Published in final edited form as:
J Comput Graph Stat. 2016 ; 25(3): 899–916. doi:10.1080/10618600.2015.1051625.

Fast Nonparametric Density-Based Clustering of Large Data Sets Using a Stochastic Approximation Mean-Shift Algorithm

Ollivier Hyrien[1] and **Andrea Baran**[1]

[1]Department of Biostatistics and Computational Biology, University of Rochester, 14642, Rochester, New York, U.S.A.

Abstract

Mean-shift is an iterative procedure often used as a nonparametric clustering algorithm that defines clusters based on the modal regions of a density function. The algorithm is conceptually appealing and makes assumptions neither about the shape of the clusters nor about their number. However, with a complexity of $O(n^2)$ per iteration, it does not scale well to large data sets. We propose a novel algorithm which performs density-based clustering much quicker than mean-shift, yet delivering virtually identical results. This algorithm combines subsampling and a stochastic approximation procedure to achieve a potential complexity of $O(n)$ at each step. Its convergence is established. Its performances are evaluated using simulations and applications to image segmentation, where the algorithm was tens or hundreds of times faster than mean-shift, yet causing negligible amounts of clustering errors. The algorithm can be combined with existing approaches to further accelerate clustering.

Keywords

Image segmentation; Large data sets; Robbins-Monro procedure

1. INTRODUCTION

Clustering is the unsupervised process of grouping observations of a same data set that share similar features. A myriad of clustering algorithms have been designed and tested for a variety of settings and applications. The most popular of them, *k*-means, belongs to a class of algorithms which perform clustering by minimizing the total amount of within-group dissimilarities defined as the sum of within-cluster sum of squares. It produces clusters that tend to be convexly shaped.

Density-based clustering is an alternative approach that does not have this limitation (Hartigan 1975, Everitt 1993, Gan et al. 2007, Klemelä 2009). It traces back to Wishart (1969)'s work on mode analysis and defines clusters as highly dense regions separated from each other by regions where the density is lower. The underlying principle is appealing and gives rise to algorithms with several attractive features: clusters may be arbitrarily-shaped (e.g., non-Gaussian or non-convex); the number of clusters is automatically determined by

Correspondence to: Ollivier Hyrien.

the number of modes of the density function; and the presence of outliers has limited influence over clustering results. By comparison, *k*-means does not enjoy any of these properties. Density-based clustering may be implemented using various techniques that lead to distinct algorithms. Broadly speaking, these algorithms can be classified into two categories: those that are *optimization-free* and those that are *optimization-based*.

Many *optimization-free* density-based clustering algorithms define clusters by measuring the connectivity between data points via a dissimilarity measure (see Kriegel et al. (2011) for a recent discussion). For example, single-linkage is an agglomerative hierarchical clustering algorithm which sequentially merges the observations that are the closest to each other (Wishart 1969, Hartigan 1975). Other algorithms proceed using the contours of a density estimate. The method of level set trees (Klemelä 2009) and the runt pruning algorithm (Stuetzle 2003) rest on this principle. DBSCAN is another popular algorithm which merges observations that are sufficiently connected if they belong to dense enough regions (Ester et al. 1996). This algorithm may not perform well when the density varies widely across clusters and when data are high-dimensional (Kriegel et al. 2011).

What we refer to as *optimization-based* algorithms are recursive procedures that construct clusters by identifying the modal regions of a density function via numerical optimization instead of the connectivity between observations. Mean-shift is a popular algorithm implementing this principle via kernel density estimation (Fukunaga and Hostetler 1975, Cheng 1995, Comaniciu and Meer 2002, Fashing and Tomasi 2005, Carreira-Perpiñan 2007). Li et al. (2007)'s algorithm constructed using finite mixture models shares a number of features with mean-shift. We note that the latter approach differs from model-based clustering where the data are described by a finite mixture, often of Gaussian distributions, and where clusters are determined by the components of the mixture instead of its modes (Hartigan 1975, Fraley and Raftery 2002).

The main limitation of mean-shift is its computational cost. With a sample of size n, the complexity of each iteration is $O(n^2)$, a major obstacle to the clustering of large data sets. By comparison, *k*-means has a complexity of $O(n)$. Several strategies can improve the computational efficiency of mean-shift, including clustering a subsample of the data set first and assigning the remaining observations to clusters via discriminant analysis or nearest-neighbor (Fraley and Raftery 2002). Observations used to compute the mean-shift may be moved toward the mode at each iteration, a technique known as blurring, which sharpens the modes of the density and accelerates convergence of the Gaussian mean-shift by reducing the number of iterations (Carreira-Perpiñan 2006a). Density estimates may be computed using random subsamples (Georgescu et al. 2003, Li et al. 2007). However, only a fraction of the information is used for clustering, making the modal regions of the density estimate variable, causing clustering errors.

In this paper, we propose a novel optimization-based algorithm, referred to as stochastic approximation mean-shift (SAMS), designed to quickly perform density-based clustering of large data sets. The proposed algorithm also uses subsampling to reduce the computing time, but it does not have the limitation of the standard mean-shift applied to a subsample of the data. In particular, by design, SAMS exhaustively uses the information contained in the data set as the algorithm progresses and it seeks to identify the same clusters as the standard mean-shift, accomplished by means of Robbins and Monro (1951)'s stochastic approximation procedure.

The proposed algorithm achieves a complexity of $O(n)$ per iteration if the number of observations used to approximate the mean-field of the stochastic approximation procedure is kept constant as n increases, making it a more competitive alternative to k-means. The algorithm includes the standard mean-shift as a particular case. Simulation studies show that SAMS behaves similarly to the standard mean-shift, even with small subsamples (e.g., < 1% of the original data set), but it is more computationally efficient (e.g., it may be 100 times faster), without inducing a substantial amount of clustering errors (e.g., < 1%). Using a general result from Delyon et al. (1999) on Robins-Monro stochastic approximation procedures, we identify a minimal set of assumptions that ensure convergence of SAMS. Although mean-shift is a generalized expectation-maximization (EM) algorithm (Carreira-Perpiñan 2007), the construction of SAMS differs from that of the stochastic approximation EM proposed by Delyon et al. (2009), in which a stochastic algorithm is used in the E-step to approximate the conditional expectation of the complete log-likelihood. SAMS uses a stochastic algorithm to approximate the M-step.

The remainder of this paper is organized as follows. The stochastic approximation mean-shift is presented in section 2. Details relevant to its implementation, including choice of tuning parameters and novel stopping rules, are also discussed in this section. Evaluation of SAMS and comparison with the regular mean-shift and existing acceleration approaches using simulated data and applications to image segmentation are presented in section 3.

2. DENSITY-BASED CLUSTERING

2.1 Preliminaries

Let $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_n)$ where $\mathbf{y}_i = (y_{i1}, \ldots, y_{ip})'$, $i = 1, \ldots, n$, denote a sample of n p-dimensional random vectors (r.v.). These vectors are assumed identically distributed with probability density function (p.d.f.) $f(\cdot)$. Let the kernel $K:R^+ \mapsto R^+$ be a non-increasing function that satisfies $K(\cdot) \geq 0$, $\int K\left(\|\mathbf{x}\|^2\right) d\mathbf{x}=1$, $\int \mathbf{x}K\left(\|\mathbf{x}\|^2\right) d\mathbf{x}=\mathbf{0}$, and $\int \mathbf{x}\mathbf{x}'K\left(\|\mathbf{x}\|^2\right) d\mathbf{x}=Id_p$, where $\mathbf{x}=(x_1,\ldots,x_p)' \in R^p$. For example, setting $K(u)=(2\pi)^{-p/2}exp(-u/2)$ gives the Gaussian kernel. The adaptive multivariate kernel density estimator of $f(\mathbf{x})$ constructed using $K(\cdot)$ is $\hat{f}(\mathbf{x})=\frac{1}{n}\sum_{i=1}^{n} h_i^{-p} K\left(\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\|^2\right)$, where h_i is a strictly positive bandwidth that modulates the smoothness of the density estimate (Scott 1992, Silverman 1998, Wand and Jones 1995).

The algorithm considered here seeks to cluster together observations that belong to a same modal region of $\hat{f}(\cdot)$. Assuming differentiability of $K(\cdot)$ on $[0, \infty)$, write $K'(u) = \partial K(u) = \partial K(u)/\partial u$ $(u \geq 0)$. The modes of $\hat{f}(\cdot)$ are then solutions to $\partial\hat{f}(\mathbf{x})/\partial\mathbf{x}=0$, where

$$\frac{\partial \hat{f}(\mathbf{x})}{\partial \mathbf{x}} = \frac{2}{n}\sum_{i=1}^{n} h_i^{-(p+2)} K'\left(\left\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\right\|^2\right)(\mathbf{x}-\mathbf{y}_i) \quad (1)$$

is the gradient of $\hat{f}(\cdot)$ evaluation at **x**. Define

$$A(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} h_i^{-(p+2)} K'\left(\left\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\right\|^2\right)\mathbf{y}_i \quad \text{and} \quad B(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} h_i^{-(p+2)} K'\left(\left\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\right\|^2\right).$$

Rearranging terms in eqn. (1) yields $\partial \hat{f}(\mathbf{x})/\partial \mathbf{x} = 2B(\mathbf{x})M(\mathbf{x})$ where $M(\mathbf{x}) = \bar{A}(\mathbf{x})/B(\mathbf{x})$ and

$$\bar{A}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} h_i^{-(p+2)} K'\left(\left\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\right\|^2\right)(\mathbf{y}_i - \mathbf{x}) = A(\mathbf{x}) - B(\mathbf{x})\mathbf{x}. \quad (2)$$

Since $B(\mathbf{x}) \neq = 0$, the modes of $\hat{f}(\cdot)$ are also solutions to the equation $M(\mathbf{x}) = 0$. To cluster a data set **Y**, one can numerically solve this equation starting from each $\mathbf{y}_i \in \mathbf{Y}$, as done by mean-shift, and cluster together all initial points that lead the algorithm to a same convergence point (a mode). The modes could be looked for by solving directly $\partial \hat{f}(\mathbf{x})/\partial \mathbf{x} = 0$, but solving $M(\mathbf{x}) = 0$ instead bestows the mean-shift with desirable properties, including: it is a steepest ascent algorithm with varying step size that depends on the gradient and on the value of $\hat{f}(\cdot)$; observations in the tail of the distribution where the density is smallest move faster toward the mode than when using $\hat{f}(\cdot)$; a single step is needed if the density is normal (Fukunaga and hostetler 1975, Cheng 1995, Fashing 2005).

