵ and therefore the function Φ_ϵ attains its global minimum in at least $k!$ points.

Note also that iterative procedure (16) can be constructed as a Gauss–Seidel iterative procedure, and in this way it will accelerate the process even more.

4.1. Convergence of the iterative process.

The following proposition can be easily checked (see also Sabo *et al*., 2012).

Proposition 2.

(*i*) *For every $i = 1,\ldots,m$ and an arbitrary $\theta \in \mathbb{R}^{kn}$, the sequence of weights $w_i^s(\theta)$, $s = 1,\ldots,k$, satisfies $0 < w_i^s(\theta) < 1$.*

(*ii*) *For an arbitrary $\theta^{(0)} \in I^{kn}$, the sequence $\left(\theta^{(t)}\right)$, defined by the iterative process* (16)*, remains in $\mathfrak{A}^k \subset I^{kn}$, and hence it is bounded.*

Proposition 3. *Let $\theta^{(0)} \in \mathbb{R}^{kn}$ be an arbitrary point. Let the sequence $\left(\theta^{(t)}\right)$ be given by the iterative process* (16)*, and let $\Phi_\epsilon\colon I^{kn} \to \mathbb{R}_+$ be the function given by* (6)*. If $\theta^{(t+1)} \neq \theta^{(t)}$, then $\Phi_\epsilon(\theta^{(t+1)}) < \Phi_\epsilon(\theta^{(t)})$.*

Similarly to a one-dimensional center-based l_1-clustering method described by Sabo *et al*. (2012), the following holds.

Theorem 1. *Let $\theta^{(0)} \in \mathbb{R}^{kn}$ be an arbitrary point, let the sequence $\left(\theta^{(t)}\right)$ be defined by the iterative process* (16)*, and let $\Phi_\epsilon\colon I^{kn} \to \mathbb{R}_+$ be the function given by* (6)*. Then*

(*i*) *the sequence $\left(\theta^{(t)}\right)$ has an accumulation point;*

(*ii*) *the sequence $\left(\Phi_\epsilon^{(t)}\right)$, where $\Phi_\epsilon^{(t)} := \Phi_\epsilon(\theta^{(t)})$, converges;*

(*iii*) *every accumulation point $\hat\theta$ of the sequence $\left(\theta^{(t)}\right)$ is a stationary point of the function Φ_ϵ, and it is obtained by the iterative process* (16) *in finitely many steps, i.e., there exists a $\mu \in \mathbb{N}$, such that $\theta^{(\mu+1)} = \theta^{(\mu)} = \hat\theta$;*

(*iv*) *if $\hat\theta_1$ and $\hat\theta_2$ are two accumulation points of the sequence $\left(\theta^{(t)}\right)$, then $\Phi_\epsilon(\hat\theta_1) = \Phi_\epsilon(\hat\theta_2)$.*

Proof. We shall prove each part separately.

(i) By Proposition 2, the sequence $\left(\theta^{(t)}\right)$ is bounded, and therefore it has an accumulation point.

(ii) By Proposition 3 the sequence $\left(\Phi_\epsilon^{(t)}\right)$ is monotonously decreasing, and by Lemma 1 the function Φ_ϵ is bounded below. Therefore, there exists a $\Phi_\epsilon^\star$, such that

$$\Phi_\epsilon^\star = \lim_{t\to\infty} \Phi_\epsilon^{(t)}.$$

(iii) Since the sequence $\Phi_\epsilon(\theta^{(t)})$ converges and $\theta^{(t)}$ belongs to $\mathfrak{A}^k$, which is a finite set, there exists a $\mu \in \mathbb{N}$ such that $\Phi_\epsilon(\theta^{(\mu+1)}) = \Phi_\epsilon(\theta^{(\mu)})$. According to Proposition 3, we have

$$\theta^{(\mu+1)} = \theta^{(\mu)} = \hat\theta. \tag{18}$$

Because

$$\theta^{(\mu+1)} = \arg\min_{\theta\in I^{kn}} g(\theta;\theta^{(\mu)}),$$

we conclude that

$$0 \in \partial g(\theta^{(\mu+1)};\theta^{(\mu)}),$$

where $\partial g(\theta;\theta^{(t)})$ is a Clarke generalized subdifferential of the function g at the point $\theta = (c^1,\ldots,c^k)$,

$$\begin{aligned} \partial g(\theta;\theta^{(t)}) = \Big\{ &(u^1,\ldots,u^k) \in \mathbb{R}^{kn} : \\ &u^s = \sum_{i=1}^{m} w_i^s(\theta^{(t)})\big(\sigma_{\lambda_1}(c_1^s,a_1^i), \\ &\ldots,\sigma_{\lambda_n}(c_n^s,a_n^i)\big), \lambda_j \in [-1,1]\Big\}, \end{aligned} \tag{19}$$

where the function σ_ζ is given by (9). From (18) it follows that

$$\begin{aligned}0 &\in \partial g(\theta^{(\mu+1)};\theta^{(\mu)})\\ &= \partial g(\theta^{(\mu)};\theta^{(\mu)})\\ &= \Big\{(u^1,\dots,u^k)\in\mathbb{R}^{kn}:\\ u^s &= \sum_{i=1}^{m} w_i^s(\theta^{(\mu)})\big(\sigma_{\lambda_1}(c_1^s(\mu),a_1^i),\\ &\quad \dots,\sigma_{\lambda_n}(c_n^s(\mu),a_n^i)\big),\lambda_j\in[-1,1]\Big\},\end{aligned}$$

which coincides with the Clarke generalized subdifferential $\partial\Phi_\epsilon(\theta^{(\mu)})$ of the function Φ_ϵ given by (8), at the point $\theta^{(\mu)}$. Therefore, $\theta^{(\mu)} = \hat{\theta}$ is a stationary point of the functionΦ_ϵ.

