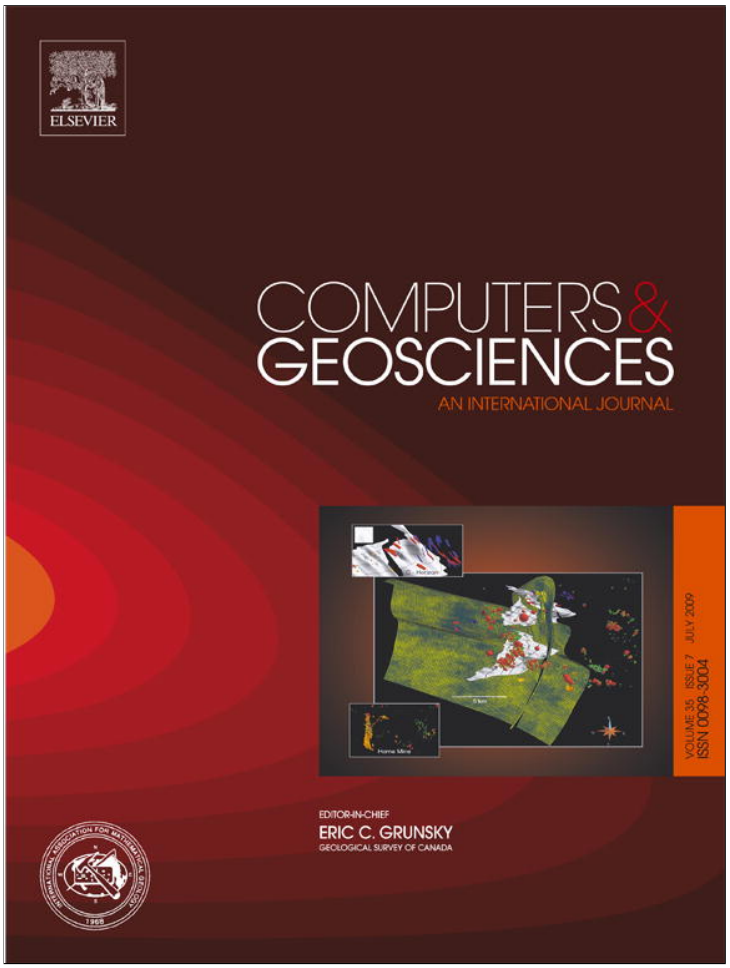
Provided for non-commercial research and education use.
Not for reproduction, distribution or commercial use.



This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

http://www.elsevier.com/copyright

Author's personal copy

Computers & Geosciences 35 (2009) 1508–1516

Contents lists available at ScienceDirect

Computers & Geosciences

journal homepage: www.elsevier.com/locate/cageo

Discovery of feature-based hot spots using supervised clustering

Wei Ding [a,*], Tomasz F. Stepinski [b], Rachana Parmar [c], Dan Jiang [c], Christoph F. Eick [c]

[a] Department of Computer Science, University of Massachusetts Boston, Boston, MA 02125-3393, USA
[b] Lunar and Planetary Institute, 3600 Bay Area Blvd., Houston, TX 77058, USA
[c] Department of Computer Science, University of Houston, Houston, TX 77204-3010, USA

ARTICLE INFO

Article history:
Received 7 September 2007
Received in revised form
6 October 2008
Accepted 11 October 2008

Keywords:
Hot spots
Clustering
Spatial data mining
Mars

ABSTRACT

Feature-based hot spots are localized regions where the attributes of objects attain high values. There is considerable interest in automatic identification of feature-based hot spots. This paper approaches the problem of finding feature-based hot spots from a data mining perspective, and describes a method that relies on supervised clustering to produce a list of hot spot regions. Supervised clustering uses a fitness function rewarding isolation of the hot spots to optimally subdivide the dataset. The clusters in the optimal division are ranked using the interestingness of clusters that encapsulate their utility for being hot spots. Hot spots are associated with the top ranked clusters. The effectiveness of supervised clustering as a hot spot identification method is evaluated for four conceptually different clustering algorithms using a dataset describing the spatial distribution of ground ice on Mars. Clustering solutions are visualized by specially developed raster approximations. Further assessment of the ability of different algorithms to yield hot spots is performed using raster approximations. Density-based clustering algorithm is found to be the most effective for hot spot identification. The results of the hot spot discovery by supervised clustering are comparable to those obtained using the G* statistic, but the new method offers a high degree of automation, making it an ideal tool for mining large datasets for the existence of potential hot spots.

© 2009 Elsevier Ltd. All rights reserved.

1. Introduction

Spatial datasets abound in geosciences, making it difficult for the research community to turn all this data into knowledge. One solution is to apply spatial data mining techniques to geospatial datasets in order to automatically discover interesting relations or places that may exist in the dataset. Existing works on spatial data mining (Koperski and Han, 1995; Munro et al., 2003; Huang et al., 2004, 2006; Zhang et al., 2004) tend to focus on discovering systematic relations between spatial variables. For example, in a spatial co-location problem (Huang et al., 2004, 2006; Zhang et al., 2004), the goal is to find subsets of features that are located together in spatial proximity, throughout the spatial extent of the dataset. In other words, the goal is to discover globally valid proximity relationships between certain features. On the other hand, less attention has been given to the discovery of feature-based hot spots in spatial datasets. The term "hot spots" is most often used to describe clustered point-event patterns. Such hot spots are determined only by arrangement of the objects' spatial coordinates without taking into account the attribute values of the data. However, in this paper, we are concerned with feature-based hot spots—localized regions of high or low attribute values.

Multi-feature hot spots—places where multiple features attain values from the tails of their respective distributions—are of special interest. Multi-feature-based hot spots are interesting because they may indicate a rare local process or an unlikely set of circumstances that forces several features to have non-average values sync with one another. For example, in a dataset describing spatial distribution of temperature, humidity, and vegetation cover across a given area, a region may be found that is characterized by high temperature, low humidity, and dense vegetation cover. Such a place is worthy of closer examination, as this particular combination of variables is unexpected (vegetation is usually poor in hot and dry places). Closer examination may reveal an *a priori* unknown factor that accounts for such a combination (possibly irrigation).

The presently popular method of finding feature-based hot spots in spatial datasets relies on the G* statistic (Getis and Ord, 1992; Ord and Getis, 1995). The G* statistic detects local pockets of spatial association. The value of G* depends on an *a priori* given scale of the packets and is calculated for each object individually. Graphical visualization of the results of G* calculations reveals hot spots (aggregates of objects with values of G* higher than expected) and cold spots (aggregates of objects with values of G* lower than expected). Note that such aggregates are not formally-defined clusters, as the G*-based method has no built-in

* Corresponding author.
E-mail addresses: ding@cs.umb.edu (W. Ding), tstepinski@lpi.usra.edu (T.F. Stepinski), rparmar@uh.edu (R. Parmar), djiang@uh.edu (D. Jiang), ceick@uh.edu (C.F. Eick).

