Time, temperature, and data cloud geometry

Hsieh Fushing* and Michael P. McAssey
University of California, Davis, California 95616, USA

(Received 20 August 2010; revised manuscript received 8 November 2010; published 7 December 2010)

We demonstrate that the geometry of a data cloud is computable on multiple scales without prior knowledge about its structure. We show that the concepts of "time" and "temperature" are beneficial for constructing a hierarchical geometry based on local information provided by a similarity measure. We design two devices for construction of this hierarchy. Along the time axis, a regulated random walk incorporated with recurrence-time dynamics detects information about the number of clusters and the corresponding cluster membership of individual data nodes. Along the temperature axis we build the geometric hierarchy of a data cloud, which consists of only a few phase transitions. The base level of the hierarchy especially exhibits the intrinsic data structure. At each chosen temperature, we form an ensemble matrix that summarizes information extracted from many regulated random walks. This device constitutes the basis for constructing one corresponding level of the hierarchy by means of spectral clustering. We illustrate the construction of such geometric hierarchies using simulated and real data.

DOI: 10.1103/PhysRevE.82.061110 PACS number(s): 02.50.Ga, 02.70.Rr, 02.70.Hm, 05.40.Fb

I. INTRODUCTION

Much of the impetus for exploratory data analysis in the sciences, from astronomy to taxonomy, comes from the problem of objectively sorting individual data nodes into homogenous clusters, and then discovering the best arrangement of these clusters into higher-level groups [1,2]. This task in modern terms is equivalent to finding the data cloud geometry and its topology [3,4]. Since our ability to visualize high-dimensional data is limited, the computation of global geometric information about a data cloud has attracted intense research attention from mathematics, computer science, neuroscience, physics, and statistics. By viewing a data cloud as a sample from a manifold, many mathematicians try to reconstruct its geometry by approximating the Laplace-Beltrami operator and the Newmann heat kernel using the discrete graph Laplacian, which is the matrix of pairwise similarity (or affinity) measurements [5–7]. Machine learning researchers and applied mathematicians also devise spectral clustering techniques based on the graph Laplacian [8–10]. From information theory and statistical physics, neuroscientists and physicists formulate optimization problems when encoding the data for transmission or modeling the data with magnetic spin [11–13]. Some computer scientists and statisticians tend to prefer modeling approaches by imposing distributional assumptions and resorting to the likelihood principle for solutions [14,15].

All of the aforementioned approaches commonly transform the data cloud geometry into the solution of an optimization problem. Then scientists adapt analytic and numerical techniques from probability theory and functional analysis in mathematics as well as from statistical mechanics in physics in order to provide algorithms and estimates for good approximate solutions to these hard optimization problems. Among these approaches, some involve the time concept and some the temperature concept, while likelihood-based approaches involve neither. Upon these observations, two questions naturally arise: (1) Is the transformation into an optimization problem absolutely necessary? (2) Is the involvement of either the time or the temperature concepts coherent with the task of finding the data cloud geometry?

This paper concludes that the answer to the first question is negative. We propose an alternative approach for extracting the data cloud geometry without involving the optimization of some *ad hoc* choice of utility or energy functions. This approach enjoys the benefit of avoiding consequential artifacts such as resulting clusters consisting of just a singleton or a pair of data nodes. However, we argue that the answer to the second question is positive. The temperature concept is necessary in order to be coherent with the multiscale nature of data cloud geometry, while the time concept is an effective apparatus for extracting clustering information at a given temperature. In fact, we show that approaches that make use of only one of the two concepts inherently miss authentic features of the data cloud geometry. Meanwhile, many existing popular clustering approaches, including statistical mixture analysis [16] and the hierarchical clustering algorithm [17], are found to be unable to provide realistic information about the data cloud geometry; due to the omission of both concepts, especially when the dimension is high. Specifically, we demonstrate that the often-asked questions—how many clusters are there and how can we devise a local-to-global approach to investigating a data cloud?—are ill-posed. Coherent answers to these questions ought to be threaded through time and temperature axes to make meaningful sense.

In this paper, we postulate the global geometry of a data cloud as a hierarchical composition which is constructed based on a varying number of clusters or communities at each of its levels. The base level of this hierarchy consists of what we call *core clusters* (equivalent to a tightly connected community in network theory [18]). Each core cluster has its intrinsic intracluster scale, while there are also heterogeneous intercluster scales among the collection of core clusters.

*Author to whom correspondence should be addressed. Hsieh Fushing, Department of Statistics, MSB 4232, University of California at Davis, CA 95616; fushing@wald.ucdavis.edu

These two types of unknown scales are bestowed by the unknown data-generating mechanism. Due to the variation in the "distances" between each pair of core clusters, a hierarchy is developed to represent the global geometry of the data cloud. We devise an algorithm to compute this hierarchical manifestation of a data cloud's global geometry. This geometry is essentially represented through an evolution of phase transitions in clustering, computed by varying the temperature from a small to a large value. Related clustering results also achieved by a sequence of phase transitions were derived through superparamagnetic clustering, also called the granular model [12,19]. However our approach is conceptually simple and computationally effective. It neither involves the extra effort of embedding inhomogeneous Potts spins onto each individual data node nor requires the tremendous need for computing the most stable configuration of spin-spin alignments under every given temperature. We demonstrate and compare our method with existing popular alternatives using both simulated and real data.

II. TEMPERATURE AND TIME CONCEPTS

Why are temperature and time both essential concepts for extracting the geometry of a data cloud? Suppose data $x_i \in \mathcal{X}_n$, $i=1,\ldots,n$ are sampled from a manifold with unknown characteristic feature V based on a distribution $P(\cdot)$ in $\mathbb{R}^p$, $p \geq 1$. From the perspective of data compression, let the cost of approximating datum x by x_v be measured by some distortion function, $d(x,x_v)$. Lossy compression using the rate distortion approach [13] or the statistical mechanics approach [11] is achieved by associating each x with a characteristic v such that the mutual information of X and V is minimized by subjecting it to a constraint on the expected distortion. The optimal solution of such a variation problem typically takes the form of a Boltzmann distribution:

$$P(x|v) \approx \exp\{-d(x,x_v)/T\}.$$

Note that the tuning parameter T plays the role of temperature in the heat kernel [7]. When T is sufficiently small, this approach leads to mathematical artifacts by creating a large range of values for v, corresponding to the one-nearest-neighbor graph. In contrast, our approach drives the same T toward zero—as in stochastic relaxation [21]—to extract the base level of the data cloud geometry.