(iv) Let $\left(\theta_1^{(t)}\right)$ and $\left(\theta_2^{(t)}\right)$ be two subsequences of the sequence $\left(\theta^{(t)}\right)$, such that

$$\hat{\theta}_1 = \lim_{t\to\infty}\theta_1^{(t)},\quad \hat{\theta}_2 = \lim_{t\to\infty}\theta_2^{(t)}.$$

Since the sequence $\left(\Phi_\epsilon^{(t)}\right)$ converges, we have

$$\begin{aligned}\Phi_\epsilon(\hat{\theta}_1) &= \lim_{t\to\infty}\Phi_\epsilon(\theta_1^{(t)}) = \lim_{t\to\infty}\Phi_\epsilon^{(t)}\\ &= \lim_{t\to\infty}\Phi_\epsilon(\theta_2^{(t)}) = \Phi_\epsilon(\hat{\theta}_2).\end{aligned}$$

■

4.2. l_1-clustering algorithm. Theorem 1 shows that, given an initial approximation $\theta^{(0)}\in\mathbb{R}^{kn}$, the iterative process (16) always converges to some stationary point which is not unique. In addition, Theorem 1(iii) gives a criterion for terminating the iterative process (16). The corresponding algorithm is given by Algorithm 1.

Remark 3. Let us mention one possibility for the choice of the smoothing parameter $\epsilon > 0$ (see also Malinen and Fränti, 2012). If we want a relative deviation $\Phi(\theta^{(0)}) - \Phi_\epsilon(\theta^{(0)})/\Phi(\theta^{(0)})$ between the function Φ and Φ_ϵ in the initial approximation $\theta^{(0)}$ to be less than the number $\delta > 0$ set in advance, then by using Lemma 1 we obtain

$$\epsilon \le \delta\frac{\Phi(\theta^{(0)})}{m\ln k}.$$

Since numbers $\exp(-\frac{1}{\epsilon}\|c^s - a^i\|_1)$ are negligible when the point a^i is not close to the center c^s, in that case the weights w_i^s from Step 2 are also negligible, so that in that sense Algorithm 1 can speed up. In accordance with Kogan (2007) and Teboulle (2007), the corresponding l_2-clustering algorithm can also be defined analogously.

Algorithm 1. l_1-clustering.

Step 1. Input $m\ge 1$, $1\le k\le m$, $\epsilon>0$, $\mathcal{A}=\{a^i\in\mathbb{R}^n\colon i=1,\dots,m\}$. Choose an initial approximation of centers $\theta^{(0)}=(c^1,\dots,c^k)$.

Step 2. For all $s=1,\dots,k$ define vectors w^s with components

$$w_i^s = \frac{\exp(-\frac{1}{\epsilon}\|c^s-a^i\|_1)}{\sum\limits_{j=1}^{k}\exp(-\frac{1}{\epsilon}\|c^j-a^i\|_1)},\quad i=1,\dots,m.$$

Step 3. Set $\theta^{(1)}=(c^1,\dots,c^k)$, where

$$c^s = \operatorname{med}(w^s,\mathcal{A}),\quad s=1,\dots,k.$$

Step 4. If $\theta^{(1)}=\theta^{(0)}$, set $\theta^{(0)}=(c^1,\dots,c^k)$ and go to Step 2. Otherwise, go to Step 5.

Step 5. According to the minimal distance principle, define a partition $\Pi=\{\pi_1,\dots,\pi_k\}$ with centers $c^1,\dots,c^k$:

$$\begin{aligned}\pi_1 &= \{a^i\in\mathcal{A}\colon \|a^i-c^1\|_1\le\|a^i-c^l\|_1,\ l=1,\dots,k\},\\ \pi_j &= \{a^i\in\mathcal{A}\setminus\bigcup_{s=1}^{j-1}\pi_s\colon \|a^i-c^j\|_1\le\|a^i-c^l\|_1,\\ &\qquad \forall\, l=1,\dots,k\},\quad j=2,\dots,k.\end{aligned}$$

5. Numerical examples

In this section, the proposed method and Algorithm 1 are tested and compared with several clustering algorithms. Special attention is paid to the comparison with the l_2-clustering algorithm (Kogan, 2007; Teboulle, 2007). In accordance with Kogan (2007), the n-dimensional data-points can be reduced to one-dimensional data by orthogonal projection onto the best line that corresponds to the main principal direction associated with data-points. In this context, the proposed method and Algorithm 1 are also compared with the one-dimensional l_1-clustering algorithm (Sabo *et al.*, 2011).

Algorithm 1 gives stationary points of the function Φ_ϵ and can be used for searching for locally optimal partition of the set $\mathcal{A}\subset\mathbb{R}^n$. In order to find a good approximation of the global minimum of the function Φ_ϵ and also a good approximation of the globally optimal partition of the set $\mathcal{A}$, in accordance with Leisch (2006), Algorithm 1 should be run multiple times with various random initializations. This approach will be used in numerical examples that are given in this section.

Alternatively, it is important to have a good initial approximation. This can be achieved (see, e.g., Pintér, 1996) by using some of global optimization methods, such

as the DIRECT method for Lipschitz global optimization (Finkel and Kelley, 2006; Grbić *et al.*, 2012; Jones *et al.*, 1993). Even after a few iterations this method will give a good initial approximation, and after that Algorithm 1 can very quickly find the global minimum of the function Φ_ϵ and the globally optimal partition. Useful numerical methods for searching for a good approximation of a globally optimal partition can be found in the works of Bagirov and Ugon (2005), Bagirov *et al.* (2011) or Scitovski and Scitovski (2013).

