Fast clustering of large datasets with sequential k-medians : a stochastic gradient approach

Hervé Cardot[(a)], Peggy Cénac[(a)] and Jean-Marie Monnez[(b)]
(a) Institut de Mathématiques de Bourgogne, UMR 5584, Université de Bourgogne,
9 Avenue Alain Savary, 21078 Dijon, France
(b) Institut Elie Cartan, UMR 7502, Nancy Université, CNRS, INRIA,
B.P. 239-F 54506 Vandoeuvre lès Nancy Cedex, France

January 24, 2011

Abstract

Clustering with fast algorithms large samples of high dimensional data is an important challenge in computational statistics. Borrowing ideas from MacQueen (1967) who introduced a sequential version of the k-means algorithm, we propose in this paper a new class of recursive stochastic gradient algorithms designed for the k-medians loss criterion. By their recursive nature, these algorithms are very fast and are well adapted to deal with large samples of data that are allowed to arrive sequentially. A particular attention is paid to the averaged versions which are known to have better performances. We prove that our stochastic gradient approach converges almost surely to the set of stationary points of the underlying loss criterion. The performance of the averaged sequential estimator is compared on a simulation study, both in terms of computation speed and accuracy of the estimations, with more classical partitioning techniques such as k-means, trimmed k-means and PAM (partitioning around medoids). Finally, this new online clustering technique is illustrated on determining television audience profiles with a sample of more than 5000 individual television audiences measured over a period of 24 hours.

keyword: averaging, high dimensional data, partitioning around medoids, recursive estimator, stochastic approximation.

1 Introduction

Clustering with fast algorithms large samples of high dimensional data is an important challenge in computational statistics and machine learning, with applications in various domains such as image analysis, biology or computer vision. There is a vast literature on clustering techniques and recent discussions and reviews may be found in Jain et al. (1999), Garcià-Escudero et al. (2010) or Croux et al. (2007). Moreover, as argued in Bottou (2010), the development of fast algorithms is even more crucial when the computation time is limited and the sample is potentially very large, since fast procedures will be able to deal with larger numbers of observations and will finally provide better estimates than slower ones.

We focus here on partitioning techniques which are able to deal with large samples of data, assuming the number k of clusters is fixed in advance. The most popular clustering methods are probably the non sequential (Forgy (1965)) and the sequential (MacQueen (1967)) versions of the

k-means algorithms. They are very fast and only require $O(kn)$ operations, where n is the sample size. They aim at finding local minima of a quadratic criterion and the cluster centers are given by the barycenters of the elements belonging to each cluster. A major drawback of the k-means algorithms is that they are based on mean values and, consequently, are very sensitive to outliers. Such atypical values, which may not be uncommon in large samples, can deteriorate significantly the performances of theses algorithms, even if they only represent a small fraction of the data. The k-medians approach is a first attempt to get more robust clustering algorithms; it was suggested by MacQueen (1967) and developed by Kaufman and Rousseeuw (1990). It consists in considering criteria based on least norms instead of least squared norms, so that the cluster centers are the spatial medians, also called geometric or L_1-medians (see Small (1990)), of the elements belonging to each cluster. Many algorithms have been proposed in the literature and the most popular one is certainly PAM (partitioning around medoids). This algorithm has been proposed by Kaufman and Rousseeuw (1990) in order to search for local minima among the elements of the sample with a computation time that is $O(kn^2)$. As a consequence, it is not very well adapted for large sample sizes. Many strategies have been suggested in the literature to reduce the computation time. For example subsampling (see *e.g* the algorithm CLARA in Kaufman and Rousseeuw (1990) or in Jin and Jung (2010)), local distances computation (Zhang and Couloigner (2005)) or the use of weighted distances during the iteration steps (Park and Jun (2008)), allow to reduce the computation time.

Trimmed k-means (see Garcià-Escudero et al. (2008, 2010) and references therein) is also a popular modification of the k-means algorithm that is more robust (see Garcià-Escudero and Gordaliza (1999)). Nevertheless, from a computational point of view, it needs to sort the data and this step requires $O(n^2)$ operations, in the worst cases, at each iteration so that its execution time can get large when one has to deal with very large samples.

Borrowing ideas from MacQueen (1967) and Hartigan (1975) who have first proposed sequential clustering algorithms and Cardot et al. (2010) who have studied the properties of stochastic gradient algorithms that can give efficient estimators of the geometric median in high dimensional space, we propose in this paper a recursive strategy that is able to estimate the cluster centers by minimizing a k-medians type criterion. One of the main advantage of our approach, compared to previous ones, is that it can be computed in only $O(kn)$ operations so that it can deal with very large datasets and is more robust than the k-means. Note also that by its recursive nature, it allows automatic update and does not need to store all the data.

The paper is organized as follows. We first fix notations and present our algorithm. In the third Section, we state the almost sure consistency of the stochastic gradient k-medians to a stationary point of the underlying objective function. The proof heavily relies on Monnez (2006). In Section4, we compare on simulations the performance of our technique with the sequential k-means, the PAM algorithm and the trimmed k-means when the data are contaminated by a small fraction of outliers. We note that applying averaging techniques (see Polyak and Juditsky (1992)) to our estimator, with a small number of different initializations points, is a very competitive approach even for moderate samples sizes with computation times that are much smaller. In Section 5, we illustrate our new clustering algorithm on two large samples, of about 5000 individuals, in order to determine profiles of television audience. Proofs are gathered in Appendix.

2 The stochastic gradient k-medians algorithm

2.1 Context and definitions

Let $(\Omega, \mathcal{A}, \mathbb{P})$ be a probability space. Suppose we have n independent realizations $Z_1, \ldots, Z_n$ of a random vector Z taking values in $\mathbb{R}^d$. The aim is to partition Ω into a finite number k of clusters $\Omega_1, \ldots, \Omega_k$. Each cluster Ω_i is represented by its center, which is an element of $\mathbb{R}^d$ denoted by θ^i. From a population point of view, the k-means and k-medians algorithms aim at finding local minima of the function g mapping $\mathbb{R}^{dk}$ to $\mathbb{R}$ and defined as follows,

$$g(x) \quad \stackrel{\text{def}}{=} \quad \mathbb{E}\left(\min_{r=1,\ldots,k} \Phi(\|Z - x^r\|)\right), \tag{1}$$

where Φ is a real, positive, continuous and non decreasing function. The particular case $\Phi(u) = u^2$, leads to the classical k-means algorithm, whereas $\phi(u) = |u|$, leads to the k-medians.

