Available online at www.sciencedirect.com
SCIENCE DIRECT
Pattern Recognition 39 (2006) 1315–1324
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

Scalable visual assessment of cluster tendency for large data sets

Richard J. Hathaway[a], James C. Bezdek[b], Jacalyn M. Huband[b,*]

[a]*Department of Mathematical Sciences, Georgia Southern University, Statesboro, GA 30460, USA*
[b]*Computer Science Department, University of West Florida, Pensacola, FL 32514, USA*

Received 12 August 2005; received in revised form 6 February 2006; accepted 10 February 2006

Abstract

The problem of determining whether clusters are present in a data set (i.e., assessment of cluster tendency) is an important first step in cluster analysis. The visual assessment of cluster tendency (VAT) tool has been successful in determining potential cluster structure of various data sets, but it can be computationally expensive for large data sets. In this article, we present a new scalable, sample-based version of VAT, which is feasible for large data sets. We include analysis and numerical examples that demonstrate the new scalable VAT algorithm.

Keywords: Clustering; Similarity measures; Cluster validity; Data visualization; Scalability

1. Introduction

Recently, extended versions of fuzzy c-means clustering algorithms for very large image [1], general object [2] and relational [3] data sets have been developed. These extended c-means algorithms have two types of useful application: (1) to provide faster clustering results when the large data set is still small enough so that application of a conventional form of c-means clustering is possible; and (2) to simply provide (any) clustering results when the data set is so large that application of a conventional version of c-means is not practical (either because of the time or space required). A requirement to run any of the extended (or conventional) forms of c-means clustering is a good choice for c, the number of clusters. The purpose of this paper is to describe, analyze and demonstrate a visual method for determining the number of clusters that can be applied to very large data sets in a computationally efficient manner. The new method is a sample-based version of the visual assessment of cluster tendency procedure from Ref. [4]. We begin with some necessary background.

Our focus is a type of preliminary data analysis related to the pattern recognition problem of clustering. Clustering or cluster analysis is the problem of partitioning a set of objects $O = \{o_1, \ldots, o_N\}$ into c self-similar subsets based on available data and some well-defined measure of (cluster) similarity. In some cases, a geometric description of the clusters (e.g. by "cluster centers" in data space) is also desired and some clustering methods produce such geometric descriptors. The type of clusters found is strongly related to the properties of the mathematical model that underlies the clustering method. All clustering algorithms will find an arbitrary (up to $1 \leqslant c \leqslant N$) number of clusters, even if no "actual" clusters exist. Therefore, a fundamentally important question to ask before applying any particular (and potentially biasing) clustering algorithm is: Are clusters present at all?

The problem of determining whether clusters are present as a step prior to actual clustering is called the *assessment of clustering tendency*. Various formal (statistically based) and informal techniques for tendency assessment are discussed in Refs. [5,6]. The technique proposed here is visual, and visual approaches for various data analysis problems have been widely studied in the last 30 years; [7,8] are standard sources for many visual techniques. The basis for the new method for large data sets developed in this article is the

* Corresponding author. Tel.: +1 850 474 2304; fax: +1 850 857 6056.
E-mail address: jhuband@cs.uwf.edu (J.M. Huband).

0031-3203/$30.00
doi:10.1016/j.patcog.2006.02.011

visual assessment of tendency (VAT) procedure from Ref. [4]. The VAT approach presents pairwise dissimilarity information about the set of objects $O=\{o_1,\ldots,o_N\}$ as a square digital image with N^2 pixels, after the objects are suitably reordered so that the image is better able to highlight potential cluster structure. To go further into the VAT approach requires some additional background on the types of data typically available to describe the set $O=\{o_1,\ldots,o_N\}$.

There are two common data representations of O upon which clustering can be based. When each object in O is represented by a (column) vector $\mathbf{x}$ in $\mathscr{R}^s$, the set $X=\{\mathbf{x}_1,\ldots,\mathbf{x}_N\}\subset\mathscr{R}^s$ is called an *object data* representation of O. The kth component of the ith feature vector (x_{ki}) is the value of the kth feature (e.g., height, weight, length, etc.) of the ith object. It is in this data space that practitioners sometimes seek geometrical descriptors (often called prototypes) of the clusters. Alternatively, when each *pair* of objects in O is represented by a relationship between them, then we have *relational data*. The most common case of relational data is when we have (a matrix of) dissimilarity data, say $R=[R_{ij}]$, where R_{ij} is the pairwise dissimilarity measure (usually a distance) $d(o_i,o_j)$ between objects o_i and o_j, for $1\leqslant i,\ j\leqslant N$. More generally, R can be a matrix of similarities based on a variety of measures [9,10].

