Data Min Knowl Disc
DOI 10.1007/s10618-008-0120-3

DECODE: a new method for discovering clusters of different densities in spatial data

Tao Pei · Ajay Jasra · David J. Hand · A.-Xing Zhu · Chenghu Zhou

Received: 5 November 2007 / Accepted: 21 October 2008

Abstract When clusters with different densities and noise lie in a spatial point set, the major obstacle to classifying these data is the determination of the thresholds for classification, which may form a series of bins for allocating each point to different clusters. Much of the previous work has adopted a model-based approach, but is either incapable of estimating the thresholds in an automatic way, or limited to only two point processes, i.e. noise and clusters with the same density. In this paper, we present a new density-based cluster method (DECODE), in which a spatial data set is presumed to consist of different point processes and clusters with different densities

Responsible editor: Charu Aggarwal.

T. Pei · A.-X. Zhu · C. Zhou (✉)
Institute of Geographical Sciences and Natural Resources Research, 11A, Datun Road Anwai, Beijing 100101, China
e-mail: zhouch@lreis.ac.cn

T. Pei
Institute for Mathematical Sciences, Imperial College, London SW7 2PG, UK
e-mail: peit@lreis.ac.cn

A. Jasra
Department of Mathematics, Imperial College, London, UK
e-mail: ajay.jasra@imperial.ac.uk

D. J. Hand
Department of Mathematics and Institute for Mathematical Sciences, Imperial College, London, UK
e-mail: d.j.hand@imperial.ac.uk

A.-X. Zhu
Department of Geography, University of Wisconsin Madison, 550N, Park Street, Madison, WI 53706-1491, USA
e-mail: axing@geography.wisc.edu; axing@lreis.ac.cn

Published online: 20 November 2008

belong to different point processes. DECODE is based upon a reversible jump Markov Chain Monte Carlo (MCMC) strategy and divided into three steps. The first step is to map each point in the data to its mth nearest distance, which is referred to as the distance between a point and its mth nearest neighbor. In the second step, classification thresholds are determined via a reversible jump MCMC strategy. In the third step, clusters are formed by spatially connecting the points whose mth nearest distances fall into a particular bin defined by the thresholds. Four experiments, including two simulated data sets and two seismic data sets, are used to evaluate the algorithm. Results on simulated data show that our approach is capable of discovering the clusters automatically. Results on seismic data suggest that the clustered earthquakes, identified by DECODE, either imply the epicenters of forthcoming strong earthquakes or indicate the areas with the most intensive seismicity, this is consistent with the tectonic states and estimated stress distribution in the associated areas. The comparison between DECODE and other state-of-the-art methods, such as DBSCAN, OPTICS and Wavelet Cluster, illustrates the contribution of our approach: although DECODE can be computationally expensive, it is capable of identifying the number of point processes and simultaneously estimating the classification thresholds with little prior knowledge.

Keywords Data mining · MCMC · Point process · Reversible jump · Nearest neighbor · Earthquake

1 Introduction

Discovering clusters in complex spatial data, in which clusters of different densities are superposed, severely challenges existing data mining methods. For instance, in seismic research, clustered earthquakes attract much attention because they can either be foreshocks (which indicate forthcoming strong earthquakes) or aftershocks (which may help to elucidate the mechanism of major earthquakes). However, foreshocks and aftershocks are often superposed by background earthquakes, and the imposed interference makes it difficult to identify them. Similar situations may arise in many fields of study, such as landslide evaluation, minefield detection, tracing violent crime and mapping traffic accidents. Therefore, although difficult, it is extremely important to discover clusters from spatial point sets in which clusters of different densities coexist with noise.

In this context, data are presumed to consist of various spatial point processes in each of which points are distributed at a constant, but different intensity. Across each process, there are, potentially many, clusters, which are mutually exclusive. Clusters of different densities then belong to different processes. As a result, to discover clusters in a point set, containing different point processes, a cluster method must not only be capable of detecting the number of point processes (cluster types), but also be capable of assigning points to different clusters; this requires the determination of thresholds for each cluster. For clarification, we use intensity when discussing a point process and density for a cluster. Recall that intensity is defined as the ratio between the number of points and the area of their support domain (Cressie 1991; Allard and Fraley 1997).

Density-based cluster methods are characterized by aggregating mechanisms based on density (Han et al. 2001). It is believed that density-based cluster methods have the potential to reveal the structure of a spatial data set in which different point processes overlap. Ester et al. (1996) and Sander et al. (1998) introduced the approaches of DBSCAN and GDBSCAN to address the detection of clusters in a spatial database according to a difference in density. Since then, many modifications have been published. However, such methods have some drawbacks. In DBSCAN and its modifications it is required to define their parameters (for example, *Eps*, the distance used to separate clusters of different densities) in an interactive way. The parameters, estimated in this fashion, may lead to classification errors in terms of class type number and membership of each point, especially in complex scenarios.

In this paper, we present a density-based cluster method for DiscovEring Clusters Of Different dEnsities (DECODE). The novelties of DECODE are 2-fold: (1) it can identify the number of point processes with little prior knowledge; (2) it can estimate the thresholds for separating point processes and clusters automatically. In developing this method, we first assume that a point set is composed of an unknown number of point processes, and each point process may be composed of different clusters. Then, we transform the multi-dimensional point set into a probability mixture using nearest neighbor theory with each probability component representing a point process. Next, we construct a Bayesian mixture model and use an MCMC algorithm to produce realizations of parameters of individual point process. Finally, the number of processes and their parameters are estimated by averaging the realizations and the points are connected to form clusters according to their spatial density-connectivity. The overall process consists of two phases (Fig. 1). The first phase is to determine the thresholds of intensity, and the second phase is to establish the mechanism to form clusters.

The rest of the paper is structured as follows. Section 2 reviews recent approaches to density-based cluster methods. In Sect. 3, we present some notions of nearest neighbor theory and derive the probability density function of the mth nearest neighbor distance, which is referred to as a distance between a point and its mth nearest neighbor. In Sect. 4, the reversible jump MCMC algorithm of the mth nearest distance is described in detail; this allows us to estimate the number of processes and their parameters. The details of the DECODE algorithm are presented in Sect. 5. Discussions of

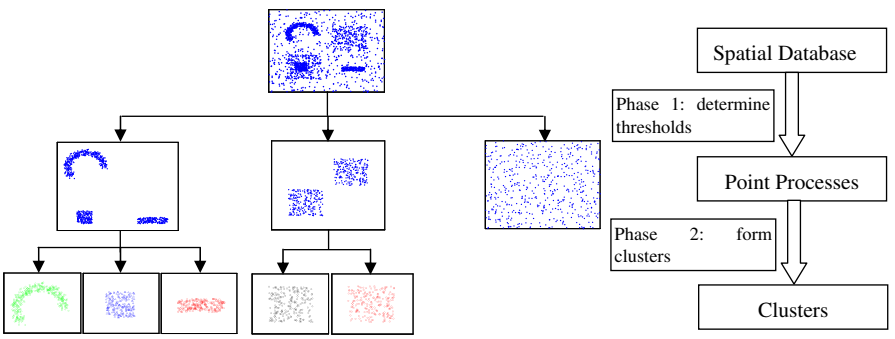

Fig. 1 Flowchart of the method for discovering clusters of different densities in spatial data

the parameters and the complexity of DECODE are given in Sect. 6. Four experiments are presented in Sect. 7 for the evaluation of DECODE. Section 8 provides a summary of this paper as well as directions for future research.

2 Related work on density-based cluster methods

The aim of clustering is to group data into meaningful subclasses (clusters) (Jain and Dubes 1988; Kaufman and Rousseeuw 1990). Clustering methods can be broadly classified into two categories: the partitional and the hierarchical. The partitional clustering methods obtain a partition of objects into clusters such that the objects in a cluster are more similar to each other than to objects in other clusters. The hierarchical cluster method is a nested sequence of partitions, it starts by placing each object in its own cluster and then merges these atomic clusters into larger clusters until some termination condition is satisfied (Kaufman and Rousseeuw 1990; Han et al. 2001). However, in recent years, density-based methods, differing from the partitional and hierarchical methods, have been proposed to classify dense objects into homogenous clusters with arbitrary shape and size and remove noise using a density criterion. Two strategies, i.e. the grid-based and the distance-based, have been adopted for finding density homogeneous clusters in density-based methods.

2.1 Grid-based clustering method

The main idea of grid-based clustering methods is to map data into a mesh grid and identify dense regions according to the density in cells. The main advantage of grid-based clustering methods is their detection capability for finding arbitrary shaped clusters and their high efficiency in dealing with complex data sets which are characterized by large amounts of data, high dimensionality and multiple densities (Agrawal et al. 1998; Han et al. 2001).

Recent approaches to grid-based clustering methods have focused on techniques for identifying dense regions which are made up of connected cells. Due to its power in estimating local densities, a kernel function is used within grid-based clustering methods to determine statistically significant clustering (Diggle 1985; Hinneburg and Keim 1998; Tran et al. 2006). However, kernel clustering methods rely upon the choice of kernel function and its parameters, and this requires extra prior knowledge about data. For this reason, techniques from "spatial scan statistics", with which a given set of predefined regions is searched over to find those containing the clusters, has been proposed to provide an efficient alternative for the scanning of clusters (Neill and Moore 2005; Neill 2006). Nevertheless, the method based on spatial scan statistics is restricted to the detection of either axis-aligned or rectangular-likewise clusters. Therefore, more work is needed to extend the method to the detection of irregularly shaped clusters.

In order to avoid computing statistics to determine whether a clustering is significant, Sheikholeslami et al. (1998) proposed WaveCluster, which applies a wavelet transform to the grid. The method is able to reveal arbitrarily shaped clusters from the wavelet approximation of the original image at different scales. Nevertheless, the

patterns become coarser and thereby distort the shape of clusters as the wavelet transform is processed. To overcome these limitations, Murtagh and Starck (1998) proposed a cluster method which expresses the structure of data through a redundant wavelet transform and identifies significant clusters according to a noise model constructed on the simulation of wavelet coefficients. The noise-model-based Wavecluster (call this method N-Wavecluster hereafter) may not only reveal the shapes and locations of clusters in different scales, without reducing the resolution, but may also eliminate the influence of noise.

Despite the computational speed, the grid-based methods suffer from a major drawback: the clustering results are sensitive to the grid partition scheme, i.e. the cell size in a grid. Choice of cell size may significantly affect the outcome of the analysis in terms of size, shape and significance of clusters.

2.2 Distance-based clustering methods

Distance-based cluster methods are able to identify dense subclasses based on the distance between a point and its neighbor, which reflects the density of the local area. As a result, the methods can avoid the computation of local density which depends on the partition scheme of the research area. Due to this advantage, many approaches have been proposed, often based on the mth nearest neighbor distance (Ester et al. 1996; Markus et al. 2000; Tran et al. 2006). Among those methods, DBSCAN is perhaps the most important and has attracted much attention in the community of spatial data mining and knowledge discovery as it is both easy to implement and can identify clusters with arbitrary shapes (Ester et al. 1996; Sander et al. 1998). Before we continue, the concept of density-connected and the significance of parameters in DBSCAN should be introduced first.

2.2.1 Concepts relating to DBSCAN

Given a point p, let $p \in D, D \subseteq R^d, d \geq 1$. There are two elements to the definition of density-connected, the *Eps*-neighborhood ($N_{Eps}(p) = \{q \in D | dist(p,q) \leq Eps\}$, $Eps > 0$) and *MinPts*, which is the minimum number of points in the *Eps*-neighborhood. A point p is said to be *density-connected* to point q with respect to (wrt) *Eps* and *MinPts* if there is a collection of points $p_1, p_2, \ldots, p_n$ (with $p_1 = q$ and $p_n = p$) so that $p_{i-1} \in N_{Eps}(p_i)$ $(i = 2, 3, \ldots, n)$ and $N_{Eps}(p_i)$ $(i = 2, 3, \ldots, n-1)$ must contain at least *MinPts* points. Based on these ideas, a cluster is defined to be a point subset M where any point $p_i \in M$ is density-connected to point $p_j \in M$ $(p_i \neq p_j)$. In a cluster, point $p_i \in M$ is denoted as a *core point* if the number of points included in $N_{Eps}(p_i)$ is not less than $MinPts$. Otherwise, it is denoted as a border point, which is density-connected with at least one core point but with less than *MinPts* points in its *Eps*-neighborhood. The cluster can be formed by extending a point into a collection of points which are all density-connected with each other. Before identifying the clusters, one has to define the parameters (*MinPts*, *Eps*) from a sorted m-dist graph in which the mth nearest distance of each point is sorted in a descending order and lined up to form a dotted curve (Ester et al. 1996).