In order to evaluate the accuracy of the proposed method and corresponding Algorithm 1, we will briefly describe several well-known indices on the basis of which it is possible to compare two different partitions of the set $\mathcal{A}$. For this purpose let us denote by $\hat{\Pi} = \{\hat{\pi}_1, \ldots, \hat{\pi}_k\}$ and $\bar{\Pi} = \{\bar{\pi}_1, \ldots, \bar{\pi}_k\}$ two partitions of the set $\mathcal{A}$ into k clusters. The confusion matrix $K = (\kappa_{ij}), i, j = 1, \ldots, k$ of the pair $(\hat{\Pi}, \bar{\Pi})$ is a $k \times k$ matrix whose ij-th entry equals the number of elements in the intersection of the clusters $\hat{\pi}_i$ and $\bar{\pi}_j$, i.e.,

$$\kappa_{ij} = |\hat{\pi}_i \cap \bar{\pi}_j|, \quad 1 \le i, j \le k.$$

The **adjusted Rand index** (Hubert and Arabie, 1985) $\mathcal{R}(\hat{\Pi}, \bar{\Pi})$ is defined as follows:

$$\mathcal{R}(\hat{\Pi}, \bar{\Pi}) = \frac{\sum_{i=1}^{k}\sum_{j=1}^{k}\binom{\kappa_{ij}}{2} - \tau_3}{\frac{1}{2}(\tau_1 + \tau_2) - \tau_3},$$

where

$$\tau_1 = \sum_{i=1}^{k}\binom{|\hat{\pi}_i|}{2}, \quad \tau_2 = \sum_{j=1}^{k}\binom{|\bar{\pi}_j|}{2},$$

$$\tau_3 = \frac{2\tau_1\tau_2}{m(m-1)}.$$

In general, $\mathcal{R}(\hat{\Pi}, \bar{\Pi}) \in [-1, 1]$ and $\mathcal{R}(\hat{\Pi}, \bar{\Pi}) = 1$ if the matching between the two partitions $\hat{\Pi}$ and $\bar{\Pi}$ is perfect.

The **Jaccard index** (Kogan, 2007) $\mathcal{J}(\hat{\Pi}, \bar{\Pi})$ is defined as follows:

$$\mathcal{J}(\hat{\Pi}, \bar{\Pi}) = \frac{\sum_{i=1}^{k}\sum_{j=1}^{k}\binom{\kappa_{ij}}{2}}{\tau_1 + \tau_2 - \sum_{i=1}^{k}\sum_{j=1}^{k}\binom{\kappa_{ij}}{2}}.$$

Similarly to the case of the adjusted Rand index, $\mathcal{J}(\hat{\Pi}, \bar{\Pi}) \in [0, 1]$ and $\mathcal{J}(\hat{\Pi}, \bar{\Pi}) = 1$ if the matching between the two partitions $\hat{\Pi}$ and $\bar{\Pi}$ is perfect.

Distance between cluster centers (Äyrämö, 2006). Let $\{\hat{C}^1, \ldots, \hat{C}^k\}$ and $\{\bar{C}^1, \ldots, \bar{C}^k\}$ be the centers of the clusters $\bar{\pi}_j$ and $\hat{\pi}_j$, with $j = 1, \ldots, k$, respectively. The distance between the sets $\hat{\mathcal{C}} = \{\hat{C}^1, \ldots, \hat{C}^k\}$ and $\bar{\mathcal{C}} = \{\bar{C}^1, \ldots, \bar{C}^k\}$ can be defined by

$$DC(\hat{\mathcal{C}}, \bar{\mathcal{C}}) = \min_{p \in \mathrm{Per}(\{1,\ldots,k\})} \sum_{j=1}^{k} \|\hat{C}^j - \bar{C}^{p(j)}\|_2^2,$$

where $\mathrm{Per}(\{1, 2, \ldots, k\})$ is the set of all permutations of the set $\{1, 2, \ldots, k\}$. Note that $DC(\hat{\mathcal{C}}, \bar{\mathcal{C}}) = 0$ if and only if $\hat{\mathcal{C}} = \bar{\mathcal{C}}$.

The **misclassification error** (Kogan, 2007) $\mathcal{E}(\hat{\Pi}, \bar{\Pi})$ is defined as follows:

$$\mathcal{E}(\hat{\Pi}, \bar{\Pi}) = \frac{m - \sum_{i}^{k} \kappa_{ir_i}}{m},$$

where

$$\kappa_{ir_i} = \max\{\kappa_{i1}, \ldots, \kappa_{ik}\}, \quad i = 1, \ldots, k.$$

The misclassification error indicates a measure of disagreement between $\hat{\Pi}$ and $\bar{\Pi}$. When the partitions coincide, $\mathcal{E}(\hat{\Pi}, \bar{\Pi})$ vanishes. Values of $\mathcal{E}(\hat{\Pi}, \bar{\Pi})$ near 1 indicates a high degree of disagreement between the partitions.

Let us mention that for $\epsilon < 0.005$, l_1 and l_2-clustering algorithms become numerically unstable. For that reason, in all of our numerical examples we take $\epsilon = 0.05$.