At a given temperature T, the heat kernel

$$K(x_i,x_j)(T) = c$$

is a device for measuring the affinity between x_i and x_j in $\mathcal{X}_n$. Presumably, the collection $\{d(x_i,x_j)\}_{j=1}^n$ contains the local geometric information about $\mathcal{X}_n$ around each point x_i. The $n \times n$ symmetric similarity matrix $W(T)=[w_{ij}(T)]$ summarizes all available local geometric information about $\mathcal{X}_n$ at the temperature T. Throughout this paper we set $w_{ii}(T)=0$ for all i, and denote the *degree* of data point (or node) x_i at temperature T by $d_i(T)=\Sigma_{j=1}^n w_{ij}(T)$. A popular approach for extracting geometric information from $W(T)$ is to study the Markov chain specified by the transition probability matrix $P(T)=D^{-1}(T)W(T)$, with the degree matrix $D(T)=\mathrm{diag}(d_1,\ldots,d_n)$. This approach is especially appealing from the point of view of graph theory: $p_{ij}(T)$ represents the probability of transition in one time step from node x_i to x_j and is an increasing function of the edge-weight $w_{ij}(T)$. It is argued through the development of the diffusion map that the discrete time process $\{P^k(T)\}_{k=1}^\infty$ integrates the local geometry into global geometry [6,7].

However, we do not use the time idea in the same fashion as in the diffusion map, because the geometric information contained in $W(T)$ and thus in $P(T)$ critically depends on the choice of T. Consider the two extremes. On one hand, if T is too high, then $\{P^k(T)\}_{k=1}^\infty$ only provides information pertaining to the whole data cloud as one single cluster at the top hierarchical level of the global geometry of $\mathcal{X}_n$. On the other hand, if T is chosen too low, then $\{P^k(T)\}_{k=1}^\infty$ becomes a matrix corresponding to the one-nearest-neighbor graph, which consists of many isolated closed loops. Therefore all Markov random walks are easily trapped within a closed loop of several data nodes. Consequently, for any T falling between the two extremes, $\{P^k(T)\}_{k=1}^\infty$ is not likely to reveal complete and realistic geometric information about $\mathcal{X}_n$, but runs the danger of missing critical geometric information about the data cloud. In this paper, we instead align the time component with that of the designed mission-oriented regulated random walk, which will be introduced below. Through this regulated random walk, which is capable of exploring the entire data cloud within a very reasonable temporal span, the time concept becomes an effective apparatus for extracting clustering information, including the number of clusters and cluster membership at any given temperature.

In contrast, the popular hierarchical clustering algorithm [17] and statistical mixture analysis do not make use of either the time or temperature concepts. Very brief reviews of both approaches can shed light on the important roles of these concepts, especially temperature, when exploring data cloud geometry. The hierarchical clustering algorithm works by modifying the distance $d(x_i,x_j)$ into one "new metric" which satisfies the ultrametric inequality condition for any two disjoint sets of data nodes. Thus a pair of sets of nodes is merged when they have the smallest ultrametric, and a new level in the hierarchy is created. A full hierarchy derived in this fashion is typically very complex with very many levels. Different choices of the ultrametric, such as the complete (pairwise maximum), the single (pairwise minimum), the median, etc., can result in characteristically different hierarchies. These consequences are chiefly attributed to the fact that the ultrametric poorly reflects the multiscale nature imbedded within the data cloud. Therefore this algorithm only works when all involved clusters are convex and well separated. It is thus prone to provide incoherent data cloud geometry when the clusters are not convex or they are convex but relatively close to each other, as in the simulated data sets used below.

As for statistical mixture analysis, much prior knowledge must be assumed, such as a range for the number of clusters, and parametric data-generating distributions for modeling and constructing its likelihood function. These assumptions are likely to be either unrealistic or wrongly imposed when the data cloud consists of a large number of high-dimensional measurements. Nonetheless, its resulting single layer of clusters hardly conveys any valuable global information regarding the data cloud geometry.

III. REGULATED RANDOM WALK

Given a p-dimensional data cloud $\mathcal{X}_n$, we construct an $n \times n$ similarity matrix $W(T)$ whose elements $w_{ij}(T)$, $i \neq j$, depend on a temperature $T>0$ through the relation $w_{ij}(T) = \exp\{-d(x_i,x_j)/T\}$, where $d(x_i,x_j)$ represents a distance (or distortion) measure between pairs of nodes in $\mathcal{X}_n$. We set $w_{ii}(T)=0$ for all i. In scientific applications, the choice of the distance measure should be based on expert knowledge about the relationship among the objects represented by the nodes. In the absence of external knowledge, the Euclidean distance may be used by default. A similarity matrix constructed in this manner may be regarded as an adjacency matrix associated with a fully connected network with weighted edges. Many alternative methods for identifying network communities are based on the adjacency matrix corresponding to a network in which two nodes are connected by an edge if and only if the distance between them falls below some threshold (or equivalently, their similarity is above some threshold), or if and only if nodes i and j are mutually among the k nearest neighbors of each other ([23]). This discretizes the connectivity among nodes, often producing disconnected components, and discards essential information about the intracluster and intercluster scales inherent in the data cloud geometry. The similarity matrix $W=W(T)$ used here preserves this information.

The degree d_i of data node i, computed as the sum of the ith row of $W(T)$, is a measure of the density of the data cloud in the vicinity of node i with respect to the temperature T. The degree matrix $D(T)$ is then the diagonal matrix whose diagonal elements D_{ii} are the corresponding degrees d_i. If desired, one may eliminate those nodes whose degrees fall below some specified threshold, since such nodes may be considered outliers or extreme points rather than integral components of the data cloud hierarchy. In our examples we do not implement this option.

Given $W(T)$ and $D(T)$, we design a regulated Markov random walk on $\mathcal{X}_n$ such that, once the walk enters a cluster, it remains within that cluster with high probability until it has been thoroughly explored before moving to another cluster. The regulated steps and their functions are described as follows.

Algorithm of the regulated random walk:

(1) Select an initial node with a relatively large degree to ensure that the random walk starts from a node that is likely to be located within a dense region of a cluster. For our algorithm we compute the sum V of the degrees of all n nodes (i.e., the *volume*), then select the smallest subset of nodes such that the sum of their degrees exceeds $0.5V$. We choose the initial node at random from this subset, with the probability of selection proportional to the degree.

(2) The random walk then moves from node to node based on the Markov transition matrix $P=D^{-1}W$, where W is the similarity matrix and D is the corresponding degree matrix. Hence element p_{ij} of P is the probability of moving from node i to node j. Under this structure, the walk tends to move from each node to one of its nearest neighbors but may visit a distant neighbor with some small probability. However, we add a feature that will bias the walk toward remaining in the vicinity of previously visited nodes: suppose node j has been visited m times. If the walk is at node i, we multiply the transition probability p_{ij} by the bias factor $e^{m/M}$, where M is defined in the next step. This necessitates maintaining a record of the number of times each node has been visited during the random walk.