Before presenting our new recursive algorithm, let us introduce now some notations and recall the recursive k-means algorithm developed by MacQueen (1967). Let us denote by I_r the indicator function,

$$I_r(z; x) = \prod_{j=1}^{k} \mathbb{1}_{\{\|z-x^r\| \leq \|z-x^j\|\}},$$

which is equal to one when x^r is the nearest point to z, among the set of points x^i, $i = 1, \ldots, k$. The k-means recursive algorithm proposed by MacQueen (1967) starts with k arbitrary groups, each containing only one point, $X_1^1, \ldots, X_1^k$. Then, at each iteration, the cluster centers are updated as follows,

$$X_{n+1}^r \quad = \quad X_n^r - a_n^r I_r(Z_n; X_n)\,(X_n^r - Z_n)\,, \tag{2}$$

where the step $a_n^r = (1+\sum_{\ell=1}^n I_r(Z_\ell; X_\ell))^{-1}$ is just the inverse of the number of elements allocated to cluster r, until iteration n. This also means that X_{n+1}^r is simply the barycenter of the elements allocated to cluster r,

$$X_{n+1}^r = \frac{1}{1+n_r}\left(X_1^r + \sum_{\ell=1}^{n} I_r(Z_\ell; X_\ell) Z_\ell\right),$$

with $n_r = \sum_{\ell=1}^n I_r(Z_\ell; X_\ell)$. The interesting point is that this recursive algorithm is very fast and can be seen as a Robbins-Monro procedure.

2.2 Stochastic gradient k-medians algorithms

Assuming Z has an absolutely continuous distribution, we have $\mathbb{P}(\left\|Z - x^i\right\| = \left\|Z - x^j\right\|) = 0$, for any $i \neq j$ and $x^i \neq x^j$. Then, the k-medians approach relies on looking for minima, that may be local, of the function g which can be also be written as follows, for any x such that $x^j \neq x^i$ when $i \neq j$,

$$g(x) = \sum_{r=1}^{k} \mathbb{E}[I_r(Z; x)\,\|Z - x^r\|]. \tag{3}$$

In order to get an explicit Robbins-Monro algorithm representation, it remains to exhibit the gradient of g. Let us write g in integral form. Denoting by f the density of the random variable Z, we have,

$$g(x) = \sum_{r=1}^{k} \int_{\mathbb{R}^d \setminus \{x^r\}} I_r(z; x)\,\|z - x^r\|\, f(z)\, dz.$$

For $j = 1, \ldots, d$, it can be checked easily that

$$\frac{\partial}{\partial x_j^r}\left(\|z - x^r\|\right) = \frac{x_j^r - z_j}{\|z - x^r\|},$$

and since

$$I_r(z;x)\frac{\left|x_j^r - z_j\right|}{\|z - x^r\|} f(z) \leq f(z), \quad \text{for } z \neq x^r,$$

the partial derivatives satisfy,

$$\frac{\partial g}{\partial x_j^r}(x) \;=\; \int_{\mathbb{R}^d \setminus \{x^r\}} I_r(z;x)\frac{x_j^r - z_j}{\|z - x^r\|} f(z)\, dz.$$

We define, for $x \in \mathbb{R}^{dk}$,

$$\nabla_r g(x) \stackrel{\text{def}}{=} \mathbb{E}\left[I_r(Z;x)\frac{x^r - Z}{\|x^r - Z\|}\right]. \tag{4}$$

We can now present our stochastic gradient k-medians algorithm. Given a set of k distinct initialization points in $\mathbb{R}^d$, $X_1^1, \cdots, X_1^k$, the set of k cluster centers is updated at each iteration as follows. For $r = 1, \ldots, k$, and $n \geq 1$,

$$\begin{aligned} X_{n+1}^r &= X_n^r - a_n^r I_r(Z_n; X_n)\frac{X_n^r - Z_n}{\|X_n^r - Z_n\|}, \\ &= X_n^r - a_n^r \nabla_r g(X_n) - a_n^r V_n^r, \end{aligned} \tag{5}$$

with $X_n = (X_n^1, \cdots, X_n^k)$, and

$$V_n^r \stackrel{\text{def}}{=} I_r(Z_n; X_n)\frac{X_n^r - Z_n}{\|X_n^r - Z_n\|} - \mathbb{E}\left[I_r(Z_n; X_n)\frac{X_n^r - Z_n}{\|X_n^r - Z_n\|}\middle| \mathcal{F}_n\right],$$

$\mathcal{F}_n = \sigma(X_1, Z_1, \ldots, Z_{n-1})$. The steps a_n^r, also called gains, are supposed to be $\mathcal{F}_n$-measurable. We denote by $\nabla g(x) = (\nabla_1 g(x), \ldots, \nabla_k g(x))'$ the gradient of g and define $V_n \stackrel{\text{def}}{=} (V_n^1, \ldots V_n^k)'$. Let A_n be the diagonal matrix of size $dk \times dk$,

$$A_n = \begin{pmatrix} a_n^1 & & & & & & \\ & \ddots & & & & & \\ & & a_n^1 & & & & \\ & & & \ddots & & & \\ & & & & a_n^k & & \\ & & & & & \ddots & \\ & & & & & & a_n^k \end{pmatrix},$$

each a_n^r being repeated d times. Then, the k-medians algorithm can be written in a matrix way,

$$X_{n+1} = X_n - A_n \nabla g(X_n) - A_n V_n, \tag{6}$$

which is a classical stochastic gradient descent.

2.3 Tuning the stochastic gradient k-medians and its averaged version

The behavior of algorithm (5) depends on the sequence of steps a_n^r, $r \in \{1, \ldots, k\}$ and the set of initialization points X_1. These two sets of tuning parameters play distinct roles and we mainly focus on the choice of the step values, noting that, as for the k-means, the estimation results must be compared for different sets of initialization points in order to get a better estimation of the cluster centers. Assume we have a sample of n realizations $Z_1, \ldots, Z_n$ of Z. Then selected estimation is the one minimizing the following empirical risk,

$$\frac{1}{n} \sum_{i=1}^{n} \sum_{r=1}^{k} I_r(Z_i; X_n) \left\| Z_i - X_n^r \right\|. \tag{7}$$

Let us denote by $n_r = \sum_{\ell=1}^{n} I_r(Z_\ell; X_\ell)$ the number of updating steps for cluster r, until iteration n, for $r \in \{1, \ldots, k\}$. A general form of a_n^r is given by

$$a_n^r = \begin{cases} a_{n-1}^r & \text{if } I_r(Z_n; X_n) = 0, \\ \dfrac{c_\gamma}{(1 + c_\alpha n_r)^\alpha} & \text{otherwise,} \end{cases} \tag{8}$$

where c_γ, c_α and $1/2 < \alpha \leq 1$ control the gain.