The VAT tool is widely applicable because it displays a reordered form of dissimilarity data, which itself can *always* be obtained from the original data for O. If the original data consists of a matrix of pairwise (symmetric) similarities $S=[S_{ij}]$, then dissimilarities can be obtained through several simple transformations. For example, we can take

$$R_{ij}=S_{\max}-S_{ij}, \tag{1}$$

where $S_{\max}$ denotes the largest similarity value. If the original data set consists of object data $X=\{\mathbf{x}_1,\ldots,\mathbf{x}_N\}\subset\mathscr{R}^s$, then R_{ij} can be computed as $R_{ij}=\|\mathbf{x}_i-\mathbf{x}_j\|$, using any convenient norm on $\mathscr{R}^s$. If the original data has missing components (is incomplete), then any existing data imputation scheme can be used to "fill in" the missing part of the data prior to processing. A discussion of various options for inexpensively handling the missing data is given in Ref. [4]. The main point here is that the dissimilarity data needed to apply VAT is available in virtually *all* numerical data sets.

The original VAT procedure is stated next. We assume that R is symmetric, and has nonnegative off-diagonal entries and zero diagonal entries. In general, the functions, arg max and arg min, in Steps 2 and 3 are set valued, so that the procedure selects any of the optimal arguments. The reordering found by VAT is stored in array $P=(P(1),\ldots,P(N))$.

VAT: Visual Assessment of (Cluster-) Tendency

Input: The user supplies the full $N\times N$ matrix of pairwise dissimilarities R.

Step 1. Set $K=\{1,2,\ldots,N\}$.
Select $(i,j)\in\arg\min_{p\in\mathrm{K},q\in\mathrm{K}}\{R_{pq}\}$.
Set $P(1)=i$; $I=\{i\}$; and $J=K-\{i\}$.

Step 2. For $t=2,\ldots,N$:
Select $(i,j)\in\arg\min_{p\in\mathrm{I},q\in\mathrm{J}}\{R_{pq}\}$.
Set $P(t)=j$; Replace $I\leftarrow I\cup\{j\}$ and $J\leftarrow J-\{j\}$.
Next t.

Step 3. Form the ordered dissimilarity matrix $\tilde{R}=[\tilde{R}_{ij}]=[R_{P(i)P(j)}]$, for $1\leqslant i,\ j\leqslant N$.

Step 4. Display $\tilde{R}$ as an intensity image, scaled so that $\max\{\tilde{R}_{ij}\}$ corresponds to white and 0 corresponds to black.

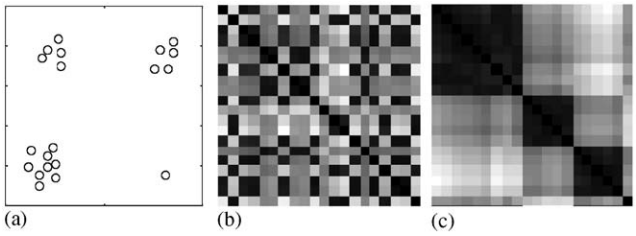

Fig. 1. (a) Object data. (b) Image for original R. (c) Image for VAT-ordered $\tilde{R}$.

The VAT ordering algorithm can be implemented in $\mathscr{O}(N^2)$ time complexity and is similar to Prim's algorithm for finding a minimal spanning tree (MST) of a weighted graph (see, for example [11] for a description of Prim's algorithm). The main differences between VAT and Prim's algorithm are that: (i) we are not interested in representing the MST, but only in finding the order in which the vertices are added as it is grown; and (ii), we specify a method for choosing the initial vertex that depends on the maximum edge weight in the underlying complete graph. (This choice of initial vertex gives nicer images, by avoiding a phenomenon known as "zigzagging", which is discussed in Ref. [4].) The permuted indices of the N objects are stored in the array P. Note that distances in $\tilde{R}$ are *not* recomputed; instead, we simply rearrange the rows (and columns) of R to construct $\tilde{R}$.

From Ref. [4], we repeat a small example in Fig. 1 to show the reader how well-separated cluster structure is indicated as dark diagonal blocks in the intensity image display of the VAT-ordered $\tilde{R}$. Fig. 1(a) gives a scatter plot of a small data set in $\mathscr{R}^2$. A display of the relational data matrix $R=[r_{ij}]=[\|\mathbf{x}_i-\mathbf{x}_j\|]$ using the Euclidean norm to convert X to R in Fig. 1(b) does not indicate the structure of the data set. After the relational matrix R is reordered by VAT and displayed as $\tilde{R}$ in Fig. 1(c), the structure is apparent. We see $c=4$ clusters in view 1(c), indicated by the four dark blocks along the main diagonal. Moreover, the size of each block corresponds directly to the number of points in each cluster. Notice the singleton! Certainly, VAT is not needed when a scatter plot such as Fig. 1(a) is possible, but we use this simple example of object data in $\mathscr{R}^2$ to help the reader correlate (visual) spatial clusters with VAT images.