Example 2. Let us choose four points $C^1 = (5, 4)$, $C^2 = (4, 6)$, $C^3 = (3, 2)$, $C^4 = (6, 6) \in \mathbb{R}^2$. Similarly as in the work of Iyigun (2007), in the neighborhood of these four points, m' points are generated from normal distributions $\mathcal{N}(C^1, \sigma)$, $\mathcal{N}(C^2, \sigma)$, $\mathcal{N}(C^3, \sigma)$ and $\mathcal{N}(C^4, \sigma)$, where

$$\sigma = \begin{pmatrix} .5 & 0 \\ 0 & .5 \end{pmatrix}.$$

Twenty outliers are also added to every subset. In that way, the set $\mathcal{A} = \{a^i \in \mathbb{R}^2 \colon i = 1, \ldots, m\} = \pi_1 \cup \pi_2 \cup \pi_3 \cup \pi_4$ is defined which consists of $m = 4(m' + 20)$ points.

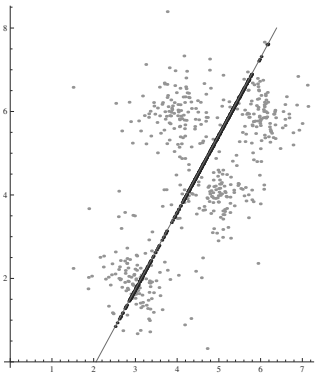


Fig. 2. 500 data points and their projections to the main principal axes.

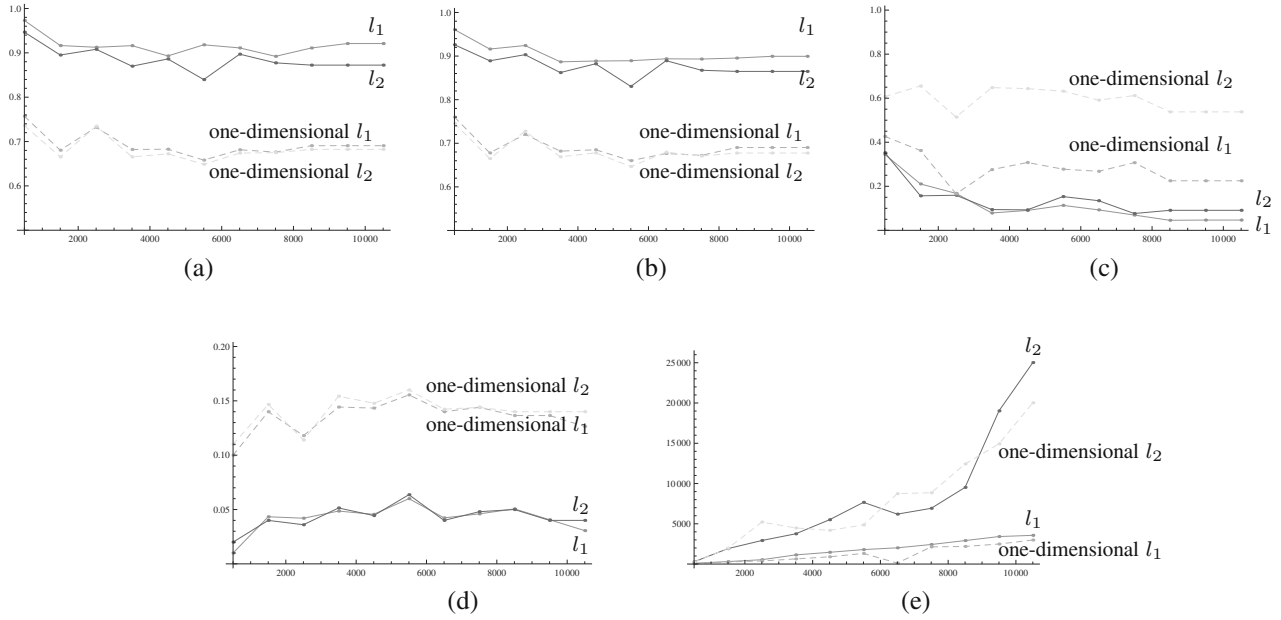


Fig. 3. Clustering algorithm comparison for a different number of data-points: adjusted rand index (a), Jaccard index (b), distance between cluster centers (c), misclassification error (d), CPU (e).

Let us write $\Pi = \{\pi_1, \pi_2, \pi_3, \pi_4\}$ and $\mathcal{C} = \{C^1, C^2, C^3, C^4\}$. For $\epsilon = 0.05$, Algorithm 1 is initiated with 100 different randomly generated initial centers. The set of centers $\mathcal{C}^\star = \{c^{1\star}, c^{2\star}, c^{3\star}, c^{4\star}\}$, i.e., the partition $\Pi^\star = \{\pi_1^\star, \pi_2^\star, \pi_3^\star, \pi_4^\star\}$ that gives the smallest value of the objective function is taken as a solution. The experiment was repeated for a different number of data points $m \in \{500, 1500, 2500, 3500, 4500, 5500, 6500, 7500, 8500, 9500, 10500\}$.

The quality of the corresponding partition is compared with the partitions obtained by (analogously with 100 various random initializations) the l_2-clustering algorithm (Kogan, 2007; Teboulle, 2007), the one-dimensional l_1-clustering algorithm (Sabo *et al.*, 2012) applied to the data obtained by orthogonal projection of the original data to the principal axes, the one-dimensional l_2-clustering algorithm applied to the data obtained by orthogonal projection of the original data to the principal axes.

