Stat Comput (2007) 17:71–80
DOI 10.1007/s11222-006-9010-y

# Clustering via nonparametric density estimation

**Adelchi Azzalini · Nicola Torelli**

Received: August 2005 / Accepted: September 2006 / Published online: 3 February 2007


**Abstract** Although Hartigan (1975) had already put forward the idea of connecting identification of subpopulations with regions with high density of the underlying probability distribution, the actual development of methods for cluster analysis has largely shifted towards other directions, for computational convenience. Current computational resources allow us to reconsider this formulation and to develop clustering techniques directly in order to identify local modes of the density. Given a set of observations, a nonparametric estimate of the underlying density function is constructed, and subsets of points with high density are formed through suitable manipulation of the associated Delaunay triangulation. The method is illustrated with some numerical examples.

**Keywords** Cluster analysis · Delaunay triangulation · Voronoi tessellation · Nonparametric density estimation · Kernel method

## 1 Clusters as regions of high density

### 1.1 Introduction

Consider the problem of grouping a set of data, represented by $d$ quantitative variables observed on $n$ subjects, to form a certain number of data clusters. Among the various alternative formulations of the concept of 'cluster' itself, one is based on the definition of a cluster as a region of high density of the underlying density function. This approach goes back to Hartigan (1975, p. 205), who stated:

> "Clusters may be thought of as regions of high density separated from other such regions by regions of low density".

Although this idea is partly explored by Hartigan (1975, Chapter 11), particularly by showing how the above concept generates a hierarchical structure of 'high-density clusters', which therefore form a tree, the mainstream development of that book as well as most of the related literature hinges on the concept of distance, in various specifications. Among the few papers pursuing Hartigan's idea, one important contribution is that of Wong and Lane (1983), who use the $k$-NN density estimate as the basis for the clustering procedure. However, the tree structure of the clusters is not examined in any detail.

One reason for this preference of the mainstream clustering literature in favour of distance based methods is arguably computational convenience. However, current computational resources allow us to reconsider Hartigan's formulation, and develop it into a viable methodology. Therefore, the quantity of interest will be the density value at each observation, instead of the distance between observations. This perspective does not imply that the method is completely independent of distances, as explained in more detail later on, but there is a shift of target in the driving criterion.

The idea of using a density estimate as the basis for clustering methods has been adopted in various recent papers, in both the statistical and the machine learning literature. Within the former group, Stuetzle (2003) presents a method closely related to the minimum spanning tree of a sample, which exploits the connection between this tree and nearest neighbour density estimation. More directly related to density estimation are the papers of Cuevas et al. (2000, 2001),

A. Azzalini (✉)
Dipartimento di Scienze Statistiche, Università di Padova, Italy
e-mail: azzalini@stat.unipd.it

N. Torelli
Dipartimento di Scienze Economiche e Statistiche, Università di Trieste, Italy
e-mail: nicola.torelli@econ.units.it

who attempt to find connected sets of the form $\hat{f} > c$ for a single level $c$ of the density. In the machine learning literature, a variant of single-level mode analysis is the method of Ester et al. (1996). A method which extends the previous one is the OPTICS algorithm by Ankerst et al. (1999), which leads to the construction of a tree of clusters; however, as pointed out by Stuetzle (2003), "the idea behind the algorithm is hard to understand".

In the rest of this section we present in a somewhat more detailed manner the idea of high-density clusters for a given density function $f(x)$, and examine some of the associated formal properties, building on a preliminary formulation of these ideas developed by Rosolin et al. (2003). These concepts are subsequently given a sample analogue, based on a nonparametric estimate $\hat{f}(x)$ of $f(x)$ and the use of a Delaunay triangulation of the observed points. Connected subsets of the triangulation represent the core parts of the clusters in our methodology. The second stage of the procedure allocates the remaining observations that are not part of the cluster cores.

### 1.2 Modes of density and some properties

Consider a $d$-dimensional density function $f(x)$, $x \in \mathbb{R}^d$. To ease exposition, assume that $f$ is differentiable everywhere; this assumption may be relaxed, with some technical complications.

For any positive constant $c$, we section the function $f$ at level $c$, thus splitting $\mathbb{R}^d$ into two sets, one having density up to $c$, and the other with density above $c$. The latter set may be connected or not. If it is not connected, then we have detected two or more regions of high density. The plot in Fig. 1 illustrates this idea in a simple case with $d = 1$, and a specific choice of $c$ leading to two connected sets. Clearly, the number of connected sets varies with $c$.

While the definition of $R(c)$ and the identification of the number of its connected components does not change with the dimension $d$, the actual evaluation of the number of components and their detection is increasingly difficult as $d$ increases above 1.

To be more specific, define

$$R(c) = \{x : x \in \mathbb{R}^d, f(x) > c\}, \quad 0 \le c \le \max f, \tag{1}$$

and denote its probability by $p_c = \int_{R(c)} f(x)\,\mathrm{d}x$. In the left plot of Fig. 1, $p_c$ is represented by the shaded area.

Also define the inverse function $c_p$, which selects the level $c$ so that $\mathbb{P}\{R(c_p)\} = p$. Note that $R(c_1)$ is the entire support of the distribution and $R(c_{0+})$ is the point or the set of points of maximal density.

For any choice of level $c$, the level set $R(c)$ consists of a certain number $m$ of connected components. Since at the same time $c$ is associated with a probability $p_c$, there is a correspondence between $p$ and the associated number $m$ of components of $R(c_p)$. We define a step function $m(p)$ which gives the number of connected components of $R(c_p)$ as $p$ varies in (0, 1). For $p = 0$ and $p = 1$, we define $m(p) = 0$. The right plot of Fig. 1 shows the function $m(p)$ corresponding to the density on the left.