We finish this section by briefly surveying the closest relatives of VAT. The earliest published reference we can find that discusses visual displays (as images) of clusters is the SHADE approach of Ref. [12]. SHADE approximates what for VAT is a nice digital image representation of clusters using a crude 15 level halftone scheme created by over striking standard printed characters. SHADE displays the lower triangular part of the complete square display. Visual identification of (triangular) patterns is more difficult than when a full, square display is used. SHADE was used *after* application of a hierarchical clustering scheme, as an alternative to visual displays of hierarchically nested clusters via the standard dendrogram. Another display technique used after application of a clustering scheme is the VCV (visual cluster validity) procedure in Ref. [13], which uses a VAT-like intensity image to represent the quality of clusterings obtained by any prototype-generating clustering method.

Closely related to SHADE, but presented more in the spirit of finding clusters rather than displaying clusters found with an outsourced algorithm is the "graphical method of *shading*" described in [14, p. 577]. Some software for visualizing distance data is available at http://www.genlab.tudelft.nl/. More recently, similarity-based intensity images, formed using kernel functions, have been used in Refs. [15,16] to provide guidance in determining the number of clusters, but no useful ordering scheme is offered there to facilitate the approach.

Huband et al. [17] proposed an upgrade of VAT to a new algorithm they called reVAT. VAT and reVAT differ importantly in several ways. First, reVAT performs a quasi-ordering of the objects, based on a threshold parameter. The quasi-ordering reduces the computational complexity to $\mathcal{O}(cN)$, where c is the number of potential clusters. Second, reVAT replaces the intensity image with a series of one-dimensional profile graphs. The profile graphs allow the algorithm to display information when the dimensions of the image may exceed the resolution of the display monitor. However, the profile graphs are not as easily interpretable as a full VAT-ordered image. Huband et al. [18] address this problem by extending reVAT to bigVAT, which uses the profile graphs to select a sample of objects. The quasi-ordered dissimilarity data of the sampled objects are displayed as a VAT-like intensity image, but this image may not be as descriptive as a VAT-ordered image.

2. sVAT: scalable visual assessment of tendency

We are interested in developing a scalable, sample-based version of VAT that can be used on large data sets. *Scalability* is often cited as a qualification for clustering algorithms for large data sets. An algorithm is *scalable* if its runtime complexity increases linearly with the number of records in the input data [19]. So ordinary VAT, with a runtime complexity of $\mathcal{O}(N^2)$, fails to be a scalable method of assessing cluster tendency. As a practical matter, applying VAT to any relational matrix with $N > 5000$ is at or beyond what the current typical PC can conveniently handle. If the form of the original data is object data X, then there is also the additional cost of computing R from X in order to do VAT.

The scalable visual assessment of tendency (sVAT) procedure proposed in this section satisfies the definition of scalability and can be efficiently applied to any sized relational data set. It produces a true VAT ordered image for the sample that is representative of the full data image, does not involve any sensitive thresholding parameter, and requires the user to only supply choices for two parameters: c', an *overestimate* of the true number of clusters, and n, the desired (approximate) sample size. In a nutshell, sVAT selects a sample of (approximately) size n from the full set of objects $O = \{o_1, \ldots, o_N\}$, and performs VAT on the sample. The sample is chosen so that it contains a cluster structure similar to the full set. This is done by first picking a set of c' *distinguished objects*, selected to provide representation of each of the clusters. Then, the remainder of the sample is built by choosing additional data near each of the distinguished objects. The new procedure is stated next, followed by an analysis of its important properties.

sVAT: Scalable Visual Assessment of Tendency

Input: The user supplies the required elements of an $N \times N$ matrix of pairwise dissimilarities R; c': an overestimate of the true but unknown number of clusters c; and an (approximate) sample size n.

Step 1. *Select the indices $m_1, \ldots, m_{c'}$ of the c' distinguished objects.*
Select $m_1 = 1$.
Initialize the search array $d = (d_1, \ldots, d_N) = (r_{1,1}, \ldots, r_{1,N})$.
For $t = 2, \ldots, c'$:
Update $d \leftarrow (\min\{d_1, r_{m_{t-1},1}\}, \min\{d_2, r_{m_{t-1},2}\}, \ldots, \min\{d_N, r_{m_{t-1},N}\})$
Select $m_t \in \arg\max_{1 \leqslant j \leqslant N} \{d_j\}$
Next t

Step 2. *Group each object in $\{o_1, \ldots, o_N\}$ with its nearest distinguished object.*
Initialize the respective distinguished objects' index sets $S_1 = S_2 = \cdots = S_{c'} = \emptyset$.
For $t = 1, \ldots, N$:
Select $k \in \arg\min_{1 \leqslant j \leqslant c'} \{r_{m_j,t}\}$ and then update $S_k \leftarrow S_k \cup \{t\}$.
Next t

Step 3. *Select some data for the sample near each of the distinguished objects.*
For $t = 1, \ldots, c'$, compute the tth group representative subsample size $n_t = \lceil n * |S_t|/N \rceil$, where $\lceil\ \rceil$ denotes the ceiling function.