Data-points for $m = 500$, the line that corresponds to the main principal axes and the projected data are shown in Fig. 2. Figure 3 (e) shows the overall CPU[1] time in seconds for the different clustering algorithms when $m \in \{500, 1500, 2500, 3500, 4500, 5500, 6500, 7500, 8500, 9500, 10500\}$. The values of the adjusted Rand index $\mathcal{R}(\Pi, \Pi^\star)$, Jaccard index $\mathcal{J}(\Pi, \Pi^\star)$, distance between clusters centers $\mathcal{D}(\mathcal{C}, \mathcal{C}^\star)$ and misclassification error $\mathcal{E}(\Pi, \Pi^\star)$ are shown in Figs. 3(a)–(d), respectively. All of these measures show that the n-dimensional l_1-clustering method is superior in comparison with the other clustering methods mentioned. Note that the one-dimensional l_1-clustering algorithm is faster, but inferior in relation to the corresponding n-dimensional algorithm. Table 1 shows the number of randomly generated initial approximations converging to the solution and illustrates that the l_1-clustering algorithm is less sensitive to the initial approximation in comparison with other methods. This means that the probability of a random choice of a good initial approximation is significantly larger in the case of the l_1-clustering algorithm.

Table 1. Number of the initial approximation converging to the solution.

Method	$m = 500$	$m = 2000$	$m = 10000$
l_1-clustering	88	82	75
l_2-clustering	8	7	3
one-dimensional l_1-clustering	45	40	39
one-dimensional l_2-clustering	3	5	1

♦

Example 3. (*Incomplete data set*) The IRIS data[2] consist of 150 four-dimensional points $\mathcal{A} = \{a^i = (a_1^i, a_2^i, a_3^i, a_4^i) \in \mathbb{R}^4 : i = 1, \ldots, 150\}$, with 50 points for each of three physically labeled classes π_1, π_2 and

[1] All calculations were done on a Pentium M processor with 1.4 GHz.

[2] UCI Machine Learning Repository, http://archive.ics.uci.edu/ml/datasets/Iris.

π_3. Let $\Pi = \{\pi_1, \pi_2, \pi_3\}$, $C^j_{\text{med}} := \text{med}(1, \pi_j)$ and $C^j_{\text{mean}} := 1/|\pi_j| \sum_{a^i \in \pi_j} a^i$, $j = 1, 2, 3$ be the medians and the means of these classes, and

$$\mathcal{C}_{\text{mean}} = \{C^1_{\text{mean}}, C^2_{\text{mean}}, C^3_{\text{mean}}\},$$

i.e.,

$$\mathcal{C}_{\text{med}} = \{C^1_{\text{med}}, C^2_{\text{med}}, C^3_{\text{med}}\}.$$

In order to examine the sensitivity of l_1 and l_2-clustering algorithms, we will consider the incomplete data set (see Hathaway and Bezdek, 2001; Simiński, 2012). Suppose that in the set $\mathcal{A}$ there are data in which there is no information about all attributes, i.e., components. In that case, the l_1-clustering algorithm will be modified in the following way:

Step 2'. For all $s = 1, \ldots, k$ define vectors w^s with components

$$w^s_i = \frac{\exp(-\frac{1}{\epsilon} \sum_{l=1}^{n} \eta^i_l |c^s_l - a^i_l|)}{\sum_{j=1}^{k} \exp(-\frac{1}{\epsilon} \sum_{l=1}^{n} \eta^i_l |c^j_l - a^i_l|)} = \frac{\exp(-\frac{1}{\epsilon} \sum\limits_{\substack{i=1 \\ \eta^i_l \in K}}^{n} \eta^i_l |c^s_l - a^i_l|)}{\sum_{j=1}^{k} \exp(-\frac{1}{\epsilon} \sum\limits_{\substack{i=1 \\ \eta^i_l \in K}}^{n} \eta^i_l |c^j_l - a^i_l|)},$$

$i = 1, \ldots, m.$

Step 3': For all $s = 1, \ldots, k$ solve the weighted median problem[3]

$$g_s(\zeta) = \sum_{i=1}^{m} w^{(s)}_i \sum_{l=1}^{n} \eta^i_l |\zeta_l - a^i| = \sum_{i=1}^{m} w^{(s)}_i \sum_{\substack{i=1 \\ \eta^i_l \in S}}^{n} \eta^i_l |\zeta_l - a^i| \to \min_{\zeta},$$

and set $\theta^{(1)} = (c^1, \ldots, c^k)$, where

$$c^s = \arg\min g_s(\zeta)$$

and

$$\eta^i_l := \eta(a^i_l) = \begin{cases} 0 & \text{if } a^i_l \text{ is missing,} \\ 1 & \text{otherwise,} \end{cases}$$

$i = 1, \ldots, m, \quad l = 1, \ldots, n,$

and $S = \{\eta^i_l \neq 0 : i = 1, \ldots, m, l = 1, \ldots, n\}$.

[3] *Mathematica*-code for solving a weighted median problem is available at `http://www.mathos.hr/seminar/Software.html`.

An analogous modification can also be done for the l_2-clustering algorithm.

Now we consider the clustering problem for the incomplete data set. For this purpose, in the data set considered we remove 10%, 20%, 30% and 40% of randomly chosen second and fourth components of the Iris data set. For such data the modification of Algorithm 1 is initiated with 100 different randomly generated initial centers, and the set of centers $\mathcal{C}^\star = \{c^{1\star}, c^{2\star}, c^{3\star}\}$, i.e., the partition $\Pi^\star = \{\pi^\star_1, \pi^\star_2, \pi^\star_3\}$ that gives the smallest value of the objective function, is taken as a solution. The result is compared with the corresponding l_2-clustering algorithm. The values of the adjusted Rand index $\mathcal{R}(\Pi, \Pi^\star)$, the Jaccard index $\mathcal{J}(\Pi, \Pi^\star)$, the distance between sets of cluster centers $\mathcal{D}(\mathcal{C}_{\text{med}}, \mathcal{C}^\star)$ (i.e., $\mathcal{D}(\mathcal{C}_{\text{mean}}, \mathcal{C}^\star)$) and the misclassification error $\mathcal{E}(\Pi, \Pi^\star)$ as the percent of removed data increased and are shown in Fig. 4. A numerical experiment described in this example illustrates that the l_1-clustering method is significantly less sensitive to the incomplete data set compared with the l_2-clustering method. ♦

Table 2. Comparison of clustering methods on the Wine data. $\mathcal{R}$: adjusted Rand index, $\mathcal{J}$: Jaccard index, $\mathcal{D}$: distance between cluster centers, $\mathcal{E}$: misclassification error, CPU: overall time in seconds necessary for the execution of the algorithm.