Adopting a theoretical point of view, one could believe that α should be set to $\alpha = 1$ with suitable adapted constants c_α and c_γ, which are unknown in practice, in order to attain the optimal parametric rates of convergence. Our experimental results on simulated data, not presented in this paper, have shown that the convergence of algorithm (5) is then very sensitive to the values of the parameters c_γ and c_α which have to be chosen very carefully. Consequently, we prefer to introduce an averaging step (see for instance Polyak and Juditsky (1992), Pelletier (2000) and Andrieu and Moulines (2006)), which does not slow down the algorithm, and provides an estimator which is much more effective. Our averaged estimator of the cluster centers is defined in a recursive way as follows, for $r \in \{1, \ldots, k\}$ and $n \geq 1$,

$$\bar{X}_{n+1}^r = \begin{cases} \bar{X}_n^r & \text{if } I_r(Z_n; X_n) = 0, \\ \dfrac{n_r \bar{X}_n^r + X_{n+1}^r}{n_r + 1} & \text{otherwise,} \end{cases} \tag{9}$$

with starting points $\bar{X}_1^r = X_1^r$, $r = 1, \ldots, k$. Then standard choices (see *e.g.* Bottou (2010) and references therein) for α and c_α are $\alpha = 3/4$ and $c_\alpha = 1$, so that one only needs to select values for c_γ.

Note that from an asymptotic point of view, it has been proved in Cardot et al. (2010) that the averaged stochastic gradient estimator of the geometric median is asymptotically efficient under classical assumptions. This means that it has the same Gaussian asymptotic distribution as the classic estimator which consists in minimizing the empirical version of (3) in the particular case $k = 1$. This has been confirmed on simulations studies and it has also been noted that the averaged algorithm is not very sensitive to the value of c_γ, provided it is not too small compared to the minimum value of the objective function.

It is also possible to consider refinements of the previous algorithm which consist in starting the averaging procedure only after a certain number of iterations of algorithm (5). Then, one has to determine when averaging starts so that it introduces another tuning parameter in the algorithm. These techniques are not used in the simulation study in order to keep our procedure as simple as possible.

3 Almost sure convergence of the algorithm

3.1 A convergence theorem

The following theorem is the main theoretical result of this paper. It states that the recursive algorithm defined in (6) converges almost surely to the set of stationary points of the objective function defined in (3), under the following assumptions.

(H1) a) The random vector Z is absolutely continuous with respect to Lebesgue measure.
b) Z is bounded: $\exists K > 0$: $\|Z\| \leq K$ a.s.
c) $\exists C$: $\forall x \in \mathbb{R}^d$ such that $\|x\| \leq K+1$, $\mathbb{E}\left[\frac{1}{\|Z-x\|}\right] < C$.

(H2) a) $\forall n \geq 1$, $\min_r a_n^r > 0$.
b) $\max_r \sup_n a_n^r < \min(\frac{1}{2}, \frac{1}{8C})$ a.s.
c) $\sum_{n=1}^{\infty} \max_r a_n^r = \infty$ a.s.
d) $\sup_n \frac{\max_r a_n^r}{\min_r a_n^r} < \infty$ a.s.

(H3) $\sum_{r=1}^{k} \sum_{n=1}^{\infty} (a_n^r)^2 < \infty$ a.s.

(H3') $\sum_{r=1}^{k} \sum_{n=1}^{\infty} \mathbb{E}\left[(a_n^r)^2 I_r(Z_n; X_n)\right] < \infty$.

Theorem 1. *Assume that X_1 is absolutely continuous and that $\|X_1^r\| \leq K$, for $r = 1, \ldots, k$. Then under Assumptions (H1a,c), (H2a,b), (H3) or (H3'), $g(X_n)$ and*

$$\sum_{r=1}^{k} \sum_{n=1}^{\infty} a_n^r \|\nabla_r g(X_n)\|^2$$

converge almost surely.
Moreover, if the hypotheses (H1b) and (H2c,d) are also fulfilled then $\nabla g(X_n)$ and the distance between X_n and the set of stationary points of g converge almost surely to zero.

A direct consequence of Theorem 1 is that if the set of stationary points of g is finite, then the sequence $(X_n)_n$ necessarily converges almost surely towards one of these stationary points because $X_{n+1} - X_n$ converges almost surely towards zero. By Cesaro means arguments, the averaged sequence $\bar{X}_n$ also converges almost surely towards the same stationary point.

3.2 Comments on the hypotheses

Note first that if the data do not arrive online and X_1 is chosen randomly among the sample units then X_1 is absolutely continuous and $\|X_1^r\| \leq K$, for $r = 1, \ldots, k$ under (H1a) and (H1b). Hypothesis (H1c) is a stronger version of a more classical hypothesis needed to get consistent estimators of the spatial median (see Chaudhuri (1996)). As noted in Cardot et al. (2010), it is closely related to small ball properties of Z and is fulfilled when

$$\mathbb{P}\left(\|Z - x\| \leq \epsilon\right) \leq c\epsilon^2,$$

for a constant c that does not depend on x and ϵ small enough. This implies in particular that hypothesis (H1c) can be satisfied only when the dimension d of the data satisfies $d \geq 2$.

Hypotheses (H2) and (H3) or (H3') deal with the stepsizes. Considering the general form of gains a_n^r given in (8), they are fulfilled when the sizes n_r of all the clusters grows to infinity at the same rate and $\alpha \in]1/2, 1]$.

4 A simulation study

We first perform a simulation study to compare our recursive k-medians algorithm with the following well known clustering algorithms : k-means (function `kmeans` in R), trimmed k-means (function `tkmeans` in the R package `tclust`, with a trimming coefficient α set to default, $\alpha = 0.05$) and PAM (function `pam` in the R package `cluster`). Our R codes are available on request.

We consider two bivariate random Gaussian vectors Z_1 (resp. Z_2) with mean vectors $\theta_1 = (2, 2)$ (resp. $\theta_2 = (-2, -2)$) and covariance matrices $\begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}$ (resp. $\begin{pmatrix} 3 & 1 \\ 1 & 2 \end{pmatrix}$).

We consider the mixture

$$Z = (1 - \epsilon)\left(\pi_1 Z_1 + \pi_2 Z_2\right) + \epsilon\delta_z$$

with $z = (-10, 10)$, $\pi_1 = 0.6$ and $\pi_2 = 0.4$. Point z is an outlier and parameter ϵ controls the level of contamination. When $\epsilon = 0$, the two components of the mixture Z are Gaussian random vectors so that any reasonable clustering process should find, when the number of clusters k is equal to 2, the cluster centers at θ_1 and θ_2. A sample of $n = 200$ realizations of Z is drawn in Figure 1.