(3) Remove a node from the random walk once it has been visited M times, where M is a predetermined threshold, and modify the transition matrix P accordingly. This ensures that, once the walk enters a cluster, it cannot remain there indefinitely. Once most of the nodes in a cluster have been visited M times, the walk is very likely to transition to another cluster. The walk may return to an earlier cluster to visit the remaining nodes without harming the effectiveness of the algorithm.

(4) We regard each step of the random walk as a time unit. Throughout the walk we maintain a record of the recurrence time for node removal. The first entry in this record is the time t_1 until the first node is removed. Each subsequent entry in the record is the time t_j *after* the $(j-1)$st node removal until occurrence of the jth node removal, $j \geq 2$. This makes $\mathcal{T}=\{t_1,t_2,\ldots\}$ a random process of recurrence times for node removal during the random walk.

(5) Terminate the random walk when every node has been visited at least once. By this point almost all nodes have been removed.

The node-removal threshold value M is empirically chosen based on the size of the data set. For large sets, we find values of 3–5 adequate, while smaller sets usually require a higher threshold. This choice allows a regulated random walk to thoroughly explore each individual cluster's regional geometry, while also traveling quickly among all the cluster regions. It is noted that the unregulated random walk based on P, i.e., our regulated walk with $M=\infty$, will take an extremely long time to visit all nodes, while the random walk with $M=1$ will visit all nodes very swiftly, but without carrying out any exploration within a cluster. Neither extreme case will provide insight about the geometry of $\mathcal{X}_n$.

A sequence plot of the recurrence time process $\mathcal{T}$ produces a profile consisting of a train of segments, with the beginning of each segment marked by a significant spike in the recurrence time when the walk enters a previously unexplored cluster. The idea behind the relationship between the spikes in the recurrence time plot and the clusters in the data cloud is simple. When the random walk enters a cluster for the first time, as at the beginning of the walk, it will tend to remain there for a long time, moving among most if not all the nodes in that cluster. Thus many steps are required before any node in that cluster is visited M times and consequently removed from the walk, provided M is not too small. Thereafter the remaining nodes in the cluster receive their Mth visit after relatively few steps, since they have each already been visited several times. Eventually most of the nodes in the cluster are removed from the walk, making it more likely for the walk to move to another cluster. Once it enters the new cluster, many steps will be required before the first node in that cluster is removed. If at any point the walk returns to an earlier cluster, the remaining nodes in that cluster will be removed after only a few steps since they are likely to have already been visited several times previously, and moreover there are fewer nodes available for visitation. If the spikes

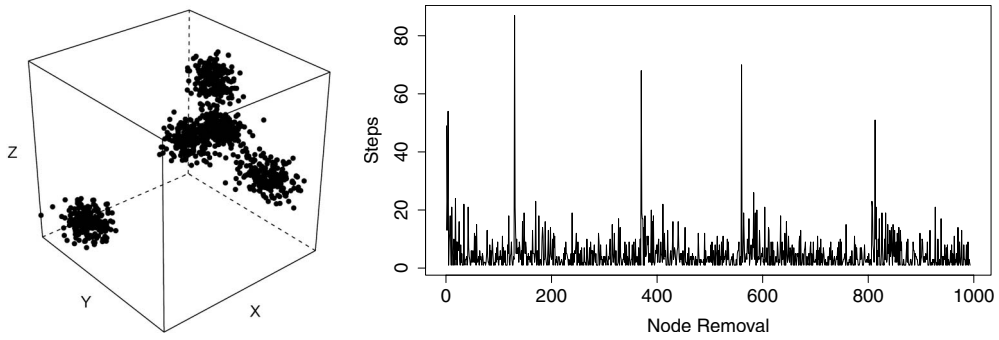

FIG. 1. Data cloud of points (x,y,z) in $\mathbb{R}^3$ consisting of five clusters, with corresponding recurrence time profile revealing a train of five segments, with node removal parameter $M=5$.

are not sufficiently pronounced when it is known that the data are clustered we can tune the parameter M by examining plots of $\mathcal{T}$ for several iterations of the regulated random walk at increasing values of M until prominent spikes occur.

Figure 1 displays a simulated data cloud in $\mathbb{R}^3$ consisting of 1000 data nodes in five equal-sized core clusters, with different intercluster distances, along with a plot of the corresponding recurrence time process for a single regulated random walk with $M=5$. In this case, the number of clusters observable in the data plot exactly matches the number of tall spikes evident in the recurrence time profile. Of course, this level of accuracy is most likely when the intercluster scale is large compared to the intracluster scale. Ideally, each individual segment between spikes in the process results from the exploration of a single cluster during the regulated random walk. However, especially when clusters are near each other, the walk may transition between clusters several times within such a segment. This does not ultimately harm the success of this procedure as long as the spikes truly mark transitions between clusters. In fact, two of these five clusters overlap to some extent, which is why two of the spikes are close together. It is likely that the random walk returned from exploring the second cluster to finish exploring the first cluster before moving to the third cluster. Yet we still see five and only five prominent spikes.

In our algorithm, we identify the locations of the spikes in the recurrence time profile by locating the top α percent of recurrence times (usually $1 \leq \alpha \leq 5$). The larger M is, the smaller α should be so that only the most prominent spikes are identified. If several of these locations are consecutive, we take only the first one. Then we partition the recurrence time profile based on these spike locations into consecutive segments, and assign node x_i to segment k if node x_i was visited a full M times during the walk, and was visited at least half of those times during segment k. Thus each individual segment of the profile identifies nodes that probably inhabit the same cluster, in as much as the nodes assigned to the same segment were visited multiple times between probable transitions from one cluster to another.

Finally, we construct an assignment matrix A, with element $a_{ij}=1$ if nodes x_i and x_j are assigned to the same segment in the profile, and $a_{ij}=0$ otherwise. We set $a_{ii}=0$ for all i. While the nonzero elements of this assignment matrix indicate probable mutual cluster membership among pairs of nodes, the likelihood of classification error is high. However, by conducting a large ensemble of such regulated random walks, the effect of these misclassifications can be attenuated. This is explored in the next section.

IV. ENSEMBLE OF REGULATED RANDOM WALKS

To obtain an accurate estimate of mutual cluster membership for each pair of nodes in the data cloud, we generate a large ensemble of N regulated random walks using the algorithm given in Sec. III, with the temperature T fixed. In practice, we set $N=1000$. We consolidate the information that was extracted from each walk and stored in each corresponding assignment matrix A by constructing an *ensemble matrix* $E=E(T)$ in which component e_{ij} is the proportion of occasions among the N random walks in which nodes x_i and x_j are assigned to the same segment in the recurrence time profile. This is easily accomplished by adding the N assignment matrices and dividing the sum by N. Hence element e_{ij} of E may be regarded as the empirical probability that, if nodes x_i and x_j are assigned to the same cluster, the assignment would be correct.