Method	$\mathcal{R}$	$\mathcal{J}$	$\mathcal{D}$	$\mathcal{E}$	CPU
l_1-clustering	0.88	0.88	0.17	0.04	270.59
l_2-clustering	0.89	0.91	0.63	0.03	810.07
one-dimensional l_1-clustering	0.52	0.61	0.84	0.19	30.44
one-dimensional l_2-clustering	0.50	0.58	0.85	0.20	75.00
complete link	0.58	0.60	–	0.16	0.02
single link	-0.01	0.33	–	0.60	0.02
average link	-0.01	0.34	–	0.60	0.02
Ward method	0.79	0.81	–	0.07	0.02

Example 4. (*Wine recognition data*[4]) The Wine data consists of 178 thirteen-dimensional points

$$\mathcal{A} = \{a^i \in \mathbb{R}^{13} : i = 1, \ldots, 178\}$$

with 59 points in class π_1, 71 points in class π_2 and 48 points in class π_3. Let $\Pi = \{\pi_1, \pi_2, \pi_3\}$,

$$C^j_{\text{med}} := \text{med}(1, \pi_j)$$

$$C^j_{\text{mean}} := \frac{1}{|\pi_j|} \sum_{a^i \in \pi_j} a^i, \; j = 1, 2, 3$$

be the medians and the means of these classes, and $\mathcal{C}_{\text{mean}} = \{C^1_{\text{mean}}, C^2_{\text{mean}}, C^3_{\text{mean}}\}$, i.e., $\mathcal{C}_{\text{med}} =$

[4] UCI Machine Learning Repository, `http://archive.ics.uci.edu/ml/datasets/Wine`.

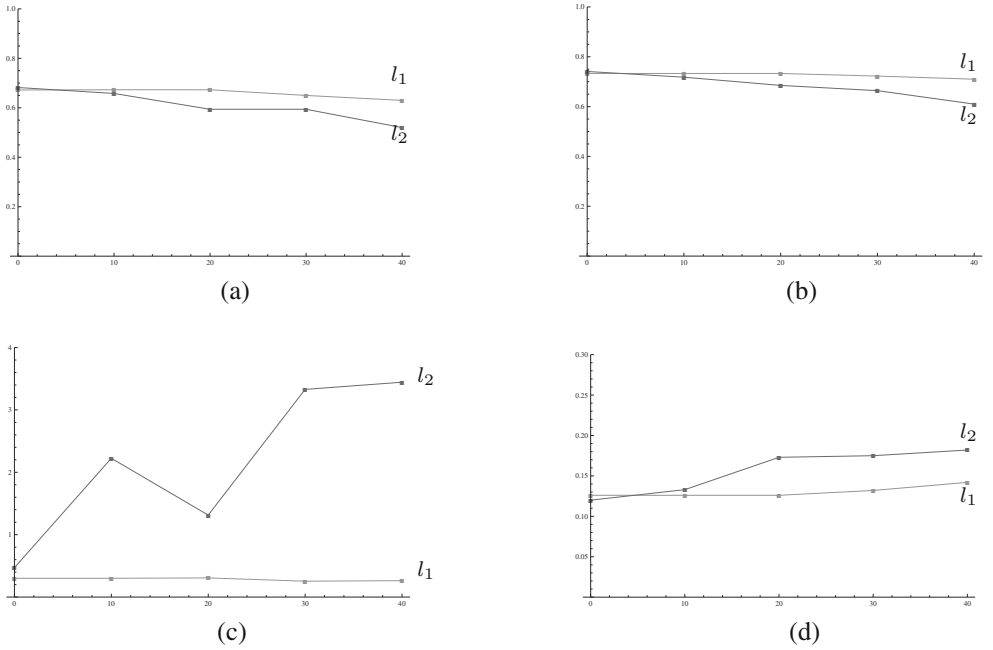


Fig. 4. Clustering algorithms comparison for a different percent of removed data: adjusted Rand index (a), Jaccard index (b), distance between cluster centers (c), misclassification error (d).

$\{C^1_{\text{med}}, C^2_{\text{med}}, C^3_{\text{med}}\}$. Algorithm 1 is initiated with 100 different randomly generated initial centers, and the set of centers $\mathcal{C}^\star = \{c^{1\star}, c^{2\star}, c^{3\star}\}$, i.e., the partition $\Pi^\star = \{\pi^\star_1, \pi^\star_2, \pi^\star_3\}$ that gives the smallest value of the objective function, is taken as a solution. The algorithm is compared with the l_2-clustering algorithm, one-dimensional l_1 and l_2-clustering algorithms, and also with several hierarchical clustering methods. The corresponding results are shown in Table 2. A significant difference between the l_1 and the l_2-algorithm with respect to the reconstruction quality is not indicated, but the l_1-clustering algorithm is faster. Note that the reconstruction quality for the Ward method is very similar to the l_1 and the l_2-algorithm. ♦