0098-3004/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cageo.2008.10.007

Author's personal copy

clustering capabilities. Instead, hot spots are inferred from visualization, utilizing the ability of the human brain to isolate "clusters" in an image.

Our proposed method offers an alternative approach for identification of hot spots. It does not rely on local statistics; instead, it is rooted in data mining methodology, and, in particular, takes advantage of the notion of supervised clustering. Supervised clustering (Eick et al., 2004) uses a fitness function in order to maximize the purity of the clusters. The fitness function is constructed to reward aggregated objects having non-average values of their features. When subjected to such a fitness function, the clustering procedure is guided toward a solution that emphasizes hot spots. Thus, in our method hot spots are identified as formal clusters of objects—visualization is not necessary for their recognition. This makes our method especially useful in the context of automated mining of large datasets for identifying potentially interesting hot spots. For example, in the presence of multiple features, our method can be set up to compile a database of all possible hot spots, including hot spots of individual features, all combinations of double-feature hot spots, etc.

Methods of finding geometrically defined hot spots have been investigated in the past both explicitly and implicitly. Because the geometrically defined hot spots are clusters with respect to spatial coordinates, their detection lies at the heart of spatial data mining and has been investigated in Murray and Estivill-Castro (1998), Openshaw (1998) and Miller and Han (2001). More explicitly, detection of hot spots using a variable resolution approach (Brimicombe, 2005) was investigated in order to minimize the effects of spatial superposition. In Tay et al. (2003), a region-growing method for the discovery of hot spots was described, which selects seed points and then grows clusters from these seed points by adding neighboring points as long as a density threshold condition is satisfied. Definition of hot spots was extended in Williams (1999) and Kulldorff (2001) to cover a set of entities that are of some particular, but crucial, importance to the experts. This is a feature-based definition, somewhat similar to, but less specific than, what we are using in the present paper. This definition was applied to relational databases of spatio-temporal domain to find important nuggets of information. As mentioned earlier, an approach to identifying feature-based hot spots based on local statistics was developed in Getis and Ord (1992) and Ord and Getis (1995). Finally, in Eick et al. (2006), feature-based hot spots are defined in a similar sense, as in this paper, but their discovery is limited to single-feature datasets.

The overall framework of using the concept of supervised clustering for the identification of hot spots is presented in Section 2.1. Section 2.2 presents a description of four conceptually different clustering algorithms considered for use within the supervised framework. We report on the effectiveness of our method in Section 3 by evaluating a case study pertaining to the spatial distribution of ground ice on Mars. The optimal clustering solutions are subjected to detailed statistical analysis, with the aim of identifying the clustering algorithm best suited to the task of finding hot spots. In Section 4, we present an ancillary method aimed at transforming a clustering solution into a segmentation solution. The difference between a cluster and a segment is that whereas a cluster is a set of objects, a segment is defined as a polygon that has an area and transparent neighborhood relations with other segments. Thus, a segmentation solution can be subjected to additional statistical analysis that is not practical for a clustering solution; the result of such an analysis allows for additional discrimination between different clustering algorithms. For the end user, the segmentation solution provides more effective visualization, and facilitates a query of identified hot spots by additional attributes related to the properties of the area they occupy. Discussion and future work directions are given in Section 5.

2. Supervised clustering methodology

2.1. Framework

The relevant dataset consists of point objects, each characterized by a list of real-valued features. The basic tenet of our approach is to use a clustering algorithm to divide a dataset O into a set of clusters $X = \{c_1, \ldots, c_k\}$, $c_i \subseteq O$, in such a way as to maximize a fitness function $q(X)$. The clusters are disjointed and contiguous but not exhaustive; some objects in O may not be assigned to any cluster. The number of clusters, k, is either set *a priori* or the best value of k is determined by clustering algorithms, depending on the capabilities of clustering techniques.

For the task of hot spot identification, the fitness function must be constructed to reward isolation of hot spots. We propose the following fitness function q:

$$q(X) = \sum_{c \in X} (i(c) \times \|c\|^{\beta}) \tag{1}$$

where $i(c)$ is the interestingness measure of a cluster c—a quantity designed to reflect the degree to which clusters can be considered hot spots. The region "size" (number of objects in the cluster) is denoted by $\|c\|$, and the quantity $(i(c) \times \|c\|^{\beta})$ is a "reward" given to a cluster c. A cluster reward is proportional to its interestingness, but a bigger cluster receives a higher reward than a smaller cluster having the same value of interestingness to reflect a preference given to larger clusters. The premium put on the size of the cluster is controlled by the user-determined value of the parameter $\beta > 0$. We seek a clustering solution X such that the sum of rewards over all of its constituent clusters is maximized.

2.2. Interestingness of clusters

An entry in a geospatial dataset has the form $(\langle spatialcoordinates \rangle, \langle feature_1 \rangle, \ldots, \langle feature_m \rangle)$, where m is the number of features. The numerical values of the features come from their respective distributions, which could have quite different functional forms. Therefore, it is necessary to normalize the values of different features to a common meaning. The two most important properties of any distribution are its center (S), which indicates the location of the bulk of the data, and its scale (σ), which indicates dispersion around the center. For features having bell-shaped distributions, S and σ are easily estimated using the mean and standard deviation, respectively. However, mean and standard deviation are biased estimates of S and σ for features with skewed distributions of their values. Thus, in general, S and σ should be calculated using more robust statistical estimators. For S, a robust estimator is the trimmed mean calculated by discarding a certain percentage of the lowest and the highest values. Note that the median is a particular example of the trimmed mean. For σ, a number of robust estimators are used, including the median absolute deviation (MAD) as well as S_n and Q_n estimators introduced by Rousseeuw and Croux (1993).

Regardless of the method used to estimate S and σ, the feature values are transformed into their z-scores, $z_j = (x_j - S_j)/\sigma_j$, $j = 1, \ldots, m$. Strictly speaking, the term "z-score" is used only in the context of S being the mean and σ being the standard deviation, but an extension of that term to data normalization using any estimate of the center and the scale is quite natural. The z-score is the number of scales of a given feature value above or below its center. In this case, the centers of all features are transformed to 0; the positive values of z indicate upward

Author's personal copy

deviations from the centers, whereas the negative values of z indicate downward deviations from the centers. The objects in the transformed database, $O = (\langle spatial\ coordinates\rangle, \langle z_1\rangle, \ldots, \langle z_m\rangle)$, are normalized inasmuch as the same values of different transformed features indicate the same amount of relative deviations from the locations of their centers.