This approach to clustering achieves a complexity of $O(n^2)$ per iteration, becoming computationally prohibitive when applied to large data sets. To design a computationally efficient algorithm, we will replace $M(\mathbf{x})$ by a function constructed using random subsamples of the data, quick to evaluate and whose expectation has the same roots as $M(\mathbf{x})$; a stochastic algorithm is invoked to eliminate the noise introduced by subsampling and to ensure convergence of the approximation to solutions of the equation $M(\mathbf{x}) = 0$ as the number of iterations increases.

2.2 SAMS: a stochastic approximation mean-shift algorithm

We integrate out the noise introduced in the algorithm by random sampling using a stochastic approximation procedure introduced by Robbins and Monro (1951) that is designed to find the roots of a vector-valued function $F(\mathbf{x})$ which cannot be computed exactly but can be estimated at any value of **x**. If the estimator of $F(\mathbf{x})$ is unbiased, the recursive sequence produced by the algorithm will converge, under certain conditions, to one of the roots of $F(\mathbf{x})$.

Let $\{Y_k\}_{k=1}^{\infty}$ denote a sequence of mutually independent random subsamples generated by sampling observations from **Y** either with or without replacement and with either equal or unequal probability. For every $k = 1, 2 \cdots$, let n_k denote the size of $\mathbf{Y}_k$, and write $\rho_k = n_k/n$ for the sampled fraction. The algorithm becomes computationally efficient if $\rho_k << 1$. For every $i = 1, \cdots, n$, let π_{ik} denote the probability of inclusion of the ith observation in the kth subsample. For example, when sampling is done without replacement and with equal probability, the probability of inclusion is $\pi_{ik} = n_k/n$. When observations are sampled with replacement and with probability p_{ik}, $i = 1, \dots, n$, at each draw, the probability of inclusion is $\pi_{ik} = 1-(1-p_{ik})^{nk}$. Let z_{ik} denote an indicator random variable equal to one if $\mathbf{y}_i \in \mathbf{Y}_k$ and 0 otherwise. Define

$$\bar{A}_k(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \frac{z_{i,2k}}{\pi_{i,2k}} h_i^{-(p+2)} K'\left(\left\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\right\|^2\right)(\mathbf{y}_i - \mathbf{x})$$

and

$$B_k(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \frac{z_{i,2k-1}}{\pi_{i,2k-1}} h_i^{-(p+2)} K'\left(\left\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\right\|^2\right),$$

which denote counterparts of $\bar{A}(\mathbf{x})$ and $B(\mathbf{x})$ constructed using the random subsamples $\mathbf{Y}_{2k}$ and $\mathbf{Y}_{2k-1}$, respectively. The functions $\bar{A}_k(\cdot)$ and $B_k(\cdot)$ are Horvitz-Thompson estimators of $\bar{A}(\cdot)$ and $B(\cdot)$ (Thompson 1992). Since $E(z_{i,2k}|\mathbf{Y}) = \pi_{i,2k}$, it follows that

$$E\left\{\bar{A}_k(\mathbf{x})\,|\boldsymbol{Y}\right\} = \frac{1}{n}\sum_{i=1}^{n} \frac{\pi_{i,2k}}{\partial_{i,2k}} h_i^{-(p+2)} K'\left(\left\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\right\|^2\right) = \bar{A}(\mathbf{x}).$$

Thus, conditional on $\mathbf{Y}$, $\bar{A}_k(\mathbf{x})$ is an unbiased estimator of $\bar{A}(\mathbf{x})$. Likewise, $E\{B_k(\mathbf{x})|\mathbf{Y}\} = B(\mathbf{x})$.

Let $\mathbf{y} \in \mathbf{Y}$ denote any of the points to be clustered. Let $\{\gamma_k\}_{k=1}^{\infty}$ and $\{\beta_k\}_{k=1}^{\infty}$ denote two decreasing sequences of positive numbers referred to as *gain coefficients*. Let $\eta = (\eta_0, \eta_1)'$ denote two real-valued constants that satisfy $0 < \eta_0 < \eta_1 < \infty$, and let $b_0 \in (\eta_0, \eta_1)$. The proposed stochastic approximation mean-shift algorithm generates a sequence $\{\mathbf{x}_k\}_{k=0}^{\infty}$ started from $\mathbf{x}_0 = \mathbf{y}$ and recursively defined, for every $k \geq 0$, by

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \gamma_{k+1}\frac{\bar{A}_{k+1}(\mathbf{x}_k)}{b_{k+1}^{\eta}}, \qquad (3)$$

where

$$b_{k+1}^{\eta} = min\left\{max\left\{b_{k+1}, \eta_0\right\}, \eta_1\right\} \qquad (4)$$

and

$$b_{k+1}{=}b_k^{\eta}{+}\beta_{k+1}\left\{B_{k+1}\left(\mathbf{x}_k\right)-b_k^{\eta}\right\}. \quad (5)$$

The sequence $\{\mathbf{x}_k\}_{k=0}^{\infty}$ converges almost surely to stationary points of $\hat{f}(\cdot)$ under mild regularity conditions. This will be proven in section 2.4. The sequence is stopped in step k if there is enough evidence that $\mathbf{x}_k$ might be close to its convergence point. How to stop the sequence is discussed in section 2.6. If the algorithm is properly set up, $\{\mathbf{x}_k\}_{k=0}^{\infty}$ converges with probability close to 1 to the mode of the modal region to which $\mathbf{x}_0$ belongs. This probability depends, for example, on how the gain coefficients $\{\gamma_k\}_{k=1}^{\infty}$ and $\{\beta_k\}_{k=1}^{\infty}$ that appear in eqns. (3) and (5) are set up. They must satisfy regularity conditions that will be given in Theorem 1. How to set these coefficients and other tuning parameters in practice is relatively easy and will be discussed in section 2.5.

To cluster a data set $\mathbf{Y}$, each observation $\mathbf{y}_i$, $i = 1, \ldots, n$, initiates a sequence $\{\mathbf{x}_k\}_{k=0}^{\infty}$, and all initial values that yield trajectories stopped within a small distance of each other are assigned to a same cluster. An overview of the algorithm is presented below.

The stochastic approximation mean-shift algorithm

- Initialize parameters (see section 2.5 for suggested values) and set $k = 0$.
- Step 1. Set $\mathbf{x}_k = \mathbf{y}_1$.
- Step 2. Generate two independent subsamples ($\mathbf{Y}_{2k-1}$ and $\mathbf{Y}_{2k}$), compute $\bar{A}_{k+1}(\mathbf{x}_k)$, $B_{k+1}(\mathbf{x}_k)$, b_{k+1} and b_{k+1}^{η}, and deduce $\mathbf{x}_{k+1}$ using eqn. (3).
- Step 3. If $\mathbf{x}_{k+1}$ meets the stopping rule (see section 2.6), then

  – if SAMS has been run for each observation to be clustered, go to Step 4;

  – otherwise return to Step 1 after having replaced $\mathbf{y}_1$ by the next point to be clustered.

  If $\mathbf{x}_{k+1}$ does not meet the stopping rule, then increment k by one, replace $\mathbf{x}_k$ by $\mathbf{x}_{k+1}$ and return to Step 2.
- Step 4. Merge together all points located within a short distance of each other, and assess whether each candidate cluster corresponds to a mode of the density estimate (see section 2.6). If they do, the procedure stops; otherwise, restart the algorithm from the location of the candidate clusters that do not identify a mode.

2.3 Connection with the standard mean-shift algorithm

The (adaptive) mean-shift developed by Comaniciu et al. (2001) seeks solutions to $\partial\hat{f}\left(\mathbf{x}\right)/\partial\mathbf{x}{=}0$ using a fixed-point iterative scheme applied to $M(\mathbf{x})$. The procedure starts from a data point to be clustered, say $\tilde{\mathbf{x}}_0 = \mathbf{y} \in \mathbf{Y}$, and produces a sequence $\{\tilde{\mathbf{x}}_k\}_{k=0}^{\infty}$ where

$$\tilde{\mathbf{x}}_{k+1}{=}A\left(\tilde{\mathbf{x}}_k\right)/B\left(\tilde{\mathbf{x}}_k\right). \quad (6)$$

This sequence converges to the mode of the modal region that **y** belongs to (Carreira-Perpiñan 2007). The procedure may be applied to the entire data set by changing the initial value. All data points **y** initiating a trajectory that converges toward a same mode are clustered together.

We remark that the recursive equation (3) is not derived by simply replacing $A(\mathbf{x})$ and $B(\mathbf{x})$ in eqn. (6) by unbiased estimators computed using random subsamples. This approach would not permit removal of the random noise, preventing the algorithm from converging. Setting $\gamma_k = \beta_k = 1$ and $\mathbf{Y}_k = \mathbf{Y}$, $k = 1, 2 \cdots$, yields $\bar{A}_{k+1}(\mathbf{x}_k) = \bar{A}(\mathbf{x}_k) = A(\mathbf{x}_k) - \mathbf{x}_k B(\mathbf{x}_k)$, giving $\mathbf{x}_{k+1} = A(\mathbf{x}_k)/B(\mathbf{x}_k)$. Thus, SAMS and mean-shift are identical under the above specifications. Setting $\gamma_k = \beta_k = 1$ and $\mathbf{Y}_k = \mathbf{Y}_1$, $k = 1, 2 \cdots$, in the stochastic approximation mean-shift would be identical to performing mean-shift on a fixed random subsample $\mathbf{Y}_1$. While this algorithm may be fast if n_1 <<, n there is no guarantee that it will behave similarly to the mean-shift.