Example 5. (*Haberman survival data set*[5]) The Haberman survival data set contains cases from the study conducted on the survival of patients who had undergone breast cancer surgery. There are two classes of survival status, i.e., the patient survived 5 years or longer and the patient died within 5 years. The data set consists of 306 examples with 3 attributes. Algorithm 1 is initiated with 100 different randomly generated initial centers, and the set of centers $\mathcal{C}^\star = \{c^{1\star}, c^{2\star}\}$, i.e., the partition $\Pi^\star = \{\pi^\star_1, \pi^\star_2\}$ that gives the smallest value of the objective function, is taken as a solution. The algorithm is compared with the l_2-clustering algorithm, one-dimensional l_1 and l_2-clustering algorithms, and also with several hierarchical clustering methods. Corresponding results are shown in Table 3. Note that the quality of a reconstructed partition obtained by the l_1-clustering method is slightly better than that of other partitions, but none of these methods are able to identify clusters efficiently. ♦

[5]UCI Machine Learning Repository, `http://archive.ics.uci.edu/ml/datasets/Haberman's+Survival`

Table 3. Comparison of clustering methods on Haberman's survival data set. $\mathcal{R}$: adjusted Rand index, $\mathcal{J}$: Jaccard index, $\mathcal{D}$: distance between cluster centers, $\mathcal{E}$: misclassification error, CPU: overall time in seconds necessary for the execution of the algorithm.

Method	$\mathcal{R}$	$\mathcal{J}$	$\mathcal{D}$	$\mathcal{E}$	CPU
l_1-clustering	-0.14	0.60	11.34	0.31	130.59
l_2-clustering	-0.14	0.38	12.13	0.31	620.18
one-dimensional l_1-clustering	-0.14	0.38	11.71	0.31	30.44
one-dimensional l_2-clustering	-0.14	0.38	12.42	0.31	75.00
complete link	-0.14	0.41	–	0.31	0.02
single link	-0.36	0.60	–	0.31	0.01
average link	-0.35	0.60	–	0.31	0.03
Ward method	-0.19	0.39	–	0.31	0.02

6. Conclusions

In this paper, we considered the iterative n-dimensional data clustering algorithm based on the l_1-optimality criterion. Robustness was shown experimentally when

outliers were to be expected among the data or data-points were incomplete, i.e., they contain data in which one or more components were missing. Experiments show that in this case the proposed algorithm is faster and superior in comparison with the corresponding l_2-algorithm (Kogan, 2007; Teboulle, 2007). The proposed iterative procedure gives stationary points of the objective function and can be used only for searching for the locally optimal partition. In order to find a good approximation of the globally optimal partition, Algorithm 1 should be run multiple times with various random initializations. Numerical experiments also show that the probability of a random choice of a good initial approximation is significantly larger in the case of the l_1-clustering algorithm. The proposed center-based l_1-clustering method has three disadvantages: (i) there is no theoretical guarantee that the globally optimal partition is found, (ii) the appropriate number of clusters should be given in advance, (iii) it is not possible to identify clusters having irregular shape.

Acknowledgment

The author would like to thank the anonymous referees and Prof. Rudof Scitovski (University of Osijek, Croatia) for their careful reading of the paper and very useful comments that significantly helped improve the paper. This work was supported by the Ministry of Science, Education and Sport, Republic of Croatia, through research grants 235-2352818-1034.

References

Angulo, J. and Serra, J. (2007). Modelling and segmentation of colour images in polar representations, *Image and Vision Computing* **25**(4): 475–495.

Äyrämö, S. (2006). *Knowledge Mining Using Robust Clustering*, Ph.D. thesis, University of Jyväskylä, Jyväskylä.

Bagirov, A.M. and Ugon, J. (2005). An algorithm for minimizing clustering functions, *Optimization* **54**(4–5): 351–368.

Bagirov, A.M., Ugon, J. and Webb, D. (2011). Fast modified global k-means algorithm for incremental cluster construction, *Pattern Recognition* **44**(4): 886–876.

Bezdek, J.C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*, Kluwer Academic Publishers, Norwell, MA.

Boyd, D.L. and Vandenberghe, L. (2004). *Convex Optimization*, Cambridge University Press, Cambridge.

Chaovalitwongse, W.A., Butenko, S. and Pardalos, P.M., (Eds.) (2009). *Clustering Challenges in Biological Networks*, World Scientific, London.

Choulakian, V. (2001). Robust q-mode principal component analysis in L_1, *Computational Statistics & Data Analysis*, **37**(2): 135–150.

Clarke, F. H., (1990). *Optimization and Nonsmooth Analysis*, SIAM, Philadelphia, PA.

Cominetti, R. and Michelot, C. (1997). Sufficient conditions for coincidence in l_1-minisum multifacility location problems, *Operations Research Letters* **20**(4): 179–185.

Cord, A., Ambroise, C. and Cocquerez, J.-P. (2006). Feature selection in robust clustering based on Laplace mixture, *Pattern Recognition Letters* **27**(6): 627–635.

Cupec, R., Grbić, R., Sabo, K. and Scitovski, R. (2009). Three points method for searching the best least absolute deviations plane, *Applied Mathematics and Computation* **215**(3): 983–994.

Duda, R., Hart, P. and Stork, D. (2001). *Pattern Classification*, Wiley, New York, NY.

Finkel, D.E. and Kelley, C.T. (2006). Additive scaling and the DIRECT algorithm, *Journal of Global Optimization* **36**(4): 597–608.

Floudas, C.A. and Gounaris, C.E. (2009). A review of recent advances in global optimization, *Journal of Global Optimization* **45**(4): 3–38.

Frąckiewicz, M. and Palus, H. (2011). KHM clustering techique as a segmentation method for endoscopic colour images, *International Journal of Applied Mathematics and Computer Science* **21**(1): 203–209, DOI: 10.2478/v10006-011-0015-0.