Randomly choose n_1 indices from S_1, n_2 indices from $S_2, \ldots,$ and $n_{c'}$ indices from $S_{c'}$. Let S denote the union of all the randomly chosen indices and define $\hat{n} = |S|$. Form R_S, the $\hat{n} \times \hat{n}$ principal submatrix of R corresponding to the row/column indices in S.

Step 4. *Apply VAT to* R_S.

First we provide some intuitive explanation regarding two parts of the sVAT algorithm. Each distinguished object selected in Step 1 effectively specifies a region in data space that will be well represented in the sample. Picking the distinguished objects to be a diverse set, as in Step 1, is intended to ensure that the sample includes representation of all the important and substantially different parts of the full data set. Step 3 gives, for $t = 1, \ldots, c'$, the subsample size $n_t = \lceil n * |S_t|/N \rceil$ corresponding to the region represented by the tth distinguished object, where the outer ceiling function is used to ensure that n_t has an integer definition. Most importantly, note that the subsample proportion n_t/n corresponding to objects near the tth distinguished object will closely (or exactly, if $n * |S_t|/N$ is an integer) match the full sample proportion $|S_t|/N$.

Next, we examine the complexity of sVAT. Steps 1 and 2 require (for fast execution) retrieval and storage of $c'N$ entries of R. The work done identifying the distinguished objects in the loop of Step 1 (min and max operations) is $\mathcal{O}(c'N)$. If the available data is object data $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subset \mathscr{R}^s$, then acquisition of the required elements of R from X has an additional complexity of $\mathcal{O}(sc'N)$. The complexity in Steps 2 and 3 prior to forming R_S is $\mathcal{O}(c'N)$. Formation of R_S from original data X requires $\mathcal{O}(s\hat{n}^2)$ additional operations, and storage of R_S involves storage of $\hat{n}^2$ numbers (regardless of the form of the original data). Finally, Step 4 executes at a cost of $\mathcal{O}(\hat{n}^2)$ operations.

Now $\hat{n} \leqslant n + c'$, and equality can only hold in this if the ceiling function rounds up for each of $n_1, \ldots, n_{c'}$ in Step 3. So the storage complexity of sVAT is $\mathcal{O}(\max\{c'N, (n + c')^2\})$ and the runtime complexity is $\mathcal{O}(\max\{c'N, (n + c')^2\})$ if the form of the original data is relational and $\mathcal{O}(\max\{sc'N, s\hat{n}^2, (n + c')^2\})$ if the original data is $X \subset \mathscr{R}^s$. Thus, sVAT is scalable since the runtime complexity increases linearly with the number of records (N) in the input data.

Scalability is important, but even more important is the ability to produce a good approximation to the full data VAT image using the sample generated sVAT image. We will give some examples in the next section to show that sVAT is indeed representative of VAT, but we continue here with a theoretical result that gives some reason for optimism about sVAT's general performance. Dunn [20] defined the notion of compact and separated clusters in the following way. For a set of objects $O = \{o_1, \ldots, o_N\}$ with corresponding relational dissimilarity data R, we say that a partitioning $O^{(1)}, O^{(2)}, \ldots, O^{(c)}$ of O is compact and separated (CS) relative to R if each of the possible intra-cluster distances is strictly less than each of the possible intercluster distances. We state this by saying that O can be partitioned into c CS clusters.

The main result below (Proposition 2) shows that if the data consists of CS clusters and there is no rounding up by the ceiling function in Step 3, then the original data set and sample will have the same proportional make up as the individual CS clusters. Based on this perfect representation in the compact and separated case, we expect the sampling to provide a reasonably good representation of the cluster structure of any full data set (whether it contains CS clusters or not). The two propositions giving the main theoretical result are next.