Our approach employs an interestingness function i on top of the transformed dataset O: for a given set of m features, the interestingness of an object $o \in O$ is measured by $z(o)$ defined as follows:

$$z(o) = z_1(o) \times \cdots \times z_m(o). \tag{2}$$

Objects with $|z(o)| \gg 0$ are in locations where the features have values from the tails of their respective distributions. The interestingness of a cluster is computed as the average interestingness of the objects belonging to it

$$i(c) = \begin{cases} \left(\left|\frac{\Sigma_{o\in c} z(o)}{\|c\|}\right| - z_{\text{th}}\right) & \text{if } \left|\frac{\Sigma_{o\in c} z(o)}{\|c\|}\right| > z_{\text{th}}, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

In Eq. (3), the threshold z_{th} is introduced to weed out (possibly large) regions with $i(c)$ close to 0, so they do not contribute to the fitness function $q(X)$. The interestingness threshold z_{th} prevents solutions from containing only large clusters of low interestingness.

2.3. Clustering algorithms

Our method works with any clustering algorithm, but not all the clustering algorithms are expected to be equally suitable for the discovery of hot spots. We evaluate which of the major approaches to clustering yields the best results. To this end we adapt four different algorithms with our fitness function $q(X)$ (Eq. 1) exemplifying representative-based, agglomerative, grid-based, and density-based approaches to clustering.

Representative-based clustering algorithms. Representative-based clustering algorithms, sometimes called prototype-based clustering algorithms in the literature, construct clusters by seeking a set of representatives; clusters are then created by assigning objects in the dataset to the closest representatives. We use a modification of the "partition around medoid" (PAM) algorithm (Kaufman and Rousseeuw, 1990), which we refer to as SPAM (Supervised PAM). SPAM starts its search with a randomly created set of representatives, and then greedily replaces representatives with non-representatives as long as $q(X)$ improves. SPAM requires a number of clusters, k, to be set *a priori*.

Agglomerative algorithms. Due to the fact that representative-based algorithms construct clusters using nearest neighbor queries, the shapes of clusters that can be discovered are limited to convex polygons (Voronoi cells). However, interesting regions, hot spots in particular, may not be restricted to convex shapes. Agglomerative clustering algorithms are capable of yielding solutions with clusters of arbitrary shapes by constructing unions of small convex polygons. We use the MOSAIC algorithm (Choo et al., 2007), which uses a set of small convex clusters as its input. In our implementation, the input is provided by the SPAM solution. The algorithm is then modified to greedily merge neighboring clusters as long as $q(X)$ improves. Gabriel graphs (Gabriel and Sokal, 1969) are used to determine which clusters are neighbors. The number of clusters, k, is then determined by the clustering algorithm itself.

Grid-based algorithms. SCMRG (Supervised Clustering using Multi-Resolution Grids) (Eick et al., 2006) is a hierarchical, grid-based method that utilizes a divisive, top down search. The spatial space of the dataset is partitioned into grid cells. Each grid cell at a higher level is partitioned further into smaller cells at the lower level, and this process continues if the sum of the rewards of the $q(X)$ of the lower level cells is not decreased. The regions returned by SCMRG usually have different sizes, because they were obtained at different levels of resolution. Moreover, a cell is partitioned further only if it improves its fitness at a lower level of resolution. The number of clusters, k, is calculated by the clustering algorithm.

Density-based algorithms. Density-based algorithms work on the idea that the influence of each data point can be modeled using an *influence function*. The clusters are extracted from an overall *density function*, a sum of the influence functions of all the data points. We adapt an SCDE (Supervised Clustering Using Density Estimation) algorithm (Jiang et al., 2007) to feature-based hot spot discovery. Each object o in our database is assigned a value of $z(o)$ (see Eq. (2); positive and negative values of $z(o)$ indicate different types of dependence among the underlying features. The influence function of object o, $f_{Gauss}(p,o)$, is defined as a weighted influence function of the product of $z(o)$ and a Gaussian kernel in order to prevent grouping points with positive and negative values of z into the same cluster

$$f_{Gauss}(p,o) = z(o)\, e^{-d(p,o)^2/2\sigma^2}. \tag{4}$$

The parameter σ determines the strength of data point influence and $d(p,o)$ is the distance between object o and p.

The density function, $\Psi(p)$ at point p, is then computed as:

$$\Psi(p) = \sum_{o\in O} f_{Gauss}(p,o). \tag{5}$$

Unlike traditional density estimation techniques, which only consider the spatial coordinates of data points, our density function also takes into account the non-spatial feature of interest $z(o)$ in its influence function. SCDE uses a hill-climbing algorithm to compute locations of the local maxima, as well as the local minima of the density function Ψ. These locales act as cluster attractors; clusters are formed by associating objects in the database with the attractors. For a maximum-derived attractor, a cluster contains all objects whose density attractor has a density $\Psi(o) > \xi_{max}$, and for a minimum-derived attractor, a cluster contains all objects whose density attractor has a density $\Psi(o) < \xi_{min}$, where the density thresholds ξ_{max} and ξ_{min} are user-defined parameters. The clusters encountered in the hill-climbing search path are greedily merged as long as $q(X)$ improves. The number of clusters, k, is the result of the calculation.

3. Example: distribution of ground ice on Mars

3.1. Dataset description

We empirically evaluate our method on a dataset that pertains to the spatial distribution of ground ice on the planet Mars. This particular case study was selected because it is the focus of one of the authors' (TFS) ongoing research into the properties of the Martian subsurface. It is widely believed (Clifford, 1993) that a significant quantity of water resides in the Martian subsurface in the form of ground ice. Ice may be the only source of water on the planet outside of polar caps, and understanding its distribution is an important goal of the planetary science community.

Two different features pertaining to ground ice can be extracted from the data; we refer to these as "shallow-ice" and "deep-ice" features, because they pertain to the abundance of ice in the shallow and deep subsurface, respectively. Values of the "shallow-ice" features are obtained remotely from orbit by the gamma-ray spectrometer (Feldman, 2004), which measures an abundance of hydrogen, a telltale sign of ice in the upper 1

Author's personal copy

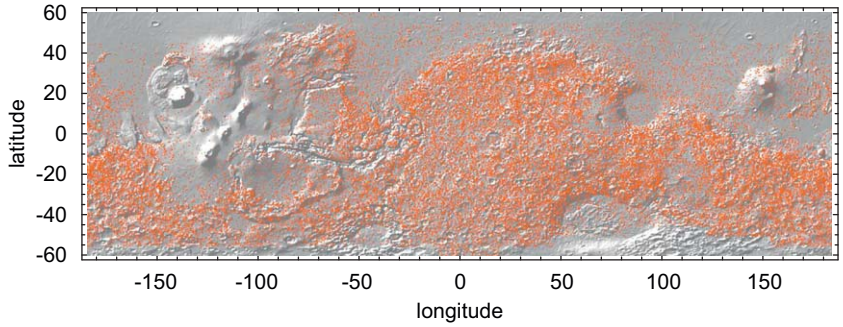


Fig. 1. Red dots shows locations of objects (craters) in our dataset. Grayscale background depicts elevation of Martian surface between longitude of −180–180° and latitude of −60–60°. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

meter of the subsurface. These measurements are translated into the percentage of ice present in the top soil by mass and reported on a grid having $5^\circ \times 5^\circ$ resolution. In the equatorial regions of Mars, values of up to 8% are reported. Values of "deep-ice" features are inferred from the spatial distribution of so-called rampart craters (Barlow, 1988). Rampart craters, which constitute about 20% of all Martian craters, are surrounded by ejecta that have patterns looking like splashes and are thought to form in locations rich in subsurface ice. The locally defined relative abundance of rampart craters can be considered a proxy for the abundance of "deep-ice," located at depths of up to few kilometers. We calculate the relative abundance of rampart craters using a $5^\circ \times 5^\circ$ moving window technique applied to the Martian crater database (Barlow, 1988).