2.2.2 Determination of process number and estimation of parameters

As discussed, when using DBSCAN to discover clusters in a complex data set, the number of processes and their parameters (*MinPts*, *Eps*) might be overestimated or underestimated by the visual and interactive way. This may result in the misclassification of points or even the misidentification of clusters or processes. Regarding the drawbacks, subsequent approaches enhance DBSCAN in two ways (although they are related to each other): one is to improve the estimation of the parameters (*MinPts*, *Eps*), another is to promote the capability of the determination of the number of processes.

To try to reduce the subjectivity in the parameter estimation, Ankerst et al. (1999) proposed an enhanced density-connected algorithm, referred to as: Ordering Points To Identify the Clustering Structure (OPTICS). OPTICS provides a graphical and interactive tool to help find the cluster structure by constructing an augmented cluster-ordering of database objects and its reachability-plot wrt Eps'(the generating distance for defining the reachability between points) and *MinPts*. Although the reachability-plot reduces the sensitivity to the input parameters to some extent, the classification is still dependent upon the manual determination of *Eps* (the clustering distance) and *MinPts*. Furthermore, it is also difficult for OPTICS to determine how many *Eps*es will be needed to determine the clusters of different densities in a complex data set. Daszykowski et al. (2001) proposed a DBSCAN-based modification to look for natural patterns of arbitrary shape. They detected noise in a subjective way, by separating a prescribed percentage of data points, which is found at the tail of the frequency curve. To make the estimation objective, Pei et al. (2006) proposed a nearest-neighbor cluster method, in which the threshold of density (equivalent to *Eps* of DBSCAN) is computed via the Expectation-Maximization (EM) algorithm and the optimum value of k (equivalent to *MinPts* of DBSCAN or m of DECODE) can be decided by the lifetime of individual k. As a result, the clustered points and noise were separated according to the threshold of density and the optimum value of k. Although the method can estimate the parameters in an automatic way, it is limited to two-process data sets.

In order to adapt DBSCAN to data consisting of multiple processes, an improvement should be made to find the difference in the mth nearest distances of processes. Roy and Bhattacharyya (2005) developed a modified DBSCAN method, which may help to find different density clusters which overlap. However, the parameters in this method, both the core-distance and the predefined variance factor α, which are used to expand a cluster and to find clusters of different densities respectively, are still defined by users interactively. Lin and Chang (2005) introduced an interactive approach called GADAC. This method adopts the diverse radii which are adjusted by a genetic algorithm and used to expand clusters of varied densities in contrast to DBSCAN's only one fixed radius. GADAC may produce more precise classification results than DBSCAN does. Nevertheless, in GADAC, the estimation of the radii is greatly dependent upon another parameter, the density threshold δ, which can only be determined in an interactive way. Pascual et al. (2006) present a density-based cluster method for finding clusters of different sizes, shapes and densities. Their method is capable of separating clusters of different densities and revealing overlapping clusters. However,

the parameters in the method, such as the neighborhood radius R, which is used to estimate the density of each point, have to be defined using prior knowledge. Moreover, their method is designed for finding Gaussian-shaped clusters and is not always appropriate for clusters with arbitrary shapes. In addition, Liu et al. (2007) proposed a method, called VDBSCAN, to identify clusters in a varied-density data set. However, the method is unable to avoid interactive and visual estimation of the parameters. Although most algorithms above can deal with data containing clusters with different densities and noise, the estimation of parameters (*MinPts* and *Eps*) is still a subjective process which is dependent upon prior knowledge.

The Penalised likelihood criterion, such as BIC (Bayesian Information Criterion), may provide a solution to estimating the number of processes and their parameters in a data set. However, the limitations of this strategy are two-fold, one is that the initial set of models can be very large, and many of those models are not of interest, so that computing resources are wasted (Andrieu et al. 2003); the other is that the number of point processes could be overestimated by BIC (Jasra et al. 2006).

In summary, when clusters with different densities and noise coexist in a data set, few existing methods can determine the number of processes and precisely estimate the parameters in an automatic way. In this article, we attempt to provide such a solution via DECODE.

3 Concepts relating to nearest neighbor cluster method

3.1 mth Nearest neighbor distance and its probability density function

Suppose that the support domain (territory) of a point process $A \subseteq R^d$ and points in the point process are denoted as $\{s_i : s_i \in A, i = 1, \ldots, n\}$. The mth nearest distance, X_m, of a point s_i is defined as the distance between s_i and its mth nearest neighbor. We use the terminology of distance order in the following context, which is referred to as the ordinance of nearest neighbor from s_i. For a homogeneous Poisson process of constant intensity (λ), points $\{s_i\}$ in the process have the equivalent likelihood to distribute in A (Cressie 1991). The cumulative distribution function (cdf) of the mth nearest distance X_m from a randomly chosen point in the process is as follows (Byers and Raftery 1998):

$$G_{X_m}(x) = P(X_m \geq x) = 1 - \sum_{l=0}^{m-1} \frac{e^{-\lambda\pi x^2}(\lambda\pi x^2)^l}{l!}. \tag{1}$$

The equation is obtained by conceiving a circle of radius x centered at a point. In detail, if X_m is greater than x, there must be one of 0,1,2, …, $m-1$ points in this circle (i.e. the 2-dimensional unit, whose area is πx^2). The probability density function (pdf) of the mth nearest distance, $f_{X_m}(x; m, \lambda)$, is the derivative of $G_{X_m}(x)$ (cdf):

$$f_{X_m}(x; m, \lambda) = \frac{e^{-\lambda\pi x^2} 2(\lambda\pi)^m x^{2m-1}}{(m-1)!}, \tag{2}$$

where x is the random variable, m is the distance order, λ is the intensity of the point process. To extend Eq. 1 to d-dimensions ($d > 2$), we only need to look at a d-dimensional unit hyper-sphere (whose volume is $\alpha_d x^d$, where $\alpha_d = 2\pi^{d/2}/d\Gamma(d/2)$) instead of a circle. The pdf of X_m can be derived from the d-dimensional *cdf* (for details, see Byers and Raftery 1998).

3.2 Probability mixture distribution of mth nearest distance

If multiple point processes with different intensities overlap in a region, the mth nearest distances of these points can be modeled via a mixture distribution. For example, in Fig. 2, the simulated data contain three point processes distributed at different intensities, which are displayed as five clusters in addition to the background noise. Each point process except the noise contains more than one cluster, moreover, the one with the highest intensity includes an embedded cluster. The histogram of their mth nearest distances is shown in Fig. 3. Note that edge effects should be considered before calculating the mth nearest distance because points near the edges of the entire research domain have larger mean of the mth nearest distance than those in the inner area. In this paper, we reduced the edge effect by transforming the data into the toroidal edge-corrected data.

The mth nearest distances, in which k point processes are assumed to exist, can be expressed as the mixture density:

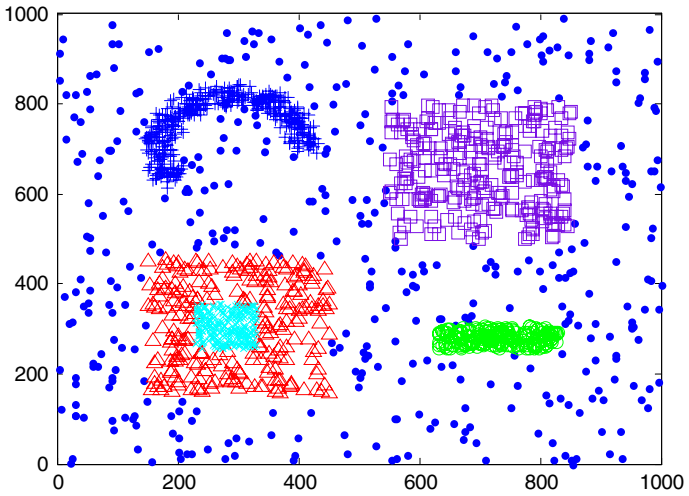

Fig. 2 Simulated data (The data are composed of three different point processes: five clusters and noise. Noise (symbolized by dots) is distributed at the lowest intensity over the whole area. Cluster 1 (symbolized by crosses), Cluster 2 (symbolized by circles) and Cluster 3 (symbolized by rotated crosses) belong to the same point process with the highest intensity while Cluster 3 is an embedded cluster. Cluster 4 (symbolized by squares) and Cluster 5 (symbolized by triangles) belong to the same point process with the medium intensity)

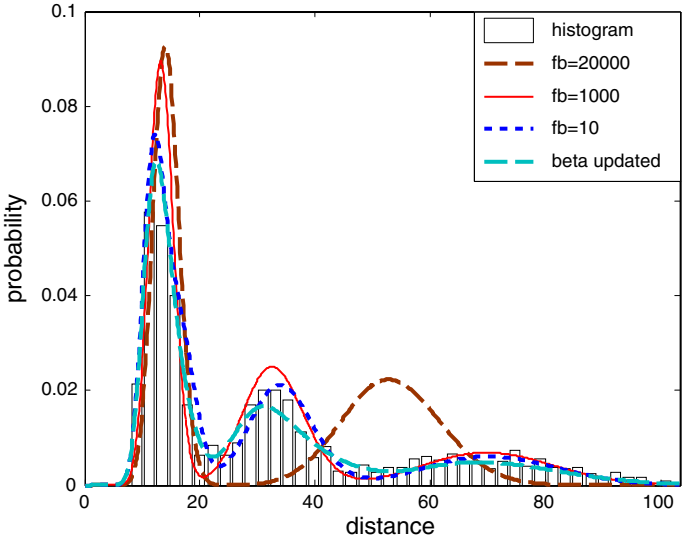

Fig. 3 The histogram of the mth nearest distance of the simulated data and the fitted mixture probability density function (β fixed with $f_b = 20,000$ (the dashed line, 2 components), β fixed with $f_b = 1,000$ (the solid line, 3 components), β fixed with $f_b = 10$ (the dotted line, 4 components), β updated (the dashed-dotted line, 9 components), where f_b is the parameter defining the expectation of λ_i which is simulated)

$$X_m \sim \sum_{i=1}^{k} w_i f(x; m, \lambda_i) = \sum_{i=1}^{k} w_i \frac{e^{-\lambda_i \pi x^2} 2(\lambda_i \pi)^m x^{2m-1}}{(m-1)!}, \tag{3}$$

where $w_i > 0$ is the proportion of the ith process with $\sum w_i = 1$, m is the distance order and λ_i is the intensity of the ith process.

Since a given point s_i in the point set provides a one-to-one correspondence with its mth nearest distance X_m, points can be classified by decomposing the mixture, i.e. determining both the number (k) of point processes and their parameters (w_i, λ_i). We use a reversible jump MCMC method to accomplish these.

4 Reversible jump MCMC strategy for decomposing of mth nearest distance mixture

4.1 Metropolis–Hastings algorithm

MCMC is a strategy for generating samples from virtually any probability distribution, $p(x)$, which is known, is point-wise up to constant; see Robert and Casella (2004) for an introduction. The method generates a Markov chain with a stationary distribution $p(x)$.

The Metropolis–Hasting kernel is the building block of most MCMC algorithms and is simulated as follows. Assume that the current state of the Markov chain is $x^{(n)}$; then sample $x^* \sim q(x^*|x^{(n)})$, where $q(x^*|x^{(n)})$ is a (essentially, up to some mathematical

requirements, arbitrary) proposal density; accept x^* as the new state with probability $\min\{1, A(x^{(n)}, x^*)\}$, where $A(x^{(n)}, x^*) = p(x^*)q(x^{(n)}|x^*)/p(x^{(n)})q(x^*|x^{(n)})$ (Andrieu et al. 2003).

In this paper, we use random walk proposal methods: the additive and multiplicative random walk. The additive random walk is expressed as: $x^* = x + \sigma u$ and the multiplicative random walk is expressed as: $x^* = x\sigma u$, where σ is the scale of the random walk and u is a random variable used to perturb the current value x. Generally the parameter σ is tuned (although it can be adaptively set—see Robert and Casella (2004) and the references therein) so that the acceptance rate is approximately 0.25.

The Metropolis–Hastings kernel described above cannot be easily constructed to deal with problems where the parameter space is of varying dimension. In particular, in our context, we are interested in sampling from a mixture (posterior) distribution with an unknown number of point processes, so that the number of intensities is not known.

4.2 Bayesian model determination by reversible jump MCMC

Green (1995) introduced reversible jump MCMC, which is essentially an adaptation of the Metropolis–Hastings algorithm. The method is designed to generate samples from probability distributions of varying dimension. More precisely, it is constructed so that it is possible to construct Markov transitions between states of different dimensions. The term 'reversible' is used because such moves are constructed in reversible pairs, for example birth and death moves, which proceed as follows.