2.4 Convergence of the algorithm

Let $\mathscr{F}=\{\mathscr{F}_k\}_{k=1}^{\infty}$, where $\mathscr{F}_k=\sigma\{\mathbf{z}_1,\ldots,\mathbf{z}_k,\mathbf{Y}\}$ and $\mathbf{z}_k=(z_{i;2k-1},z_{i;2k})_{i=1,\ldots,n}$, $\quad k=1,2\cdots$, be a sequence of increasing sigma-algebras generated by $\mathbf{z}_1, \ldots, \mathbf{z}_k$ and $\mathbf{Y}$. The sequence $\{\mathbf{x}_k\}_{k=1}^{\infty}$ can be written as $\mathbf{x}_{k+1} = \mathbf{x}k + \gamma_{k+1} h(\mathbf{x}k) + \gamma_{k+1}(\mathbf{x}_k)$, where $h(\mathbf{x}_k) := E\left\{\frac{\bar{A}_{k+1}(\mathbf{x}_k)}{b_{k+1}^{\eta}}|\mathscr{F}_k\right\} = \bar{A}(\mathbf{x}_k)C(\mathbf{x}_k)$ is the *mean field*, where $C(\mathbf{x}_k)=E\left\{1/b_{k+1}^{\eta}|\mathscr{F}_k\right\}$, and where the stochastic perturbations $e_{k+1}(\mathbf{x}_k)=\frac{\bar{A}_{k+1}(\mathbf{x}_k)}{b_{k+1}^{\eta}} - h(\mathbf{x}_k) \quad (k=1,2\cdots)$ form a sequence of martingale differences. Thus, the algorithm defined by (3) is a special case of the Robbins-Monro procedure (Robbins and Monro 1951, Kushner and Yin 2003). Its convergence is established in the following Theorem.

Theorem 1—*Assume that:* (*A*.1) $\{\gamma_k\}_{k=1}^{\infty}$ *is a positive sequence such that:* $\sum_{k=1}^{\infty}\gamma_k=\infty$ *and* $\sum_{k=1}^{\infty}\gamma_k^2<\infty$; (*A*.2) $K(\cdot)$ *is p time continuously differentiable on* R^+; (*A*.3) $sup_{u\in R^+} K'(u^2)u<\infty$; (*A*.4) $\underset{i=1,\ldots,n;k=1,2\cdots}{inf} \pi_{ik}>0$. *Then, w.p.1, conditional on* $\mathbf{Y}$, $lim\ sup_{k\to\infty} d(\mathbf{x}_k,\mathscr{L}_{\hat{f}})=0$, *where* $\mathscr{L}_{\hat{f}}=\left\{\mathbf{x}\in R^p{:}\partial\hat{f}(\mathbf{x})/\partial\mathbf{x}=0\right\}$ *is the set of critical points of* $\hat{f}(\cdot)$.

A proof of Theorem 1 is given in the Appendix. We note that convergence holds for *n*. Assumption (A.1) is satisfied if $\underset{k\to\infty}{lim} k^{\alpha_\gamma}\gamma_k=\gamma_\star$ for some constants $a_\gamma \in (1/2; 1]$ and $\gamma_\bigstar \in (0; \infty)$. The sequence of gain coefficients $\{\beta_k\}_{k=1}^{\infty}$ need not satisfy any regularity conditions in order for $\{\mathbf{x}_k\}_{k=1}^{\infty}$ to coverage because of the bounds imposed on b_k^{η} by η_0 and η_1. However, in order for b_k to behave as $B(\mathbf{x}_k)$, a desirable property so the procedure mimics the standard mean-shift, $\{\beta_k\}_{k=1}^{\infty}$ should satisfy conditions identical to those of Assumption (A.1). The Gaussian kernel satisfies Assumptions (A.2) and (A.3). More generally, when

Assumption (A.2) holds, Assumption (A.3) is verified if $K'(u) = O(u^{-1/2})$. In particular, it holds if $\int \mathbf{x}\mathbf{x}' K\left(\|\mathbf{x}\|^2\right) d\mathbf{x} = Id_p$.

The sequence $\{\mathbf{x}_k\}_{k=1}^{\infty}$ converges to one of the roots of $\partial \hat{f}(\mathbf{x}) / \partial \mathbf{x}$ by ascending the surface of $\hat{f}(\cdot)$. Trajectories started from points of a same modal region of the kernel density $\hat{f}(\cdot)$ are expected to be clustered together. An important point to make is that convergence of $\{\mathbf{x}_k\}_{k=1}^{\infty}$ is independent of the sampled fraction ρ_k, as long as it satisfies Assumption (A.4). In particular, it does not have to grow with n, and the complexity of each iteration is $O(n)$ if n_k is kept constant as n increases. In practice, the sequence is subject to random perturbations, and points located close to the boundary of a modal region are the most likely to jump to a diffrent modal region. This is preventable by properly sizing $\mathbf{Y}_\mathrm{k}$ during the initial steps.

2.5 Details about implementing the algorithm

The algorithm depends on several parameters, including the bandwidths h_i, the sampled fractions ρ_k, the bounds η_0 and η_1, and the gain coefficients γ_k and β_k. The default values suggested below worked satisfactorily in applications or provided a good starting point.

2.5.1 Bandwidth selection (h_i)—SAMS and the standard mean-shift identify the same clusters provided they use identical bandwidths h_i. Thus, the selection of h_i to run SAMS should not depend on the size of the subsamples used to construct the functions $\bar{A}_{k+1}$ and B_{k+1}. Guidelines for bandwidth selection for mean-shift discussed in Comaniciu et al (2001) apply directly to SAMS, and we set $h_i = \lambda_i \alpha_1$, where $\lambda_i = \left\{\hat{\beta} / \hat{f}(\mathbf{y}_i)\right\}^{\alpha_2}$, $\hat{\beta} = \prod_{j=1}^{n} \tilde{f}(\mathbf{y}_j)^{1/n}$, and where $\tilde{f}(\mathbf{y}_i)$ is a *pilot* density estimate of $f(\mathrm{y}_i)$, which can be computed using a fixed bandwidth kernel density estimator (Silverman 1998). The parameter $\alpha_1 > 0$ can be set equal to the bandwidth used in constructing the pilot density estimate. A simple data-driven rule is to set $\alpha_1 = \hat{\sigma} n^{-1/(p+4)}$, where $\hat{\sigma}^2$ denotes the sample variance (Scott 1992). The parameter $\alpha_2 \in [0; 1]$ controls the sensitivity of the estimator with respect to the variation in the *pilot* density estimate. The sensitivity is the highest when $\alpha_2 = 1$, whereas the method reverts to a fixed bandwidth estimator with single bandwidth when $\alpha_2 = 0$. Breiman et al. (1977) suggested using $\alpha_2 = 1/p$. Based on our experience with running the algorithm in various settings, we find this rule reasonable, but adjustments may be needed. For example, in the presence of highly dense clusters, the algorithm performed better with smaller values of α_2 (e.g., $\alpha_2 = 1/2p$ or smaller).

2.5.2 Sampled fractions (ρ_k)—The choice of ρ_k is driven by the sample size n and the size of the smallest cluster. Empirical evidence suggests that sampled fractions ranging between 0.1% and 1% may be sufficient to run SAMS when $n \simeq 10^5 – 10^6$ and with clusters accounting for ≥1% of the data. When the value of ρ_k becomes too small, the algorithm may identify either too many or too few clusters. We can calibrate the sampled fractions by running SAMS on a training set and retain a value of ρ_k that preserves cluster assignment, measured by the proportion of pairs of observations that are assigned to a same cluster. The values of ρ_k can also be decreased after a few steps, once $\mathbf{x}_k$ has moved away from the boundary of its modal region.

2.5.3 Choice of η—η_0 prevents the jumps of $\{\mathbf{x}_k\}_{k=1}^{\infty}$ from being too large, useful during the initial steps of the algorithm where b_{k+1} may take small values. The role of η_1 is solely technical and can be selected arbitrarily large. In applications, we have used $\eta_0 = 10^{-3}$ and = $\eta_1 10^{50}$.

2.5.4 Sequences of gain coefficients $\{\gamma_k\}_{k=1}^{\infty}$ and $\{\beta_k\}_{k=1}^{\infty}$—Gain coefficients $\{\gamma_k\}_{k=1}^{\infty}$ that satisfy Assumption (A.1) may be formulated as

$$\gamma_k = \frac{\alpha_{\gamma,0}}{max(k-k_0,1)^{\alpha_{\gamma,1}}} \quad (k=1,2,\cdots) \qquad (7)$$

where $\alpha_{\gamma,0} > 0$, $\alpha_{\gamma,1} \in (1/2, 1]$ and $k_0 \in \{1, 2 \cdots\}$. In applications, we set $\alpha_{\gamma,0} = 1$, and use $\alpha_{\gamma,1} = 0.51$ to reduce the probability that the algorithm will stop prematurely before approaching its convergence point, say $\mathbf{x}_\infty$.