For the purpose of our evaluation, the values of shallow-ice $\langle feature_1 \rangle$ and deep-ice $\langle feature_2 \rangle$ are reported at the locations of 35 927 craters catalogued between latitudes of -60° and 60° as shown on Fig. 1. Both features have unimodal distributions. The statistics of the shallow-ice distribution are: mean = 4.11, median = 3.88, standard deviation = 1.11, $S_n = 0.93$. The statistics of the deep-ice distribution are: mean = 0.21, median = 0.17, standard deviation = 0.19, $S_n = 0.09$. The deep-ice feature has a distribution somewhat skewed to the right; nevertheless, for our evaluation purposes, we use the mean and the standard deviation (the best known estimates) to normalize the features. The transformed dataset is $O = (\langle longitude, latitude \rangle_i, \langle z_1 \rangle_i, \langle z_2 \rangle_i)$, $i = 1, \ldots, 35\,927$, where z_1 is the z-score of the shallow-ice variable and z_2 is the z-score of the deep-ice variable.

Three different types of hot spots are potentially present in this dataset: hot spots of shallow-ice, hot spots of deep-ice, and double-feature hot spots that take into account the values of both features. In this dataset, the discovery of double-feature hot spots is most interesting, because planetary scientists are interested in knowing the locations on the surface of Mars where extreme values of shallow- and deep-ice abundances coincide, or where a high/low combination of the two ground ice indicators is present. Such knowledge provides insight into the history of water on Mars.

Table 1
Parameters of clustering algorithms used in our experiments.

Algorithm	Parameters	
	$\beta = 1.01$	$\beta = 1.2$
SPAM	$k = 2000$	$k = 807$
MOSAIC	Input is a SPAM clustering	
SCMRG	None	
SCDE	$\sigma = 0.1$ $\xi_{max} = 1$ $\xi_{min} = -1$	$\sigma = 1.2$ $\xi_{max} = 1.5$ $\xi_{min} = -1.5$

Table 2
Statistics of selected properties calculated on population of clusters obtained by using four clustering algorithms.

	$\beta = 1.01/\beta = 1.2$ SPAM	SCMRG	SCDE	MOSAIC
$q(X)$	13502/24265	4129/34614	14709/39935	14047/59006
# of clusters	2000/807	1597/644	1155/613	258/152
Statistics of objects in clustering solutions				
Max.	93 /162	523/2685	1258/3806	4155/5542
Mean	18/45	15/45	25/49	139/236
Std.	10/25	31/201	80/193	399/717
Skewness	1.38/1.06	9.52/10.16	9.1/13.44	6.0/5.24
Statistics of rewards in clustering solutions				
Max.	197/705	743/6380	671/9488	3126/16461
Mean	10/46	9/54	12/65	94/694
Std.	15/66	35/326	38/415	373/2661
Skewness	5.11/4.02	13.8/13.95	10.1/19.59	6.24/4.69
Statistics of $\sqrt{\lvert z \rvert}$ in clustering solutions				
Max.	2.7/2.45	2.85/2.31	2.95/2.94	1.24/1.01
Mean	0.6/0.57	0.74/0.68	0.95/0.97	0.44/0.40
Std.	0.38/0.36	0.31/0.26	0.47/0.47	0.24/0.22
Skewness	1.14/1.34	1.58/1.88	1.28/1.31	0.73/0.40

3.2. Hot spot discovery results

The supervised clustering method (see Section 2) has been applied to the problem of identification of double-feature hot spots in the Martian ground ice dataset. Clustering solutions, seeking to maximize $q(X)$, are computed using four different clustering algorithms described in Section 2.3. In the experiments, the threshold value of $i(c)$ in Eq. (3) is set to $z_{\text{th}} = 0.15$. In order to accommodate the interest of domain scientists in finding the strongest hot spots (characterized by highest values of $|z(o)|$) even if they are small, we perform clustering with $\beta = 1.01$. Clustering with $\beta = 1.2$ is of interest in cases where larger but possibly weaker hot spots need to be identified. The other parameters used in our experiments are given in Table 1.

The results of the experiments are summarized in Table 2. This table is divided into four sections. The first section gives the overall properties of clustering solutions: the total reward and the number of clusters. The SPAM algorithm requires an *a priori* setting of k, which is chosen to be a value that is of the same order of magnitude as the values of k yielded by the SCMRG and SCDE

Author's personal copy

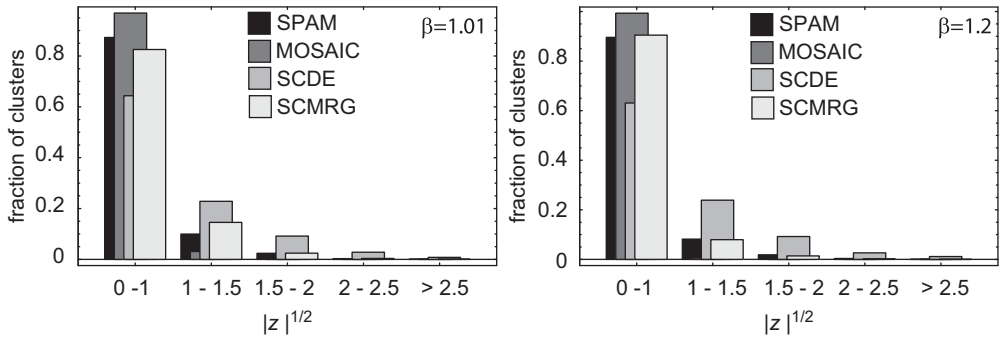


Fig. 2. Distribution of $\sqrt{|z|}$ for clustering solutions obtained using four clustering algorithms.