Proposition 1. *Suppose that the set of objects* $O = \{o_1, \ldots, o_N\}$ *represented by the relational dissimilarity matrix* R *can be partitioned into* $c \geqslant 1$ CS *clusters, and that* $c' \geqslant c$. *Then Step* 1 *of sVAT will select at least one distinguished object from each cluster.*

Proof. The result is trivially true for the case of $c = 1$. Now, suppose that O can be partitioned into $c \geqslant 2$ compact and separated clusters $O^{(1)}, O^{(2)}, \ldots, O^{(c)}$. When needed, we will indicate the cluster of a datum as a superscript in parentheses; e.g., $o_7^{(2)}$ indicates that object 7 is in (CS) cluster 2. Let $d(o_i, o_j) = r_{ij}$ denote the dissimilarity between o_i and o_j for $1 \leqslant i, j \leqslant N$. Since the clusters are compact and separated, it is true that

$$r_{k,p} = d(o_k^{(i)}, o_p^{(i)}) < d(o_k^{(i)}, o_j^{(h)}) = r_{k,j} \quad \text{for all } 1 \leqslant i \neq h \leqslant c, \quad \text{and applicable } k, p, j. \tag{2}$$

Assume that we first select object o_1, and that it happens to belong to $O^{(1)}$. Then the search array d in Step 1 is initialized as row 1 of R:

$$d = (r_{1,1}, \ldots, r_{1,N}) = (d(o_1, o_1), \ldots, d(o_1, o_N)).$$

Now applying (2) with $i = 1$, we see that the max element in d must correspond to a datum in $O^{(2)}, \ldots, O^{(c)}$ (but not in $O^{(1)}$), which implies that the second distinguished object chosen will belong to a cluster other than $O^{(1)}$. This completes the proof for the case of $c = 2$, and we now continue to the more general case of $c \geqslant 3$.

Suppose that a maximum value occurs in the second entry of d and that the next distinguished object selected is o_2 (i.e., $m_2 = 2$ in Step 1); and for further notational simplicity suppose that o_2 belongs to cluster $O^{(2)}$. The search array d is now updated to

$$\begin{aligned} z &= (\min\{r_{1,1}, r_{2,1}\}, \ldots, \min\{r_{1,N}, r_{2,N}\}) \\ &= (\min\{d(o_1, o_1), d(o_2, o_1)\}, \ldots, \min\{d(o_1, o_n), \\ &\quad d(o_2, o_n)\}). \end{aligned}$$

Suppose that a maximum entry is found in the third spot and that o_3 is the third object selected ($m_3 = 3$). We will prove by contradiction that o_3 cannot belong to $O^{(1)}$ or $O^{(2)}$.

The proof by contradiction is begun by assuming that o_3 *does* belong to either $O^{(1)}$ or $O^{(2)}$, say $O^{(1)}$. Selection of o_3 implies that

$$\min\{d(o_1, o_3), d(o_2, o_3)\} \geqslant \min\{d(o_1, o_j), d(o_2, o_j)\} \quad \text{for all } j = 1, \ldots, n. \tag{3}$$

But (3) implies that

$$d(o_1, o_3) \geqslant \min\{d(o_1, o_j), d(o_2, o_j)\} \quad \text{for all } j = 1, \ldots, n. \tag{4}$$

Now, let $j \geqslant 4$ be any index such that o_j is in neither $O^{(1)}$ nor $O^{(2)}$. (At least one such j exists since $c \geqslant 3$ and for $k = 1, 2, 3$ we have $o_k \in O^{(1)} \cup O^{(2)}$.) Without loss of generality we suppose that $j = 4$ satisfies (4) with $o_4 \in O^{(3)}$, and that $d(o_1, o_4) \leqslant d(o_2, o_4)$. Then (4) gives

$$d(o_1, o_3) \geqslant d(o_1, o_4), \tag{5}$$

where $o_1 \in O^{(1)}$, $o_3 \in O^{(1)}$, and $o_4 \in O^{(3)}$. But (5) contradicts (2) for $i = k = 1$, $p = h = 3$, and $j = 4$. The conclusion is that the third object chosen must be in a previously unrepresented cluster, and the argument extends to the fourth chosen, etc. It immediately follows that we have at least one distinguished object from each compact and separated cluster if $c' \geqslant c$. □

Proposition 2. *Suppose that the set of objects* $O = \{o_1, \ldots, o_N\}$ *represented by the relational dissimilarity matrix* R *can be partitioned into* $c \geqslant 1$ CS *clusters, and that* $c' \geqslant c$. *Further suppose that* $n|S_t|/N$ *in Step* 3 *of sVAT is an integer for* $t = 1, \ldots, c'$. *Then the proportion of objects in the sVAT sample from cluster* $O^{(i)}$ *equals the proportion of objects in the population from cluster* $O^{(i)}$, *for* $i = 1, \ldots, c$.