The function $m(p)$ enjoys some useful properties related to the modes of the density function, which is why we refer to it as the 'mode function'. To fix notation, denote by $\tilde{x}_1, \ldots, \tilde{x}_M$ the set of modes of $f$; without loss of generality, assume that the points $\tilde{x}_j$ are distinct and $f(\tilde{x}_1) \ge f(\tilde{x}_2) \ge \cdots \ge f(\tilde{x}_M)$. To avoid technical complications in the subsequent discussion, we assume that there

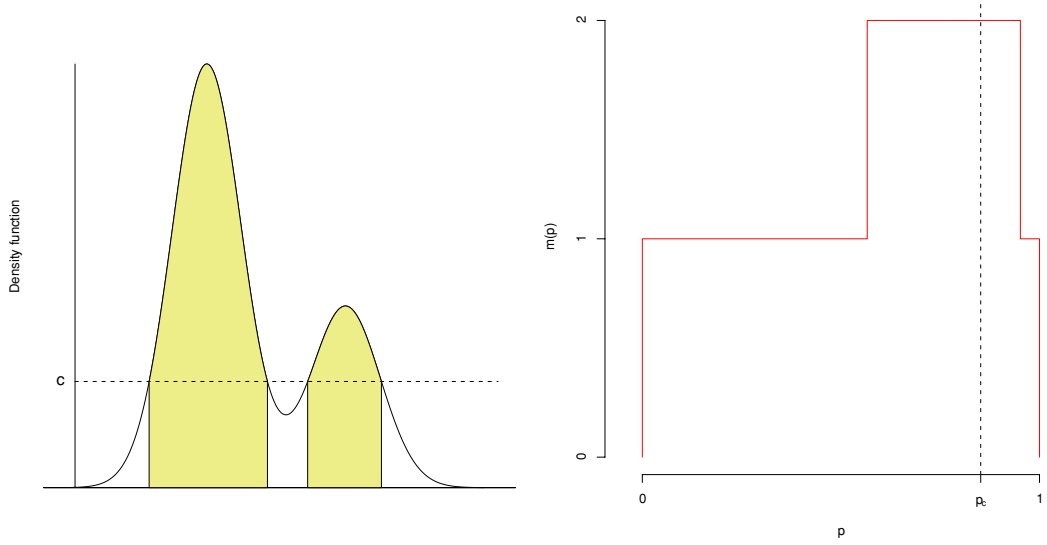


**Fig. 1** Density function and set $R(c)$ for a given $c$ (*left*) and corresponding function $m(p)$ (*right*)

are no saddle points at which $f(x)$ takes on the same value as any $f(\tilde{x}_j)$. The following properties may then be obtained immediately:

1. For all $p \in (0, 1)$, we have $m(p) \geq 1$.
2. If $f$ is unimodal, that is, $M = 1$, then $m(p) \equiv 1$ for $0 < p < 1$.
3. The total number of increments of $m(p)$, counted with their multiplicity, is equal to the number of modes, $M$; a similar statement holds for the number of decrements.
4. The increment of $m(p)$ at a given point $p$ equals the number of modes whose ordinate is $c_p$.

The increments of $m(p)$ correspond to the appearance of one or more modes of $f$, whereas the decrements correspond to the fusion of two or more groups associated with existing modes. As $c$ varies, the connected components of $R(c)$ generate a hierarchical structure which may be represented in the form of a tree. Since these facts have been pointed out by Hartigan (1975, Section 11.13) and further discussed by Stuetzle (2003), we do not replicate the details here. However, the tree structure is illustrated in the subsequent numerical examples.

As an aid to interpretation of the tree function, note that, if $p'$, $p''$ (where $p' < p''$) are discontinuity points of $m(p)$, and $m(p)$ is constant in the interval $(p', p'')$, then

$$p'' - p' = \int_{R(c_{p''}) \setminus R(c_{p'})} f(x)\,\mathrm{d}x.$$

Hence, the difference $p'' - p'$ gives an indication of how prominent the modes of level $c_{p'}$ are with respect to the others, where the idea of prominence refers not to the height of the mode, but to the probability mass associated with the regions $R(\cdot)$. Figure 2 illustrates this fact for the two non-boundary discontinuity points of the right panel of Fig. 1, by showing with two levels of grey the areas corresponding to $p'$ and $p'' - p'$.

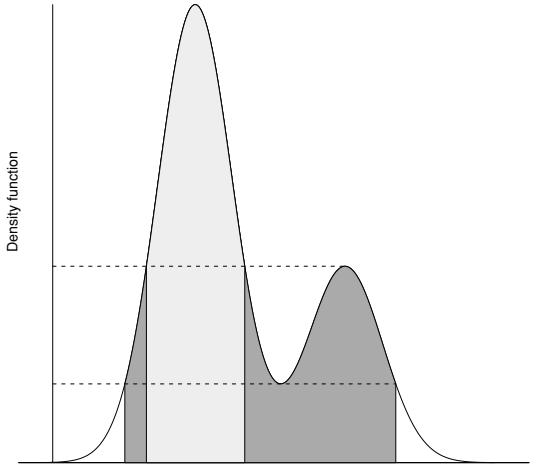


**Fig. 2** Density function and areas associated with two discontinuity points of the mode function in right panel of Fig. 1

## 2 Spatial tessellation and connected components

### 2.1 Voronoi tessellation and Delaunay triangulation

For the subsequent development we need to recall briefly a few basic concepts of spatial tessellation; for an extensive treatment, readers are referred to the treatise of Okabe et al. (1992).