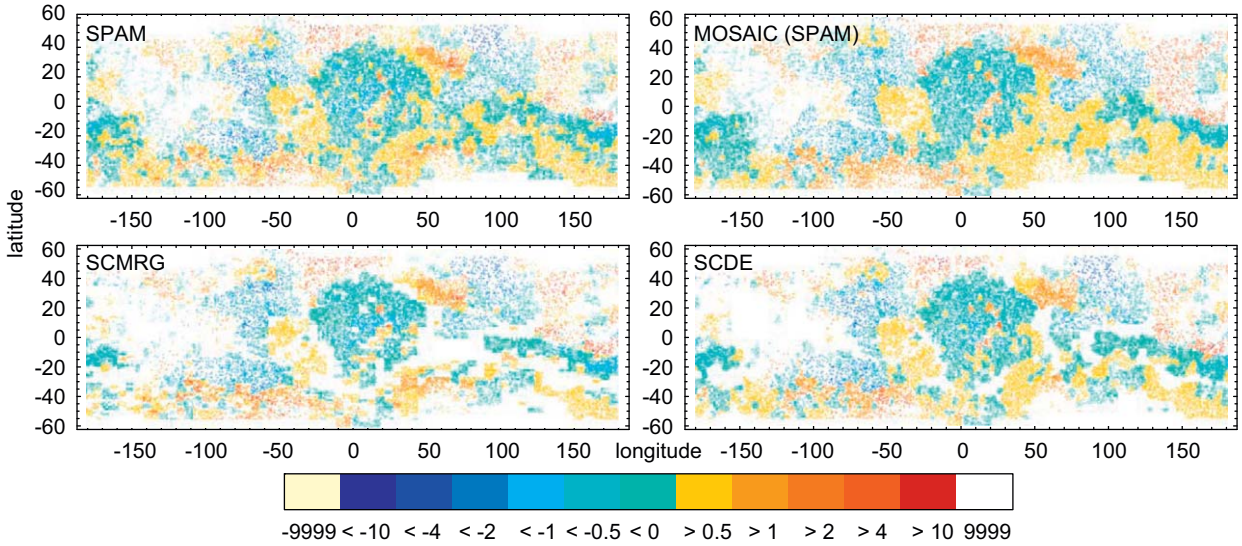


Fig. 3. Clustering solutions for Martian ground ice case study obtained by clustering algorithms as indicated and assuming $\beta = 1.01$. Legend indicates mean z value for each cluster. Objects not belonging to any cluster are not shown. (For interpretation of the references to colors in this figure legend, the reader is referred to the web version of this article.)

algorithms. Due to its agglomerative character, the MOSAIC algorithm always produces a significantly smaller number of clusters regardless of the size of its input provided by SPAM. The remaining three sections of Table 2 give statistics performed on the population of the constituent clusters. Statistics of three different properties are calculated: cluster size $\|c\|$, its reward $(i(c) \times \|c\|^{\beta})$, and $\sqrt{|z|}$, the square root of the absolute value of the mean interestingness of objects within the cluster. For the end-user, $\sqrt{|z|}$ is an intuitive measure of how hot-spot-like a given cluster is. For example, $\sqrt{|z|} = 1$ indicates a cluster where, on average, both features have values one dispersion scale from their centers. The distribution of each property within its population is summarized by four statistical measures: maximum value, mean, standard deviation, and skewness. Skewness describes the amount of asymmetry in the distribution. Large positive values of skewness indicate distribution with a tail toward larger values of a variable.

These statistical measures are not intended to determine the "best" clustering algorithm from the point of view of cluster definition, but rather to help in identification of the algorithm most suitable for discovery of hot spots. Recall that hot spots are clusters characterized by the high values of reward and $\sqrt{|z|}$. The solution that provides more such clusters, and especially the clusters with the highest values of $\sqrt{|z|}$, is the most suitable. This is the solution having large values of skewness for the reward and $\sqrt{|z|}$ properties, as the large value of skewness indicates the existence of more outliers (hot spots). In addition, a suitable solution has larger values of the mean and standard deviation for the reward and $\sqrt{|z|}$ properties, as they indicate the existence of higher valued outliers (stronger hot spots). In Table 2, the MOSAIC solution is separated from other solutions because it is not directly statistically comparable with other solutions, due to a significantly smaller number of clusters. Although it provides an interesting alternative to the other algorithms, we do not include it in the statistics-based comparison. The analysis of Table 2 indicates that among SPAM, SCMRG, and SCDE algorithms, the SCDE algorithm is the most suitable for the discovery of hot spots.

To further compare different solutions, Fig. 2 shows the side-by-side comparison of histograms of $\sqrt{|z|}$, constructed from outputs of each clustering algorithm. Regardless of the algorithm used, most clusters are characterized by $0 < \sqrt{|z|} < 1$ (the leftmost group of bars in the histograms)—they are obviously not the hot spots. The possible hot spots are the clusters fulfilling the $\sqrt{|z|} > 1$ criterion (the rightmost four groups of bars in the histograms). Inspection of Fig. 2 reveals that the SCDE clustering solution has the most clusters in every bin of $\sqrt{|z|}$ that could potentially be associated with hot spots. We rank the SCMRG algorithm as the second best for $\beta = 1.01$ and the SPAM algorithm as the second best for $\beta = 1.2$.

Fig. 3 graphically shows the clustering solutions ($\beta = 1.01$), which are summarized in Table 2. Objects (craters) are color-coded according to the mean z-values of clusters to which they belong. The hot spots are in the locations where objects coded by either deep red or deep blue colors are present. In the red-coded

Author's personal copy

hot spots, the two features have values from the same-side tails of their distributions (high-high or low-low). In the blue-coded hot spots, the two features have values from the opposite-side tails of their distributions (high-low or low-high). Although Fig. 3 shows the location of the hot spots, this type of figure would be difficult to interpret by the end-user because hot spots are visualized as clouds of objects, instead of the actual areas as expected by most end-users. We address this issue in the next section.

4. Transforming clusters to segments

For the purpose of an effective visualization, further statistical analysis, and large-scale data mining, it is convenient to transform the clustering solution to the "segmentation solution". We define the segmentation solution as a raster (image) representation of the original clustering solution. Some loss of information is incurred by transforming a clustering solution into a raster, but the benefits outweigh some loss of accuracy.

Let $R_{i,j} = label(i\,dx, j\,dy)$, $i = 1,\ldots,N_x$, $j = 1,\ldots,N_y$, be a raster having dimensions of (N_x, N_y) and covering the entire spatial extent of the dataset. The raster is an array of constituent pixels (cells) each having an area of $dx \times dy$. Segments in the raster are the single-connected regions, consisting of a number of pixels constructed to represent the clusters. Segmentation is achieved in the following manner. First, dataset objects, each marked by its associated cluster label, c_i, $i = 1 \ldots k$, are grouped into pixels to which they spatially belong. Second, each pixel is assigned a label equal to the cluster label of the majority of objects within it. Pixels with no objects are assigned the 9999 label, and pixels with the majority of objects not belonging to any cluster are assigned the −9999 label. Finally, the entire raster is divided into a set of segments using the connected components algorithm (Alnuweiri and Prasanna, 1992). A connected component is a maximal region of connected pixels which have the same label. We are using the notion of 8-connectivity, wherein two pixels are adjacent if one pixel lies in any of the eight positions surrounding the other pixel. Each segment is a polygon with a clearly defined area and neighborhood relations with other segments. The character of the segmentation solution depends on the selected size of the pixels. For our case study, we have chosen $dx = dy = 5^\circ$, resulting in a raster with $N_x = 72$ and $N_y = 24$.