Proof. First notice that if $n * |S_t|/N$ is always an integer for $t = 1, \ldots, c'$, then the ceiling function in Step 3 never rounds up and the size of the sample is exactly $\hat{n} = |S| = n$. By Proposition 1, each CS cluster is represented by at least 1 distinguished object. First consider the case that the ith of the CS clusters $O^{(i)}$ is represented by only 1 distinguished object, say object m_t. Then since the clusters are CS, the inequalities of (2) imply that all objects in cluster $O^{(i)}$ will be associated with distinguished object m_t, and that no objects outside of $O^{(i)}$ will be associated with distinguished object m_t. This implies that the proportion of the population made up by points in $O^{(i)}$ is $|S_t|/N$. The proportion of the sample made up by points in $O^{(i)}$ is the ratio of $n_t = n * |S_t|/N$ to n, which is just $|S_t|/N$.

Now consider the case that the ith of the CS clusters, $O^{(i)}$, is represented by more than 1 distinguished object. We can illustrate the general situation and minimize notational difficulties without loss of generality by assuming that $O^{(i)}$ is represented by exactly two distinguished objects, say objects m_r and m_s. Again using the inequalities in (2) we have that each object in $O^{(i)}$ must be associated with either m_r or m_s, and that no object outside of cluster $O^{(i)}$ is associated with either m_r or m_s. So the proportion of the population made up by points in cluster i is the sum $|S_r|/N + |S_s|/N$. But the proportion of the sample made up by points in $O^{(i)}$ is the sum of the ratios $n_r = n * |S_r|/N$ to n and $n_s = n * |S_S|/N$ to n, which agrees with the population proportion and completes the proof. (Note that the choice of tie-breaking strategies employed in Steps 1 and 2 of sVAT are irrelevant to the argument in this proof.) □

If $n * |S_t|/N$ in Step 3 of sVAT is an integer for $t = 1, \ldots, c'$, we get perfect cluster representation in the sample. In cases when this is not true, the sample and full data set proportions are still very nearly equal. For example, suppose that the sole distinguished object for CS cluster $O^{(1)}$ is object m_1 and that $n|S_1|/N$ is not an integer. Then the absolute value of the difference between the sample proportion and full data set proportion for CS cluster $O^{(1)}$ is

$$\begin{aligned} &|\lceil n * |S_1|/N \rceil/\hat{n} - |S_1|/N| \\ &\quad = |\lceil n * |S_1|/N \rceil/\hat{n} - \hat{n} * |S_1|/N/\hat{n}| \\ &\quad < (1 + c' * |S_1|/N)/\hat{n}, \end{aligned} \tag{6}$$

where the inequality uses the fact that $n \leqslant \hat{n} \leqslant n + c'$, the triangle inequality, and an optimization argument for maximizing piecewise linear functions of the form $f(\hat{n}) = |\alpha - \beta\hat{n}|$. If the particular cluster has q corresponding distinguished objects, then the bound on the difference in sample and full data set proportions becomes

$$q(1 + c' * |S_1|/N)/\hat{n}. \tag{7}$$

From (6) and (7) we see that as long as c' is not too large and $\hat{n}$ is not too small (both realistic assumptions for very large real worlds data sets) then the sample and full data set proportions will closely agree. As stated before the propositions, we expect the good sample representation in the CS cluster case to carry over to the general case, and we will examine this using several numerical examples in the next section.

3. Numerical examples

All computations are done using MATLAB on a PC with 1024 MB ram and 3.0 Ghz P4 chip. The first five examples use relational data calculated from sets of two-dimensional object data. VAT and sVAT images are not needed to deduce the cluster structure of two-dimensional object data sets, since a scatter plot is computationally cheaper and more informative. But we use this type of data, and include scatter plots, so that the reader is able to compare apparent visual structure in a scatter plot with the appearance of the corresponding VAT or sVAT image. The relational data used in the first five examples is generated from the object data $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ using the Euclidean norm $\| \ \|_2$ by

$$R = [r_{jk}] = [d(o_j, ok)] = \left[\sqrt{\|\mathbf{x}_j - \mathbf{x}_k\|_2}\right]. \tag{8}$$