Suppose the current state of our Markov chain is $(x_1, \ldots, x_k) \in R^d$ ($x_{1:k}$ for short) and we have the choice of either 'birth' or 'death', selected with probabilities $b(x_{1:k})$ and $d(x_{1:k}) = 1 - b(x_{1:k})$, respectively. Suppose that we select the birth, and it consists of increasing dimension by 1, such a move is completed as follows. Sample $u \in R$ from a probability density q and set the proposed state of the chain as $x^*_{1:k+1} = f(x_{1:k}, u)$ with $f : R^k \times R \to R^{k+1}$, which is an invertible and differential mapping. This state is accepted as the new state of the chain with probability of $\min\{1, A(x_{1:k}, x^*_{1:k+1})\}$, where

$$A(x_{1:k}, x^*_{1:k+1}) = \frac{p(x^*_{1:k+1})}{p(x_{1:k})} \frac{1}{q(u)} \frac{d(x^*_{1:k+1})}{b(x_{1:k})} \left| \frac{\partial f}{\partial (x_{1:k}, u)} \right|.$$

The Jacobian is present to take into account the change of variables (see Green 1995). The reverse death in state $x^*_{1:k+1}$ is performed by inverting f (to determine u) and inverting the formula for A above (to compute the acceptance probability).

4.3 Bayesian statistical model and reversible jump algorithm

4.3.1 Priors

We will use a Bayesian statistical model. This requires us to specify prior probability distributions for the unknown parameters. We assume that the priors on the intensities

are taken to be i.i.d (independently and identically distributed) for each point process with $\lambda_i|\beta \sim ga(\alpha, \beta)$ and $w_i \sim dirichlet(\delta)$, where $ga(.)$ and $dirichlet(.)$ denote the Gamma and Dirichlet distributions, respectively. β is taken to be either random ($ga(g, h)$) or fixed. The prior parameters ($\alpha,\ g,\ h$) can be set as in (Richardson and Green 1997). The prior distribution on k is uniform on the range $\{1, \ldots, k_{\max}\}$, with $k_{\max}$ a pre-specified value.

4.3.2 Reversible jump algorithm

The reversible jump MCMC strategy is as follows.

(1) Initialize the parameters ($k, \lambda_{1:k},\ w_{1:k}, \beta$).
(2) Specify the sweep times, i.e. the number of simulations that the algorithm will generate.
(3) Perform the following moves.
Update λ_i with a log-normal random walk

$$\lambda_i^{(n+1)} = \lambda_i^{(n)} e^{u_i \sigma}\ (u_i \sim N(0,1),\ i = 1, \ldots, k);$$

Accept $\lambda_{1:k}^{(n+1)}$ according to $A(\lambda_{1:k}^{(n)}, \lambda_{1:k}^{*})$;
where

$$A(\lambda_{1:k}^{(n)}, \lambda_{1:k}^{*}) = \min\left\{1, \frac{\prod_{j=1}^{M}\sum_{i=1}^{k}\left[w_i^{(n)} f\left(x_j|\lambda_j^{(n+1)}\right)\right]\cdot\prod_{i=1}^{k}\left(\lambda_i^{(n+1)}\right)^{\alpha}\cdot e^{-\beta\left(\sum_{i=1}^{k}\lambda_i^{(n+1)}\right)}}{\prod_{j=1}^{M}\sum_{i=1}^{k}\left[w_i^{(n)} f\left(x_j|\lambda_j^{(n)}\right)\right]\cdot\prod_{i=1}^{k}\left(\lambda_i^{(n)}\right)^{\alpha}\cdot e^{-\beta\left(\sum_{i=1}^{k}\lambda_i^{(n)}\right)}}\right\}.$$

Update w_i with an additive normal random walk on the logit scale.

$$w_i^{(n+1)} = \left[\frac{w_i^{(n)}}{1-\sum_{j=1}^{k-1} w_j^{(n)}}\left(1-\sum_{j=1}^{k-1} w_j^{(n+1)}\right)\right] e^{\sigma u_i}\ \ (u_i \sim N(0,1),\ i = 1, \ldots, k);$$

Accept $w_{1:k}^{(n+1)}$ according to $A(w_{1:k}^{(n)}, w_{1:k}^{*})$;
where $A(w_{1:k}^{(n)}, w_{1:k}^{*}) = \min\left\{1, \frac{\prod_{j=1}^{M}\sum_{i=1}^{k}\left[w_i^{(n+1)} f\left(x_j|\lambda_j^{(n+1)}\right)\right]\cdot|\bullet|_{q\left(w_{1:k}^{(n)}|w_{1:k}^{(n+1)}\right)}}{\prod_{j=1}^{M}\sum_{i=1}^{k}\left[w_i^{(n)} f\left(x_j|\lambda_j^{(n)}\right)\right]\cdot|\bullet|_{q\left(w_{1:k}^{(n+1)}|w_{1:k}^{(n)}\right)}}\right\}$,
$|\bullet|_{q\left(w_{1:k}^{(n)}|w_{1:k}^{(n+1)}\right)}$ is the Jacobian matrix for $q(w_{1:k}^{(n)}|w_{1:k}^{(n+1)})$, and $|\bullet|_{q(w_{1:k}^{(n+1)}|w_{1:k}^{(n)})}$ is the Jacobian matrix for $q(w_{1:k}^{(n+1)}|w_{1:k}^{(n)})$.
Update β (β is taken to be either random or constant)
Sample β from its full conditional probability function, $\beta \sim ga(g + k\alpha, h + \sum_{i=1}^{k}\lambda_i)$(random).
(4) Update k with the birth-and-death step

Make a random choice between birth and death at the probability b_k and d_k, respectively.

For a birth move:

Sample $\nu \sim U(0,1)$;
$\lambda_{k+1}|\beta \sim ga(\alpha,\beta)$, $w_{k+1}^* \sim be(1,k)$ and set $w_i^{(n+1)} = w_i^{(n)}(1-w_{k+1}^*)$ $(i=1,2,\ldots,k)$;
if $\nu < A_{birth}$ and $k < k_{\max}$
accept the proposed state and let $k = k+1$;
else
reject the proposal;
end

For a death move:

Choose a point process, with uniform probability, to die. Assume that the jth point process is chosen, then remove its $\lambda_j^{(n)}$, $w_j^{(n)}$ and make $w_i^{(n+1)} = w_i^{(n)}/(1-w_j^{(n)})$ $(i \neq j)$;
Sample $\nu \sim U(0,1)$;
if $\nu < A_{death}$ and $k > 1$
accept the proposal and let $k = k-1$;
else
reject the proposal;
end

Here A_{birth} is $\min\left\{1, \frac{p(k+1,\theta^{(k+1)}|x)\cdot j(k+1|\theta^{(k+1)})}{p(k,\theta^{(k)}|x)\cdot j(k|\theta^{(k)})q_1(u^{(1)})}\left|\frac{\partial(\theta^{(k+1)})}{\partial(\theta^{(k)},u^{(1)})}\right|\right\}$, that is, $A_{birth} = \min\left\{1, \frac{p\left(x|\lambda_{1:k+1}^{(n+1)},w_{1:k+1}^{(n+1)},k+1\right)\frac{1}{k}}{p(x|\lambda_{1:k}^{(n)},w_{1:k}^{(n)},k)\frac{1}{k}}\cdot B(k\delta,\delta)^{-1}w^{\delta-1}(1-w)^{k(\delta-1)}(k+1)\ \frac{d_{k+1}}{(k+1)b_k}\cdot\frac{(1-w)^{k-1}}{Be(w;1,k)}\right\}$ (Green 1995). k is the number of point processes at the current step, $p(x|\lambda_{1:k+1}^{(n)}, w_{1:k+1}^{(n)}, k+1) = \prod_{j=1}^{M}\sum_{i=1}^{k+1}\left[w_i^{(n+1)} f\left(x_j|\lambda_j^{(n+1)}\right)\right]$ (M is the number of the points in the data set) is the likelihood function of the mixture conditioned on $(\lambda_{1:k+1}^{(n+1)}, w_{1:k+1}^{(n+1)}, k+1)$, so is $p(x|\lambda_{1:k}^{(n)}, w_{1:k}^{(n)}, k) = \prod_{j=1}^{M}\sum_{i=1}^{k}[w_i^{(n)} f(x_j|\lambda_j^{(n)})]$, w is the weight of newly born point process, b_k is the probability of proposing a birth move at state k, d_{k+1} is the probability of proposing a death move at state k+1, $B(.,.)$ is the beta function. The birth move is performed for $k = 1,2,\ldots k_{max}-1$, where k_{max} is the maximal number of point processes that could be existed in the mixture. $A_{death} = 1/A_{birth}$ with k being replaced by $k-1$, and the parameters in $A_{death}$are the same as those in the A_{birth}. The death move is performed for $k = 2,3,\ldots k_{max}$.

5 DECODE: DiscovEring clusters of different dEnsities

Based upon the ideas of the mth nearest distance reversible jump MCMC, we propose the algorithm DECODE for discovering the clusters of different densities in a spatial data as follows.

(1) Compute the mth nearest distance of each point (X_m)
(2) Run the mth nearest distance reversible jump MCMC for analyzing the mixture of the mth nearest distances.

(3) Check if the algorithm converges. If yes, then go to the next step; otherwise, go back to step (2).
(4) Pick out the number of point processes with the highest posterior probability and compute the parameters of each point processes.
(5) Estimate the thresholds for classifying clusters using the formula of $Eps_i^* = \sqrt{\frac{\ln \frac{w_i}{w_{i+1}} + m \ln \frac{\lambda_i}{\lambda_{i+1}}}{\pi(\lambda_i - \lambda_{i+1})}}$ $(i = 1, 2, \ldots, k-1)$, where Eps_i^* is the threshold between the ith point process and the $(i+1)$th, and can be determined by computing the intersection between the pdf of the ith point process and that of the $(i+1)$th.
(6) Extend the clusters of different densities from the data set with Eps_i^* $(i = 1, 2, \ldots, k-1)$ and m according to their density-connectivity. Each pair of parameters (Eps_i^*, m) will lead to a set of clusters with the same density.

Some points should be noted before running the algorithm above. In DECODE, Step (2) is the key step in which the number of point processes and the thresholds (parameters) for classifying point processes and clusters are determined. The details of this step have been described in Sect. 4. Step (3) is used to check if the algorithm has converged. The algorithm is considered to have converged if the plot of cumulative occupancy fractions becomes stable as the sweep time increases. The curve of the cumulative occupancy fractions of j indicate the ratio between the times of simulations in which the point process number generated by the algorithm is no more than j and the total times having been processed. In Step (4), the posterior probability of k is obtained by computing the ratio between the number of simulations which contains k point processes and the total sweep times. The parameters of each process with the maximum posterior probability of k are estimated by averaging the parameters of simulations which contain k processes.

The mechanism of extending clusters is same as that in (Ester et al. 1996; Sander et al. 1998). The only difference between DECODE and DBSCAN is that in DECODE border points are not considered as members of clusters whereas in DBSCAN they are.

6 Some aspects of the model

6.1 Analysis of sensitivity to prior specification

Among the parameters in DECODE, β should be highlighted because it has a significant impact on the results (Richardson and Green 1997). As has been seen in the algorithm (Sect. 4.3.2), two strategies can be applied to the selection of β in Step 3. One is to update β constantly during the process, the other one is to fix β. For fixing β we set $\beta = f_b \lambda_{\max}$, where $\lambda_{\max} = \left(\frac{m(2m!)}{2^m (m!)^2 \min(X_m)}\right)^2$ (Thompson 1956) and $\lambda_{\max}$ is the intensity corresponding to $\min(X_m)$, so that in the birth move λ_i is sampled from a constant density function $ga(\alpha, \beta)$. In this context, f_b and $\lambda_{\max}$ define the expectation (that is $f_b \lambda_{\max}$ if $\alpha = 1$, this is because the expectation of $ga(\alpha, \beta)$ is $\alpha\beta$) of λ_i which is sampled. Therefore, f_b is independent of input data (i.e. the data of the mth nearest

distance). That is to say, once appropriate values of f_b are determined, they could apply to other problems.

To determine an appropriate range of values for f_b, we compared results generated using various values of f_b. Taking the simulated data in Fig. 2 as an example, we ran the MCMC algorithm for 100,000 sweeps at various values of f_b, and also ran the algorithm for updated β. As shown in Fig. 4, all of the plots of the cumulative occupancy fractions at different f_b become stable along with the increase of sweep time. Therefore, we may say that the algorithm (appears) to converge in all cases. We took the burn-in to be 50,000 sweeps. The corresponding posterior probabilities of k are shown in Table 1.

As shown in Table 1, we find that updating β produces the highest posterior probability at $k = 9$, thereby it leads to a model with more point processes. In contrast, fixing β indicates mixtures with fewer point processes from 5 ($f_b = 2$) to 1 ($f_b = 250{,}000$).