The gain coefficients γ_k defined in eqn. (7) are deterministic and do not adapt to the distance between $\mathbf{x}_k$ and $\mathbf{x}_\infty$. The concern raised here is that these coefficients may shorten the jumps of $\mathbf{x}_k$ too quickly, slow down convergence, and generate unnecessary clusters. Ideally, $\mathbf{x}_k$ should make bigger jumps when remote from $\mathbf{x}_\infty$ and smaller jumps otherwise. This adaptive behavior can be achieved using a multidimensional version of Kesten's procedure (Kesten 1958, Delyon and Juditsky 1993) that decreases γ_k only when $(\mathbf{x}_k - \mathbf{x}_{k-1})'$ $(\mathbf{x}_{k-1} - \mathbf{x}_{k-2})$ is negative, expected to occur more frequently as $\mathbf{x}_k$ approaches $\mathbf{x}_\infty$. Setting $s_0 = 1$, this procedure yields the following gain coefficients for SAMS:

$$\begin{cases} \gamma_{k+1} = \gamma\left(s_{k+1}\right) \\ s_{k+1} = s_k + \mathbf{1}_{\left\{\bar{A}_{k+1}(\mathbf{x}_k)'\bar{A}_k(\mathbf{x}_{k-1})<0\right\}}, \end{cases} \qquad (8)$$

where $\gamma(\cdot)$ is a deterministic real-valued function (e.g., $\gamma(s) = \gamma_0 s^{-\alpha_\gamma}$, where $\gamma_0 \in R_*^+$, and $\alpha_\gamma \in (0.5; 1]$; in practice, we set $\gamma_0 = 1$ and $\alpha_\gamma = 0.51$), and where $\mathbf{1}_{\{\cdot\}}$ is the indicator function. When $p = 1$, the sequence $\{s_k\}_{k=1}^{\infty}$ counts the number of times two consecutive values of the sequence $\left\{\bar{A}_{k+1}\left(\mathbf{x}_k\right)\right\}_{k=1}^{\infty}$ (hence of the sequence $\{\mathbf{x}_{k+1} - \mathbf{x}_k\}_{k=1}^{\infty}$) have opposite signs. Robbins-Monro procedures constructed with the gain coefficients defined in (8) retain the almost sure convergence of their regular (non-adaptive) version while improving the performance of the algorithm (Delyon and Juditsky 1993). We also observed improvements in numerical studies.

2.6 Stopping rules and convergence criterion

To stop the algorithm, one can run SAMS for a fixed number of iterations, assess convergence, and decide to continue or not. Another strategy is to stop the algorithm as soon as there is evidence that $\mathbf{x}_k$ might be close to its limit. Such evidence may come from the gradient $\bar{A}(\mathbf{x}_k)$, which converges to a r.v. centered about $\mathbf{0}$ as $\mathbf{x}_k$ converges. Thus, to monitor convergence for SAMS, we propose to compute the sequence $\bar{\mathbf{a}}_k = (\bar{a}_{k1}, \ldots, \bar{a}_{kp})'$, $k = 1, 2 \cdots$, started from $\bar{\mathbf{a}}_0 = \bar{A}_1(\mathbf{x}_0)$ and recursively defined as $\bar{\mathbf{a}}_{k+1} = \bar{\mathbf{a}}_\mathrm{k} + \phi_k\{\bar{\mathbf{A}}_{k+1}(\mathbf{x}_k) - \bar{\mathbf{a}}_k\}$. Under mild conditions, $\bar{\mathbf{a}}_k \xrightarrow{a.s.} 0$ as $k \to \infty$, and we stop SAMS in step k if $\sup_{l=1,\ldots,p} |\bar{a}_{kl}| < \epsilon_l$, where $\epsilon = (\epsilon_1, \ldots, \epsilon_p)$ are pre-specified thresholds. We note that $\bar{\mathbf{a}}_k$ is computed at no additional cost since all necessary quantities are available from SAMS. In applications, we set $\phi_k = 1k^{\alpha\phi}$ and $a_\phi = 0.75$. To properly scale ϵ to the magnitude of $\bar{A}_k(\mathbf{x})$, we compute $\bar{A}_1(\mathbf{y}_i) = (\bar{A}_{11}(\mathbf{y}_i), \ldots, \bar{A}_{1p}(\mathbf{y}_i))'$ for all $i = 1, \ldots, n$, and set ϵ_l equal to the empirical 5th percentile of $\{A_{1l}(\mathbf{y}_i)\}$ for every $l = 1, \ldots, p$.

An alternative stopping rule, free of the magnitude of $\bar{A}_k(\mathbf{x})$, is designed using a sequence $\left\{\bar{s}_k\right\}_{k=1}^{\infty}$, initialized at $\bar{s}_0 = \bar{s}_1 = 0$ and defined as $\bar{s}_{k+1} = \bar{s}_\mathrm{k} - 1\ \delta_{k+1}\ (\bar{s}_k - \mathbf{1}_{\{\bar{A}_{k+1}(\mathbf{x}_k)'\bar{A}_k(\mathbf{x}_{k-1}) < 0\}})$, where $\{\delta_k\}_{k=1}^{\infty}$ is a sequence of positive numbers converging to 0 as $k \to \infty$. In practice, we set $\delta_k = 1/k^{a\delta}$ with $a_\delta \in (1/2, 1]$. When $a_\delta = 1$, we have $\bar{s}_k = s_k/k$, and $\bar{s}_k$ is the proportion of times that Kesten's procedure increased in the first k steps. When $a_\delta \in (1/2, 1)$, $\bar{s}_k$ estimates the proportion of times the procedure increased, but the most recent steps of the algorithm receive more weight than the first ones. Under mild regularity conditions (e.g., $\alpha_\delta \in (1/2, 1]$), $\bar{s}_k \xrightarrow{a.s.} P\left\{lim_{k\to\infty}\bar{A}_{k+1}(\mathbf{x}_k)'\bar{A}_k(\mathbf{x}_{k-1}) < 0\right\} = 1.2$ and $2k^{\alpha_\delta/2}\left(\bar{s}_k - 1/2\right) \xrightarrow{D} N(0,1)$. Thus, we propose $\bar{s}_k \simeq 1/2$ is strong. Specifically, we build an approximate one-sided 95% confidence interval $\bar{s}_k$ as $[\bar{s}_k - z_{0.95}/2k^{a\delta}, 1]$, where z_a is the ath-percentile of the standard normal distribution, and stop SAMS in step k if $\bar{s}_k - z_{0.95}/2k^{\alpha\delta} > 1/2 - \Delta$, where Δ is a fixed threshold. In practice, we have used $a_\delta = 0.95$ and $\Delta \simeq 0.15$ with success.

The first (gradient-based) stopping rule uses local information about the density estimate, while the second one solely rests on the expected behavior of sign $[\bar{A}_{k+1}\ (\mathbf{x}k)'\bar{A}_k(\mathbf{x}_{k-1})]$ as $\mathbf{x}k$ approaches its limit. We found that the first stopping rule becomes effective within fewer steps but it requires the thresholds ϵ to be properly set. The second rule may require more iterations before it becomes effective, but it is more robust and easier to set.

Once all sequences $\{\mathbf{x}_k\}_{k=1}^{\infty}$ have been stopped, the data are merged to form candidate clusters. The sequence $\{\mathbf{x}_k\}_{k=1}^{\infty}$ usually converges to a mode, but convergence to a saddle point occasionally occurs. Let $\hat{\mathbf{x}}$ be the location of a candidate cluster defined (e.g.) as the average final position of the trajectories for all points assigned to that cluster. The hessian of $\hat{f}(\mathbf{x})$ is

$$H\left(\mathbf{x}\right)=\frac{4}{n}\sum_{i=1}^{n}\left\{h_i^{-(p+4)}K''\left(\left\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\right\|^2\right)\left(\mathbf{y}_i-\mathbf{x}\right)\left(\mathbf{y}_i-\mathbf{x}\right)'-\frac{h_i^{-(p+2)}}{2}K'\left(\left\|\frac{\mathbf{x}-\mathbf{y}_i}{h_i}\right\|^2\right)Id_p\right\},$$

To verify that $\hat{\mathbf{x}}$ is a local maximum (a mode) for $\hat{f}(\cdot)$, we check whether $\lambda_{max}(H(\mathbf{x})) < 0$, where $\lambda_{max}(H(\mathbf{x}))$ denotes the largest eigenvalue of $H(\mathbf{x})$. Failure to meet this criterion would prompt restarting SAMS from $\hat{\mathbf{x}}$ until convergence toward a mode occurs.

3. NUMERICAL EXAMPLES AND APPLICATIONS

3.1 Simulated data

We found it instructive to first evaluate SAMS in a simple two-dimensional ($p = 2$) setting using simulated data. We generated a data set consisting of 100,000 observations using a finite mixture model with 6 components, some of which were non-normal, and mixing proportions ranging between 1% and 39%. We constructed a density estimate based on a Gaussian kernel using $a_1 = 0.05$ and $a_2 = 0.5$ (Fig. 1, panel A), and first clustered the data set using the standard mean-shift. Six clusters closely matching the 6 components of the mixture were identified after 100 iterations. Panel C displays trajectories started from 1,000 randomly selected points. The color of each trajectory matches cluster assignment.

We ran SAMS by sampling $\rho_k = 0.4\%$ (4 out of 1,000) of the data points at each iteration. The gain coefficients $\{\gamma_k\}_{k=1}^{\infty}$ were constructed using Kesten's procedure with $\gamma(s) = 1/s^{a_\gamma}$ and used $a_\gamma = 0.51$. We set $\beta_k = 1/k^{0.51}$, $\eta_0 = 0.001$ and $\eta_1 = 10^{50}$. The bandwidths h_i were identical to those used for the standard mean-shift. The algorithm ran for 100 iterations.