Table 3 summarizes the results of segmentation solutions obtained by the aforementioned transformation. This table has a format similar to that of Table 2; the MOSAIC solution, which is not part of the statistics-based comparison, is set aside and the table is divided into five sections. The first section gives a number of segments in each solution. Note that the number of segments is significantly smaller than the number of clusters (see Table 2) in corresponding solutions. Because of the relatively large pixel size, small clusters are merged into larger segments. The remaining four sections of Table 3 give statistics calculated on the population of segments. Statistics of four different properties are calculated: segment size (in pixels), quality of conversion from clusters to segments, the uniformness of the interestingness of the objects within the segment, and contrast between the interestingness of a given segment and that of its neighbors. The conversion quality factor is calculated using the formula given by Shufelt (1999)

$$Q = \frac{100TP}{TP + FP + FN}, \tag{6}$$

where *TP* (true positive) is the number of objects in a segment that have the same label as the segment, *FP* (false positive) is the number of objects in the segment that have labels different than the segment, and *FN* (false negative) is the number of objects that have the same label as the segment but are located in different segments. The purpose of calculating the conversion quality is to compare how well different clustering solutions are transformed into segmentation solutions. A segment's uniformness and contrast are properties borrowed from the field of image processing. Uniformness of a segment is encapsulated by the standard deviation of the objects' *z*-values in this segment. Uniformness could be, in principle, calculated also for the original clusters, but since clusters are smaller than segments the statistics would be worse. To calculate the contrast r_i of a given segment s_i, we first identify the segment's neighbors s_j, $j = 1,\ldots,neigh$, where *neigh* is the number of neighboring segments, using a region adjacency graph (RAG) (Sonka et al., 1998). The RAG is an undirected graph whose nodes correspond to segments and branches connect adjacent segments. Segments with labels 9999 and −9999 do not count as neighbors. Second, we calculate the percentage of segment's boundary with each of its neighbors, $w_{i,j}$, $j = 1,\ldots,neigh$, where $\sum_j w_{i,j} = 1$. Third, we measure the dissimilarity between s_i and s_j using quantity $B_{i,j}$ (Sarkar et al., 2000)

$$B_{i,j} = (m_i - m_j)^2 (n_i n_j/(n_i + n_j)), \tag{7}$$

where $m_{i,j}$ are the mean values of *z* in the two segments and $n_{i,j}$ are the number of objects in them. Finally, the contrast between the segment and its neighborhood is calculated as the weighted average of pairwise dissimilarities

$$r_i = B_{i,1} w_{i,1} + \cdots + B_{i,neigh} w_{i,neigh}. \tag{8}$$

The purpose of calculating uniformness and contrast is to compare the separability of segments in different solutions. Ideally, we would like a solution in which the segments would be uniform with respect to the interestingness of its constituting objects, and the values of the segments' interestingness would "stand out" from the interestingness of its neighbors. Thus, the solutions with larger values of uniformness and contrast are more desirable.

Table 3
Statistics of selected properties calculated on the population of segments derived from four clustering solutions.

	$\beta = 1.01/\beta = 1.2$ SPAM	SCMRG	SCDE	MOSAIC
# of segments	1040/642	592/274	565/392	172/117
Statistics of segment sizes (in pixels)				
Max.	10/17	27/108	32/99	152/261
Mean	1.43/2.3	1.61/4.46	2.0/3.03	8.7/13.0
Std.	0.98/2.13	2.0/12.07	2.67/6.05	19.2/34.8
Skewness	3.47/3.12	7.98/6.93	6.86/11.03	4.52/4.81
Statistics of clusters-to-segments conversion quality				
Max.	94/88	100/100	100/100	90/88
Mean	44/51	43/52	54/59	58/59
Std.	15/14	18/23	16/16	16/18
Skewness	0.52/0.09	1.19/0.7	0.22/− 0.12	− 0.13/− 0.29
Statistics of segments uniformness				
Max.	2.87/3.43	4.60/3.89	3.67/3.3	2.49/1.34
Mean	0.40/0.43	0.52/0.54	0.55/0.62	0.45/0.4
Std.	0.38/0.39	0.47/0.51	0.42/0.46	0.35/0.3
Skewness	2.01/2.21	3.05/2.90	2.02/1.72	2.28/1.35
Statistics of segments contrast				
Max.	1050/1746	1172/862	921/887	1308/1129
Mean	27/38	37/42	48/59	55/52
Std.	81/107	99/120	104/123	152/166
Skewness	8.00/8.67	6.07/4.59	4.18/3.52	6.09/5.36