Given a set $S$ of points $x_1, x_2, \ldots, x_n$ of $\mathbb{R}^d$, the Voronoi tessellation is defined as the partition of $\mathbb{R}^d$ formed by $n$ sets $V(x_1), \ldots, V(x_n)$ such that, for a generic point $x$ of $\mathbb{R}^d$, $x \in V(x_i)$ if $x_i$ is the closest element of $S$. Identification of the closest point requires the specification of a distance function; this is usually assumed to be the Euclidean distance, and it is the one adopted here.

The regions $V(x_1), \ldots, V(x_n)$ are polyhedra in $\mathbb{R}^d$, possibly unbounded. The facets of these polyhedra are formed by polygons of $\mathbb{R}^{d-1}$, and their edges by lines in $\mathbb{R}^{d-2}$.

From the Voronoi tessellation, a second tessellation can be formed as follows. Any two elements $x_i$ and $x_j$ of $S$ are connected by a line segment if the corresponding elements $V(x_i)$ and $V(x_j)$ of the Voronoi tessellation share a portion of their boundary facets. These segments partition the space into a set of new polyhedra. In many cases, these are simplices, i. e., polyhedra formed by $d + 1$ vertices in $\mathbb{R}^d$; otherwise, they may be broken down arbitrarily into simplices. The result is the associated Delaunay triangulation; the term triangulation relates to the fact that the simplices are triangles when $d = 2$. From a computational viewpoint, the Delaunay triangulation may be obtained directly, i.e., without first constructing the Voronoi tessellation.

These concepts are illustrated in the left panel of Fig. 3, which displays the Voronoi tessellation and Delaunay triangulation for a set of points in $\mathbb{R}^2$.

### 2.2 Sample analogues of $R(c)$

Assume that a random sample $S = \{x_1, x_2, \ldots, x_n\}$ is drawn from a distribution with density $f(x)$, with $x \in \mathbb{R}^d$. From this sample, a nonparametric estimate $\hat{f}(x)$ of $f(x)$ is obtained. For subsequent development of the method, it does not really matter which specific estimator is adopted, provided that $\hat{f}$ is positive and finite, at least at the observed points. Clearly, it is sensible to adopt a method with good statistical properties; among these, an important requirement is consistency of the estimation method as $n \to \infty$, for the reason explained at the end of this section. The corresponding natural choice for estimating $R(c)$ is given by

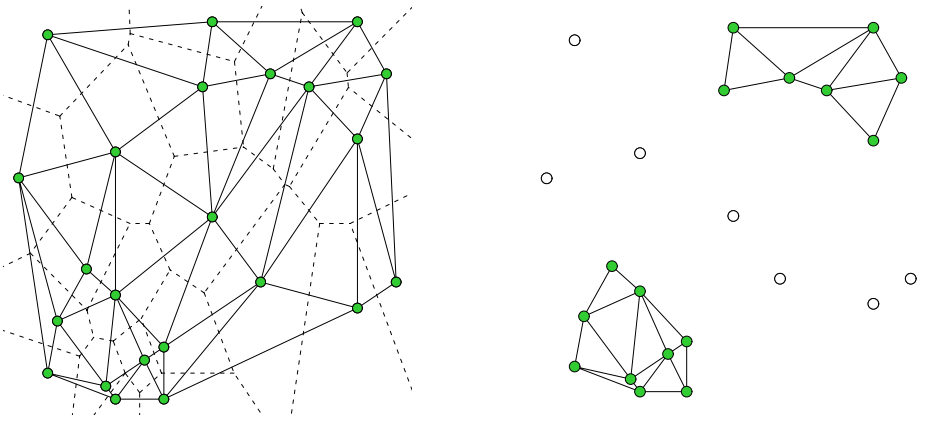

**Fig. 3** The left plot displays an example of Voronoi tessellation (dashed lines) for a set of points when $d = 2$, and superimposed Delaunay triangulation (continuous lines). The right plot removes edges of some points from the original Delaunay triangulation, keeping points with $\hat{f} > c$ for some threshold $c$

$$\hat{R}(c) = \{x : x \in \mathbb{R}^d, \hat{f}(x) > c\}, \quad 0 \le c \le \max \hat{f}.$$

Since we are interested only in allocation of the data points, and not of all points in $\mathbb{R}^d$, it is natural to restrict attention to elements of $\hat{R}(c)$ corresponding to the data points, and then to consider only

$$S(c) = \{x_i : x_i \in S, \hat{f}(x_i) > c\}, \quad 0 \le c \le \max \hat{f}, \qquad (2)$$

with associated relative frequency

$$\hat{p}_c = |S(c)|/n$$

where $|\cdot|$ denotes the cardinality of a set. It is easy to show that $\hat{p}_c$ converges almost surely to $p_c$, using theoretical results ensuring that, under mild conditions, $\hat{f}$ converges uniformly to $f$ as $n \to \infty$; for the latter fact, see for instance Nadaraya (1965) and Devroye and Wagner (1980).

We recall the argument presented by Wong and Lane (1983) concerning the property of 'strong set consistency' for set $\hat{R}(c)$ which is shown to be a consistent estimate of $R(c)$ as $n \to \infty$ provided that a uniformly strongly consistent estimate $\hat{f}$ is adopted. As a consequence, the associated tree structure of the connected components of $R(c)$ is estimated consistently.

### 2.3 Search for connected components

To find the empirical analogue of the mode function $m(p)$, we must define the sample analogue of the connected components of $R(c)$. To this end, we consider the Delaunay triangulation of the sample points after removing the sample points $x_i \notin S(c)$ and all edges with at least one vertex among these points. This step is illustrated in the right-hand panel of Fig. 3, where the open circles represent observations with density at or below a given level $c$.

After removal of edges, some groups of points are connected by a sequence of remaining edges. We call these groups the connected components of $S(c)$. In most cases, each group consists of points so that each pair of points in the group shares one facet of the original Delaunay triangulation, and the union of these facets forms a polyhedron in $\mathbb{R}^d$. In the case of Fig. 3, we obtain two connected components and two polyhedra.