Gan, G., Ma, C. and Wu, J. (2007). *Data Clustering: Theory, Algorithms, and Applications*, SIAM, Philadelphia, PA.

Grbić, R., Nyarko, E.K. and Scitovski, R. (2012). A modification of the direct method for Lipschitz global optimization for a symmetric function, *Journal of Global Optimization*, **57**(4): 1193–1212, DOI: 10.1007/s10898-012-0020-3.

Grbić, R., Scitovski, K., Sabo, K. and Scitovski, R. (2013). Approximating surfaces by the moving least absolute deviations method, *Applied Mathematics and Computation* **219**(9): 4387–4399.

Gurwitz, C. (1990). Weighted median algorithms for l_1 approximation, *BIT* **30**(2): 301–310.

Hathaway, R.J. and Bezdek, J.C. (2001). Fuzzy c-means clustering of incomplete data, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* **31**(5): 735–744.

Hubert, L. and Arabie, P. (1985). Comparing partitions, *Journal of Classification* **2**(1): 193–218.

Jain, A. (2010). 50 years beyond k-means, *Pattern Recognition Letters* **31**(8): 651–666.

Jajuga, K. (1987). A clustering method based on the L_1-norm, *Computational Statistics & Data Analysis* **5**(4): 357–371.

Jajuga, K. (1991). L_1-norm based fuzzy clustering, *Fuzzy Sets and Systems* **39**(1): 43–50.

Iyigun, C. (2007). *Probabilistic Distance Clustering*, Ph.D. thesis, Graduate School, Rutgers, New Brunswick, NJ.

Jones, D.R., Perttunen, C.D. and Stuckman, B.E. (1993). Lipschitzian optimization without the Lipschitz constant, *Journal of Optimization Theory and Applications* **79**(1): 157–181.

Jörnsten, R. (2004). Clustering and classification based on the L_1 data depth, *Journal of Multivariate Analysis* **90**(1): 67–89.

Kogan, J. (2007). *Introduction to Clustering Large and High-Dimensional Data*, Cambridge University Press, Cambridge.

Leisch, F. (2006). A toolbox for k-centroids cluster analysis, *Computational Statistics & Data Analysis* **51**(2): 526–544.

Li, X. Hu, W., Wang, H. and Zhang, Z. (2010). Linear discriminant analysis using rotational invariant L_1 norm, *Neurocomputing* **73**(13–15): 2571–2579.

Scitovski, R. and Scitovski, S. (2013). A fast partitioning algorithm and its application to earthquake investigation, *Computers and Geosciences* **59**(1): 124–131.

Simiński, K. (2012). Neuro-rough-fuzzy approach for regression modelling from missing data, *International Journal of Applied Mathematics and Computer Science* **22**(2): 461–476, DOI: 10.2478/v10006-012-0035-4.

Späth, H. (1976). L_1-cluster analysis, *Computing* **16**(4): 379–387.

Späth, H. (1987). Using the L_1-norm within cluster analysis, *in* Y. Dodge (Ed.), *Proceedings of the First International Conference on Statistical Data Analysis Based on the L_1-Norm and Related Methods, University of Neuchatel/Switzerland, August 31–September 04, 1987*, Elsevier, Amsterdam, pp. 427–434.

Malinen, M.I. and Fränti, P. (2012). Clustering by analytic functions, *Information Sciences* **217**(1): 31–38.

Meng, D., Zhao, Q and Xu, Z. (2012). Improve robustness of sparse PCA by L_1-norm maximization, *Pattern Recognition* **45**(1): 487–497.

Pintér, J.D. (1996). *Global Optimization in Action (Continuous and Lipschitz Optimization: Algorithms, Implementations and Applications)*, Kluwer Academic Publishers, Dordrecht.

Ruszczynski, A (2006). *Nonlinear Optimization*, Princeton University Press, Princeton/Oxford, NJ.

Sabo, K. and Scitovski, R. (2008). The best least absolute deviations line—properties and two efficient methods, *ANZIAM Journal* **50**(2): 185–198.

Sabo, K., Scitovski, R. and Vazler, I. (2011). Searching for a best LAD-solution of an overdetermined system of linear equations motivated by searching for a best LAD-hyperplane on the basis of given data, *Journal of Optimization Theory and Applications* **149**(2): 293–314.

Sabo, K., Scitovski, R. and Vazler, I. (2012). One-dimensional center-based l_1-clustering method, *Optimization Letters* **7**(1): 5–22

Sabo, K., Scitovski, R., Vazler, I. and Zekić-Sušac, M. (2011). Mathematical models of natural gas consumption, *Energy Conversion and Management* **52**(3): 1721–1727.

Teboulle, M. (2007). A unified continuous optimization framework for center-based clustering methods, *Journal of Machine Learning Research* **8**(1): 65–102.

Vardi, Y., Zhang, C. H. (2000). The multivariate L_1-median and associated data depth, *Proceedings of the National Academy of Sciences, United States of America* **97**(4): 1423–1426.

Vazler, I., Sabo, K. and Scitovski, R. (2012). Weighted median of the data in solving least absolute deviations problems, *Communications in Statistics—Theory and Methods* **41**(8): 1455–1465.

Zhang, J., Peng, L., Zhao, X. and Kuruoglu E.E. (2012). Robust data clustering by learning multi-metric l_q-norm distances, *Expert Systems with Applications* **39**(1): 335–349.

Kristian Sabo, an associate professor at the Department of Mathematics, University of Osijek, was born in 1975. He obtained his Ph.D. degree in 2007 from the Department of Mathematics, University of Zagreb, in the field of applied and numerical mathematics. His fields of interest are cluster analysis, least absolute deviations problems and applications.

Received: 5 April 2013
Revised: 27 July 2013