Let $E^*=E^*(T)$ denote the unknown matrix that contains the true mutual cluster membership, whose element $e^*_{ij}=1$ if nodes x_i and x_j do, in fact, belong to the same cluster at the scale corresponding to T, and $e^*_{ij}=0$ otherwise. The meaning of "cluster" here depends on the temperature T since at higher temperatures the clusters may be conglomerations of the core clusters identified at lower temperatures. The ensemble matrix $E(T)$ can thus be taken as a perturbation of $E^*(T)$ at the level of the hierarchy of global geometry of $\mathcal{X}_n$ corresponding to T. By employing appropriate permutations, E^* can be represented as a 0–1 block-diagonal matrix, with each block indicating the members of one common cluster. The eigenvalue corresponding to a block is equal to one less than the cluster size. Thus the number of nonzero eigenvalues of $E^*(T)$ is equal to the number of clusters in the data cloud at the scale corresponding to T.

Hence the dominating eigenvalues of the ensemble matrix $E(T)$ resulting from N regulated random walks should provide evidence for the number of clusters perceptible at temperature T. We first obtain a matrix B corresponding to E, which is the diagonal matrix whose diagonal elements are the sums of the corresponding rows in E. Then we normalize E to produce a positive semidefinite matrix $I-B^{-1/2}EB^{-1/2}$, whose eigenvalues must be nonnegative. To normalize these

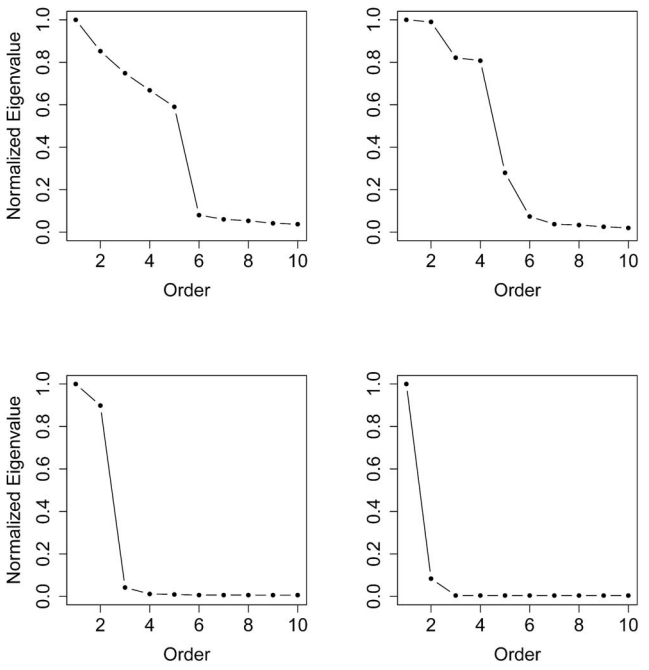

FIG. 2. First ten normalized eigenvalues of the normalized ensemble matrix resulting from 1000 regulated random walks on the data cloud from Fig. 1, at four increasing temperatures, with spectral gap evident at the fifth value (top left), the fourth value (top right), the second value (bottom left) and the first (bottom right).

eigenvalues we divide each of them by the largest value among them, then subtract the results from one. Finally, we plot the normalized eigenvalues in decreasing order. From this eigenplot we look for a significant spectral gap—a point at which the normalized eigenvalues drop significantly from values near one toward zero. We denote the location of the spectral gap in the eigenplot, if any, by $C(T)$. The spectral gap, or absence thereof, serves as the chief source of information about the number of clusters in the data cloud at temperature T and is most robust when the temperature is low. If no gap is evident, and the eigenvalues gradually decrease from one toward zero, then we can infer that, at the scale corresponding to the chosen temperature, any macroscopic clustering in the geometry of the data cloud is not visible at that scale. This often occurs when the temperature is very low so that only a large number of microscopic clusters are identified. If the gap occurs immediately after the first eigenvalue, then the data cloud may be viewed as a single cluster at that scale. This eventually occurs when the temperature climbs sufficiently high even when multiple clusters are found at lower temperatures.

Figure 2 displays four eigenplots, each obtained from the ensemble matrix resulting from 1000 regulated random walks on the data cloud displayed in Fig. 1 at one of four successive temperatures. At the lowest temperature (top left panel), the spectral gap occurs at the fifth normalized eigenvalue, thus correctly identifying the presence of five core clusters in the data cloud. As the temperature increases, the location of the gap gradually decreases to four (top right panel) and then to two (bottom left panel), signifying three levels in the geometric hierarchy. The two clusters that are closest together merge into a conglomerate cluster, resulting in four clusters, and then this conglomerate merges with the two clusters that are nearby to form a larger conglomerate, resulting in two clusters. Finally, all clusters merge into one, so that the eigengap occurs after the first eigenvalue (bottom right panel). Hence a hierarchy of clustering has been identified.

It is readily apparent that the ensemble matrix E is used here in place of the similarity matrix W for the purposes of spectral clustering, with the normalized graph Laplacian $I-D^{-1/2}WD^{-1/2}$ replaced by the normalized ensemble matrix $I-B^{-1/2}EB^{-1/2}$. Both the similarity matrix and the ensemble matrix contain information about the local geometry of $\mathcal{X}_n$, but the latter matrix is much more informative about the global geometry. The ensemble matrix contains clustering information extracted from multiple regulated random walks, which depends very little on the size, shape, or density of the individual clusters. Meanwhile, clustering based on the similarity matrix has been shown to be highly susceptible to such features [24] since the information about the local geometry is based solely on the relative proximity of nodes in the data cloud and cannot take into account more general geometric characteristics.

Once the number of clusters $C(T)$ has been estimated at a specified temperature, the spectral clustering algorithm of Ng, *et al.* [9], is applied to the ensemble matrix, using the location $C(T)$ of the spectral gap in the corresponding eigenplot for the number of clusters k. In this algorithm, the k-means procedure is applied to the last k normalized eigenvectors (corresponding to the largest k normalized eigenvalues) of the normalized ensemble matrix $I-B^{-1/2}EB^{-1/2}$ to assign the corresponding data points among k clusters. The spectral clustering algorithm of Shi and Malik [8] may also be implemented, with the last k eigenvectors of $I-B^{-1}E$ used as the basis for discriminating among clusters. Once the cluster assignments are made, we may then form an empirical assignment matrix corresponding to that temperature, where element $(i,j)=1$ if the spectral clustering algorithm assigns nodes i and j to the same cluster, and $(i,j)=0$ otherwise. In the ideal situation, this matrix will match the true cluster assignment matrix $E^*(T)$.