The intuitive idea behind this scheme is that, for large $n$, each polyhedron approximates a corresponding connected component of the unobserved set $\hat{R}(c)$. The idea has some connection to the 'slice plot' put forward by Bowman and Foster (1993), for selecting groups of observations formed of a given fraction of the sample and having maximal estimated density.

### 2.4 Empirical mode functions and cluster tree

The above steps must be performed for a range of values of $c$, ideally all $0 < c < \max \hat{f}$, but in practice a finite grid of values is considered. Alternatively, a set of equally spaced values of $p$ $(0 < p < 1)$ may be scanned, yielding effectively the same outcome if the grid is sufficiently fine; this is the route adopted in the following.

For each selected value of $p$, the set $S(c_p)$ is obtained and the number of its connected components is determined, as defined above, obtaining the value of empirical mode function $\hat{m}(p)$.

As already noted at the end of Section 1.2, the increments of mode function $\hat{m}(p)$ as $p$ ranges from 0 to 1 correspond to the appearance of new clusters and, *vice versa*, the decrements correspond to the merging of clusters. Specifically, a value of $p$ corresponding to an increment of $\hat{m}(p)$ denotes the 'birth' of as many clusters as the increment of $\hat{m}(p)$, and the elements of $S$ comprising these clusters may be identified. Similarly, if $p^*$ is a value of $p$ where $\hat{m}(p)$ decreases, two or more clusters are merging. In this case, comparison of the elements constituting $S_n(c')$ and $S_n(c_{p^*})$, where $c' < c_{p^*}$, allows us to detect which groups are merging at level $c_{p^*}$ and which are their components. Proceeding in this fashion

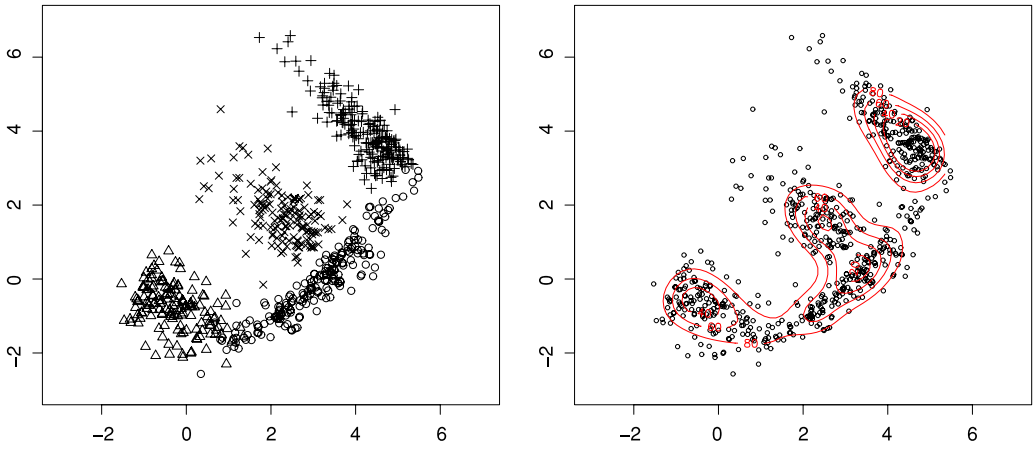


**Fig. 4** A sample of size 700 of simulated data from four sub-populations. Right plot superimposes contour levels of nonparametric kernel estimate of the density

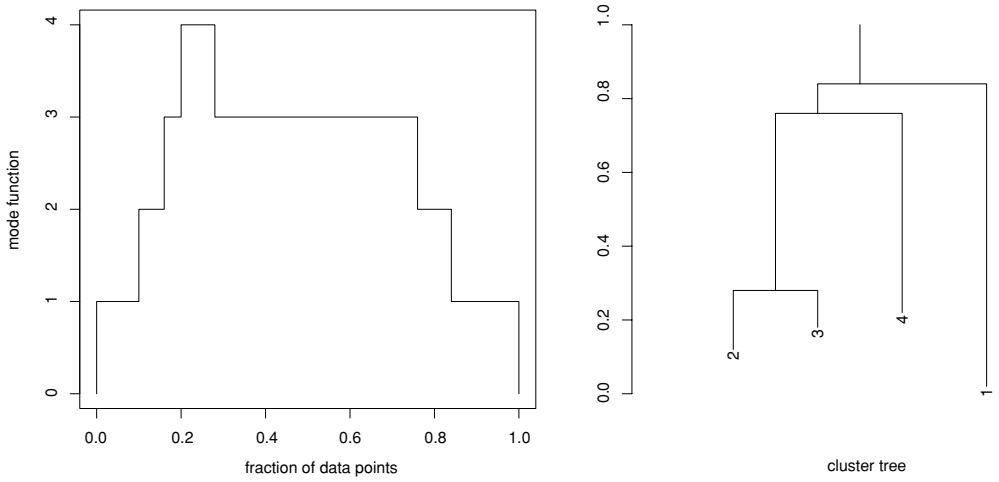


**Fig. 5** Simulated data: mode function (*left*) and corresponding cluster tree (*right*)

sequentially from $p = 0$ to $p = 1$, the whole tree structure of the clusters is identified.

To illustrate the above procedure, consider the two-dimensional data displayed in Fig. 4, left panel. There are 700 simulated data points sampled from a mixture of four subpopulations, whose sizes are 200, 150, 200 and 150, respectively. These subpopulations were chosen so that the presence of four groups is clearly visible, but at the same time there is a non-negligible overlap between some groups. The right panel of Fig. 4 also shows a nonparametric kernel estimate $\hat{f}$ of the density function. We defer until later discussion of the choice of the smoothing parameter.