Author's personal copy

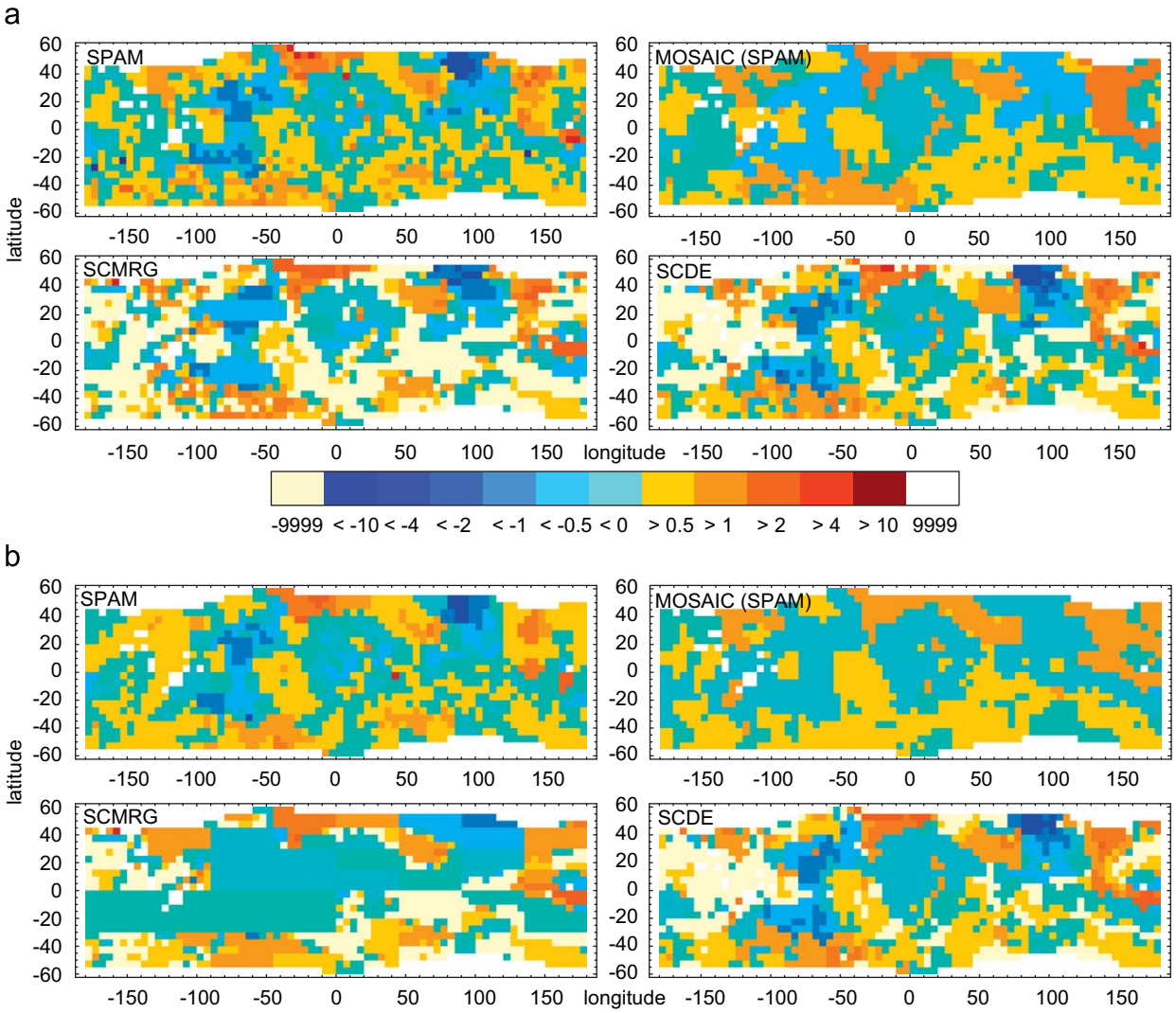


Fig. 4. Segmentation solutions for Martian ground ice case study derived from clustering algorithms as indicated; $\beta = 1.01$ (top two rows) and $\beta = 1.2$ (bottom two rows). Legend indicates mean z-values of segments. Code −9999 indicates regions where objects are left out of clusters, code 9999 indicates regions where no objects are present. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

An analysis of Table 3 indicates that among segmentation solutions stemmed from clustering solutions using SPAM, SCMRG and SCDE algorithms (remember that the MOSAIC-derived solution is set apart), the SCDE-derived solution has significantly better conversion quality than the other two solutions. The SCDE-derived solution is also characterized by the best uniformness and contrast. Thus, the SCDE algorithm not only yields clustering best suited to hot spot discovery (see Section 3.2 and Table 2), but its clustering solution also converts most cleanly to the raster, and the corresponding segmentation solution has the most desirable properties.

Fig. 4 shows segmentation solutions summarized in Table 3. There is a general correspondence between the clustering solutions (Fig. 3) and the segmentation solutions (Fig. 4), but the segmentation solutions are easier to work with for the end-user accustomed to working with maps. The SCMRG-based solutions are only useful for small values of β, as larger values of β lead to formation of large, boxy segments that are not effective in isolating hot spots. Likewise, all MOSAIC-based solutions are too coarse for identification of hot spots with the resolution required by the ground ice on Mars application and set by the domain experts. The hot spots are the segments coded by either deep red or deep blue colors.

We have applied the ESRI ArcGIS implementation of the Getis-Ord's G^* algorithm to our case study dataset, setting the distance scale to 200 km. The color-coded visualization of the result corresponds closely to our SCDE-derived raster shown in Fig. 4. Thus, both methods zero in on to the same hot spots. However, our method provide a means for filing discovered hot spots for further analysis (perhaps as a part of larger data mining procedure) without visualization. The segments in the raster representation are sorted by the descending value of $\sqrt{|z|}$ and then a number of segments at the top of the list are saved as hot spots. The specific threshold for being considered a hot spot needs to be established by a domain expert. For example, the SCDE-derived raster (see Fig. 4, $\beta = 1.01$) has five segments with $\sqrt{|z|} \geqslant 2.5$ and 15 segments with $2 \leqslant \sqrt{|z|} < 2.5$. These segments are saved with all associated information for further analysis by the domain experts to find what particular set of geological circumstances led to their existence.

5. Discussion and future work

This paper presents and examines a method to identify feature-based hot spots using the supervised clustering technique.

Author's personal copy

It offers an alternative to an existing method based on the G^* statistic. Its biggest advantage is the inherent ability to output hot spots as clusters or polygons. Therefore, the method is well-suited for mining large datasets in order to identify all sort of potential hot spots.

Because our method depends on clustering, we have examined four different types of clustering algorithms in order to identify an algorithm most suitable for identifying hot spots. The density-based clustering algorithm SCDE has been found to be best suited to this task. Moreover, the SCDE solution converts best to the raster, and its segmentation-based equivalent has the best separation properties, important advantages from the point of view of the end-user who requires effective visualization. The SCDE solution corresponds closely to the results of hot spot analysis performed using the Getis–Ord G^* algorithm.

This conclusion was reached on the basis of examining a particular dataset (ground ice on Mars). However, the SCDE algorithm should provide the best performance for other datasets as well. Recall that in the original density-based algorithm, the clusters are extracted from the density function Ψ, which provides a continuous approximation to the density of the set of point objects. Our modification (see Eqs. (4) and (5)) provides a continuous approximation to the density of "interestingness" of the set of point objects. Thus, the density function itself provides visual indication for the location of the hot spots even *before* any clustering is carried out. In fact, we could provide the end-user with visualization of the hot spot locations on the basis of Ψ alone. However, automating the discovery process requires clustering. The effectiveness of the SCDE algorithm in producing a clustering solution that best identifies the hot spots is most likely due to its reliance on Ψ, a built-in advantage over all other methods. In our case study, the SCDE algorithm took ~500 s to complete (using a 3.2 GHz CPU), whereas SCMRG took ~3.5 s, SPAM took ~50 000 s, and MOSAIC took ~155 000 s. Thus, the SCMRG algorithm is significantly faster than SCDE and, on this basis, could present an alternative to SCDE when searching for hot spots in a very large dataset for small values of β.

Although our method is designed to find hot spots across multiple variables, we have evaluated its effectiveness using a dataset with only two variables. The reason for such a choice is twofold. First, we want the case study to provide a relatively simple illustration of the principles behind our method. Second, a pairwise search for hot spots in a database with multiple variables is the most logical choice for initial exploration of data by our method. Our finding that the SCDE is the most effective in finding hot spots should not depend on the number of variables. The overall quality of clustering will somewhat decrease with the increasing number of variables because of the increased loss of information when multiple values of variables are aggregated in a single value (see Eq. (2)).