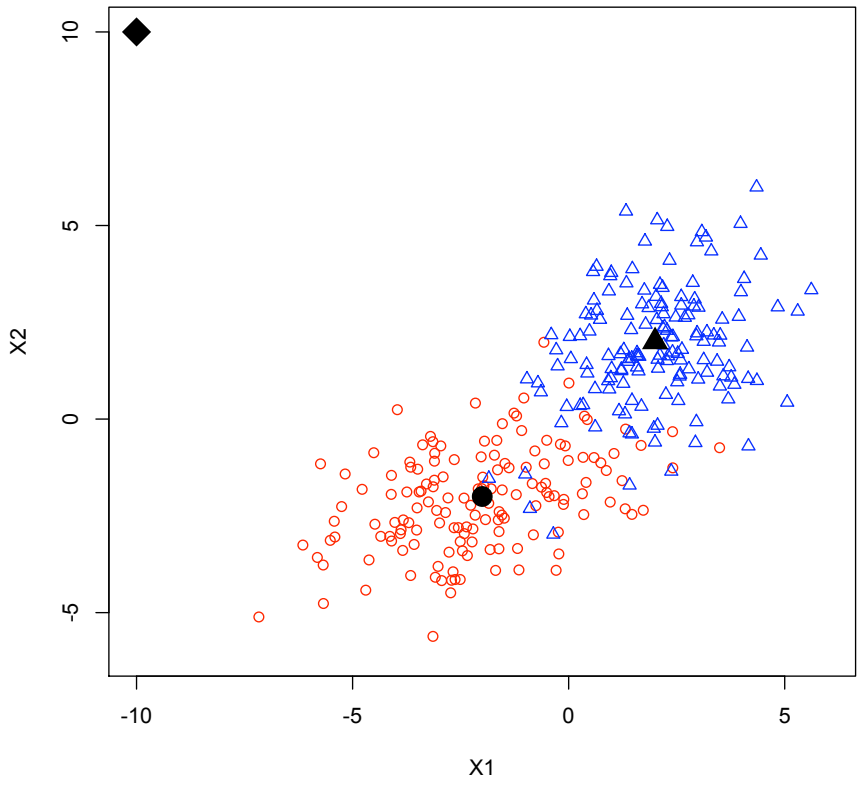

Figure 1: A sample of $n = 200$ realizations of Z. An outlier is located at position (-10,10). The centers θ_1 and θ_2 of the two natural clusters are denoted by a black triangle and a black circle.

As argued in Section 2.3, we only consider the averaged estimator, defined in (9). It depends on α, c_α and c_γ. The descent parameter c_γ plays the most important role in the convergence of the

averaged algorithm and we fixed $c_\alpha = 1$ and $\alpha = 3/4$. To evaluate the sensitivity of our recursive procedure, we consider the set $\{1, 2, 5, 7, 10\}$ of values for the parameter c_γ. For the k-means, trimmed k-means and k-medians clustering procedures, a set on 10 initial starting points X_1 is randomly chosen among the individuals of the sample. We select the estimate of the cluster centers which minimizes the corresponding empirical criterion (equation (7), for the k-medians).

4.1 Estimation of the centers of the clusters

To compare the performances of the different estimation procedures, we draw 1000 samples of sizes $n = 250$, $n = 500$ and $n = 2000$ and measure the estimation errors of the cluster centers by

$$\min\left(\sqrt{\left\|\hat{\theta}_1 - \theta_1\right\|^2 + \left\|\hat{\theta}_2 - \theta_2\right\|^2}, \sqrt{\left\|\hat{\theta}_1 - \theta_2\right\|^2 + \left\|\hat{\theta}_2 - \theta_1\right\|^2}\right),$$

where $\hat{\theta}_1$ and $\hat{\theta}_2$ are the estimated cluster centers.

The first quartile (Q1), the median value and the third quartile (Q3) of the estimation errors are presented in Table 1 when there is no contamination, *i.e* $\epsilon = 0$. For small sample sizes, the best performances are obtained for the Mac Queen algorithm and the trimmed k-means. The recursive k-medians algorithm also gives interesting estimates, provided the value of c_γ is not too small, $c_\gamma = 1$ or not too large, $c_\gamma = 10$. As the sample size increases, the performances of the stochastic gradient k-medians get better. When $n = 2000$, the recursive k-medians gives the best results for all the considered values of c_γ. This means that the averaged procedure is not very sensitive to the tuning parameter c_γ provided the sample size is large enough and, as noted in Cardot et al. (2010), the value of c_γ is at least of the same order as the minimum value of the objective function to be minimized.

Table 1: Estimation errors of the clusters centers for different sample sizes, when there is no contamination by outliers, $\epsilon = 0$.

	n=250			n=500			n=2000		
Estimator	[Q1	median	Q3]	[Q1	median	Q3]	[Q1	median	Q3]
$c_\gamma = 1$	0.30	0.41	0.58	0.22	0.30	0.41	0.14	0.19	0.26
$c_\gamma = 2$	0.28	0.39	0.51	0.21	0.29	0.40	0.14	0.19	0.26
$c_\gamma = 5$	0.28	0.38	0.53	0.21	0.29	0.41	0.14	0.20	0.26
$c_\gamma = 7$	0.29	0.42	0.57	0.22	0.31	0.43	0.14	0.20	0.27
$c_\gamma = 10$	0.35	0.51	0.75	0.26	0.37	0.53	0.15	0.21	0.29
PAM	0.34	0.44	0.57	0.25	0.33	0.44	0.16	0.21	0.28
Trimmed k-means	0.26	0.36	0.50	0.20	0.28	0.42	0.16	0.23	0.30
MacQueen	0.26	0.35	0.47	0.22	0.30	0.37	0.18	0.22	0.27

We can note in Table 2 that the performances of the MacQueen algorithm are affected by the presence, even in a very small proportion ($\epsilon = 0.02$), of outliers. The trimmed k-means is the most effective approach when the sample size is not too large, *i.e.* $n = 250$ or $n = 500$. When the sample size is larger, $n = 2000$, the best estimations are obtained with the trimmed k-means and the recursive k-medians algorithms, for values of c_γ ranging from 1 to 10. Once again, as the sample size increases, the stochastic k-medians becomes less sensitive to the choice of c_γ.