From the estimate $\hat{f}$, mode function and cluster tree are obtained as described above. The result of this process is displayed in Fig. 5. The cluster tree reflects the original idea of Hartigan (1975), except that the vertical axis refers to the fraction of data points included, instead of density level. For each selected level $p$, the number of branches extending above $p$ indicates the number of modes with density above level $c_p$, and the corresponding number of connected components.

The construction of the cluster tree allocates a subset of observations to groups. The proportion of allocated points lies between the lowest and highest branching points of the tree. In the example examined here, this proportion is 52.4%, which is between the levels 28% and 84% in the right plot of Fig. 5. The left panel of Fig. 6 displays the four identified groups using four different graphical symbols, and simple dots to represent unallocated points. The numbering indicated on the plot agrees with the ordering of the mode levels, with 1 corresponding to the highest and 4 to the lowest mode.

## 2.5 Classification of unallocated points

The procedure described so far creates $M$ groups of points, which we call "cluster cores", and it leaves a number of points unlabeled. Allocation of the unlabelled points to existing

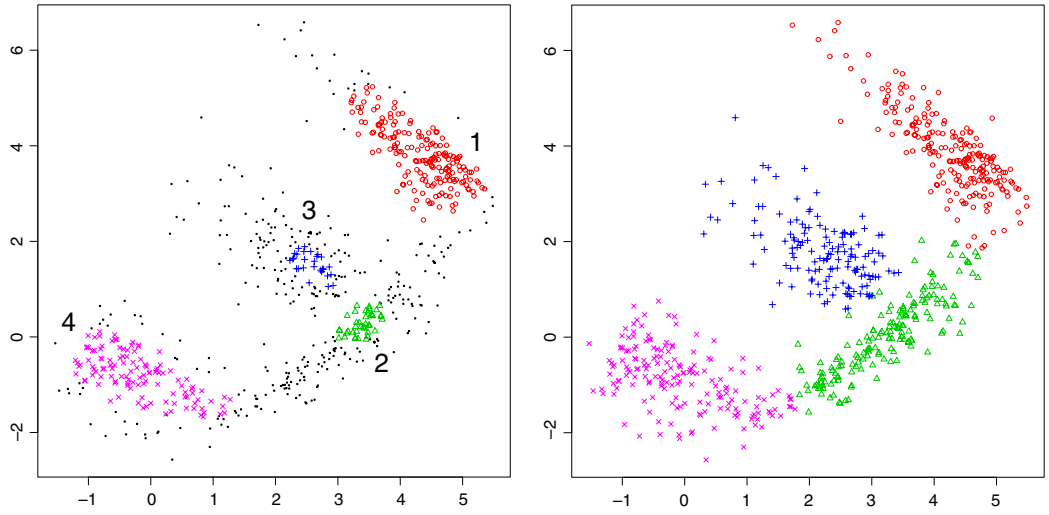


**Fig. 6** Simulated data: allocation of points after initial stage (*left*) and at final stage (*right*), with groups denoted as follows: $1 = \bigcirc$, $2 = \triangle$, $3 = +$, $4 = \times$

groups is essentially a classification problem, although of a rather peculiar type. The unusual aspect is that the unlabelled points are not positioned randomly in the $d$-dimensional space, but are inevitably on the outskirts of the $M$ existing groups.

There is a wide choice of classification methods. To remain within the same kind of approach followed so far, we focus on methods based on nonparametric density estimation. The basic idea is as follows: for an unallocated data point $x_0$, compute the estimated density $\hat{f}_j(x_0)$ based on the data already assigned to group $j$ (for $j = 1, 2, \ldots, M$), and assign $x_0$ to the group with highest $r_j(x_0) = \hat{f}_j(x_0)/\max_{k \neq j} \hat{f}_k(x_0)$. The actual implementation of this idea may take various forms:

(a) Estimate $M$ density functions $\hat{f}_j(\cdot)$ once and for all, and classify all unlabelled points using these estimates.
(b) Proceed sequentially: once a point $x_0$ has been assigned to group $j'$, say, re-estimate $\hat{f}_{j'}(\cdot)$ before allocating the next data point.
(c) In order to save computation, proceed in a block-sequential manner, allocating several points at a time, before updating the estimates $\hat{f}_j(\cdot)$.

All the above options may be combined with the use of some prior distribution $\pi_j$, $j = 1, \ldots, M$. In this context, this distribution can reasonably only be given by the relative frequencies obtained in the first stage. In this approach, the classification rule is based on comparison of $\hat{f}_j(\cdot)\,\pi_j$.

In the subsequent numerical work, we adopted the intermediate strategy described in (c), in the following manner. First, we sorted unallocated points according to ratios $r_j(x)$, and split into five blocks of equal size. Next, we allocated the points in the block with highest density to the existing groups. We then reestimated the density, followed by allocation of the second block of points, and so on. The right panel of Fig. 6 shows the final outcome after allocation of all points for the simulated data described above.

**Table 1** Main steps of the method

```
(1) Compute Delaunay Triangulation (DT).
(2) Compute f̂(x_i), for i = 1,...,n.
(3) For each p belonging to grid of points
    in (0,1),
  (a) remove points of low density (f̂(x_i) < c_p)
      from DT,
  (b) determine connected sets of retained
      points,
  (c) compute m(p).
(4) Build cluster tree and form the initial
    clusters.
(5) Allocate remaining points to these
    clusters.
```

The major steps of the method are summarised in Table 1.

## 3 Further aspects

### 3.1 Nonparametric density estimation

The method described so far is not linked to any specific method for nonparametric density estimation. The only restriction is that $\hat{f}(x_i) < \infty$ for $i = 1, \ldots, n$, which is satisfied by all estimation techniques known to us. Obviously, a method with good general properties should be used.