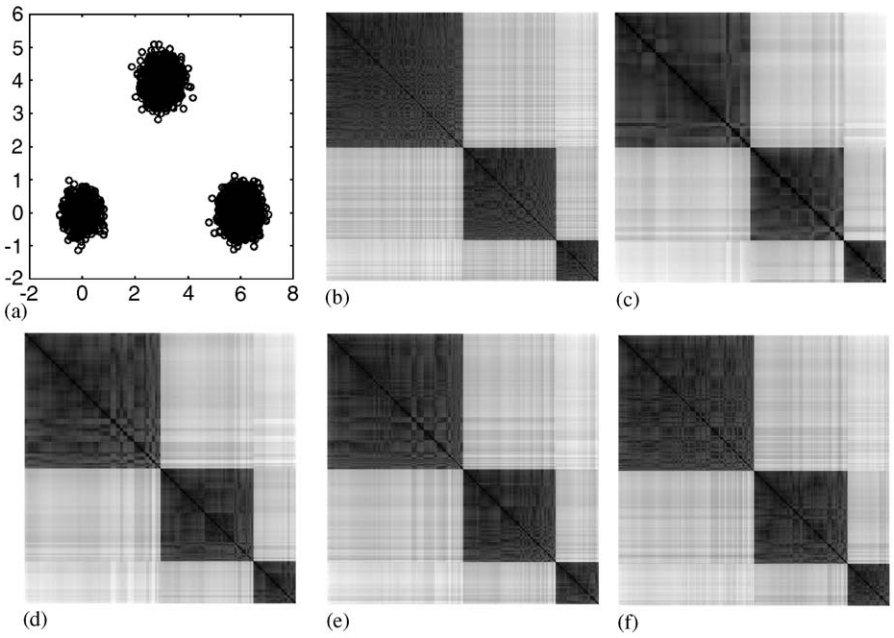

Fig. 2. Scatter plot and images for normal mixture with $\sigma^2 = 0.1$ and $c' = 5$. (a) Scatter plot of X. (b) VAT; $N = 5000$ (156 s). (c) sVAT; $n = 100$ (0.3 s). (d) sVAT; $n = 300$ (0.8 s). (e) sVAT; $n = 500$ (1.8 s). (f) sVAT; $n = 1000$ (6.5 s).

The square root is used because it transforms image intensities to lighter shades, resulting in better visual acuity of the displayed images.

The first example uses a data set consisting of $N = 5000$ observations from a mixture of three normal distributions with the following component parameters: mixing proportions $\alpha_1 = 0.15$, $\alpha_2 = 0.35$ and $\alpha_3 = 0.50$; means $\mu_1 = (0, 0)^T$, $\mu_2 = (3, 4)^T$ and $\mu_3 = (6, 0)^T$; and covariance matrices $\Sigma_1 = \Sigma_2 = \Sigma_3 = \sigma^2 I$, where I is the 2×2 identity matrix and $\sigma^2 = 0.1$. This data consists of three CS clusters, as indicated in the scatter plot shown in Fig. 2(a). The VAT image for the full data set is displayed in Fig. 2(b). A series of corresponding sVAT images using $c' = 5$ distinguished features is shown in Figs. 2(c–f), corresponding to sample sizes of $n = 100, 300, 500$ and 1000, respectively. All four sVAT images are in close agreement with the VAT image, indicating the presence of three clusters, of varying cardinality. The computation times included in the figure demonstrate the economy of sVAT. sVAT runs $156/0.3 = 520$ times faster than VAT for $n = 100$; the visual information about cluster structure in the input data in views 2(b) and (c) is practically the same.

The second example is identical to the first except that the three covariance matrices $\Sigma_1 = \Sigma_2 = \Sigma_3 = \sigma^2 I$ use $\sigma_2 = 1.0$. The three clusters, which are shown in Fig. 3(a), are *not* compact and separated, so the result of Proposition 2 is not guaranteed. However, we see that the sVAT images in Figs. 3(c–f) capture the same gross structure as the VAT image in Fig. 3(b). Comparing Fig. 3 with Fig. 2, we see less contrast between the diagonal blocks and off diagonal elements in the images of Fig. 3. This indicates that the clusters are less well separated in the example of Fig. 3. Note that sVAT images may produce the diagonal blocks in a permuted order from those obtained in the full data VAT image. The main point of this second example is to demonstrate that sVAT can closely approximate VAT even when the clusters are not compact and separated. Comparing views 3(b) and (f) shows that when sVAT uses 20% ($n = 1000$) of the full data ($N = 5000$), its runtime is still 22 times faster than VAT while producing a qualitatively equivalent image.

The third example repeats the type of experiment done in the previous two examples using a data set consisting of $N = 5000$ points that are uniformly distributed in $[0, 1] \times [0, 1]$. This data set does not have a well-defined cluster structure like the earlier examples, although every clustering algorithm will always find "clusters" in the data. The standard sequence of images shown in Fig. 4 demonstrates good visual agreement between the sVAT and VAT images, all of which (correctly) fail to have well formed dark diagonal blocks indicating clusters. In summary, the first three examples show agreement between sVAT and VAT images for CS clusters, partly overlapping clusters, and a complete lack of clusters.