V. UNVEILING HIERARCHICAL DATA CLOUD GEOMETRY THROUGH MULTIPLE SCALES

To discover the complete hierarchy of the global geometry of data cloud $\mathcal{X}_n$, we run the algorithm over a wide range of temperature values, from very low to very high. The resulting collection of ensemble matrices typically crystallizes into just a few different phases corresponding to the hierarchical structure. See Blatt *et al.* [12], Kushnir *et al.* [10], and Sharon *et al.* [20] for similar results involving the use of multiscale approaches to detect hierarchical clustering among data. A sufficiently high temperature gives rise to the phase of one cluster—the entire cloud $\mathcal{X}_n$. A sufficiently low temperature, say T_1, brings out the intrinsic intercluster scale by identifying the collection of core clusters. We may identify a suitable value for T_1 by reducing the temperature until the average degree of the similarity matrix (normalized by its

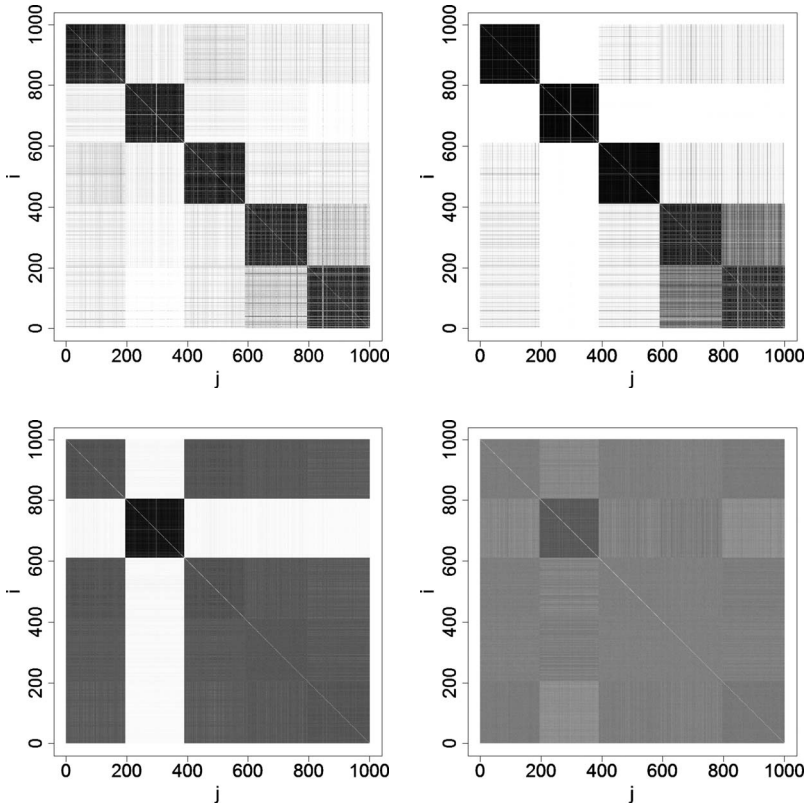

FIG. 3. Resulting phase transitions for the ensemble matrix based on 1000 regulated random walks on the data cloud from Fig. 1, at four temperatures. The darker the shade of element (i,j), the closer the value of $e_{i,j}$ is to one.

row-wise maximum) is nearly equal to 1. Under such a temperature the regulated random walk will take the trajectory of the one-nearest-neighbor path. Thus additional phases will not arise when the temperature drops below T_1. In contrast, eigenvalues equal to zero for the corresponding normalized graph Laplacian have very high multiplicity since it is associated with the one-nearest-neighbor graph.

Figure 3 displays graphical representations of the ensemble matrix $E(T)$ based on 1000 regulated random walks on the five-cluster data cloud shown in Fig. 1 at the four different temperatures corresponding to the eigenplots in Fig. 2. The darkness of a pixel at point (i,j) in the grid reflects the value of component $e_{i,j}$ on the unit interval. Since we have organized the true cluster membership of the simulated nodes in consecutive order, the resulting block-diagonal pattern at the lower temperature, shown by the grid in the first panel, makes the base level of the geometric hierarchy apparent. As the temperature increases, the two nearest clusters merge into a conglomerate cluster as is evident from the merger of the two blocks at the bottom-right of the grid in the second panel. When the temperature increases further, the four nearest clusters become conglomerate, as seen in the third panel. Eventually, as the temperature climbs, the entire grid will become dark gray, as the fourth panel shows.

To illustrate that the resulting collection of empirical assignment matrices corresponding to levels of the geometric hierarchy are indeed crystallized by means of gradual phase transitions along the temperature axis, Fig. 4 displays representations of the empirical assignment matrices resulting from performing spectral clustering on the first three ensemble matrices given in Fig. 3, but with k set to $5=C(T_1)$ in each case. Once again we note the progression through the geometric hierarchy from five core clusters at the lowest level (top left panel), to three core clusters and one conglomerate cluster at the intermediate level (top right panel), to one core cluster and one big conglomerate cluster at the highest level (bottom left panel). Thus by varying the temperature we can recover any hierarchy of clusters present in the data cloud.

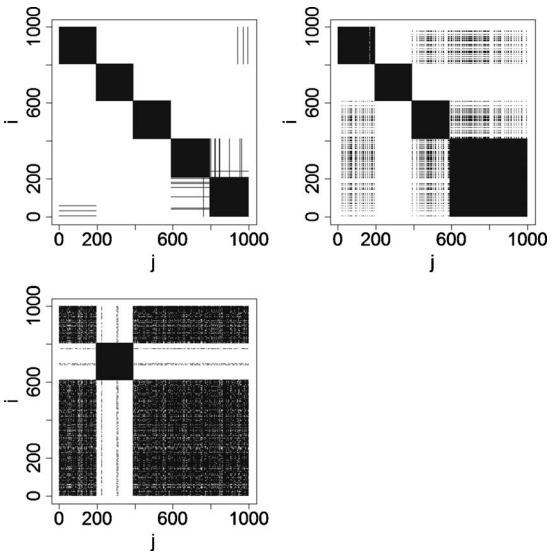

FIG. 4. Resulting phase transitions for the empirical assignment matrix based on performing spectral clustering on the first three ensemble matrices of Fig. 3, with five clusters specified in each case. If element (i,j) is shaded then nodes i and j are assigned to the same cluster at the respective temperature.