Among the many possible alternatives, we chose a kernel method with Gaussian kernel and constant smoothing parameter $h = (h_1, \ldots, h_d)^\top$. The critical issue is the choice of $h$. There is a vast specialised literature dealing with this problem, but it is aimed at minimizing the mean square error

or some related quantity. This is not the point for us, because the relevant question is how well $\hat{f}$ performs when used in our clustering method.

A choice of $h$ targeted to the specific problem would therefore be desirable. However, this is a project by itself and is not tackled here. In the present paper, we take a simpler route, adopting a standard, computationally inexpensive, procedure. Specifically, we choose $h$ resulting in the asymptotically optimal integrated mean squared error under the assumption of normality, which is known to be

$$h_j = \left(\frac{4}{(d+2)\,n}\right)^{1/(d+4)} \sigma_j\,, \tag{3}$$

where in practice the standard deviation $\sigma_j$ of the $j$-th variable ($j = 1, \ldots, d$) must be replaced by an estimate; see, for instance, Bowman and Azzalini (1997, p. 32). We adopted this choice both for the first estimate which produces the cluster tree and for the subsequent classification stage of unallocated points.

While this strategy may appear naive, it does produce sensible results in virtually all cases on which we have tested the method; this conclusion is based on the examples reported here plus some additional simulated data. Empirically, it was observed that it is often advantageous to shrink $h$ slightly towards zero; in the subsequent numerical work, we adopted a shrinkage factor of 3/4 as an overall reasonable choice.

### 3.2 Algorithmic and computational aspects

For the actual implementation of the method, the following software components were used. For the computation of the Delaunay triangulation, we used a public domain implementation of the 'Quickhull' algorithm by Barber et al. (1996), available at `http://www.qhull.org/download/`

This package was interfaced with R (R Development Core Team, 2004), which was adopted as the general computing environment for our work. To find the connected components of a graph, we used the R package `spdep`. The remainder of the code was written in R and Fortran-77.

The computational complexity of the various algorithmic components is as follows.

(i) Barber et al. (1996) state that the Quickhull algorithm for finding the convex hull of a set of $n$ points in $\mathbb{R}^d$ requires at most $O(n \log n)$ operations if $d \leq 3$, and $O(n^m/m!)$ where $m = \lfloor d/2 \rfloor$ for $d > 3$. Our numerical experiments using the publicly available implementation to obtain the Delaunay triangulation is that the computing time increases less than quadratically in $n$ for any fixed $d$, but it increases more than exponentially in $d$ for fixed $n$.

(ii) Computation of $\hat{f}(x)$ at the $n$ observed points requires $O(d\,n^2)$ operations.

(iii) The complexity of the 'depth-first search' algorithm implemented in the R package `spdep` is $O(n)$; this is to be multiplied by the size of the grid used to search over $p$.

(iv) Order $O(n)$ also holds for our portion of R code, which keeps track of membership lists.

These facts effectively impose some restrictions on the size of the problems to which our method can be applied. The most severe limitation is due to the combined effect of $d$ and $n$ on the Quickhull algorithm.

## 4 Examples

### 4.1 Four simulated groups of data

The result of our method applied to the data displayed in Fig. 4 is reported in Table 2 and compared with results produced by a few popular clustering methods, namely $k$-means and two variants of hierarchical clustering, complete linkage and Ward's method. We also tried other popular methods, such as hierarchical clustering with single linkage and average linkage, but their performance was inferior to those reported here.

For the density-based method, the smoothing parameter $h = (h_1, h_2)^\top$ was chosen as described in Section 3.1, with a

**Table 2** Simulated data: number of objects cross-classified by true partitions (row labels) and clusters obtained by density-based method and some alternatives (column labels); ARI denotes the adjusted Rand index

| | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| $k$-means (best outcome) | | | | |
| 1 | 160 | 26 | 14 | 0 |
| 2 | 0 | 150 | 0 | 0 |
| 3 | 0 | 0 | 200 | 0 |
| 4 | 22 | 0 | 0 | 128 |
| ARI = 0.787 | | | | |
| Hierarchical (complete linkage) | | | | |
| 1 | 125 | 68 | 7 | 0 |
| 2 | 0 | 150 | 0 | 0 |
| 3 | 0 | 0 | 131 | 69 |
| 4 | 124 | 1 | 0 | 25 |
| ARI = 0.471 | | | | |
| Hierarchical (Ward method) | | | | |
| 1 | 125 | 68 | 7 | 0 |
| 2 | 0 | 150 | 0 | 0 |
| 3 | 0 | 0 | 200 | 0 |
| 4 | 0 | 1 | 0 | 149 |
| ARI = 0.767 | | | | |
| Density-based method | | | | |
| 1 | 166 | 21 | 13 | 0 |
| 2 | 0 | 150 | 0 | 0 |
| 3 | 0 | 0 | 200 | 0 |
| 4 | 10 | 0 | 0 | 140 |
| ARI = 0.841 | | | | |