Table 2: Comparison of the estimation errors of the centers of the clusters for different sample sizes, when $\epsilon = 0.02$.

	n=250			n=500			n=2000		
Estimator	[Q1	median	Q3]	[Q1	median	Q3]	[Q1	median	Q3]
$c_\gamma = 1$	0.31	0.43	0.58	0.23	0.32	0.42	0.13	0.20	0.26
$c_\gamma = 2$	0.29	0.40	0.54	0.21	0.30	0.41	0.14	0.19	0.26
$c_\gamma = 5$	0.29	0.40	0.57	0.21	0.30	0.42	0.14	0.20	0.26
$c_\gamma = 7$	0.31	0.44	0.61	0.23	0.32	0.45	0.14	0.20	0.27
$c_\gamma = 10$	0.41	0.58	0.85	0.27	0.40	0.57	0.15	0.22	0.30
PAM	0.34	0.45	0.59	0.24	0.33	0.43	0.15	0.21	0.26
Trimmed k-means	0.26	0.37	0.51	0.19	0.26	0.39	0.14	0.20	0.27
MacQueen	0.47	0.61	0.77	0.43	0.52	0.64	0.40	0.45	0.51

4.2 Computation time

The R codes of all the considered estimation procedures call C routines and provide the same output. Mean computation times, for 100 runs, various sample sizes and numbers of clusters are reported in Table 3. From a computational point of view, the recursive k-means based on the MacQueen algorithm as well as the averaged stochastic k-medians algorithm are always faster than the others and the gain increases as the sample size gets larger. For example, when $k = 5$ and $n = 2000$ our procedure is approximately 30 times faster than the trimmed k-means and 350 times faster than the PAM algorithm. This also means that when the allocated computation is fixed and the dataset is very large, the recursive k-medians algorithm can deal with sample sizes that are 30 times larger than the trimmed k-means and 350 times larger than the PAM algorithm.

Table 3: Comparison of the mean computation time in seconds, for 100 runs, of the different estimators for various sample sizes and number of clusters k.

	n=250			n=500			n=2000		
Estimator	k=2	k=4	k=5	k=2	k=4	k=5	k=2	k=4	k=5
k-medians	0.33	0.35	0.36	0.45	0.47	0.48	1.14	1.25	1.68
PAM	1.38	3.34	4.21	5.46	15.12	20.90	111	302.00	596.00
Trimmed k-means	2.20	5.45	6.76	5.32	11.19	13.48	22.97	42.72	51.00
MacQueen	0.21	0.29	0.31	0.25	0.43	0.50	0.60	1.38	1.76

When the sample size and the dimension increase the computation time is even more critical. For instance, when $d = 1440$ and $n = 5422$ as in the example of Section 5.2, sequential estimation procedures are at least 1000 times faster than the trimmed k-means. It takes about 3.0 seconds for our averaged k-medians to converge, 5.5 seconds for the sequential k-means and more than 5700 seconds for the trimmed k-means.

5 Determining television audience profiles with k-medians

The Médiamétrie company provides every day the official estimations of television audience in France. Television consumption can be measured both in terms of how long do people watch each channel and when do they watch television. Médiamétrie has a panel of about 9000 individuals

equipped at home with sensors that are able to record and send the audience of the different television channels. Among this panel, a sample of around 7000 people is drawn every day and the television consumption of the people belonging to this sample is sent to Médiamétrie at night, between 3 and 5 AM. Online clustering techniques are then interesting to determine automatically, the number of clusters being fixed in advance, the main profiles of viewers and then relate these profiles to socio-economic variables. In these samples, Médiamétrie has noted the presence of some atypical behaviors so that robust techniques may be helpful.

5.1 Clustering aggregated audiences

We first consider aggregated data and deal with the cumulated audiences of the different channels. We focus our study on two days of september 2010, a sunday and a tuesday. We have initial samples with size $n_S = 7337$ for the sunday and with size $n_T = 7173$ for the tuesday and we observe, for each individual i in the sample, a vector $Z_i \in \mathbb{R}^7$ whose components correspond to the cumulated audience, measured in hours, for each television channel (numbered from 1 to 7 for confidentiality reasons). Note that about 28 % of the individuals in the two samples did not watch television at all during these two days. They form a particular cluster and have not been taken into account in the clustering procedure, so that we consider, in the following, subsamples of the initial samples which have sizes $n_S = 5202$ and $n_T = 5188$.

We perform our recursive averaged clustering procedure and consider $k = 6$ clusters. The centers of the clusters are presented in Figure 2 and are sorted according to the total cumulated audience. They have been estimated with a tuning parameter $c_\gamma = 1$ and 100 different initial starting points for the algorithm. We have noted that the results do not vary much when c_γ take values around 1, which lead to the smallest values of the empirical version of (3).

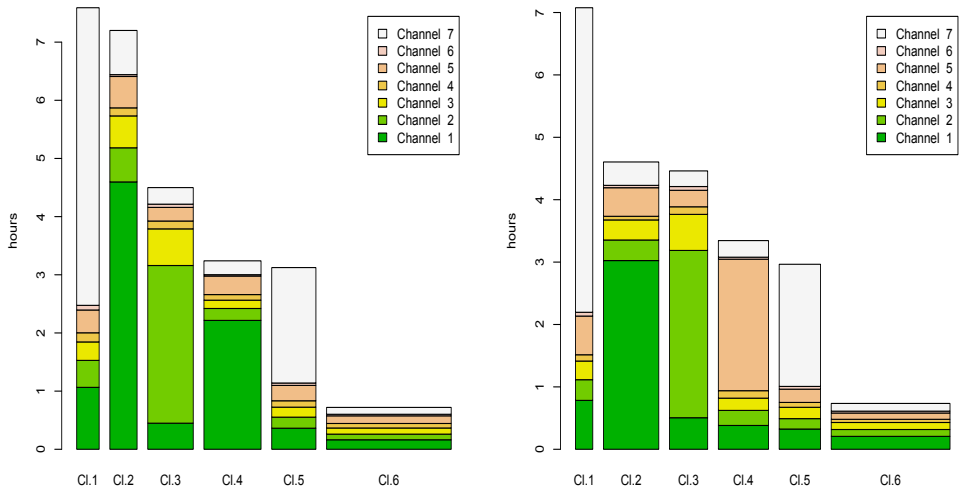

Figure 2: Cumulated audience for the different clusters denoted by Cl.1 - Cl.6. Bar widths are proportional to cluster sizes. For confidentiality reasons, the television channels have been renamed. On the left, the cluster spatial medians during the sunday. On the right, the cluster spatial medians during the tuesday.

When analyzing and comparing the cluster centers for the sunday and the tuesday, the first surprising fact is the similarity between some clusters of the two different days whereas one could expect very different profiles. For example, Cluster 1, which represents people watching television a long period of time during the day, has approximately the same distribution along the different television channels for the sunday and the tuesday. It is a profile of people that have high levels of television consumption and spend around 5 hours in front of the channel 7. This profile represents about 7.0% of the sample for the sunday and 5.1% for the tuesday. Cluster 6, which represents about 38% of the sample, for both days, is also nearly unchanged and corresponds to people that do not spend much time watching television. The third cluster centers, in terms of sample sizes, are also very stable for both days. People belonging to these clusters, which represent about 14.2% of the sample for the sunday and 11.9% of the sample for the tuesday, correspond to a profile of people that mainly watch channel 2 and a rather long period of time during these two days. Cluster 5 is also unchanged, it represents about 13% of the sample for both days and is very similar to cluster 1, up to a scale factor. The main difference between the two studied days essentially comes from cluster 4. For the tuesday, it corresponds to a group of people mainly watching channel 5 and there is no equivalent group during the sunday.