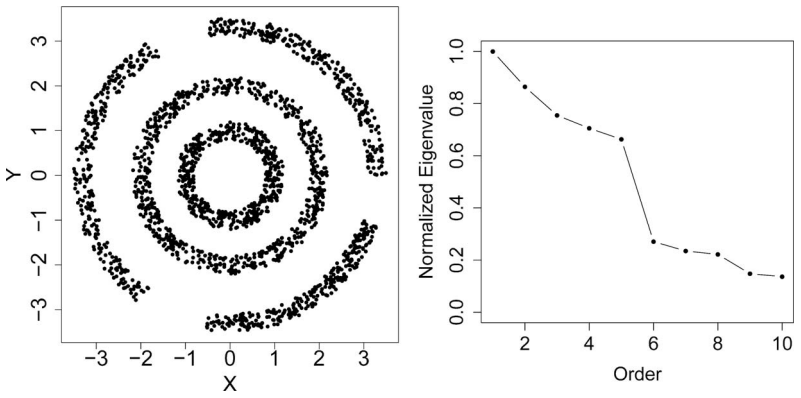

FIG. 5. Simulated data cloud consisting of 2000 points (x,y) in $\mathbb{R}^2$ among five nonconvex clusters, and the first 10 normalized eigenvalues of the normalized ensemble matrix based on 1000 regulated random walks on this cloud at a sufficiently low temperature.

Based on the illustration given in Fig. 4, we now can reliably depict the mechanism underlying the occurrence of the sequential phase transitions as follows. At the lowest temperature T_1, the node removal device in our regulated random walk permits it to break away from the closed loops characterizing the one-nearest-neighbor graph, and the random walk always travels to the nearest neighbor among the remaining nodes. This is how the core clusters are identified. As the temperature increases, the barrier among a small collection of core clusters, which share the smallest intercluster scale, becomes less prohibitive than at the lower temperature. Hence it is increasingly likely that their nodes are consecutively visited in an intercluster fashion by the regulated random walk. The shadows seen in Fig. 4 are manifestations of such a phenomenon. As the temperature sufficiently increases, this small collection of core clusters will consequently first merge into a conglomerate cluster. This process occurs in a discrete fashion due to the discreteness involved in the choice of the spectral gap in the eigenplot and the k-means clustering involved in spectral clustering. This first conglomerate cluster that emerges, together with the remaining well-separated core clusters, then form the next level of the hierarchy. This is how the first phase transition occurs. When the temperature increases further, its interaction with the current smallest intercluster scale will produce the phase transition and thus determine the formation of the next level of the hierarchical geometry. Finally, when the temperature is high enough to break all the barriers set by all the intercluster scales that previously constrained the regulated random walk's travel trajectory, there will be no distinct cluster memberships but only one cluster of the entire data cloud.

With the above mechanistic reasoning, we conclude that our computational algorithm is an effective approach for unveiling hierarchical data cloud geometry. It performs remarkably well in conjunction with the spectral clustering algorithm to correctly identify the number of clusters and cluster membership at each hierarchical level, and thus to reveal the global geometry of a data cloud. Other multiscale approaches may succeed in identifying features of hierarchical structure in a convex data cloud but lack the flexibility to handle general cases. Some such approaches are based on an adjacency matrix obtained by establishing the presence of an edge by the k-nearest-neighbor criterion, or by a threshold on the distance (or similarity) between nodes. The hierarchy is discovered by varying the value of k or of the threshold in order to determine critical values at which phase transitions occur. In [22], for example, network nodes are connected by an edge if the similarity measure between them exceeds a threshold $S_{\min}$. The value of $S_{\min}$ is increased until it falls into a critical range in which the network breaks into separate components. The weakness of any clustering method that bases the connectivity among nodes solely on the distance or similarity measure is exposed when the cluster membership is not coherent with that measure, such as when the clusters are not convex. The ensemble of regulated random walks transforms the information contained in the similarity matrix into information about mutual cluster membership regardless of the geometric structure of the data cloud. This quality will be demonstrated in Sec. VI.

VI. NUMERICAL ILLUSTRATIONS AND AN ANALYSIS OF REAL DATA

Most clustering techniques work well when the clusters are convex, but tend to fail when the clusters are not convex because such techniques are limited by their dependence on the distance measure. The regulated random walk, however, is robust against nonconvexity of the clusters. To illustrate, we generate 2000 data nodes in $\mathbb{R}^2$ configured in three concentric rings, with the outer ring broken into three pieces and the gap between the middle and outer rings wider than the gap between the middle and inner rings, as shown in Fig. 5. Hence there are five distinct, nonconvex clusters.

We apply an ensemble of 1000 regulated random walks on this data cloud at a sufficiently low temperature. The first ten normalized eigenvalues of the resulting normalized ensemble matrix at the lowest temperature are also displayed in Fig. 5. A discernible spectral gap occurs after the fifth normalized eigenvalue. Thus we are justified in concluding that our method recognizes five clusters among the data.

We then apply the spectral clustering algorithm, with five clusters specified, to the resulting ensemble matrix at this and two higher temperatures. We use five different colors to signify the five different cluster assignments of the nodes at each temperature and display the results in Fig. 6. For the lowest temperature all data points are correctly assigned among the five core clusters, while for the middle temperature the two inner rings become one conglomerate cluster and the three pieces of the outer ring begin to merge into another conglomerate cluster. At the highest temperature all three rings begin to merge into one whole cluster.

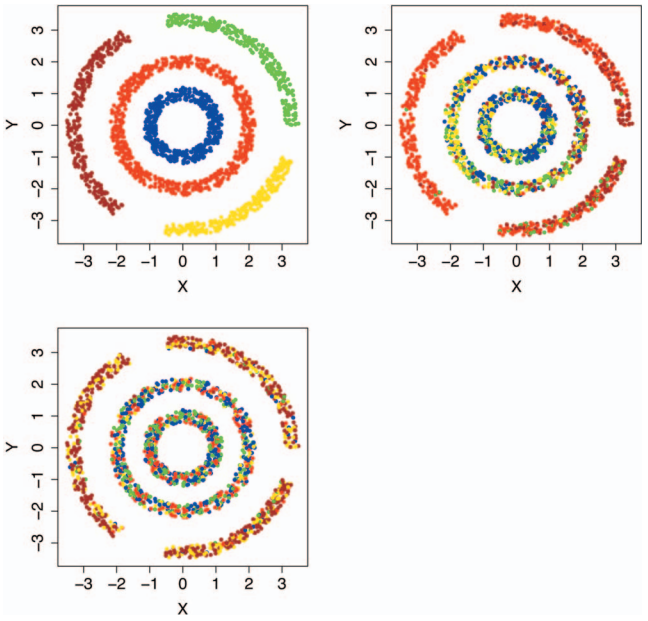

FIG. 6. (Color) Cluster assignments of data nodes from the data cloud given in Fig. 5 based on the spectral clustering algorithm, with five clusters specified each time, applied to the ensemble matrices obtained from 1000 regulated random walks on simulated nonconvex clusters for increasing sequence of three temperatures.