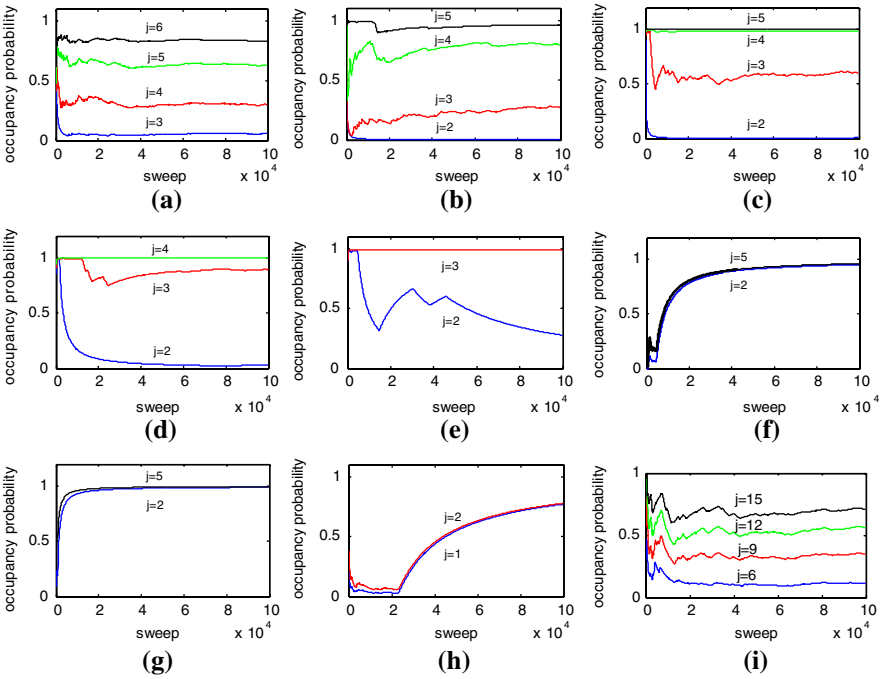

Fig. 4 The cumulative occupancy fractions of j at different f_b: **a** $f_b = 2$; **b** $f_b = 10$; **c** $f_b = 50$; **d** $f_b = 200$; **e** $f_b = 1{,}000$; **f** $f_b = 5{,}000$; **g** $f_b = 20{,}000$; **h** $f_b = 250{,}000$; **i** updating β (with $\delta = 1$, $\alpha = 1$, $h = 10/(\max(X_m) - \min(X_m))$, $g = 0.2$)

Table 1 The point process number (k) with the maximum posterior probability at different f_b

f_b	2	10	50	200	1,000	5,000	20,000	250,000	β Updated
k with the maximum posterior probability	5	4	3	3	3	2	2	1	9

This is due to Lindley's paradox (For details, see Richardson and Green 1997). It was also found that models with 3 point processes are acquired when $f_b = 50$, 200 and 1,000 and these are in accordance with the truth.

From the results, we find that: (1) updating β tends to produce a model with more point processes while fixing β suggests mixtures with fewer point processes; (2) f_b is the parameter which significantly influences the results when fixing β. The sensitivity of the results to the prior parameters has been addressed in many papers. Interested readers may refer to Richardson and Green (1997) for details. Based on these facts, we decide to fix β. From the relationship between f_b and k with the maximum Posterior probability, shown in Table 1, we can infer that the appropriate value of f_b can be selected from [50 1000] for this specific probability mixture problem. m is another parameter that should be chosen before running DECODE. The appropriate value of m can be chosen according to the size of clusters which users intend to discover.

The MCMC step in DECODE has several parameters, such as δ, α, σ, but the method is insensitive to those parameters. Usually, we let $\sigma = 0.1, \delta = 1$ and $\alpha = 1$. A similar observation has been made by other researchers (Richardson and Green 1997; Jasra et al. 2006). Therefore, DECODE is a robust and automatic cluster method which needs less prior knowledge of the target data set compared with other density based clusters methods.

6.2 The evaluation of the complexity of DECODE

For each sweep, time is mostly spent on the computation of $A(\lambda_{1:k}^{(n)}, \lambda_{1:k}^{*})$, $A(w_{1:k}^{(n)}, w_{1:k}^{*})$ and $A_{birth}(A_{death})$. The complexity of $A(\lambda_{1:k}^{(n)}, \lambda_{1:k}^{*})$ is $O(M \cdot k)$, that of $A(w_{1:k}^{(n)}, w_{1:k}^{*})$ is $O(M \cdot k + (k-1)k!)$ and that of A_{birth} is $O(M \cdot k)$, where M is the number of points of the data set, k is the number of point processes that the algorithm determines. So the total complexity of the algorithm is $O(T(Mk + (k-1)k!))$, where T is the sweep times. Usually, T is more than 50,000 to ensure the convergence of the algorithm. Therefore, the running time of the algorithm is largely dependent upon the sweep times.

To evaluate the efficiency of the algorithm we tested the program (which is coded in Matlab) on the platform Windows XP, with several simulated data sets, which have varying numbers of processes and points. Table 2 lists the CPU time spent on the classification of these data. We found that the run time of DECODE could be roughly approximated by $O(T(Mk + (k-1)k!))$. According to the analysis above, the algorithm is a time-consuming process when the number of sweep times is expected to be large.

Table 2 The CPU time of DECODE spent on different data

Number of points	250	500	1,000	1,500	1,500	500	500
Number of components	2	2	3	3	3	4	5
Sweep time	100,000	100,000	100,000	100,000	50,000	100,000	100,000
CPU time (s)	309	625	1,875	2,843	1,406	1,320	1,898

7 Experiments and analysis

7.1 Experiment 1

To evaluate our algorithm, four experiments have been conducted, two on simulated data, one using a regional seismic catalog of Southwestern China and one using a strong earthquake catalog of China and its adjacent areas.

The data of Experiment 1, displayed in Fig. 2, are distributed at different intensities in a $1,000 \times 1,000$ rectangle. In the data, three Poisson processes were simulated, the one with high density includes 3 clusters (i.e. Cluster 1 (200 points), Cluster 2 (200 points) and Cluster 3 (377 points)), the one with medium density includes two clusters (i.e. Cluster 4 (220 points) and Cluster 5 (220 points)) and the one with low density is noise (527 points) distributed over the whole region. We first applied DECODE to the data and then compared it with three state-of-the-art density-based cluster methods, i.e. DBSCAN, OPTICS and N-Wavecluster.

We took $\sigma = 0.1$, $\delta = 1$ for the updating of λ_i and w_i, $\alpha = 1$ for the prior probabilities of λ_i, and let $f_b = 500$ and $m = 10$ for running DECODE. The thresholds, estimated by DECODE, are indicated by crosses in Fig. 5a ($Eps_1^* = 19.65$, $Eps_2^* = 46.58$) and very close to the true values ($Eps_1 = 19.28$, $Eps_2 = 47.90$). The classification under the thresholds, shown in Fig. 6a, indicates 3 processes and 5 clusters, and also clearly reveals the structures of clusters, including the embedded ones. The validation shows that 98 mismatches are produced by DECODE. These results demonstrate that DECODE is capable of dealing with the clusters with various topological relationship with each other.

The thresholds for classifying data were estimated to be $Eps_1^* = 25.28$, $Eps_2^* = 45.21$ and $Eps_1^* = 15.0$, $Eps_2^* = 40.0$ by the m-dist plot of DBSCAN (Fig. 5a) and the reachability-plot of OPTICS (Fig. 5b), respectively. The difference between the estimations and the true values is obvious. We then applied the thresholds estimated by the m-dist plot to the data. From the classification result (Fig. 6b) we find that the data are divided into 8 clusters by DBSCAN and the false clusters are hidden in the cluster symbolized by squares. The number of misclassified points is 150. DBSCAN increases the error rate by more than 1/2. Similar to that of DBSCAN, classification under the thresholds estimated by OPTICS indicates 9 clusters and even more misclassified points (Fig. 6c). The results imply that DBSCAN and OPTICS would not only overestimate the number of clusters but also increase the number of misclassified points due to the error in estimating thresholds, which is done in a subjective and visual way. As a result, determining thresholds through visual estimation could be unrealistic especially when multiple point processes exist in a point set.

The N-Wavecluster was then applied to the data. The wavelet representations at different scales are shown in Fig. 5c. We find that the representation at Scale 6 could reflect the structure of clusters. The classification is enlarged in Fig. 6d. It is found that: (1) N-Wavecluster only identifies four clusters and neglects Cluster 3, the embedded cluster; (2) it overestimates Cluster 1 and 2 and underestimates Cluster 4 and 5. The number of total misclassified points is 302.

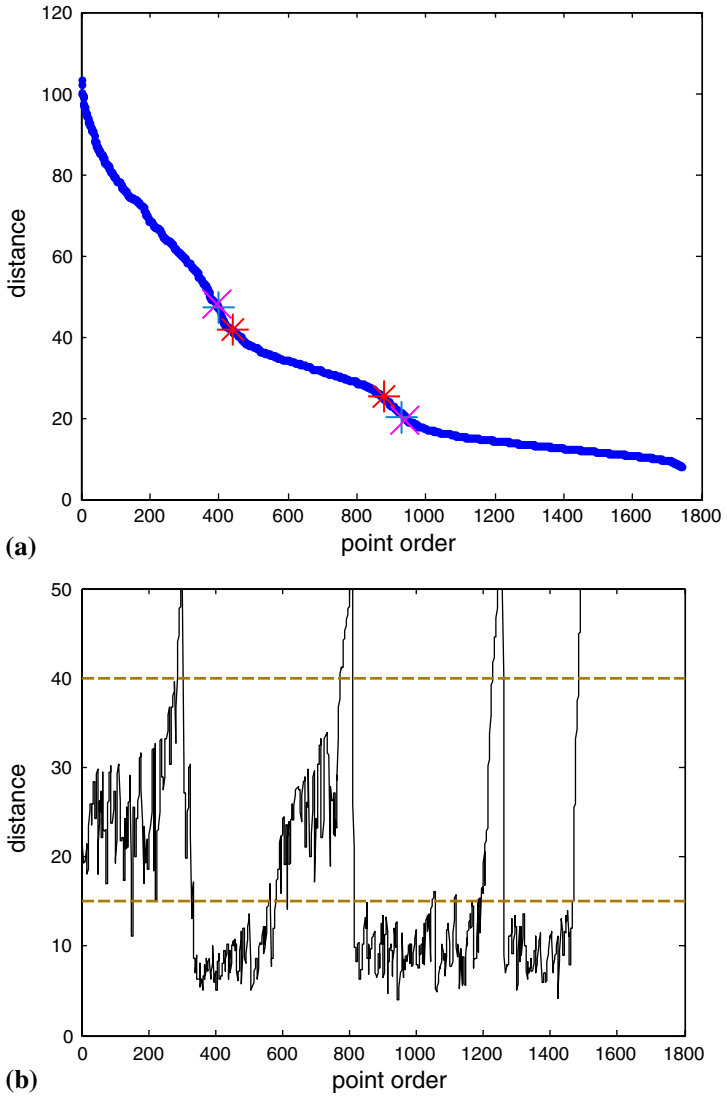

Fig. 5 Determination of thresholds for classifying points: **a** the sorted m-dist graph ($MinPts = 10$) with the thresholds determined by DECODE and a visual trial (the rotated crosses represent the true values of thresholds ($Eps_1 = 19.28$, $Eps_2 = 47.90$), the crosses represent the thresholds ($Eps_1^* = 19.65$, $Eps_2^* = 46.58$) determined by the reversible jump MCMC at $f_b = 500$ and the asterisks represent the thresholds ($Eps_1^* = 25.28$, $Eps_2^* = 45.21$) from an interactive estimation); **b** The reachability-plot ($MinPts = 10$, $Eps' = 50$) with the thresholds ($Eps_1^* = 15$, $Eps_2^* = 40$ indicated by two dashed lines) determined by a visual trail; **c** The multi-resolution wavelet representation of the cluster space

The comparison shows that DECODE outperforms the other three cluster methods for this data set in terms of cluster number, misclassified point number and removal of noise.