5.2 Clustering temporal profiles

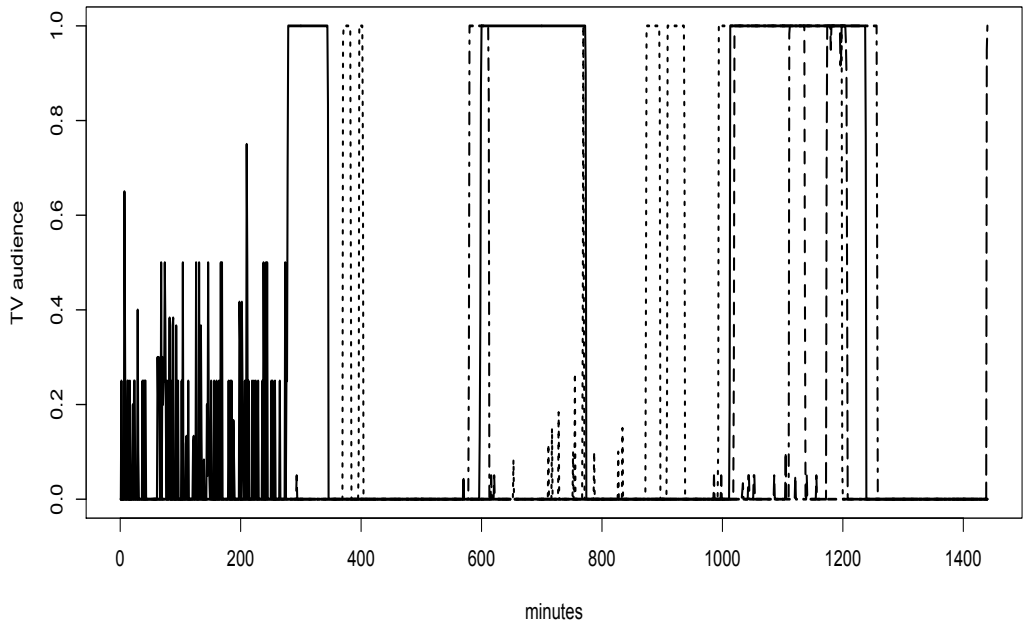

Figure 3: A sample of 5 observations of individual audience profiles measured every minute over a period of 24 hours.

Another interesting question is to build profiles of the evolution along time of the total audience. We have now a sample of $n = 5422$ individual audiences, aggregated along all television channels, measured every minute over a period of 24 hours during the 6th september 2010. An observation

Z_i is a vector in $[0,1]^d$, with $d = 1440$, each component giving the fraction of time spent watching television during the corresponding minute of the day. A sample of 5 individual temporal profiles is drawn in Figure 3.

The cluster centers, estimated with our averaged algorithm for $k = 5$, with a parameter $c_\gamma = 0.2$ and 100 different starting points, are drawn in Figure 4. They have been ordered in decreasing order according to their sizes and labelled Cl.1 to Cl.5. Cluster 1 (Cl.1) is thus the largest cluster and it contains about 35% of the sample. It corresponds to individuals that do not watch television much during the day, with a cumulated audience of about 42 minutes. At the opposite, cluster 5, which represents about 12% of the sample, is associated to high audience rates during nearly all the day with a cumulated audience of about 592 minutes. Clusters 2, 3 and 4 correspond to intermediate consumption levels and can be distinguished according the audience during the evening and at night. For example cluster 4, which represents 16% of the sample, is related to people watching television late at night, with a cumulated audience of about 310 minutes.

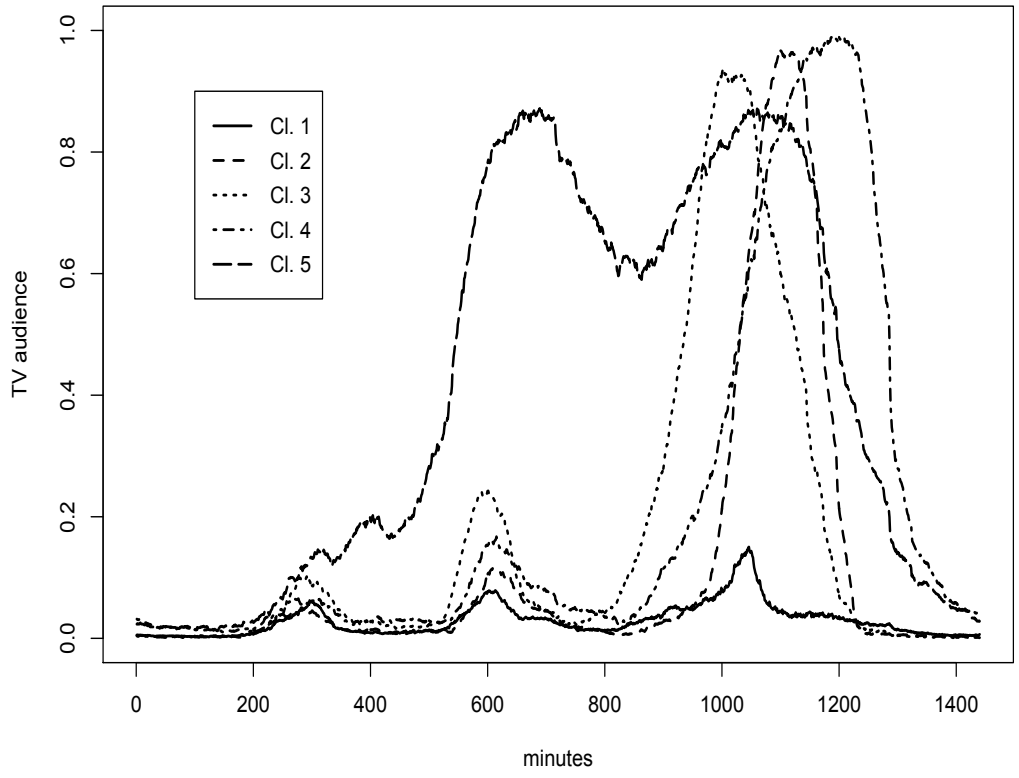

Figure 4: Cluster centers for temporal television audience profiles measured every minute over a period of 24 hours.