Let C_i^{MS} and $C_i^{SAMS}\left(\rho\right), \quad i=1,\ldots,n$, denote the clusters that $\mathbf{y}_i$ is assigned to by the standard mean-shift and by SAMS with a sampled fraction ρ. We computed a clustering error rate as $R\left(\rho\right)=\sum_{i=1}^{n}\mathbf{1}_{\left\{C_i^{SAMS}(\rho)\neq C_i^{MS}\right\}}/n$, equal to the proportion of observations assigned to different clusters by the two algorithms. Thus, mean-shift was the gold standard.

Panel E displays trajectories of $\{\mathbf{x}_k\}_{k=1}^{\infty}$ when $\rho_k = 0.004$ started from the 1,000 points plotted in panel C. Each point was clustered 100 times. SAMS found an average of 6.1 clusters (standard deviation (SD) = 0.4). The occasional additional cluster(s) were very small and the average clustering error rate was $\bar{R}(0.004) = 0.008$ (SD = 0.005); thus, on average 8 out of 1,000 data points were assigned to a cluster different from the one they were assigned to by mean-shift.

We repeated the above analyses with $a_\gamma = 0.75$ (results not shown). The algorithm found an average of 6.8 clusters (SD = 0.6) and the average clustering error rate increased to 0.015, mostly because a few trajectories stopped prematurely. By restarting the algorithm from the location of the candidate clusters, some of them ultimately merged to form 6 final clusters.

We performed similar analyses using fixed (non-adaptive) gain coefficients of the form: $\gamma_k = 1/k^{a_\gamma}$, with $a_\gamma = 0.51$, 0.75, or 1. Overall, SAMS identified larger numbers of clusters than with Kesten's method because the gain coefficients decreased too quickly, preventing $\mathbf{x}_k$

from approaching its limit within 100 iterations. The issue could also be resolved by restarting the algorithm from the location of the candidate clusters.

The average time per iteration is plotted against ρ_k in panel E. With $\rho_k = 0.001$, SAMS was 750 times faster than mean-shift. The average error rate increased to 0.059 (SD = 0.038).

In panel F, several trajectories started from the same six data points display the random behavior of SAMS. For comparison, the trajectories produced by the standard mean-shift are also plotted. The difference with mean-shift and the variability across repeats for SAMS were small, even with a sampled fraction of 0.4%.

We stopped SAMS used the stopping rules defined in section 2.6. We subsampled $\rho_k = 0.4\%$ of the data in each iteration, and assessed performances by repeating clustering 100 times for each stopping rule. For the stopping rule based on the gradient $\bar{\mathbf{a}}_k$, we defined ϵ using the 5th percentiles of the gradient at the first iteration, and set $\alpha_\phi = 0.75$. The algorithm performed on average 69.5 iterations per trajectory (SD = 4.1), and resulted in an average clustering error rate of 0.0087 (SD = 0.006). For the stopping rule based on $\bar{s}_k$, we set $\Delta = 0.15$ and $\alpha_\delta = 0.95$. The algorithm performed on average 53.8 iterations per trajectory (SD = 8.0), and resulted in an average clustering error rate of 0.0114 (SD = 0.010). We also combined the two strategies, running first 50 iterations of the gradient stopping rule described above, followed by the second stopping rule until the trajectory was stopped. This resulted in an average clustering error rate of 0.0106 (SD = 0.013), and performed on average 57.2 iterations per trajectory (SD = 5.7).

Regions of the sample space where points are not properly clustered due to the random nature of the algorithm can be detected using a variety of metrics. Here, we used $\rho_k = 0.1\%$ and calculated the number of clusters in the neighborhood of each point using k-nearest neighbor ($k = 20$; *Metric 1*); and the L^2-distance traveled by each sequence $\{\mathbf{x}_l\}_{l=1}^{\infty}$ until stopped (*Metric 2*) (panels G-J). The points that were not properly clustered belonged to the upper tail of the distribution of each metrics and were easily located using the suggested metrics for reclustering.

Theoretically, mean-shift converges linearly (Carreira–Perpiñan 2007), while one can show that SAMS will converge at a rate of $k^{-1/2}$ under appropriate conditions (e.g., Pflug 1996). This asymptotic rate, however, does not necessarily capture the transient behavior of the algorithm. In figure 2, we empirically compared the speed of convergence of SAMS with that of the standard mean-shift (see figure legend for detail). These results indicate that, with the exception of a few points, SAMS and mean-shift collapsed the data toward the modes at approximately the same rates, while the cost of each iteration was hundred of times higher for the regular mean-shift.

3.2 Application to image segmentation

We ran SAMS and the standard mean-shift on three images to evaluate their performances for image segmentation (figure 3). For the *cameraman* (panel A.1), the data set consisted of 65, 536 (= 256 × 256) 3-dimensional observations $\mathbf{y}_i = (y_{i1}, y_{i2}, y_{i3})$, where y_{i1} and y_{i1} identify the location of the ith pixel in the image and where y_{i3} is the color (on a gray scale) of the pixel. We segmented the image using the standard mean-shift algorithm (panel A.2). Five clusters were identified. We next used SAMS. The bandwidth parameters were identical to those used to run the standard mean-shift. When randomly sampling 0.2% (2 out of 1000) of the pixels at each iteration, we obtained 5 clusters (the error rate was 1.8%) (panel A.3). Thus, SAMS performed almost identically to the standard mean-shift but was 305 times faster. The algorithm achieved similar performances with the other two images (see figure 3 for detail).

3.3 Comparison and combination of SAMS with alternative acceleration approaches

The computational limitation of mean-shift as a clustering tool has been widely acknowledged and several effective methods proposed to speed it up. These methods fall into three categories.

In the first category, acceleration is accomplished by reducing the number of iterations needed to cluster each data point. The blurring mean-shift mentioned in the introduction (Fukunaga and Hostetler 1975, Carreira-Perpiñan 2006a) and the spatial discretization mean-shift (Carreira-Perpiñan 2006b) are two such examples. The latter algorithm was proposed for image segmentation, and, unlike SAMS, may not be suitable for arbitrary data sets. In the second category, acceleration is achieved by reducing the number of points on which a full run of the algorithm is performed. One example is the agglomerative mean-shift (Yuan et al 2012) in which points and subclusters are gradually merged into temporary subclusters based on an appropriate metric. We note that methods from these two categories can be used in conjunction with SAMS to further accelerate clustering, and thus do not compete with each other. In previous applications, we observed that SAMS could cluster hundreds of times faster than the standard mean-shift while causing negligible clustering errors (e.g., < 1%). When combining SAMS with an agglomerative mean-shift, gradually merging subclusters within each step, the resulting procedure further accelerated SAMS by > 70 times without noticeably increasing the clustering error rates.

Algorithms from the third category accelerate clustering by reducing the computational cost of each iteration. SAMS is one of them. Other similar procedures include the mean-shift sparse expectation-maximization (EM) (Carreira-Perpiñan 2006b). This algorithm proceeds from expressing mean-shift as a (generalized) EM algorithm (Carreira-Perpiñan 2007) and accelerates clustering by performing a partial E-step. This approach is supported by a reformulation of the E-step of the EM algorithm proposed by Neil and Hinton (1998). We applied this algorithm to previously analyzed data, and were able to speed up the clustering process by a factor of 2 to 4 compared to mean shift, lower than the accelerations reported for SAMS, while achieving similar clustering error rates. Another algorithm is the spatial neighbourhood mean-shift (Carreira-Perpiñan 2006b), but it does not converge to a mode of $\hat{f}(\cdot)$.

The stochastic gradient mean-shift algorithm proposed by Yuan and Li (2009) also uses stochastic optimization, but differently from SAMS. In step $k + 1$, it computes an online version of $\breve{\mathbf{x}}_{k+1} = \breve{A}_k(\breve{\mathbf{x}}_k$, where $\breve{A}_k(\cdot)$ and $\breve{B}_k(\cdot)$ approximate the functions $A(\cdot)$ and $B(\cdot)$ used by mean-shift (eqn. (6)) defined as $\breve{A}_k\left(\breve{\mathbf{x}}_k\right) = \frac{1}{k}\sum_{i=1}^{k} h_i^{-(p+2)} K'\left(\|\frac{\breve{\mathbf{x}}_{i-1}-\mathbf{y}_i}{h_i}\|^2\right)\mathbf{y}_i$ and $\breve{B}_k\left(\breve{\mathbf{x}}_k\right) = \frac{1}{k}\sum_{i=1}^{k} h_i^{-(p+2)} K'\left(\|\frac{\breve{\mathbf{x}}_{i-1}-\mathbf{y}_i}{h_i}\|^2\right)$. Thus, $\breve{A}_k(\breve{\mathbf{x}}_k)$ and $\breve{B}_k(\breve{\mathbf{x}}_k)$ are sequential averages of k values of the kernel, each evaluated using one of the previous locations $\breve{\mathbf{x}}_i$, i = 0; ..., k–1. Convergence of $\breve{\mathbf{x}}_k$ as $k \to \infty$ holds only if the sample size increases ($n \to \infty$). By comparison, SAMS converges as $k \to \infty$ even if n is fixed. In practice, to circumvent this limitation, the stochastic gradient mean-shift is combined with blurring: the first iteration computes $\left\{\breve{\mathbf{x}}_k\right\}_{k=1}^{n}$ for each data point using $\mathbf{Y}$ to build $\hat{f}(\cdot)$, and replaces $\mathbf{Y}$ by the output of the first iteration; a second iteration is conducted similarly, and so on. The resultant algorithm reduces the total number of iterations compared to mean-shift, but each iteration has a cost of O(n2). Thus, accelerations reported by Yuan and Li (2009) (2 to 3 times faster) were substantially lower than those achieved by SAMS.