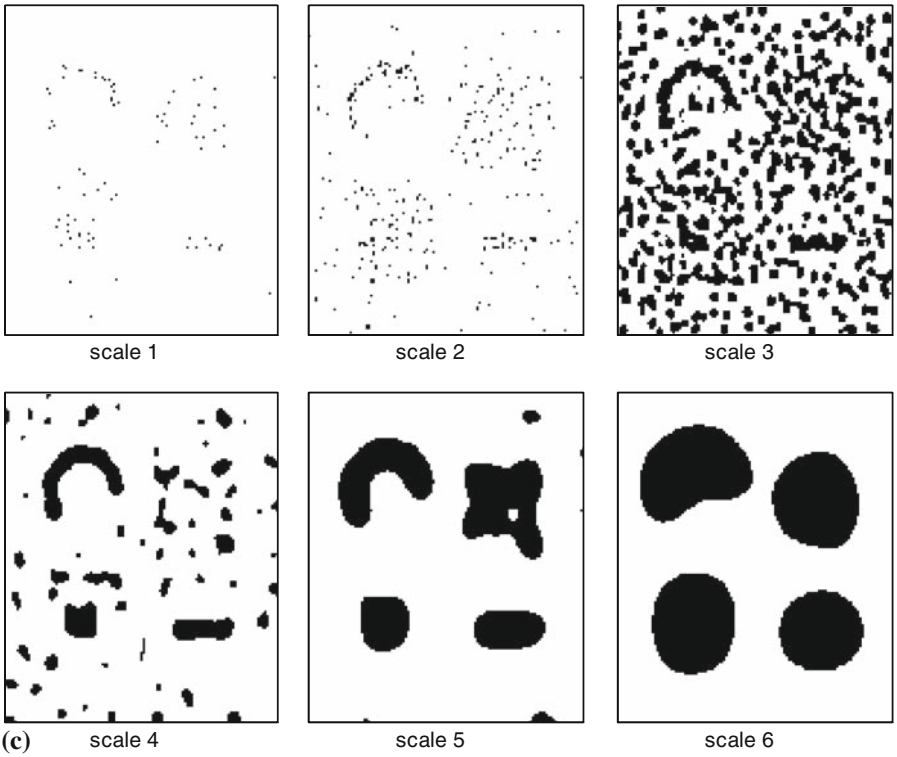

Fig. 5 continued

7.2 Experiment 2

In this experiment, we used a data set, which contains Gaussian clusters and noise, to test DECODE, DBSCAN, OPTICS and N-Wavecluster and to compare their efficiencies on the identification of clusters. The data of Experiment 2, shown in Fig. 7, lie within a $1,000 \times 1,000$ rectangle. There are two Gaussian clusters and Poisson noise in the point set. One of the Gaussian clusters (320 points), with high density, is distributed in a circle centered at (200, 300) with a diameter of 100 and the other Gaussian cluster (130 points), with low density, is distributed in a circle centered at (500, 600) with a diameter of 360. The Poisson noise points (262 points) are distributed over the whole area.

After setting σ, δ, λ_i, w_i and α to the same values as those set in Experiment 1 and $m = 18$, we ran DECODE on the point set. The performance of DECODE, shown in Fig. 8, indicates that the algorithm has converged (Fig. 8a) and the point set is composed of three processes (Fig. 8b). Figure 9a displays the histogram of the mth distances and the fitted curve, from which thresholds for classifying data are derived ($Eps_1^* = 23.2$, $Eps_2^* = 92.7$). The derived thresholds are very close to the true thresholds ($Eps_1 = 22.6$, $Eps_2 = 91.3$). The classification of DECODE, shown in Fig. 10a, reveals that DECODE automatically detects the number of point processes as

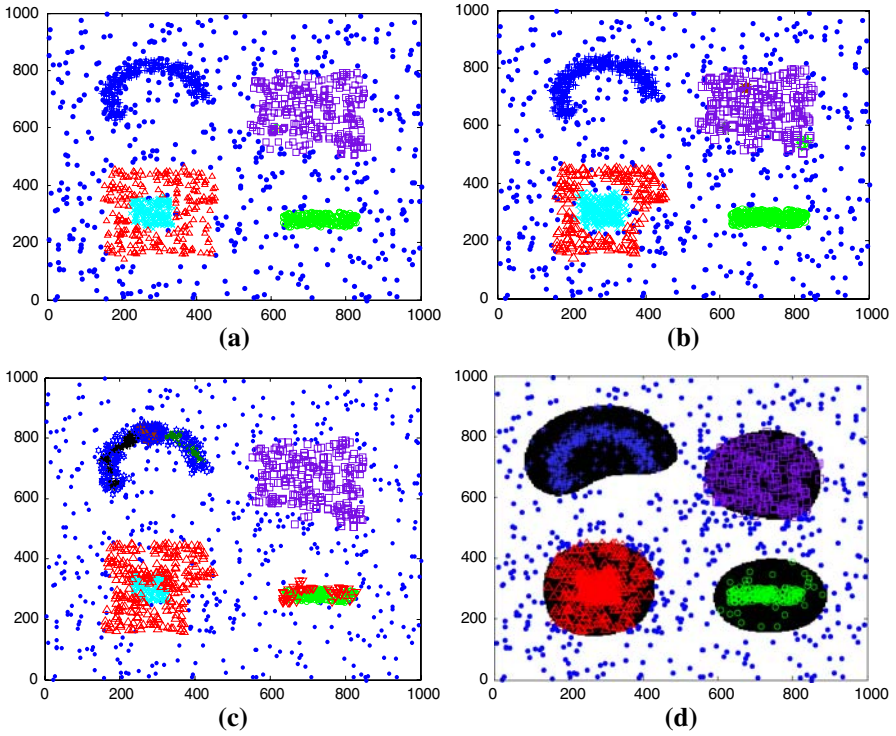

Fig. 6 The comparison between DECODE, DBSCAN, OPTICS and N-Wavecluster: **a** DECODE; **b** DBSCAN; **c** OPTICS; **d** N-Wavecluster

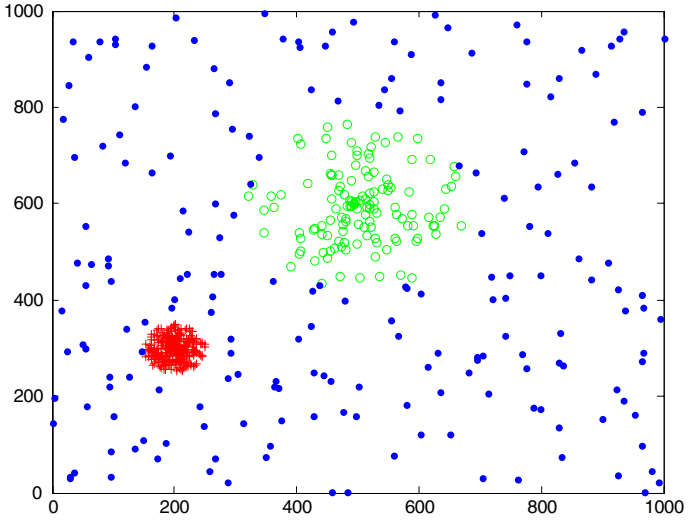

Fig. 7 The simulated data of Experiment 2 (Cluster 1, with high density, are symbolized by crosses; Cluster 2, with medium density, are symbolized by circles; noise, with low density, is symbolized by dot)

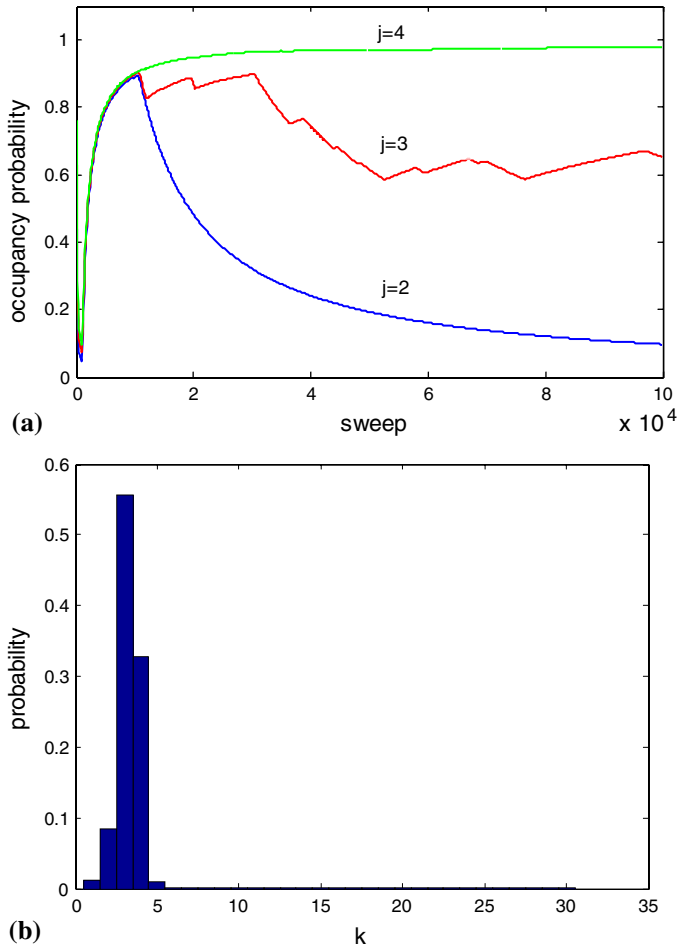

Fig. 8 Performance of DECODE on Gaussian clusters: **a** the cumulative occupancy probability; **b** the posterior probability of k

well as that of clusters. In addition, it classifies the points into two distinctive clusters and noise very successfully with only 19 misclassified points.

We then ran the other three methods, i.e. DBSCAN, OPTICS, N-Wavecluster, to estimate the thresholds for classifying points. The m-dist graph of DBSCAN (with $MinPts = 18$) is displayed in Fig. 9b. We find that it is not easy to determine the number of thresholds and whether there is any threshold between cluster of low density and noise. As a result, only one threshold ($Eps_1^* = 27$) could be detected and estimated. The classification result (Fig. 10b) shows that only the cluster with high density is identified and the number of misclassified points is 132.

According to the reachability-plot (with $MinPts = 18$ and $Eps' = 100$) constructed by OPTICS, shown in Fig. 9c, we may identify two "valleys" from the plot and estimate the threshold between clusters and noise, nevertheless, it is still difficult to determine if there is any threshold between clusters. When the threshold between clusters and

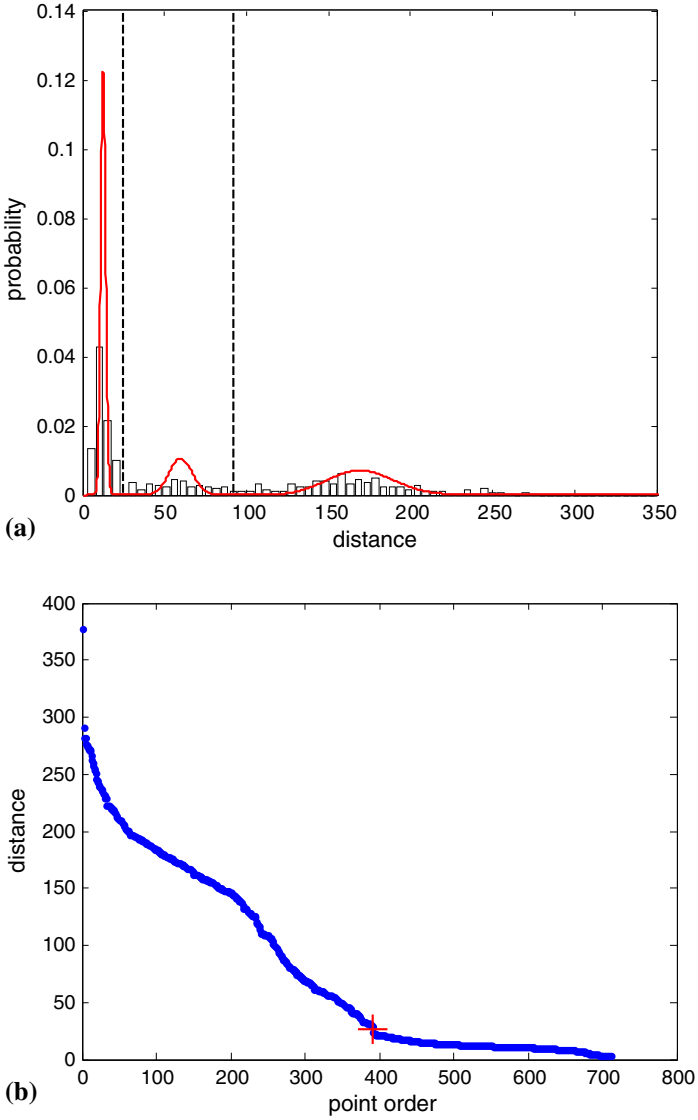

Fig. 9 Determination of thresholds for classifying Gaussian clusters and noise: **a** the histogram of the mth nearest distance ($m = 18$) and the fitted curve (the thresholds ($Eps_1^* = 23.2$, $Eps_2^* = 92.7$), estimated by DECODE, are indicated by vertically dashed lines while the true thresholds are $Eps_1 = 22.6$, $Eps_2 = 91.3$); **b** the m-dist graph ($MinPts = 18$) with the threshold (indicated by a cross); **c** the reachability-plot ($MinPts = 18$, Eps'=100) with the threshold ($Eps_1^* = 60$ indicated by a horizontally dashed line); **d** The multi-resolution wavelet representation of the cluster space

noise was set to 60 ($Eps_1^* = 60$), which we think is the optimum one, from the classification result of OPTICS (Fig. 10c) we found that the Cluster 1 is overestimated while Cluster 2 is underestimated. The number of misclassified points is 46.