Appendix : Proof of Theorem 1

The proof of Theorem 1 relies on the following light version of the main theorem in Monnez (2006), section 2.1.

Theorem 2 (Monnez (2006))**.** *Assuming*

$(A1a)$ *g is a non negative function;*

$(A1b)$ *There exists a constant $L > 0$ such that, for all $n \geq 1$,*

$$g(X_{n+1}) - g(X_n) \leq \langle X_{n+1} - X_n, \nabla g(X_n) \rangle + L \left\| X_{n+1} - X_n \right\|^2 \quad a.s.;$$

$(A1c)$ *The sequence (X_n) is almost surely bounded and ∇g is continuous almost everywhere on the compact set containing (X_n);*

$(A2)$ *There exists four sequences of random variables (B_n), (C_n),(D_n) and (E_n) in $\mathbb{R}^+$ adapted to the sequence $(\mathcal{F}_n)$ such that a.s.:*

$(A2a)$ $\left\| \sqrt{A_n} \mathbb{E}[V_n | \mathcal{F}_n] \right\|^2 \leq B_n g(X_n) + C_n$ *and* $\sum_{n=1}^{\infty} (B_n + C_n) < \infty$*;*

$(A2b)$ $\mathbb{E}[\|A_n V_n\|^2 | \mathcal{F}_n] \leq D_n g(X_n) + E_n$ *and* $\sum_{n=1}^{\infty} (D_n + E_n) < \infty$*;*

$(A4)$ $\sup_n a_n^r < \min(\frac{1}{2}, \frac{1}{4L})$ *a.s.,* $\sum_{n=1}^{\infty} \max_r a_n^r = \infty$ *a.s. and*

$$\sup_n \frac{\max_r a_n^r}{\min_r a_n^r} < \infty \quad a.s.$$

then the distance of X_n to the set of stationary points of g converges almost surely to zero.

Proof of Theorem 1.

Step 1: proof of $(A1b)$

Let $A = X_n$ and $B = X_{n+1}$. Since X_n is absolutely continuous with respect to Lebesgue measure, $\sum_{r=1}^{k} I_r(Z; A) = 1$ a.s. and one gets

$$g(B) = \mathbb{E}\left[\min_r \|Z - B^r\|\right] = \mathbb{E}\left[\sum_{r=1}^{k} I_r(Z; A) \min_j \left\|Z - B^j\right\|\right],$$

and it comes

$$g(B) \leq \sum_{r=1}^{k} \mathbb{E}\left[I_r(Z; A) \|Z - B^r\|\right],$$

which yields

$$g(B) - g(A) \leq \sum_{r=1}^{k} \mathbb{E}\left[I_r(Z; A) \left(\|Z - B^r\| - \|Z - A^r\|\right)\right].$$

The application $x \mapsto \|z - x^r\|$ is a continuous function whose gradient

$$\nabla_r \|z - x^r\| = \frac{x^r - z}{\|x^r - z\|}$$

is also continuous for $x^r \neq z$. Then almost surely for $d \geq 2$, there exists $C^r = A^r + \mu^r(B^r - A^r)$, $0 \leq \mu^r \leq 1$, such that

$$\|Z - B^r\| - \|Z - A^r\| = \langle B^r - A^r, \nabla_r \|Z - C^r\| \rangle.$$

Consequently for all $d \geq 2$,

$$g(B) - g(A) \leq \sum_{r=1}^{k} \mathbb{E}\left[I_r(Z; A) \langle B^r - A^r, \nabla_r \|Z - C^r\| \rangle \right],$$

so that

$$\begin{aligned} g(B) - g(A) &\leq \sum_{r=1}^{k} \mathbb{E}\left[I_r(Z; A) \langle B^r - A^r, \nabla_r \|Z - C^r\| - \nabla_r \|Z - A^r\| \rangle \right] \\ &+ \sum_{r=1}^{k} \mathbb{E}\left[I_r(Z; A) \langle B^r - A^r, \nabla_r \|Z - A^r\| \rangle \right] \stackrel{\text{def}}{=} (1) + (2) \end{aligned}$$

On the one hand

$$(2) = \sum_{r=1}^{k} \langle B^r - A^r, \nabla_r g(A) \rangle = \langle B - A, \nabla g(A) \rangle,$$

and on the other hand

$$(1) \leq \sum_{r=1}^{k} \|B^r - A^r\| \, \mathbb{E}\left[\left| \nabla_r \|Z - C^r\| - \nabla_r \|Z - A^r\| \right| \right],$$

hence since

$$\left| \nabla_r \|Z - C^r\| - \nabla_r \|Z - A^r\| \right| = \left\| \frac{C^r - Z}{\|C^r - Z\|} - \frac{A^r - Z}{\|A^r - Z\|} \right\| \leq 2 \frac{\|C^r - A^r\|}{\|A^r - Z\|},$$

it comes, with (H1c)

$$(1) \leq 2 \sum_{r=1}^{k} \|B^r - A^r\| \, \|C^r - A^r\| \, \mathbb{E}\left[\frac{1}{\|Z - A^r\|} \right] \leq 2C \sum_{r=1}^{k} \|B^r - A^r\|^2 = 2C \|B - A\|^2.$$

Consequently, one gets

$$g(B) - g(A) \leq \langle B - A, \nabla g(A) \rangle + 2C \|B - A\|^2.$$

Step 2: Proof of the assertion: $\forall n \geq 1$**, for all** $r = 1, ...k$**,** $\|X_n^r\| \leq K + 2 \sup_n a_n^r$

Let us prove by induction on n that for all $n \in \mathbb{N}^*$, for all $r = 1, \ldots, k$, $\|X_n^r\| \leq K + 2 \sup_n a_n^r$. This inequality is trivial for the case $n = 1$: $\|X_1^r\| \leq K$. Let $n \in \mathbb{N}^*$ such that $\|X_n^r\| \leq K + 2 \sup_n a_n^r$, $\forall r \in \{1, \ldots, k\}$. Let $r \in \{1, \ldots, k\}$. First we assume that $\|X_n^r\| \leq K + a_n^r$. Then it comes

$$\left\| X_{n+1}^r \right\| \leq \|X_n^r\| + a_n^r I_r(Z_n; X_n) \leq \|X_n^r\| + a_n^r \leq K + 2a_n^r.$$