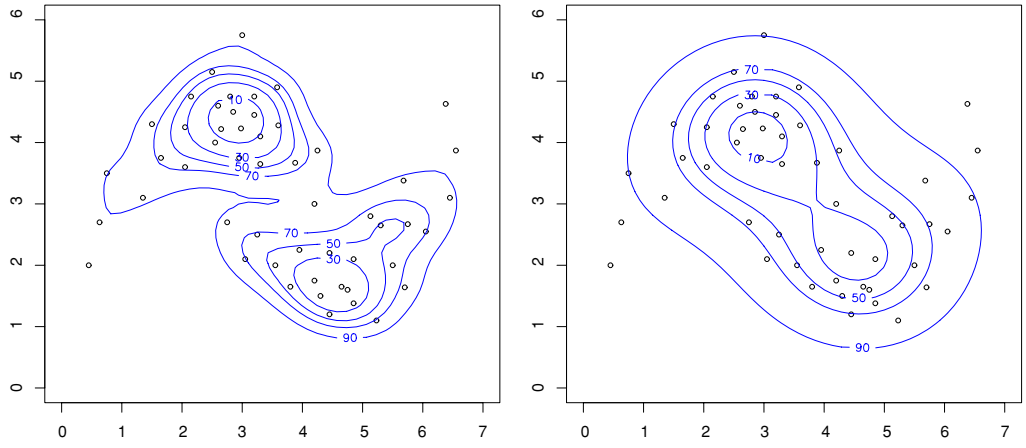


**Fig. 7** Two crescent-shaped clusters and contour levels of kernel density estimates for two choices of smoothing parameter

shrinkage factor of 3/4. While for the density-based method the number of clusters is selected by the procedure and turned out to have been estimated correctly at four, for the other methods this value was set by us. For $k$-means, the actual outcome varies with the initial centroids which are randomly selected, so that we ran the algorithm 20 times, with different starting values, and reported the best outcome.

To assess the performance of the various methods, we evaluated the adjusted Rand index, ARI, proposed by Hubert and Arabie (1985) and used among others by Stuetzle (2003) for comparing competing clustering techniques. The ARI gives a measure of agreement between two groupings, independent of the labelling of the groups. In this case, one of the groupings represents the true classification, so that the ARI value is a measure of performance of the clustering procedure. Higher values of ARI correspond to better performance.

### 4.2 Two crescent-shaped clusters

A classical challenge for a clustering procedure is identification of groups that are not linearly separable, such as the two crescent-shaped groups shown in Fig. 1 of Wong and Lane (1983). Since the exact numerical values were not available to us, we reconstructed their data as closely as possible by starting from the published plot.

The proposed method identifies the existence of two clusters for a wide range of the shrinkage factor introduced in Section 3.1. Specifically, we obtained two clusters for all values between 0.45 and 1.6. Figure 7 illustrates the effect of using multipliers 0.6 and 1.5 for $h$ on the shape of the contour lines of the estimated density. It is reassuring that the number of identified clusters remains fixed at two within such a wide range of values.

### 4.3 Olive oil composition

A more substantial example is provided by the data presented by Forina et al. (1983) and subsequently analysed by various authors to illustrate clustering techniques; see, for instance, Stuetzle (2003). The data represent eight chemical measurements on $n = 572$ specimens of olive oil produced in various areas of Italy. There are nine areas from which these specimens originate, but they are naturally grouped into three macro-areas: Centre-North, South, and Sardinia. The purpose of our analysis is to test whether the clustering algorithm based on the chemical measurements is able to reconstruct the geographical origin of the oil specimens.

Since the raw data are of compositional nature, totalling 10000, the additive log-ratio transform (ALR) was adopted, as advocated by Aitchison (1986). If $z_j$ $(j = 1, \ldots, 8)$ denotes the $j$-th chemical measurement, the ALR transform is

$$y_j = \log z_j / z_k, \qquad (j \neq k),$$

where $k$ refers to an arbitrary but fixed variable, whose choice is essentially irrelevant.

A practical complication in our case is that the $z_j$ do not exactly total 10000 in all cases, due to measurement errors. A second difficulty is the presence of some 0's, which is the value recorded when the actual measurement is below instrument sensitivity level. To overcome these problems, we added 1 to all raw data, and normalized them by dividing each number by the corresponding row sum $\sum_j (z_j + 1)$.

Since the resulting data matrix $y$ is of size $572 \times 7$, which is too large to be handled by the Quickhull algorithm, we considered the first five principal components, computed after scaling the variables. These five components account for 96% of total variability.

To estimate the density function, the smoothing parameter $h$ was chosen as described in Section 3.1, with a shrinkage factor of 3/4. The resulting mode function and cluster tree are shown in Fig. 8. It is reassuring that these plots indicate the existence of three groups, in agreement with the actual partition into three geographical macro-areas.

Table 3 reports the cross-classification frequencies of the actual geographical macro-areas and of the results obtained

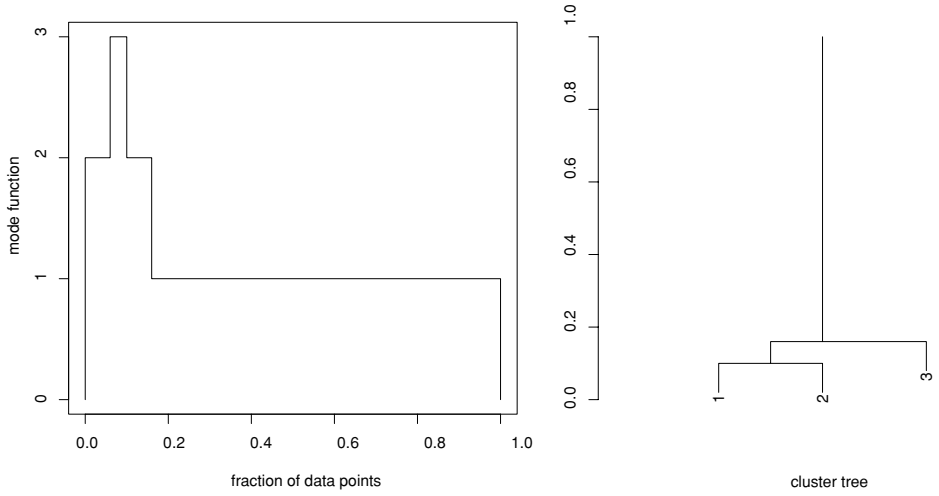


**Fig. 8** Olive oil data: mode function (*left*) and corresponding cluster tree (*right*)

**Table 3** Olive oil data: number of objects cross-classified by geographical macro-area and clusters obtained by the density-based methods and some alternatives; ARI denotes the adjusted Rand index