4. DISCUSSION

We have proposed a stochastic approximation mean-shift algorithm for density-based clustering and illustrated its performance on simulated data and in the context of image segmentation. The algorithm is designed to process data sets that are too large to be efficiently clustered using nonparametric algorithms such as mean-shift, where it can substantially reduce computing times. To gain efficiency, each iteration of the algorithm is constructed using subsamples instead of the entire data set. The noise that arises from random subsampling is gradually eliminated by means of a Robbins-Monro procedure. We showed that the algorithm converges almost surely to stationary points of the density estimate, as does the mean-shift. In applications to simulated data and image segmentation, we found that the procedure is considerably faster than the mean-shift without sacrificing much of its accuracy. Just as for the mean-shift, the number of clusters does not need to be specified before running the stochastic approximation procedure, and it is automatically determined by the number of modes of the density estimate. The number of modes depends on the tuning parameters used to construct the kernel density estimate, and it increases with parameters that produce rougher estimates.

Algorithms that can cluster in high dimensions are highly desirable. It is well-known that the performance of kernel density estimators degrades as the dimension increases, caused by the sparseness of the data in high-dimensional spaces. In principle, the sample size must grow at least exponentially fast with the dimension of the data in order to maintain equivalent accuracy of estimation (Scott 1992, Silverman 1998). However, several authors have reported empirical evidence of acceptable behavior of multivariate density estimators in high-dimensions (Scott 1992). It has also been observed that mode finding procedures and density-based clustering algorithms performed well in dimensions 50 and 100 (Scott 1992, Georgescu et al 2003). Since we are not interested in estimating the density itself but rather its modal regions, the curse of dimensionality is not necessarily deleterious to modal clustering. We evaluated the performance of SAMS in multi-dimensional settings using simulated data sets of dimension varying between 4 and 100. These simulations (not shown) indicated that SAMS was able to perform well in high-dimensions, even with small sampled fractions. Key to successful clustering was a good separation between clusters. Strategies based on dimension reduction could improve resolution.

ACKNOWLEDGEMENTS

This work was supported by NIH R01 Grants NS039511, CA134839, and AI069351. The authors gratefully acknowledge the comments by an anonymous reviewer and an associate editor.

APPENDIX: PROOF OF THEOREM 1

Delyon et al. (1999) established the convergence of a general Robbins-Monro procedure under assumptions that are appropriate when the mean field has multiple roots; that is, when $\hat{f}_K(\cdot)$ is multimodal, which we expect to be the case in clustering applications. Convergence of the sequence $\{\mathbf{x}_k\}_{k=1}^{\infty}$ may be investigated using their work. Thus, to prove Theorem 1, it is sufficient to show that Assumptions (A1-A3) imply Assumptions (SA0-SA4) used in Delyon et al. (1999)'s Theorem 2. We verify these assumptions in Lemma 1. The convergence of $\{\mathbf{x}_k\}_{k=1}^{\infty}$ stated in our Theorem 1 follows from applying their Theorem.

Lemma 1

Assume that assumptions (A1-A4) hold and $0 < \eta_0 < \eta_1 < \eta_1 \infty$. *Then:* (SA0) *w.p.1, for all* $k \in N^* = \{1,2 \cdots\}$, $\mathbf{x}_k$ *belongs to an open subset* $\mathscr{X} \subset R^p$; (SA1) $\{\gamma_k\}_{k=1}^{\infty}$ is a decreasing sequence of positive numbers such that $\sum_{k=1}^{\infty}\gamma_k=\infty$; (SA2) $h(\cdot)$ *is continuous on* $\mathscr{X}$ and there exists a continuously differentiable function $V:\mathscr{X} \to R$ *such that*: (i) *for all* $\mathbf{x} \in \mathscr{X}$, $F(\mathbf{x}) = \langle \frac{\partial V(\mathbf{x})}{\partial x}, h(\mathbf{x})\rangle \leq 0$; (ii) $int\{V(\mathscr{L}_F)\}=\varnothing$, *where* $\mathscr{L}_F := \{\mathbf{x} \in \mathscr{X}: F(\mathbf{x})=0\}$; (SA3) *w.p.1,* $clos(\{\mathbf{x}_k\}_{k=1}^{\infty})$ is a compact subset of R^p; (SA4) *w.p.1,* $\lim_{K\to\infty}\sum_{k=1}^{K}\gamma_k e_k(\mathbf{x}_k)$ *exists and is finite.*

Proof of Lemma 1

The smoothness properties satisfied by the kernel $K(\cdot)$ imply that $\sup_{\mathbf{x}\in R^p}|\bar{A}_k(\mathbf{x})| < \infty$ and $\sup_{\mathbf{x}\in R^p}|\bar{A}_k(\mathbf{x})| < \infty$, $k = 1, 2, \cdots$. It is also clear from the definition of b_{k+1}^{η} that $0<\eta_1^{-1} \leq C(\mathbf{x}_k) \leq \eta_0^{-1}<\infty$. Hence, $\sup_{\mathbf{x}\in R^p}|C(\mathbf{x})| < \infty$ for every $\eta_0 > 0$. Also, the jumps of $\mathbf{x}_k$ are almost surely finite.

Firstly, (SA0) is satisfied with $\chi = R^p$, and (SA1) is implied by Assumption (A1) of our Theorem 1. Define next $V(\mathbf{x}) = -\hat{f}(\mathbf{x})$. Then, for every $\mathbf{x} \in R^p$, $\partial V(\mathbf{x})/\partial\mathbf{x} = -2\bar{A}(\mathbf{x})$, and $F(\mathbf{x}) = \langle \frac{\partial V(\mathbf{x})}{\partial \mathbf{x}}, h(\mathbf{x})\rangle = -2C(\mathbf{x}) \| \bar{A}(\mathbf{x}) \|^2 \leq 0$, which establishes (SA2i). Let $\mathscr{L}_{\nabla V} := \{\mathbf{x} \in \mathscr{X}: \frac{\partial V(\mathbf{x})}{\partial \mathbf{x}}=0\}$. It follows from Assumption (A.2) that $V(\cdot)$ is p times continuously differentiable. Sard's Theorem implies that $V(\mathscr{L}_{\nabla V})$ has Lebesgue measure 0 in R^p. To show that $V(\mathscr{L}_F)$ also has Lebesgue measure 0, it suffices to prove that $\mathscr{L}_F=\mathscr{L}_{\nabla V}$. This identity holds because $F(\mathbf{x}) = -\frac{C(\mathbf{x})}{2}\|\frac{\partial V(\mathbf{x})}{\partial \mathbf{x}}\|^2$ and because $\sup_{\mathbf{x}\in R^D} C(\mathbf{x}) > 0$ when η_0. Hence, we have verified (SA2ii). In order to prove that (SA3) holds true, recall that the jumps of $\{\mathbf{x}_k\}_{k=1}^{\infty}$ are almost surely finite. Hence, $clos\left(\{\mathbf{x}_k\}_{k=1}^{\infty}\right)$ is almost surely a compact subset of R^p if and only if limp sup $\|\mathbf{x}_k - \mathbf{x}_0\}\}^2 < \infty$. Write $c_0 = \sup_{u \in \mathrm{R}^+} K'(u^2)u$ and $\pi^{\min}{}_{i=1,\ldots,n;k=1,2\ldots}\ \boldsymbol{\pi}_{ik}$. Then:

$$\begin{aligned}\|\mathbf{x}_k - \mathbf{x}_0\|^2 \leq \sum_{l=1}^{k}\|\mathbf{x}_l - \mathbf{x}_{l-1}\|^2 \quad &\leq \sum_{l=1}^{k}\frac{\gamma_l^2}{\eta_0^2}\left\|\frac{1}{n}\sum_{i=1}^{n}\frac{z_{i,2k}}{\pi_{i,2k}}h_i^{-(p+2)}K'\left(\left\|\frac{\mathbf{x}_l-\mathbf{y}_i}{h_i}\right\|^2\right)(\mathbf{y}_i - \mathbf{x}_l)\right\|^2\\ &\leq \frac{1}{v_{min}^{2p}n^2\eta_0^2\pi^{min}}\sum_{l=1}^{k}\gamma_l^2 \sup_{i=1,\ldots,n}\left\{K'\left(\left\|\frac{\mathbf{x}_l-\mathbf{y}_i}{h_i}\right\|^2\right)^2\left\|\frac{\mathbf{y}_i-\mathbf{x}_l}{h_i}\right\|^2\right\}\\ &\leq \frac{c_0^2}{v_{min}^{2p}n^2\eta_0^2\pi^{min}}\sum_{l=1}^{k}\gamma_l^2.\end{aligned}$$

Assumption (A.1) and the facts that $n_l \geq 1$, $c_0 < \infty$, $v_{\min} > 0$, $\eta_0 > 0$ and $\pi^{\min} > 0$ imply $\sup_{k=1,\ldots,\infty}\|\mathbf{x}_k - \mathbf{x}_0\|^2 < \infty$, hence lim sup $\|\mathbf{x}_\mathrm{k} - \mathbf{x}_0\|^2$ establishing (SA3). For the final step of the proof, notice that ($\mathbf{x}_k$, b_k) is $\mathscr{F}_k$-measurable and $E\left\{e_{k+1}(\mathbf{x}_k)\,|\mathscr{F}_k\right\} = 0$ (a.s.). Moreover

$$E\left\{\|\sum_{k'=1}^{l}\gamma_{k'+1}e_{k'+1}(\mathbf{x}_{k'})\|^2\right\} \leq \sum_{k'=1}^{k}\gamma_{k'+1}^2 E\left\{\left\|\frac{\bar{A}_{k'+1}(\mathbf{x}_{k'})}{b_{k'+1}^{\eta}}\right\|^2\right\} \leq \sum_{k'=1}^{k}\frac{\gamma_{k'+1}^2}{\eta_0^2}E\left\{\|\bar{A}_{k'+1}(\mathbf{x}_{k'})\|^2\right\}.$$