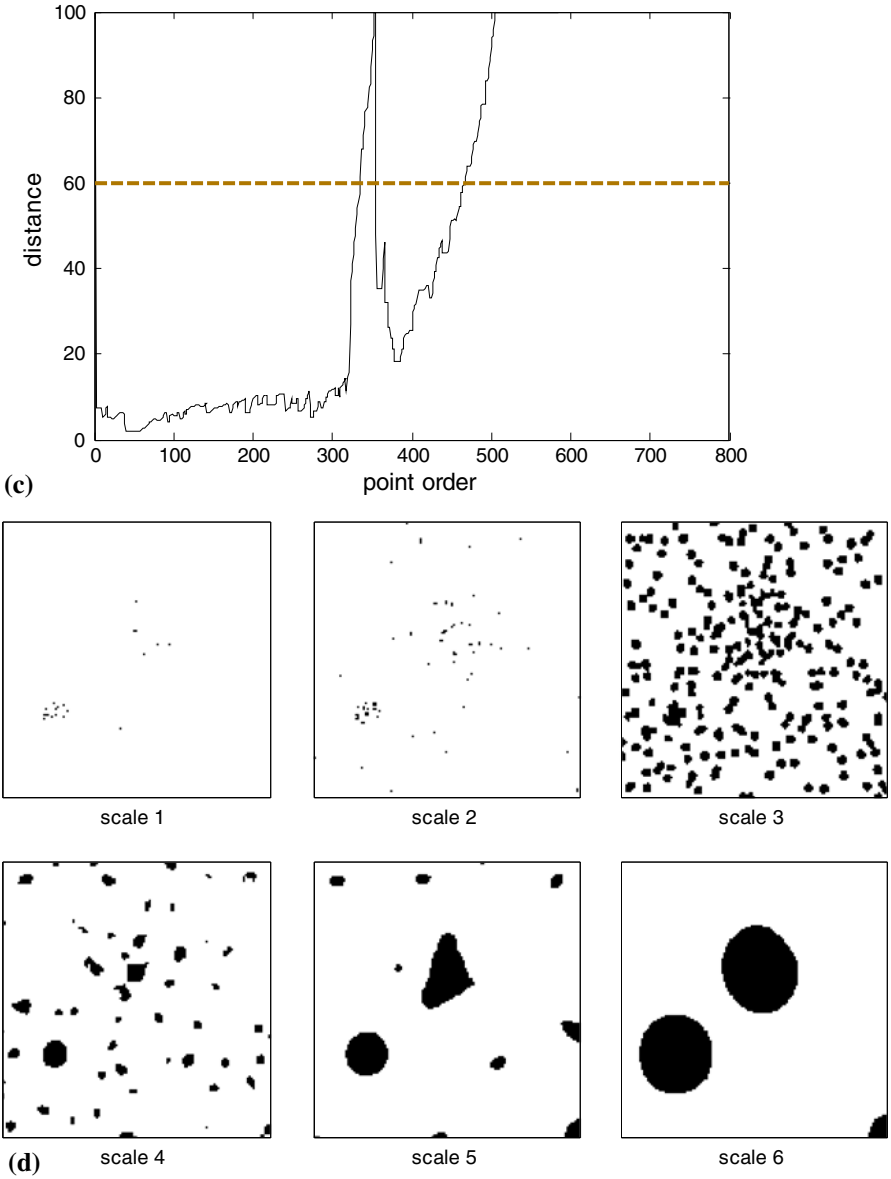

Fig. 9 continued

The wavelet representations at different scales, produced by N-Wavecluster, are shown in Fig. 9d. It appears that the representation at Scale 6 might reflect the structure of clusters. The enlarged classification result, shown in Fig. 10d, clearly delineates the cluster space. However, N-WaveCluster produced three clusters, among which the one at the right-bottom (symbolized by squares) is obviously a false positive.

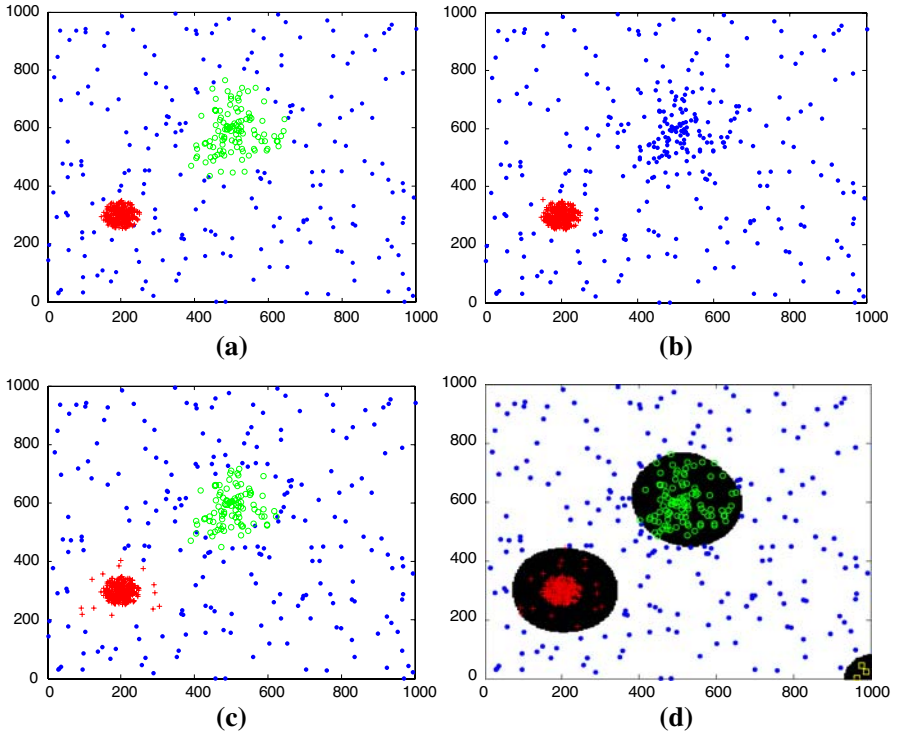

Fig. 10 The comparison between DEOCODE, DBSCAN, OPTICS and N-Wavecluster on classification of Gaussian clusters: **a** DECODE ($m = 18$); **b** DBSCAN ($MinPts = 18$, $Eps_1^* = 27$); **c** OPTICS ($MinPts = 18$, $Eps_1^* = 60$); **d** N-Wavecluster

Moreover, similar to OPTICS, Cluster 1 is overestimated and Cluster 2 is underestimated. The number of misclassified points is 53.

The comparison on Experiment 2 implies that the DECODE is capable of dealing with Gaussian clusters and present superior performance on parameter estimation and point classification.

7.3 Experiment 3

In the two following experiments, DECODE was applied to two seismic data sets to evaluate its efficiency on real data. In Experiment 3, we will use DECODE to identify the clustered earthquakes (seismic anomaly) in a seismic catalog. Clustered earthquakes are a swarm of earthquakes that are generated at a higher rate (time) and are distributed in a higher intensity (space) (Matsu'ura and Karakama 2005). As opposed to clustered earthquakes, background earthquakes are a number of small earthquakes that occur at a stable rate and are randomly located in a certain area (Wyss and Toya 2000 ; Pei et al. 2003). As a result, clustered earthquakes and background earthquakes can be seen as two point processes with different rates and intensities (Kagan and Houston 2005; Zhuang et al. 2005). Clustered earthquakes could be either precursors

that induce strong earthquakes or offspring that are triggered by strong earthquakes. The former are referred to as foreshocks and the latter as aftershocks. Foreshocks can indicate locations of strong earthquakes while aftershocks may help to understand the mechanism of the strong earthquakes (Reasenberg 1999; Umino et al. 2002). In this context, the key task is to identify clusters from the locations of earthquakes. Note that objects here are earthquakes and the attributes are the coordinates of the epicenter of each earthquake. However, the clustered earthquakes are difficult to extract because of the interference of background earthquakes. In this regard, the density-based cluster methods can be employed to separate the clustered earthquakes and background earthquakes. In the seismic case and hereafter, we only make the comparison between DECODE and OPTICS. The reason why we only choose OPTICS for the comparison is that: (1) this paper mainly discusses the parameter estimation problem of the distance-based method while N-Wavecluster is a grid-based method; (2) OPTICS provides a more efficient tool, the reachability-plot, for the estimation of the thresholds.

Our research area is located from 100° to 107° E and from 27° to 34° N. We selected the seismic records from two sources: Earthquake Catalogue in West China (1970–1975, M $\geq$ 1) (Feng and Huang 1980) and Earthquake Catalogue in West China (1976–1979, M $\geq$ 1) (Feng and Huang 1989). Note: M is the unit of the seismic record and represents the magnitude on the Richter scale. From these two sources, we selected the records between 15 February 1975 and 15 August 1976, with the magnitudes greater than 2.

Before running DECODE, we set σ, δ, α to the same values as those in Experiment 1 and also set $f_b = 500$. We chose $m = 12$ for calculating the mth nearest distance for two reasons. The first is because the value of m decides the minimum number of points in a cluster, and a value of m between 10 and 15 is large enough to help find an appropriate size of an anomaly. The second is to compare the result with that we achieved in Pei et al. (2006). We ran DECODE for 100,000 sweeps with 50,000 sweeps taken as the burn in. After the algorithm had converged, we displayed the posterior probabilities of k in Fig. 11a, which implies that the maximal posterior probability is acquired at $k = 2$. The histogram of the mth nearest distance and its fitted pdf are displayed in Fig. 11b. The classification result is shown in Fig. 12a. The clustered seismic data are divided into three groups, as A, B and C.

We then estimated the threshold for classification with the reachability-plot of seismic data (Fig. 13). Classification under the threshold ($Eps_1^* = 3.5 \times 10^4(m)$) produces four clusters (Fig. 12b). There are two discrepancies between the result produced by OPTICS and that by DECODE: one is that OPTICS produces one more cluster (Cluster D), the other is that Cluster A, B and C contain more earthquakes compared with those identified by DECODE.

Next, we evaluated these two classification results by the seismic records thereafter. The seismic catalog, recorded after 15 August 1976, was selected from the same sources as that of the data in the experiment. It shows that the anomalies identified by DECODE are more accurate indicators of the location of forthcoming strong earthquakes and offspring of strong earthquakes than those identified by OPTICS. From Fig. 12a, three seismic anomalies are detected by DECODE. Anomaly A are the foreshocks of the Songpan earthquake, which hit Songpan county at (32°42′ Nm 104°06′ E) on 16 August 1976 and was measured as 7.2M. Anomaly B are the aftershocks of the

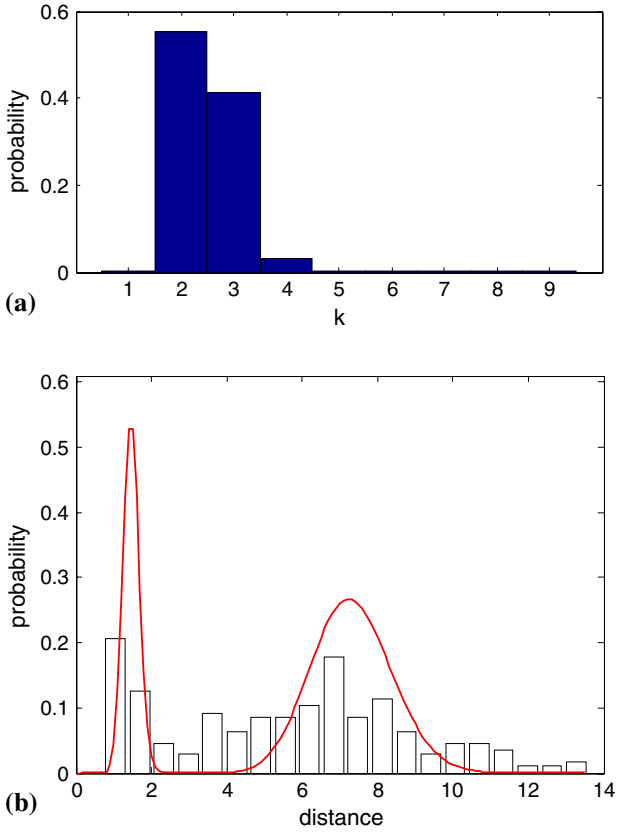

Fig. 11 Results of DECODE on the regional seismic catalog: **a** the posterior probability of k; **b** the histogram of the mth nearest distance ($m = 12$) and the fitted curve

Kangding–Jiulong event ($M = 6.2$), which occurred at (29°26′ N, 101°48′ E) on 15 January 1975. Anomaly C are the aftershocks of the Daguan event ($M = 7.1$), which struck at (28°06′ N 104°00′ E) on 11 May 1974. The detected anomalies are also similar to those in (Pei et al. 2006) both in terms of location and size. The only difference is that the numbers of earthquakes in the anomalies in Fig. 12a are slightly underestimated. This is because DECODE does not treat border points as members of the clusters while the approach in Pei et al. (2006) did. We also confirmed that earthquakes, which are classified as background earthquakes in Fig. 12a and appeared as clustered earthquakes in Pei et al. (2006), are border points. We then analyzed the result produced by OPTICS (Fig. 12b). The seismic records show that Cluster D is a false positive and other seismic anomalies are overestimated by OPTICS (some earthquakes in Cluster B and C have been confirmed as background earthquakes).