| | 1 | 2 | 3 |
|---|---|---|---|
| | $k$-means (best outcome) | | |
| South | 282 | 0 | 41 |
| Sardinia | 0 | 97 | 1 |
| Centre-North | 0 | 55 | 96 |
| ARI = 0.663 | | | |
| | Hierarchical (complete linkage) | | |
| South | 107 | 216 | 0 |
| Sardinia | 98 | 0 | 0 |
| Centre-North | 88 | 0 | 63 |
| ARI = 0.280 | | | |
| | Hierarchical (Ward method) | | |
| South | 109 | 214 | 0 |
| Sardinia | 98 | 0 | 0 |
| Centre-North | 114 | 0 | 37 |
| ARI = 0.232 | | | |
| | Density-based method | | |
| South | 294 | 0 | 29 |
| Sardinia | 0 | 98 | 0 |
| Centre-North | 0 | 17 | 134 |
| ARI = 0.792 | | | |

by the same clustering techniques considered in Table 2, with the constraint to form three clusters.

## 5 Final remarks

We conclude by recalling the main features of the proposed method, and mention some aspects which require further investigation.

- In contrast to most clustering techniques, the method is not directly linked to the idea of distance between observations. However, distances do enter in two steps: (i) non-parametric estimation of the density function, and (ii) Delaunay triangulation.
- Another aspect different from most clustering techniques is that the number of clusters is selected by the method. It is not an input value, as in $k$-means, and is not left undetermined, as in hierarchical clustering.
- In principle, the choice of smoothing parameter is critical. However, our numerical experience indicates that its effect on the final outcome is limited, and does not reflect the instability of the actual density estimate. Selection of the smoothing parameter is a point to be explored in more detail, possibly using variable bandwidth.
- Allocating points not belonging to cluster cores is naturally of an incremental nature. This fact allows for the introduction of a degree of confidence in the allocation, by giving lower confidence to points which are allocated last, compared with those in the cluster cores and those allocated in earlier stages. Another option is non-allocation of some points when this confidence degree is low, if appropriate for the problem at hand.
- The most critical computational step of the whole procedure is obtaining the Delaunay triangulation. At the current stage of development in computational geometry, the value of $d$ which can be handled in reasonable computing time is not very high.

As an overall conclusion, the method compares favourably with some established clustering techniques, as illustrated by the examples. While it is already satisfactory in many ways, there are aspects of the method which can be further improved. Exploration of possible refinements mentioned above is left for future work.

**Acknowledgments** A preliminary version of this work was presented at the 'International Conference in Honour of Sir David Cox on the Occasion of his 80th Birthday' held in Neûchatel (CH) on 15–17 July 2004. We would like to thank participants in the ensuing discussion

for their useful remarks. Further useful comments have been kindly offered to us by Prof. David Hand. We also thank a reviewer of the paper for a detailed revision leading to a much improved presentation of the material.

## References

Aitchison J. 1986. The Statistical Analysis of Compositional Data. Chapman & Hall, London.

Ankerst M., Breuning M.M., Kriegel H.P., and Sander J. 1999. OPTICS: ordering points to identify the clustering structure. In: International Conference on Management of Data (SIGMOD'99), ACM, pp. 49–60.

Barber C.B., Dobkin D.P., and Huhdanpaa H. 1996. The Quickhull algorithm for convex hulls. ACM Trans. Math. Software 22: 469–483.

Bowman A. and Foster P. 1993. Density based exploration of bivariate data. Statistics and Computing 3: 171–177.

Bowman A.W. and Azzalini 1997. Applied Smoothing Techniques for Data Analysis. Claredon Press, Oxford.

Cuevas A., Febrero M., and Fraiman R. 2000. Estimating the number of clusters. Canad. J. Stat. 28: 367–382.

Cuevas A., Febrero M., and Fraiman R. 2001. Cluster analysis: a further approach based on density estimation. Computational Statistics & Data Analysis 36: 441–459.

Devroye L.P. and Wagner T.J. 1980. The strong uniform consistency of kernel density estimates. In: Multivariate Analysis, North-Holland, Vol. 5, pp. 59–77.

Ester M., Kriegel H.P., Sander J., and Xu X. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the 2nd International Conference on Knowledge Discovery in Data Mining (KDD-96), Portland, OR, USA. ACM, pp. 226–231.

Forina M., Armanino C., Lanteri S., and Tiscornia E. 1983. Classication of olive oils from their fatty acid composition. In: H. Martens and H. J. Russwurm (Eds.), Food Research and Data Analysis, Applied Science Publishers: London, pp. 189–214.

Hartigan J.A. 1975. Clustering Algorithms. J. Wiley & Sons, New York.

Hubert L. and Arabie P. 1985. Comparing partitions. Journal of Classification 2: 193–218.

Nadaraya É.A. 1965. On non-parametric estimates of density functions and regression curves. Theory Probability its Appl. (Transl. Teorija Verojatnostei i ee Primenenija) 10: 186–190.

Okabe A., Boots B.N., and Sugihara K. 1992. Spatial Tessellations: Concepts and Applications of Voronoi Diagrams. J. Wiley & Sons, New York.

R Development Core Team 2004. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria 3-900051-07-0.

Rosolin T., Azzalini A., and Torelli N. 2003. Detecting clusters via non-parametric density estimation. In: Convegno SIS analisi statistica multivariata per le scienze economico-sociali, le scienze naturali e la tecnologia, Napoli, Italy. Società Italiana di Statistica, RCE edizioni.

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

Wong A.M. and Lane T. 1983. The *k*th nearest neighbour clustering procedure. Journal of the Royal Statistical Society, Series B 45: 362–368.