Now in the case when $K + a_n^r < \|X_n^r\| \leq K + 2 \sup_n a_n^r$, one gets

$$\|X_n^r\| > K + a_n^r \geq \|Z_n\| + a_n^r,$$

and then

$$\|X_n^r - Z_n\| \geq |\|X_n^r\| - \|Z_n\|| > a_n^r.$$

Since for $I_r(Z_n; X_n) = 0$, $X_{n+1}^r = X_n^r$, it remains to deal with the unique index r such that $I_r(Z_n; X_n) = 1$. In that case,

$$X_{n+1}^r = X_n^r - a_n^r \frac{X_n^r - Z_n}{\|X_n^r - Z_n\|} = \left(1 - \frac{a_n^r}{\|X_n^r - Z_n\|}\right) X_n^r + a_n^r \frac{Z_n}{\|X_n^r - Z_n\|}.$$

By (H1b) and from the inequalities $a_n^r / \|X_n^r - Z_n\| < 1$ and $\|Z_n\| \leq K < \|X_n^r\|$, it comes,

$$\left\|X_{n+1}^r\right\| < \left(1 - \frac{a_n^r}{\|X_n^r - Z_n\|}\right) \|X_n^r\| + a_n^r \frac{\|X_n^r\|}{\|X_n^r - Z_n\|} = \|X_n^r\|,$$

which leads to $\left\|X_{n+1}^r\right\| \leq K + 2 \sup_n a_n^r$ and concludes the proof by induction.

Step 3: Proof of $(A1c)$

From the integral form

$$\frac{\partial g}{\partial x_j^r}(x) = \int_{\mathbb{R}^d \setminus \{x^r\}} I_r(z; x) \frac{x_j^r - z_j}{\|z - x^r\|} f(z) dz,$$

it is easy to see that $\frac{\partial g}{\partial x_j^r}$ is a continuous function of x.

Step 4: Proof of $(A2a)$

The definition of V_n^r implies that $\mathbb{E}[V_n^r | \mathcal{F}_n] = 0$ and hence it comes $\mathbb{E}[V_n | \mathcal{F}_n] = 0$.

Step 5: Proof of $(A2b)$

$$\begin{aligned}
\mathbb{E}\left[\|A_n V_n\|^2 | \mathcal{F}_n\right] &= \sum_{r=1}^{k} \mathbb{E}\left[(a_n^r)^2 \|V_n^r\|^2 | \mathcal{F}_n\right] \\
&\leq \sum_{r=1}^{k} (a_n^r)^2 \, \mathbb{E}\left[I_r(Z_n; X_n) \frac{\|X_n^r - Z_n\|^2}{\|X_n^r - Z_n\|^2} \Big| \mathcal{F}_n\right] \\
&\leq \sum_{r=1}^{k} (a_n^r)^2.
\end{aligned}$$

Hence assuming (H3), it comes

$$\mathbb{E}\left[\sum_{n=1}^{\infty} \mathbb{E}\left[\|A_n V_n\|^2 | \mathcal{F}_n\right]\right] < \infty.$$

In the case when (H3') holds instead of (H3), one has

$$\mathbb{E}\left[\sum_{n=1}^{\infty} \mathbb{E}\left[\|A_n V_n\|^2 | \mathcal{F}_n\right]\right] \leq \sum_{n=1}^{\infty} \sum_{r=1}^{k} \mathbb{E}\left[(a_n^r)^2 I_r(Z_n; X_n)\right] < \infty.$$

Consequently,

$$\sum_{n=1}^{\infty} \mathbb{E}\left[\|A_n V_n\|^2 \,|\mathcal{F}_n\right] < \infty \quad \text{a.s,}$$

which concludes the proof. □

Acknowledgements. We thank the Médiamétrie company for allowing us to illustrate our sequential clustering techniques with their data.

References

Andrieu, C., Moulines, É., 2006. On the ergodicity properties of some adaptive MCMC algorithms. Ann. Appl. Probab. 16 (3), 1462–1505.

Bottou, L., 2010. Large-scale machine learning with stochastic gradient descent. In: Lechevallier, Y., Saporta, G. (Eds.), Compstat 2010. Physica Verlag, Springer., pp. 177–186.

Cardot, H., Cénac, P., Zitt, P.-A., 2010. Efficient and fast estimation of the geometric median in Hilbert spaces with an averaged stochastic gradient approach. Tech. rep., Institut de Mathématiques de Bourgogne.

Chaudhuri, P., 1996. On a geometric notion of quantiles for multivariate data. J. Amer. Statist. Assoc. 91 (434), 862–872.

Croux, C., Gallopoulos, E., Van Aelst, S., Zha, H., 2007. Machine learning and robust data mining. Computational Statistics and Data Analysis 52, 151–154.

Forgy, E., 1965. Cluster analysis of multivariate data: efficiency vs. interpretability of classifications. Biometrics 21, 768–769.

Garcià-Escudero, L., Godaliza, A., 1999. Robustness properties of k-means and trimmed k-means. Journal of the American Statistical Association 94, 956–969.

Garcià-Escudero, L., Godaliza, A., Matràn, C., Mayo-Iscar, A., 2008. A general trimming approach to cluster analysis. Annals of Statistics 36, 1324–1345.

Garcià-Escudero, L., Godaliza, A., Matràn, C., Mayo-Iscar, A., 2010. A review of robust clustering methods. Adv. Data Anal. Classif. 4, 89–109.

Hartigan, J., 1975. Clustering algorithms. John Wiley & Sons, New York.

Jain, A., Marty, M., Flynn, P., 1999. Data clustering: a review. ACM Computing surveys 31, 264–323.

Jin, S., Jung, B.-C., 2010. Sample based algorithm for k-spatial medians clustering. Korean Journal of Applied Statistics 23, 367–374.

Kaufman, L., Rousseeuw, P., 1990. Finding groups in data: an introduction to cluster analysis. John Wiley & Sons, New York.

MacQueen, J., 1967. Some methods for classification and analysis of multivariate observations. In: Proc. Fifth Berkeley Sympos. Math. Statist. and Probability (Berkeley, Calif., 1965/66). Univ. California Press, Berkeley, Calif., pp. Vol. I: Statistics, pp. 281–297.

Monnez, J.-M., 2006. Almost sure convergence of stochastic gradient processes with matrix step sizes. Statist. Probab. Lett. 76 (5), 531–536.

Park, H.-S., Jun, C.-H., 2008. A simple and fast algorithm for k-medoids clustering. Expert Systems with Applications 36, 3336–3341.

Pelletier, M., 2000. Asymptotic almost sure efficiency of averaged stochastic algorithms. SIAM J. Control Optim. 39 (1), 49–72 (electronic).

Polyak, B., Juditsky, A., 1992. Acceleration of stochastic approximation. SIAM J. Control and Optimization 30, 838–855.

Small, C. G., 1990. A survey of multidimensional medians. International Statistical Review / Revue Internationale de Statistique 58 (3), 263–277.

Zhang, Q., Couloigner, A., 2005. A new and efficient k-medoid algorithm for spatial clustering. Lecture Notes in Computer Science 3482, 181–189.