7.4 Experiment 4

In order to evaluate their efficiencies with multi-process real data, in this experiment we used DECODE and OPTICS to identify clusters in the strong earthquake data. As

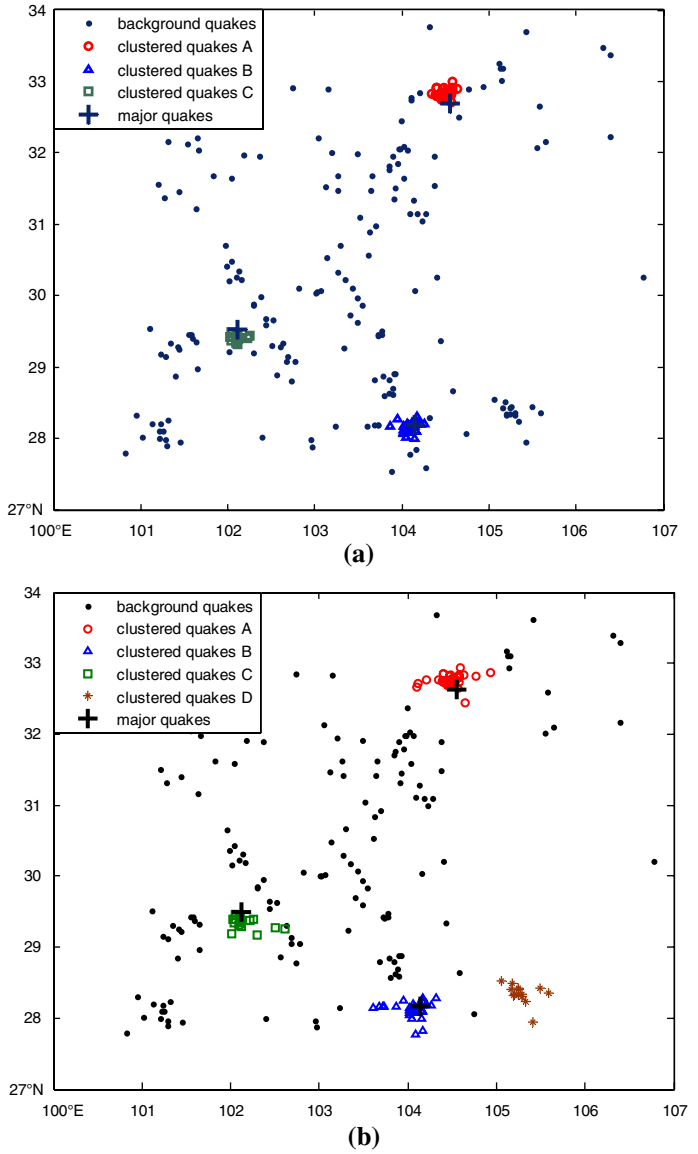

Fig. 12 Comparison on seismic anomaly detection between DECODE and OPTICS: **a** DECODE; **b** OPTICS

the distribution of strong earthquakes are, to a large extent, decided by global tectonics (Zhuang et al. 2005; Zhou et al. 2006), the strong earthquakes on a large scale, which is similar to small earthquakes on a small scale (in Experiment 3), is not completely randomly distributed. As a result, strong earthquakes can be seen as a mixture in which different point processes overlap. The identification of clustered strong earthquakes may help to understand the tectonic activities and to make a quakeproof plan.

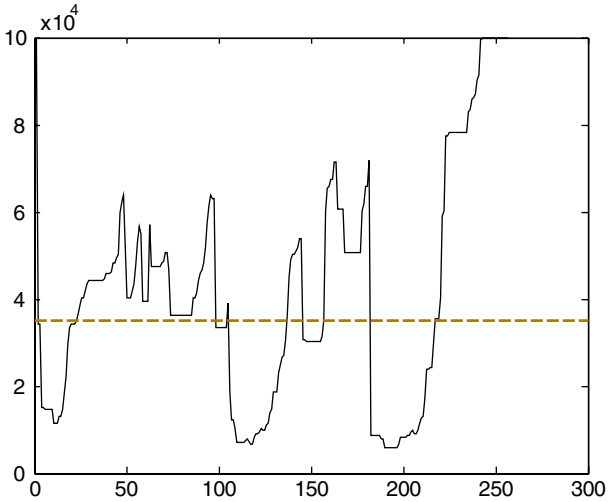

Fig. 13 The reachability-plot of the regional seismic data ($Eps_1^* = 3.5 \times 10^4(m)$, indicated by a horizontally dashed line)

The research area is located from 65° to 125° E and from 20° to 55° N. The area covers most of the land area of China and adjacent areas except for the eastern part of Northeastern China. The seismic data were selected from Chinese Seismic Catalog (1831BC–1969AD) (Gu 1983) and the China Seismograph Network (CSN) Catalog (China Seismograph Network 2008) and include records between 1 January 1900 and 31 December 2000. All included earthquakes were measured on the Richter Scale with the magnitudes greater than 6.5M.

We first set all parameters at the same values as those in Experiment 3 and ran DECODE for 100,000 sweeps taking 50,000 sweeps to be the burn in. We showed the result of DECODE on the strong earthquake data in Fig. 14 after DECODE had converged. The posterior probabilities of k shown in Fig. 14a indicate that the strong earthquakes are composed of five different point processes. The fitted mixture pdf of the mth nearest distance can be seen in Fig. 14b. Figure 15a shows the classification result, revealing five earthquake clusters with different densities and background earthquakes. It was also found that the clusters of strong earthquakes are concentrated in three areas. The first is in the northwestern part of the Uigur Autonomous Region (in northwest China) and the adjacent area outside China and consists of Cluster 1 and 2. The second is in southwestern China and the adjacent area in Southeast Asia and consists of Cluster 3 only. The third is in the island of Taiwan and the sea around it and consists of Cluster 4 and 5. These three areas are the most intensive seismic regions in China (Fu and Jiang 1997).

Next, we used OPTICS to classify the seismic data. The threshold ($Eps_1^* = 2.4 \times 10^5\ (m)$) was estimated from the reachability-plot ($MinPts = 12$, $Eps' = 6 \times 10^5\ (m)$), which appears to be the optimum value (see Fig. 16). Within the similar places in Fig. 15a, OPTICS produces three clustering areas, each of which contains only one cluster (see Fig. 15b). The difference between classifications generated by the two

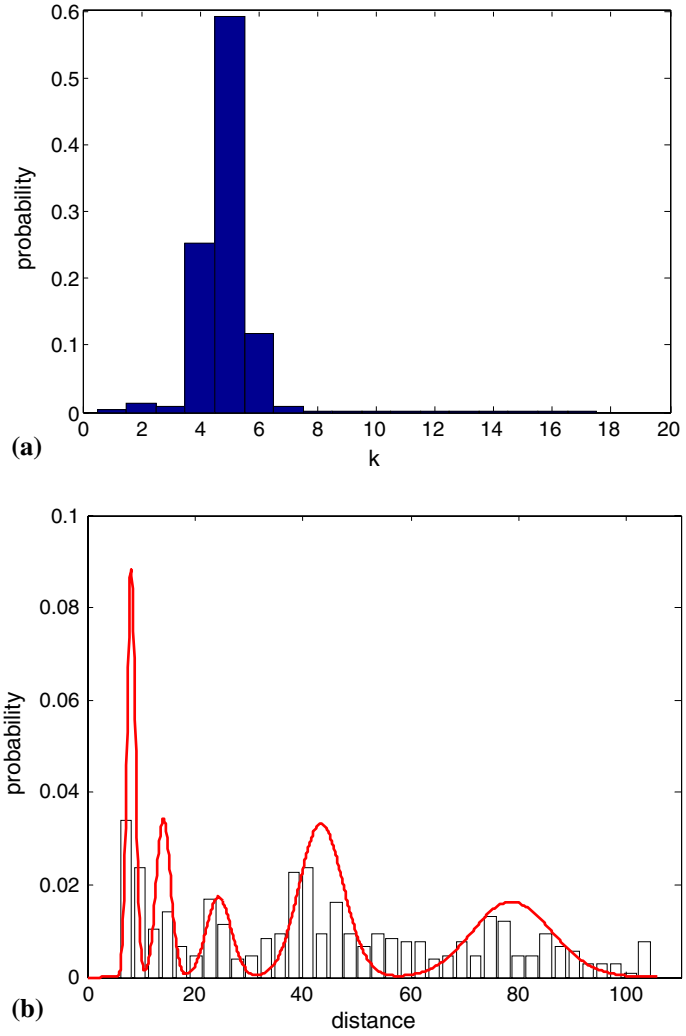

Fig. 14 Results of DECODE on the strong earthquake data: **a** the posterior probability of k; **b** the histogram of the mth nearest distance ($m = 12$) and the fitted curve

algorithms lies in two aspects: the first is the sizes of clustering areas and clusters; the second is the number of clusters (3 clusters are produced by OPTICS and 5 by DECODE). Regarding the first aspect, for instance, Cluster 2 in Fig. 15b, which is generated by OPTICS, is significantly smaller than Cluster 3 in the corresponding area of Fig. 15a, which is generated by DECODE.

As the distribution of the clustered strong earthquakes is in accordance with the pattern of tectonics and tectonic stress in East and Southeast Asia (Jiao et al. 1999), DECODE and OPTICS can be compared by analyzing the consistency between the tectonics and clusters detected by them. We first discuss the classification generated by DECODE. Earthquakes are concentrated in the first area and the second area because

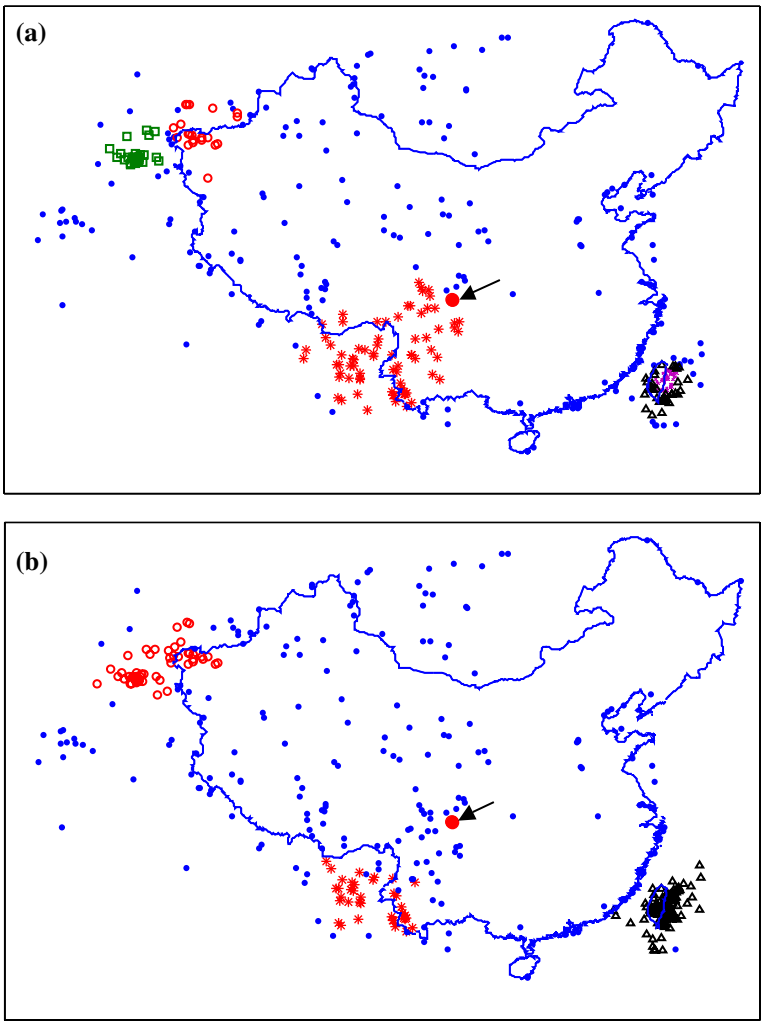

Fig. 15 Comparison on strong earthquake classification between DECODE and OPTICS: **a** DECODE (Strong clustered earthquakes in China and the adjacent areas, where Cluster 1 are symbolized by squares, Cluster 2 are symbolized by circles, Cluster 3 are symbolized by asterisks, Cluster 4 are symbolized by triangles, Cluster 5 are symbolized by rotated crosses, background earthquakes are symbolized by dots.); **b** OPTICS (Cluster 1 are symbolized by circles, Cluster 2 are symbolized by asterisks, Cluster 3 are symbolized by triangles, background earthquakes are symbolized by dots) (The Wenchuan earthquake is indicated by an arrow in both figures)

the Indian Plate is moving to the north and colliding with the Sino-Eurasian Plate (Ghosh 2002).[1] Research has shown that stress caused by the collision is concentrated on the first area and the second area in Fig. 15a and the maximum value of the stress is

[1] The updated seismic records of China also support our result since the Wenchuan earthquake (indicated by an arrow in Fig. 15), which occurred at 103.4°E, 31.0°N on 12 May, 2008, with the magnitude measured as 8.0 (China Seismograph Network (CSN) Catalog), is caused by the collision and located at the edge of Cluster 3 in Fig. 15a.