Since $\sup_{\mathbf{x} \in R^D}|\bar{A}_{k'+1}(\mathbf{x})| < c_1 < \infty$, we deduce that $E\left\{\|\sum_{k'=1}^{k}\gamma_{k'+1}e_{k'+1}(\mathbf{x}_{k'})\|^2\right\} < \infty$. Hence, the sequence $\left\{\sum_{k'=1}^{k}\gamma_{k'+1}e_{k'+1}(\mathbf{x}_{k'})\right\}_{k=1}^{\infty}$ defines a zero-mean, square-integrable martingale and adapted to the filtration $\mathscr{F}$. Moreover, we have almost surely that:

$$\sum_{k=0}^{\infty}E\left\{\|\gamma_{k+1}e_{k+1}(\mathbf{x}_k)\|^2|\mathscr{F}_k\right\} \leq \sum_{k=0}^{\infty}\gamma_{k+1}^2 E\left\{\left\|\frac{\bar{A}_{k+1}(\mathbf{x}_k)}{b_{k+1}^{\eta}}\right\|^2|\mathscr{F}_k\right\} \leq \sum_{k=0}^{\infty}\frac{\gamma_{k+1}^2}{\eta_0^2}\sup_{\mathbf{x}\in R^p}E\left\{\|\bar{A}_{k+1}(\mathbf{x})\|^2|\mathscr{F}_k\right\},$$

which is finite. Finally, a Strong Law of Large Numbers for Martingales (Hall and Heyde 1980) entails that $\exists c_2 \in R{:}\sum_{k=1}^{K}\gamma_{k+1}e_{k+1}(\mathbf{x}_k)\xrightarrow{a.s.}c_2$, which establishes Lemma 1.

REFERENCES

Breiman L, Meisel W, Purcell E. Variable kernel estimates of multivariate densities. Technometrics. 1977; 19:135–144.

Carreira-Perpinan, M. Fast nonparametric clustering with Gaussian blurring mean-shift.; Proceedings of the 23rd International Conference on Machine Learning; 2006a. p. 153-160.

Carreira-Perpiñan MA. Acceleration strategies for Gaussian mean-shift image segmentation. I.E.E.E. Transactions on Computer Vision and Pattern Recognition. 2006b; 1:1160–1167.

Carreira-Perpiñan MA. Gaussian mean-shift is an em algorithm. I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence. 2007; 29:767–776.

Cheng Y. Mean shift, mode seeking, and clustering. I.E.E.E. on Pattern Analysis and Machine Intelligence. 1995; 17:790–799.

Comaniciu D, Meer P. Mean-shift: A robust approach toward feature space analysis. I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence. 2002; 24:603–619.

Comaniciu D, Ramesh V, Meer P. The variable bandwidth mean-shift and data-driven scale selection. 8th I.E.E.E. International Conference on Computer Vision. 2001; 1:438–445.

Delyon B, Juditsky A. Accelerated stochastic approximation. SIAM Journal of Optimization. 1993; 4:868–881.

Delyon B, Lavielle M, Moulines E. Convergence of a stochastic approximation version of the EM algorithm. The Annals of Statistics. 1999; 27:94–128.

Ester, M., Kriegel, HP., Sander, J., Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise.. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD); Portland Oregon. 1996. p. 226-231.

Everitt, BS. Cluster Analysis. Third Edition. New-York; Arnold: 1993.

Fashing M, Tomasi C. Mean shift is a bound optimization. I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence. 2005; 27:471–474.

Fraley C, Raftery AE. Model-based clustering, discriminant analysis, and density estimation. Journal of the American Statistical Association. 2002; 458:611–631.

Fukunaga K, Hostetler LD. The estimation of the gradient of a density function, with application in pattern recognition. I.E.E.E. Transaction on Information Theory. 1975; 21:32–40.

Gan, G., Ma, C., Wu, J. Data Clustering: Theory, Algorithms, and Applications. SIAM; Philadelphia: 2007.

Georgescu B, Shimshoni I, Meer P. Mean shift based clustering in high dimensions: a texture classification example. Computer Vision (Proceedings of the Ninth IEEE International Conference on). 2003; 1:456–463.

Hall, P., Heyde, CC. Martingale Limit Theory and its Application. Academic Press; New York: 1980.

Hartigan, JA. Clustering Algorithms. John Wiley & Sons; New York: 1975.

Kesten H. Accelerated stochastic approximation. Annals of Mathematical Statistics. 1958; 29:41–59.

Klemelä, J. Smoothing of Multivariate Data: Density Estimation and Visualization. John Wiley & Sons; New York: 2009.

Kriegel HP, Kröger P, Sander J, Zimek A. Density-based clustering. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 2011; 1:231–240.

Kushner, HJ., Yin, G. Stochastic Approximation and Recursive Algorithms and Applications. second edition. Springer; New York: 2003.

Li J, Ray S, Lindsay BG. A nonparametric statistical approach to clustering via mode identification. Journal of Machine Learning Research. 2007; 8:1687–1723.

Neal, RM., Hinton, GE. A view of the EM algorithm that justifies incremental, sparse, and other variants.. In: Jordan, MI., editor. Learning in Graphical Models. MIT Press; 1998. p. 355-368.

Pflug, GC. Optimization of Stochastic Models. Kluwer Academic Publishers; Dordrecht, Netherlands: 1996.

Ruppert, D. Tech. Report No 781, School of Operation Research and Industrial Engineering. Cornell University; New York: 1988. Efficient estimators from a slowly convergent Robbins-Monro process..

Robbins H, Monro S. A stochastic approximation method. The Annals of Mathematical Statistics. 1951; 22:400–407.

Scott, DW. Multivariate Density Estimation: Theory, Practice and Visualization. John Wiley & Sons, Inc; New York: 1992.

Silverman, BW. Density Estimation for Statistics and Data Analysis. Chapman & Hall/CRC; Boca Raton: 1998.

Stuetzle W. Estimating the cluster tree of a density by analyzing the minimal spanning tree of a sample. Journal of Classification. 2003; 20:25–47.

Thompson, SK. Sampling. John Wiley & Sons, Inc.; New York: 1992.

Wand, MP., Jones, MC. Kernel Smoothing. Chapman & Hall/CRC; Boca Raton: 1995.

Wishart, D. Mode Analysis: a generalization of nearest neighbor which reduces chaining effects.. In: Cole, AJ., editor. Numerical Taxonomy. Academic Press; 1969. p. 282-311.

Yuan X-T, Hu B-G, He R. Agglomerative mean-shift clustering. I.E.E.E. Transactions on Knowledge and Data Engineering. 2012; 24

Yuan X-T, Li SZ. Stochastic gradient kernel density mode-seeking. Proc. I.E.E.E. Int. Conf. Computer Vision and Pattern Recognition. 2009

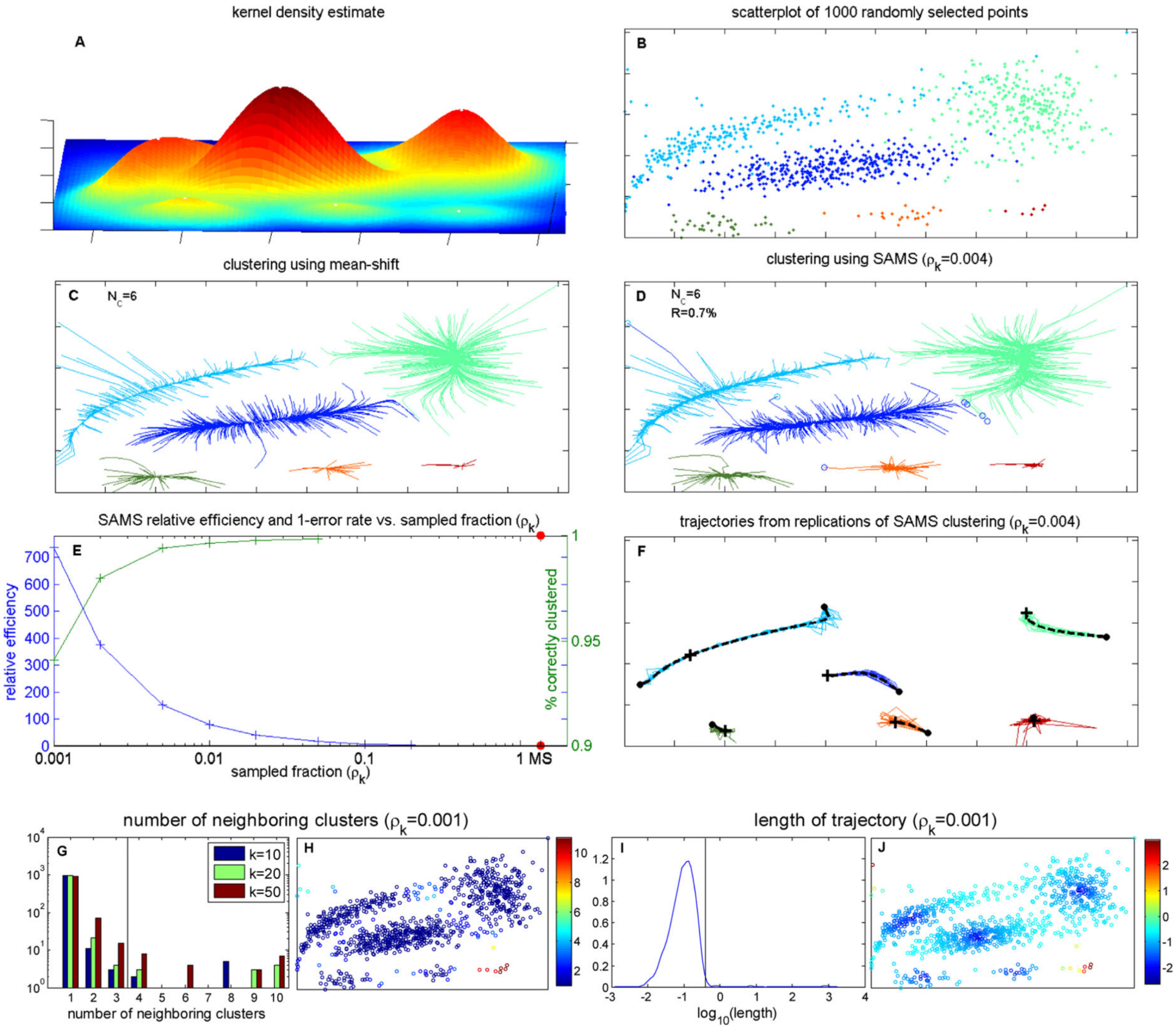


Figure 1.