The method described here uses datasets consisting of point objects (see Section 2.1); however, some datasets of interest are polygon-based and are given in the form of shapefiles. The shapefiles can be incorporated into our framework by converting them into point-based features. Frequently, the points are already defined (like the locations of craters in our case study), and the conversion reduces to reading off the values of shapefiles at those predefined points. When no reference points are available, they need to be defined. One possibility is to use the centroids of polygons as reference points, but other methods of point-based representation may be necessary to insure that the created points are located inside each polygon's area or that elongated polygons are represented by multiple points.

The future work will examine the possibility of using different fitness functions. Recall that we are using a fitness function (see Eq. (1)) that maximizes the sum of rewards from all the clusters. Because hot spots are clusters of high interestingness, maximizing the total interestingness of the clustering solution is a good, but perhaps not optimal, choice for a fitness function. Inspired by our calculation of uniformness and contrast for the segments in the visualization raster (see Section 4), we plan on investigating an effectiveness of a different fitness function based on optimizing an "energy function" defined as the sum of clique potentials (Sarkar et al., 2000). Such a solution will maximize the uniformness of cluster mean z-values, while at the same time maximizing the contrast of mean z-values between neighboring clusters. Such a fitness function will yield a solution where hot spots are clearly isolated from other regions. A method to calculate contrast for clusters as opposed to segments needs to be implemented to produce proper input for such a fitness function.

Acknowledgments

We acknowledge support by the National Science Foundation under Grant IIS-0430208. A portion of this research is conducted at the Lunar and Planetary Institute, which is operated by the USRA under Contract CAN-NCC5-679 with NASA. This is LPI Contribution no. 1437.

References

Alnuweiri, H., Prasanna, V., 1992. Parallel architectures and algorithms for image component labeling. IEEE Transactions on Pattern Analysis and Machine Intelligence 14, 1014–1034.

Barlow, N.G., 1988. Crater size-distribution and a revised Martian relative chronology. Icarus 75 (20), 285–305.

Brimicombe, A.J., 2005. Cluster detection in point event data having tendency towards spatially repetitive events. In: Proceedings of the Eighth International Conference on GeoComputation (CD), Michigan, pp. 1–11.

Choo, J., Jiamthapthaksin, R., Chen, C.S., Celepcikay, O.U., Giusti, C., Eick, C.F., 2007. MOSAIC: a proximity graph approach for agglomerative clustering. In: Proceedings of the Ninth International Conference on Data Warehousing and Knowledge Discovery, Regensburg, Germany, pp. 1–10.

Clifford, S.M., 1993. A model for the hydrological and climatic behavior of water on Mars. Journal of Geophysical Research 98 (E6), 10973–11016.

Eick, C.F., Vaezian, B., Jiang, D., Wang, J., 2006. Discovering of interesting regions in spatial data sets using supervised clustering. In: Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases. ACM, Irvine, California, pp. 1–10.

Eick, C.F., Zeidat, N., Zhenghong, Z., 2004. Supervised clustering—algorithms and benefits. In: Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence, Boca Raton, Florida, USA, pp. 774–776.

Feldman, W.C., 2004. Global distribution of near-surface hydrogen on Mars. Journal of Geophysical Research 109, E09006.

Gabriel, K.R., Sokal, R.R., 1969. A new statistical approach to geographic variation analysis. Systematic Zoology 18, 259–278.

Getis, A., Ord, J.K., 1992. The analysis of spatial association by use of distance statistics. Journal of Geographical Analysis 24, 189–206.

Huang, Y., Pei, J., Xiong, H., 2006. Mining co-location patterns with rare events from spatial data sets. Geoinformatica 10 (3), 239–260.

Huang, Y., Shekhar, S., Xiong, H., 2004. Discovering co-location patterns from spatial datasets: a general approach. IEEE Transactions on Knowledge and Data Engineering (TKDE) 16 (12), 1472–1485.

Jiang, D., Eick, C.F., Chen, C., 2007. On supervised density estimation techniques and their application to clustering. In: Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems, Seattle, Washington, pp. 1–4.

Kaufman, L., Rousseeuw, P.J., 1990. Finding Groups in Data: An Introduction to Cluster Analysis. Wiley, New York, 386pp.

Koperski, K., Han, J., 1995. Discovery of spatial association rules in geographic information databases. In: Egenhofer, M.J., Herring, J.R. (Eds.), Proceedings of the Fourth International Symposium on Advances in Spatial Databases, London, UK, vol. 951, pp. 47–66.

Kulldorff, M., 2001. Prospective time periodic geographical disease surveillance using a scan statistic. Journal of The Royal Statistical Society Series A 164, 61–72.

Miller, H.J., Han, J., 2001. Geographic Data Mining and Knowledge Discovery. CRC Press, Boca Raton, FL, 372pp.

Author's personal copy

Munro, R., Chawla, S., Sun, P., 2003. Complex spatial relationships. In: Proceedings of the Third IEEE International Conference on Data Mining, Washington, DC, USA, p. 227.

Murray, A.T., Estivill-Castro, V., 1998. Cluster discovery techniques for exploratory spatial data analysis. International Journal of Geographical Information Science 12, 431–443.

Openshaw, S., 1998. Geocomputation: A Primer, Wiley, Chichester, pp. 95–115 (Chapter Building automated geographical analysis and explanation machines).

Ord, J.K., Getis, A., 1995. Local spatial autocorrelation statistics: distributional issues and an application. Geographical Analysis 27 (4), 286–306.

Rousseeuw, J., Croux, C., 1993. Alternatives to the median absolute deviation. Journal of American Statistical Association 88, 1273–1283.

Sarkar, A., Biswas, M., Sharma, K., 2000. A simple unsupervised MRF model based image segmentation approach. IEEE Transactions on Image Processing 9, 801–812.

Shufelt, J.A., 1999. Performance evaluation and analysis of monocular building extraction from aerial imagery. IEEE Transactions on Pattern Analysis and Machine Intelligence 21, 311–326.

Sonka, M., Hlavac, V., Boyle, R., 1998. Image Processing, Analysis, and Machine Vision. In: Brooks and Cole Publishing, Pacific Grove, CA, 770pp.

Tay, S.C., Hsu, W., Lim, K.H., 2003. Spatial data mining: clustering of hot spots and pattern recognition. In: Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Toulouse, France, vol. 6, pp. 3685–3687.

Williams, G.J., 1999. Evolutionary hot spots data mining—an architecture for exploring for interesting discoveries. In: Proceedings of the Third Pacific-Asia Conference on Methodologies for Knowledge Discovery and Data Mining, vol. 1. Springer, London, UK, pp. 184–193.

Zhang, X., Mamoulis, N., Cheung, D.W., Shou, Y., 2004. Fast mining of spatial collocations. In: Proceedings of the 10th ACM International Conference on Knowledge Discovery and Data Mining, Seattle, WA, USA, pp. 384–393.