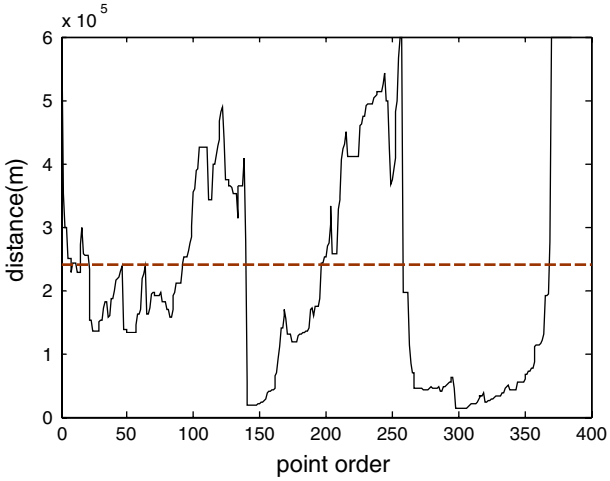

Fig. 16 The reachability-plot of the global strong earthquake data ($Eps_1^* = 2.4 \times 10^5(m)$, indicated by a horizontally dashed line)

found around the center of Cluster 1 in Fig. 15a (Jiao et al. 1999). The same situation also occurs in the island of Taiwan and the adjacent sea (the third area). This area is a stress concentrated region caused by the collision between the Philippine Sea Plate and the Sino-Eurasian Plate. It was also found that the denser cluster (Cluster 5 in Fig. 15a) is located in the eastern part of the island of Taiwan and the sea along its east coast. This is also consistent with the stress convergence point of the collision (Jiao et al. 1999), which is believed to form the Central Mountains extended northeasterly in the central east part of the island of Taiwan (Cheng 2002; Zhuang et al. 2005).

We then analyzed the result produced by OPTICS and found that it is inconsistent with the regional tectonics. In other words, OPTICS significantly underestimated Cluster 2 in Fig. 15b within the stress concentrated area (the second area in Fig. 15a) and failed to discern the inhomogeneity in clustering areas (Cluster 1 and 3 in Fig. 15b, which correspond to the first and the third area in Fig. 15a). This may be due to the estimation error caused by OPTICS in terms of the number of thresholds and values of thresholds.

Experiments on seismic data also show that DECODE is capable of identifying seismic clusters in real data sets of varied densities.

8 Conclusion and future work

Clusters and noise are regarded as arising from different spatial point processes, with different intensities. Present density-based cluster methods have made substantial progress in distinguishing clustered points from noise and grouping them into clusters. However, when several point processes overlap in a restricted area, few cluster methods can detect the number of point processes and group points into clusters in an accurate and objective way. In this paper, we have presented a new cluster method (DECODE),

which is based upon a reversible jump MCMC strategy, to discover clusters of different densities. The contribution of DECODE is that it is capable of determining the number of cluster types and the thresholds for identifying clusters of different densities with little prior knowledge. In fact, the method has only two parameters, i.e. f_b and m, left for a user to define. The experiments show that the appropriate value of f_b is between 50 and 1,000 and that the cluster result has low sensitivity to the values of f_b over this range. m can be decided according to the minimum cluster size which one intends to discover. The comparison studies on four experiments appear to show that DECODE outperforms DBSCAN OPTICS and N-Wavecluster in terms of the estimation of the number of point processes and their thresholds.

Although only seismic data have been used in the applications, we believe other spatial data, such as landslides, criminal venues and traffic accidents, may easily be analyzed with our method due to their resemblance in the data formation and the spatial distribution.

One limitation of the method is that the complexity of the algorithm is largely dependent on the sweep times, which are usually taken to be more than 50,000 times to ensure the convergence of the MCMC process. Subsequent work will be focused on finding more efficient algorithm to speed up the convergence.

Acknowledgement This study was funded through support from the National Key Basic Research and Development Program of China (Project Number: 2006CB701305), a grant from the State Key Laboratory of Resource and Environment Information System (Project Number: 088RA400SA) and the 'one Hundred Talents' Program of Chinese Academy of Sciences. Ajay Jasra was supported by a Chapman Fellowship, and David Hand was partially supported by a Royal Society Wolfson Research Merit Award.

References

Agrawal R, Gehrke J, Gunopulos D, Raghavan P (1998) Automatic subspace clustering of high dimensional data for data mining applications. In: Proceedings of the ACM SIGMOD '98 international conference on management of data, Seattle, WA, USA, pp 94–105

Allard D, Fraley C (1997) Nonparametric maximun likelihood estimation of features in spatial point process using voronoi tessellation. J Am Stat Assoc 92:1485–1493. doi:10.2307/2965419

Andrieu C, Freitas DN, Doucet A, Jordan IM (2003) An introduction to MCMC for machine learning. Mach Learn 50:5–43. doi:10.1023/A:1020281327116

Ankerst M, Breunig MM, Kriegel H-P, Sander J (1999) OPTICS: ordering points to identify the clustering structure. In: Proceedings of ACM-SIGMOD'99 international conference on management data, Philadelphia, USA, pp 46-60

Byers S, Raftery AE (1998) Nearest-neighbor clutter removal for estimating features in spatial point processes. J Am Stat Assoc 93:577–584. doi:10.2307/2670109

Cheng KH (2002) An analysis of tectonic environment and contemporary seismicity of frontal orogeny in central Taiwan area. Seismol Geol 24(3):400–411

China Seismograph Network (CSN) catalog available online at: http://www.csndmc.ac.cn. Accessed in 2008

Cressie NAC (1991) Statistics for spatial data, 1st edn. Wiley, New York

Daszykowski M, Walczak B, Massart DL (2001) Looking for natural patterns in data Part 1. Density-based approach. Chemom Intell Lab Syst 56:83–92. doi:10.1016/S0169-7439(01)00111-3

Diggle PJ (1985) A kernel method for smoothing point process data. Appl Stat 34:138–147. doi:10.2307/2347366

Ester M, Kriegel H-P, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proc. 2nd int. conf. on knowledge discovery and data mining, Portland, OR, pp 226–231

Feng H, Huang DY (1980) Earthquake catalogue in West China (1970—1975,M≥1). Seismological Press, Beijing (in Chinese)
Feng H, Huang DY (1989) Earthquake catalogue in West China (1976—1979,M≥1). Seismological Press, Beijing (in Chinese)
Fu ZX, Jiang LX (1997) On large-scale spatial heterogeneties of great shallow earthquakes and plates coupling mechanism in Chinese mainland and its adjacent area. Earthq Res China 13(1):1–9 (in Chinese)
Ghosh SC (2002) The raniganj coal basin: an example of an Indian Gondwana rift. Sediment Geol 147(Sp. Iss.):155–176
Green PJ (1995) Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. Biometrika 82:711–732. doi:10.1093/biomet/82.4.711
Gu GX (1983) Chin seismic catalog (1831 BC-1969 AD). Science Press, Beijing
Han JW, Kamber M, Tung AKH (2001) Spatial clustering methods in data mining. In: Miller HJ, Han JW (eds) Geographic data mining and knowledge discovery. Taylor & Francis, London, pp 188–217
Hinneburg A, Keim DA (1998) An efficient approach to clustering in large multimedia databases with noise. In: Proceedings of the knowledge discovery and data mining, pp 58–65
Jain AK, Dubes RC (1988) Algorithms for clustering data. Prentice Hall, Englewood Cliffs
Jasra A, Stephens DA, Gallagher K, Holmes CC (2006) Bayesian mixture modelling in geochronology via Markov chain Monte Carlo. Math Geol 38:269–300. doi:10.1007/s11004-005-9019-3
Jiao MR, Zhang GM, Che S, Liu J (1999) Numerical calculations of tectonic stress field of Chinese mainland and its neighboring regions and their applications to explanation of seismic activity. Acta Seismologica Sin 12(2):137–147. doi:10.1007/s11589-999-0018-1
Kagan YY, Houston H (2005) Relation between mainshock rupture process and Omori's law for aftershock moment release rate. Geophys J Int 163:1039–1048
Kaufman L, Rousseeuw P (1990) Finding groups in data: an introduction to cluster analysis. Wiley, New York
Lin CY, Chang CC (2005) A new density-based scheme for clustering based on genetic algorithm. Fundam Inform 68:315–331
Liu P, Zhou D, Wu NJ (2007) VDBSCAN: varied density based spatial clustering of applications with noise. In: Proceedings of IEEE international conference on service systems and service management, Chengdu, China, pp 1–4
Markus MB, Kriegel H-P, Raymond TN, Sander J (2000) LOF: identifying density-based local outliers. In: Proceedings of ACM SIGMOD of 2000 international conference on management of data, vol 29, pp 93–104
Matsu'ura RS, Karakama I (2005) A point-process analysis of the Matsushiro earthquake swarm sequence: the effect of water on earthquake occurrence. Pure Appl Geophys 162:1319–1345. doi:10.1007/s00024-005-2672-0
Murtagh F, Starck JL (1998) Pattern clustering based on noise modeling in wavelet space. Pattern Recogn 31(7):847–855. doi:10.1016/S0031-3203(97)00115-5
Neill DB (2006) Detection of spatial and spatio-temporal clusters. Ph.D. Thesis of University of South Carolina
Neill DB, Moore AW (2005) Anomalous spatial cluster detection. In: Proceeding of KDD 2005 workshop on data mining methods for anomaly detection, Chicago, Illinois, USA, pp 41–44
Pascual D, Pla F, Sanchez JS (2006) Non parametric local density-based clustering for multimodal overlapping distributions. In: Proceedings of intelligent data engineering and automated learning (IDEAL2006), Spain, Burgos, pp 671–678
Pei T, Yang M, Zhang JS, Zhou CH, Luo JC, Li QL (2003) Multi-scale expression of spatial activity anomalies of earthquakes and its indicative significance on the space and time attributes of strong earthquakes. Acta Seismologica Sin 3:292–303. doi:10.1007/s11589-003-0033-6
Pei T, Zhu AX, Zhou CH, Li BL, Qin CZ (2006) A new approach to the nearest-neighbour method to discover cluster features in overlaid spatial point processes. Int J Geogr Inf Sci 20:153–168. doi:10.1080/13658810500399654
Reasenberg PA (1999) Foreshock occurrence rates before large earthquakes worldwide. Pure Appl Geophys 155:355–379. doi:10.1007/s000240050269
Richardson S, Green PJ (1997) On Bayesian analysis of mixtures with an unknown number of components. J Roy Stat Soc Ser B-Methodol 59:731–758
Robert CP, Casella G (2004) Monte Carlo statistical methods, 2nd edn. Springer, New York

Roy S, Bhattacharyya DK (2005) An approach to find embedded clusters using density based techniques. Lect Notes Comput Sci 3816:523–535. doi:10.1007/11604655_59

Sander J, Ester M, Kriegel H, Xu X (1998) Density-based clustering in spatial databases: the algorithm GDBSCAN and its applications. Data Min Knowl Discov 2:169–194. doi:10.1023/A:1009745219419

Sheikholeslami G, Chatterjee S, Zhang A (1998) WaveCluster: a multi-resolution clustering approach for very large spatial databases. In: Proceedings of the 24th international conference on very large data bases, New York City, NY, pp 428-439

Thompson HR (1956) Distribution of distance to nth nearest neighbour in a population of randomly distributed individuals. Ecology 27:391–394. doi:10.2307/1933159

Tran TN, Wehrensa R, Lutgarde MCB (2006) KNN-kernel density-based clustering for high-dimensional multivariate data. Comput Stat Data Anal 51:513–525. doi:10.1016/j.csda.2005.10.001

Umino N, Okada T, Hasegawa A (2002) Foreshock and aftershock sequence of the 1998 M $\geq$ 5.0 Sendai, northeastern Japan, earthquake and its implications for earthquake nucleation. Bull Seismol Soc Am 92:2465–2477. doi:10.1785/0120010140

Wyss M, Toya Y (2000) Is background seismicity produced at a stationary Poissonian rate. Bull Seismol Soc Am 90:1174–1187. doi:10.1785/0119990158

Zhang GM, Ma HS, Wang H, Wang XL (2005) Boundaries between active-tectonic blocks and strong earthquakes in the China mainland. Chin J Geophys 48:602–610

Zhou CH, Pei T, Li QL, Chen JB, Qin CZ, Han ZJ (2006) Database of Integrated Catalog of Chinese earthquakes and Its Application. Water and Electricity Press, Beijing (in Chinese)

Zhuang JC, Chang CP, Ogata Y, Chen YI (2005) A study on the background and clustering seismicity in the Taiwan region by using point process models. J Geophys Res Solid Earth 110(B05S18). doi:10.1029/2004JB003157