This simulation study demonstrates the effectiveness of the ensemble matrix derived from the regulated random walks in identifying the correct number of clusters and in assigning data points to the correct clusters, even when the clusters are not convex. At the lowest temperature, the regulated random walk tends to remain within a core cluster, regardless of its shape, until the majority of its points have been deleted, and then moves to the next cluster. This feature makes the method robust against nonconvexity of clusters and gives it the advantage over other approaches. For instance, when the hierarchical clustering algorithm is applied to this simulated data, we find that for most choices of ultrametic, except for the single one, the clustering results consist of splitting the middle ring into pieces which are bound with the three outer pieces.

We also apply our approach to another simulated data set, which is the example presented by [24] involving a long narrow strip and a Gaussian ball in $\mathbb{R}^2$, as shown in Fig. 7,

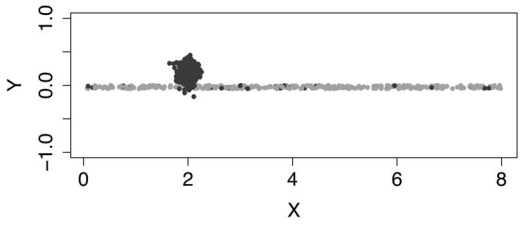

FIG. 7. Clustering of a data cloud consisting of a Gaussian ball and a long narrow strip of points (x,y) in $\mathbb{R}^2$, based on the second eigenvector of the normalized ensemble matrix after 1000 regulated random walks at a moderately high temperature.

but using 700 points instead of 1400 points. They show that spectral clustering based on the second eigenvector of the normalized graph Laplacian merely splits the data cloud down the middle so that the entire left half of the strip is clustered with the ball. We find that the hierarchical clustering algorithm also works very poorly on this simulated data even with the single ultrametric.

We apply our approach over a range of temperatures and find that the eigenplots pertaining to the range from low to medium temperatures indicate more than two clusters are present. However, making use of the prior knowledge that there are two clusters, we gradually increase the temperature and perform our ensemble of regulated random walks at each step until the eigenplot indicates two clusters. When we perform spectral clustering on the resulting ensemble matrix, we find that the misclassification rate is an amazingly low 4.4%. The cluster assignments are indicated in Fig. 7 by the different shades, demonstrating the remarkable performance of our method.

For a practical application, we also analyze electroencephalogram (EEG) recordings taken from nine tetrodes distributed on three regions of a rat's brain while it is within a cage. We select blocks of recordings from each of three velocity periods and partition each block into intervals of 150 EEG recordings (equivalent to 0.1 s) each. For each tetrode we compute the proportion of EEG measurements within each interval which fall into the top 5% and into the bottom 5% of all measurements at that tetrode over the entire recording epoch. This gives us 2590 data points of 18 dimensions (nine upper extreme proportions, and nine lower extreme proportions). The first 990 points pertain to the high velocity block, the second 600 points pertain to the medium velocity block, and the remaining 1000 points pertain to the low velocity block. We apply our clustering method to determine whether extreme EEG recordings on the centisecond level are influenced by the rat's locomotion. If there is no influence, the nodes in the three blocks should be distributed among the resulting clusters in about the same proportion as the relative sizes of the blocks.

We identify a sufficiently low temperature and then obtain the normalized eigenvalues of the normalized ensemble matrix from 1000 regulated random walks on the data cloud. We discern that the spectral gap appears to occur between the third and sixth eigenvalue, so we apply spectral clustering with both $k=3$ and $k=6$ to the ensemble matrix. When $k=3$, 582 of the 864 points (or 67%) assigned to the first cluster belong to the low-velocity block, while 687 of the 1317 points (or 52%) assigned to the third cluster belong to the high-velocity block. Yet the low-velocity block comprises only 39% of the data, while the high-velocity block comprises only 38%. This provides evidence that the occurrence of extreme values in the EEG is influenced by the rat's movement. More significantly, for $k=6$, 79% of the nodes assigned to the first cluster belong to the low-velocity block. This analysis using our clustering method brings out a significant relationship between an animal's locomotion and its neurological activity. That is, when extreme values of the EEG occur it is much more likely that the animal is relatively still, perhaps asleep.

Next we apply our procedure to the iris data set [25], which has become somewhat of a benchmark for clustering

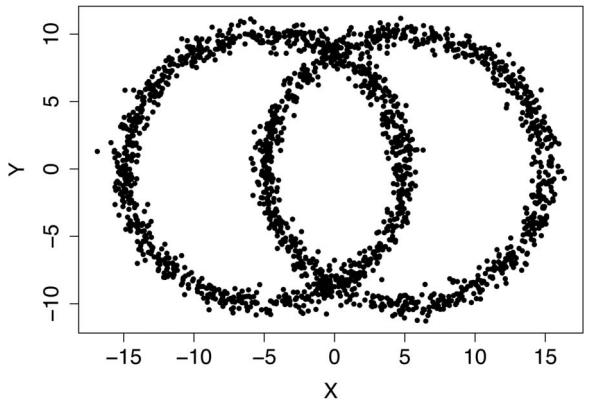

FIG. 8. Data cloud consisting of two intersecting rings of points (x,y) in $\mathbb{R}^2$.

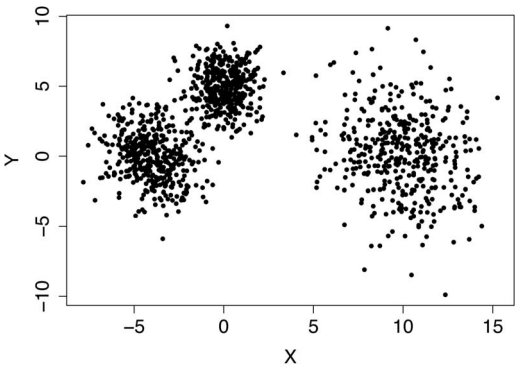

FIG. 9. Data cloud consisting of points (x,y) in $\mathbb{R}^2$ among three clusters of different densities.

procedures. These data include measurements on the length and width of both sepal and petal for 150 iris blooms split evenly among three species (setosa, versicolor, and virginica). The measurements for the versicolor and virginica clusters overlap considerably, making accurate classification difficult. We gradually increase the temperature and run an ensemble of regulated random walks until the eigenplot indicates two or three clusters. We apply spectral clustering to the resulting ensemble matrix with three clusters specified to assign the data among them. We find that all 50 setosa irises and all 50 virginica irises are correctly classified. But 25 of the 50 versicolor irises are misclassified as virginica, and two as setosa. Nevertheless, the overall misclassification rate is a low 18%. This is comparable, for example, to the result of the superparamagnetic clustering procedure of Blatt *et al.* [12], in which 125 irises were correctly classified and 25 were left unclassified. Hence the effectiveness of our approach is strongly confirmed, given that the versicolor and virginica species are difficult to distinguish by any method.