Comparison of mean-shift (MS) and SAMS clustering. Kernel density estimate (A). One thousand randomly selected points from the original data set to be clustered (B). Trajectories obtained using MS (C) and SAMS ($\rho_k = 0.004$) (D). Points incorrectly clustered by SAMS (taking MS as gold standard) are circled. Average time to run an iteration using MS and SAMS for varying $\rho \in [0.001,1]$ compared in (E). SAMS ran multiple times started from the same initial points to assess its stability (F); dashed lines show mean-shift trajectories. Bottom panels: SAMS ran with $\rho_k = 0.001$. Two metrics were used to locate incorrectly clustered points: the diversity of clusters in the k-nearest neighbors to a point for $k = 10, 20, 50$ (G); the length of the trajectory (L^2-distance) the point traveled during clustering (I). Points with ≥ 4 clusters in the k-nearest neighbors ($k = 20$) or a trajectory length with log value ≥ −0.4 were reclustered, decreasing the clustering error rate slightly from ~ 0.059 to 0.039.

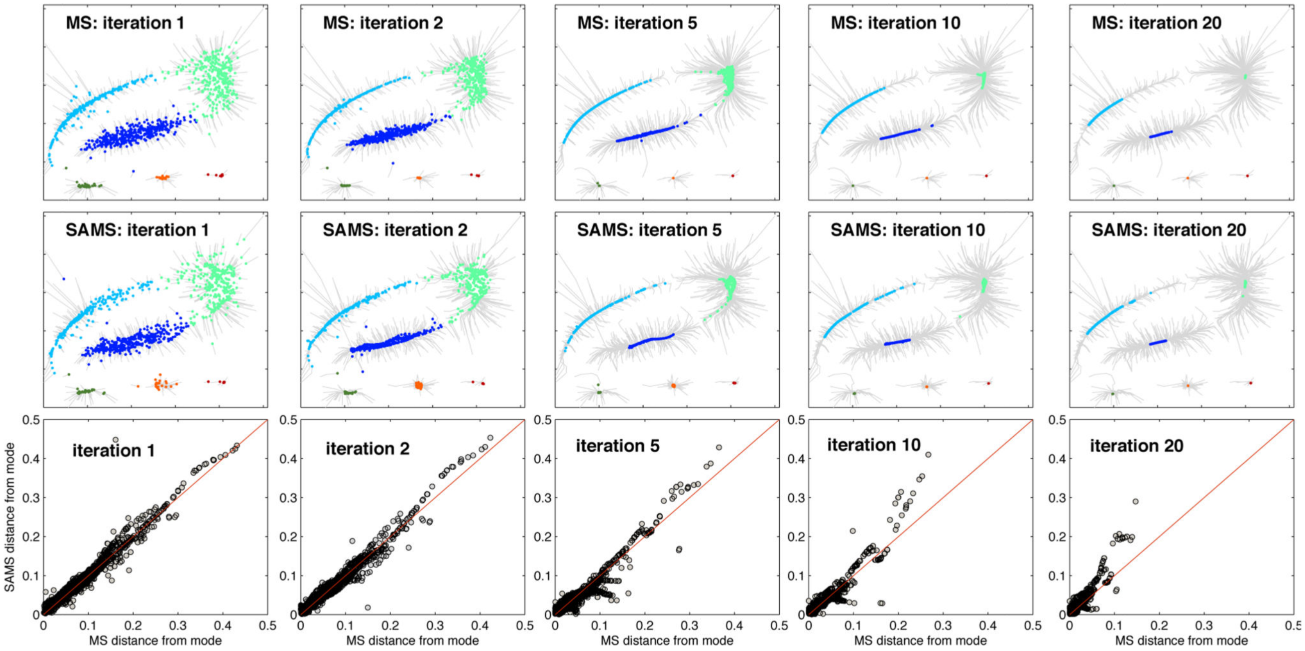
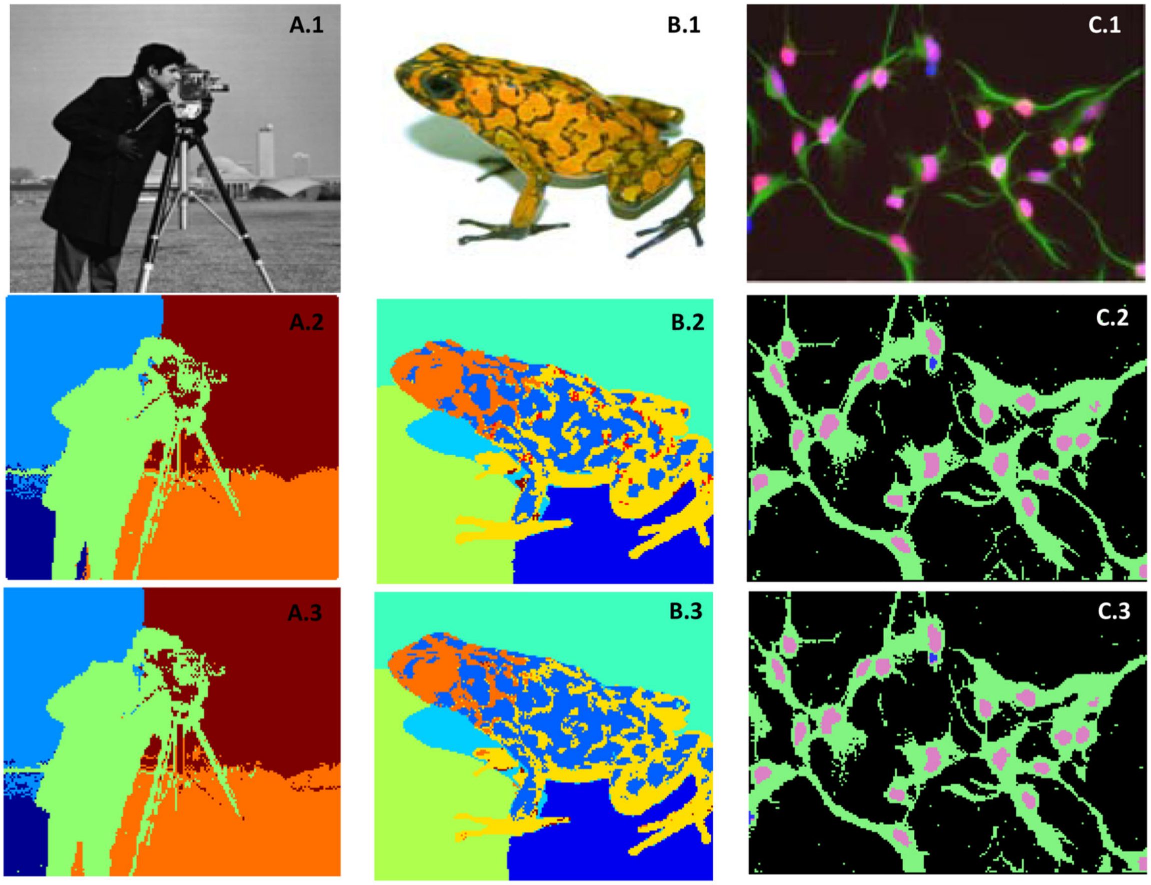


Figure 2.

Empirical comparison of the rate of convergence of mean-shift (MS) and SAMS (ρ_k = 0.4%). Each dot shows the location of a data point after 1, 2, 5, 10, and 20 iterations (left to right) of MS (top) and SAMS (middle). Points were color coded based on cluster assignment. The grey lines show the trajectories generated by each algorithm. Bottom: distance to the final mode for SAMS vs. MS. The results indicate that the two algorithms convergenced at virtually identical rates for most points. The computational cost per iteration was considerably smaller with SAMS.



A.4

method	MS	SAMS (mean (SD))
ρ	1	0.004
N_c	5	5.2 (0.39)
RE	1	151.0 (0.70)
R	0	0.019 (0.011)
Size	256x256	
α_1, α_2	0.14, 0.50	

B.4

method	MS	SAMS (mean (SD))
ρ	1	0.001
N_c	9	10.6 (0.60)
RE	1	594.5 (0.90)
R	0	0.057 (0.014)
Size	1000x1000x3	
α_1, α_2	0.11, 0.12	

C.4

method	MS	SAMS (mean (SD))
ρ	1	0.01
N_c	4	4.3 (0.91)
RE	1	59.0 (0.09)
R	0	0.0038 (0.0018)
Size	484x620x3	
α_1, α_2	0.03, 0.35	

Figure 3.

Image segmentation. Three images (A.1, B.1, C.1) were segmented with mean-shift (MS) (A.2, B.2, C.2) and SAMS (A.3, B.3, C.3). For each image, the sample fraction (ρ), the number of clusters identified for each method (N_c) and the classfication error rate (R) of the SAMS algorithm compared to the MS algorithm are included in the summary table (A.4, B. 4, C.4). For SAMS, 100 replications of clustering was performed so R and N_c are reported as a mean and standard deviation. The relative effiency (RE), measured as times faster than MS, and the values of the tuning parameters (α_1, α_2) are also included in the tables. (image sources: B.1: Santos J.C. (2009), PLOS Biol.; C.1: Davies J. et al. (2008), J. Biol. 7:24).