To test the performance of our method on clusters that cross each other in space, we apply it to the two intersecting rings shown in Fig. 8, which were successfully distinguished by the method in [10]. In this case, the eigenvalue plots resulting from ensembles of regulated random walks from low to high temperatures do not reveal a spectral gap until the highest temperatures are reached, at which point the gap occurs at the first eigenvalue. That is, our procedure fails to detect the two rings as separate clusters, but only detects the entire cloud as a single cluster. One might reasonably contend that the perception of two clusters here requires prior knowledge that the clusters are circular in shape, and that without such prior knowledge there is no legitimate basis for claiming that there are two clusters. Further investigation is needed to determine whether the regulated random walk can be enhanced to be able to distinguish intersecting clusters of this type if it is known *a priori* that the clusters are circular.

Finally, we test our method on clusters of different densities. We consider the three clusters in $\mathbb{R}^2$ shown in Fig. 9, which were each generated from a bivariate normal with a distinct mean and covariance. At the lowest temperature, our ensemble of regulated random walks results in a matrix with three strong eigenvalues followed by a large gap. Hence the presence of three clusters is correctly identified. When spectral clustering is then performed on this ensemble matrix with three clusters specified, we obtain a very low misclassification rate of 2.17%. As the temperature increases the two clusters on the left merge, and then the entire cloud merges so that the hierarchy of the geometry is accurately discovered. Hence different cluster densities are not an obstacle to our approach at all.

VII. DISCUSSION

We perceive the global geometry of a data cloud as a hierarchy composed of core clusters within conglomerate clusters. An algorithm is devised to construct this hierarchy of global information by incorporating both temperature and time concepts and without causing mathematical artifacts. As we thread through the hierarchy by varying the temperature, the regulated random walk incorporates the time concept through the recurrence times of node removal. Consequently, through an ensemble of such walks, this algorithm reveals realistic information about the number of clusters and the cluster membership. The composition of clusters constituting the base level of the hierarchy is especially important in real-world applications. This geometric point of view of a data cloud and the idea of using the ensemble of regulated random walks on its structure is similarly applicable for discovering the hierarchy of communities in a complex network.

Since the point of view of global geometry for a data cloud is intrinsically genuine, and our technique cleverly addresses the most fundamental questions, we anticipate that the ideas and the algorithm presented here will be recognized as a major improvement in the exploration of data clouds and complex network structures. For instance, this hierarchical global structural information can be taken as the basis for the study of system dynamics in biology and in social networking. We anticipate that this approach will provide researchers with a valuable tool for analyzing clustered data clouds when the intrinsic clusters vary in size, shape, and density.

We find that for $n \leq 2000$, the computational demand of performing a large ensemble of random walks on a set of n nodes over a range of temperatures is manageable, since the task is easily run in parallel on multiple processors. As n increases, however, the memory allocation for multiple $n \times n$ data structures eventually bogs down the processing.

Our R code, which is available upon request, may be optimized in the hands of skilled programmers using either R or other platforms. Nevertheless, techniques for very large n are required. We are currently developing one such technique which involves partitioning the data among blocks of manageable size such that each node has an equal probability of assignment to any block, then performing our clustering method to each block separately. The combination of the blockwise results into a final result presents some challenges that are yet to be resolved. The idea of a multi-grid approach as used in [20] is another potential methodology for handling very large data sets.

ACKNOWLEDGMENTS

The authors wish to thank Dr. Loren Frank and Margaret Carr at the University of California, San Francisco, for provision of the EEG data. This work was supported in part by the NSF under Grants No. HSD 0826844 and No. DMS-1007219 and by the NIH under Grant No. 1R01AG025218-01A2.

[1] P. H. A. Sneath and R. R. Sokal, *Numerical Taxonomy: The Principles and Practices of Numerical Classification* (Freeman, San Francisco, 1973).
[2] J. L. Stack and F. Murtagh, *Astronomical Image and Data Analysis* (Springer, New York, 2002).
[3] G. Carlsson, Bull. Am. Math. Soc. **46**, 255 (2009).
[4] P. Niyogi, S. Smale, and A. Weinbeger, Discrete Comput. Geom. **39**, 419 (2008).
[5] M. Belkin and P. Niyogi, Neural Comput. **15**, 1373 (2003).
[6] R. R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker, Proc. Natl. Acad. Sci. U.S.A. **102**, 7426 (2005).
[7] R. R. Coifman and S. Lafon, Appl. Comput. Harmon. Anal. **21**, 5 (2006).
[8] J. Shi and J. Malik, IEEE Trans. Pattern Anal. Mach. Intell. **22**, 888 (2000).
[9] A. Ng, M. Jordan, and Y. Weiss, in *Advances in Neural Information Processing Systems*, edited by T. Dietterich, S. Becker, and Z. Ghahramani (MIT Press, Cambridge, 2002), Vol. 14, pp. 849–856.
[10] D. Kushnir, M. Galun, and A. Brandt, Pattern Recogn. **39**, 1876 (2006).
[11] K. Rose, E. Gurewitz, and G. C. Fox, Phys. Rev. Lett. **65**, 945 (1990).
[12] M. Blatt, S. Wiseman, and E. Domany, Phys. Rev. Lett. **76**, 3251 (1996).
[13] S. Still and W. Bialek, Neural Comput. **16**, 2483 (2004).
[14] C. Fraley and A. Raftery, Comput. J. **41**, 578 (1998).
[15] R. Tibshirani, G. Walther, and T. Hastie, J. R. Stat. Soc. Ser. B (Stat. Methodol.) **63**(Part 2), 411 (2001).
[16] D. M. Titterington, S. M. M. Smith, and U. E. Markov, *Statistical Analysis of Finite Mixture Distributions* (Wiley, New York, 1985).
[17] S. C. Johnson, Psychometrika **32**, 241 (1967).
[18] M. Girvan and M. Newman, Proc. Natl. Acad. Sci. U.S.A. **99**, 7821 (2002).
[19] M. Blatt, S. Wiseman, and E. Domany, Neural Comput. **9**, 1805 (1997).
[20] E. Sharon, M. Galun, D. Sharon, R. Basri, and A. Brandt, Nature **442**, 810 (2006).
[21] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, Science **220**, 671 (1983).
[22] A. Góes-Neto, M. V. C. Diniza, L. B. L. Santos, S. T. R. Pinho, J. G. V. Miranda, T. P. Thierry Petit Lobao, P. Ernesto, E. P. Borges, C. N. El-Hani, and R. F. S. Andrade, Biosystems **101**, 59 (2010).
[23] U. von Luxburg, Stat. Comput. **17**, 395 (2007).
[24] B. Nadler and M. Galun, in *Advances in Neural Information Processing Systems*, edited by B. Schölkopf, J. Platt, and T. Hoffman, (MIT Press, Cambridge, MA, 2007), Vol. 19, pp. 1017–1024.
[25] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis* (Wiley, New York, 1973).