The previous examples all used $c' = 5$ distinguished features. How sensitive is sVAT to the number of distinguished features c'? Does simple random sampling from the full data set, which is equivalent to taking $c' = 1$, suffice? For sufficiently large sample sizes n, the law of large numbers guarantees that the cluster composition of a randomly chosen sample will be close to the cluster composition of the population; but the next example shows how the proper choice

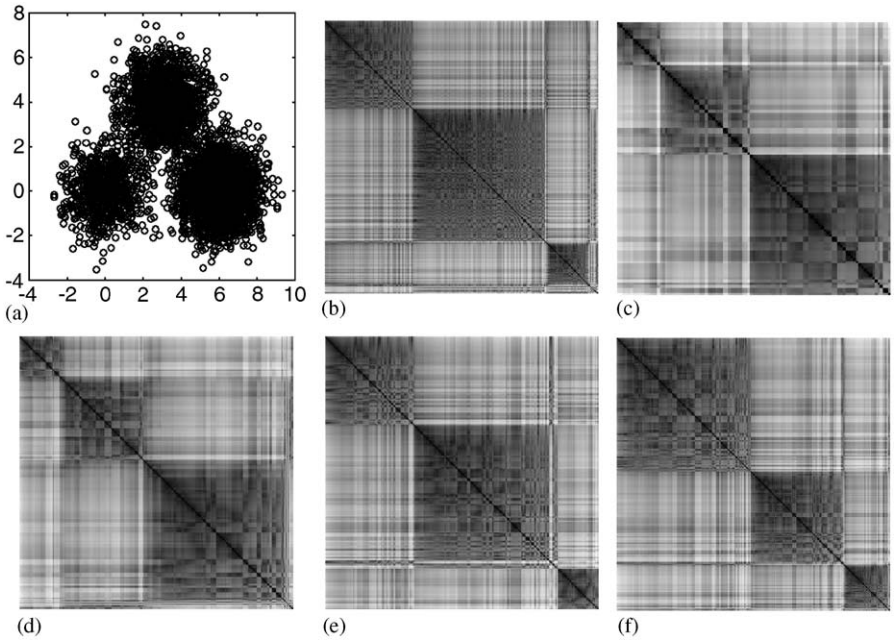

Fig. 3. Scatter plot and images for normal mixture with $\sigma^2 = 1.0$ and $c' = 5$. (a) Scatter plot of X. (b) VAT; $N = 5000$ (155 s). (c) sVAT; $n = 100$ (0.3 s). (d) sVAT; $n = 300$ (0.8 s). (e) sVAT; $n = 500$ (1.9 s). (f) sVAT; $n = 1000$ (6.9 s).

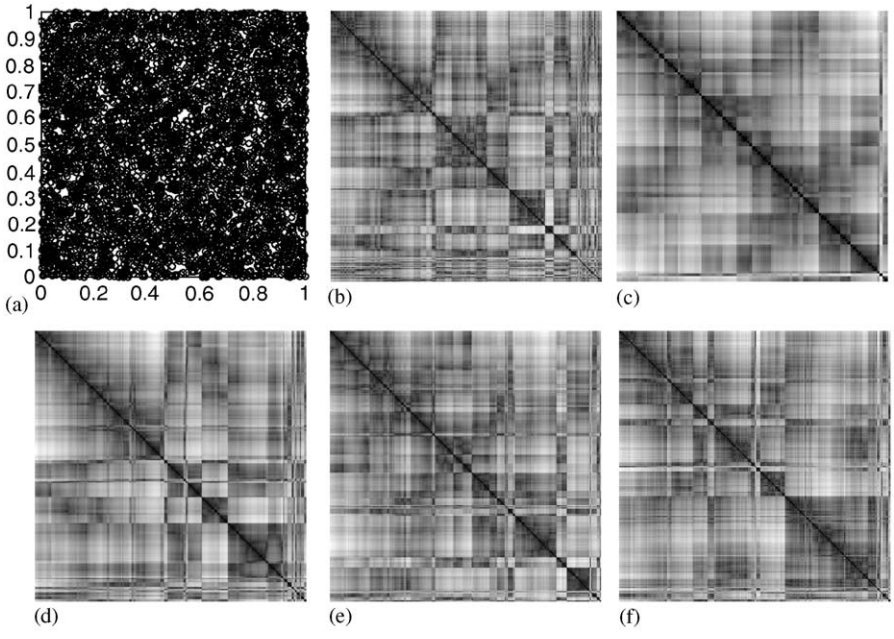

Fig. 4. Scatter plot and images for uniformly distributed data with $c' = 5$. (a) Scatter plot of X. (b) VAT; $N = 5000$ (161 s). (c) sVAT; $n = 100$ (0.3 s). (d) sVAT; $n = 300$ (0.8 s). (e) sVAT; $n = 500$ (1.8 s). (f) sVAT; $n = 1000$ (6.5 s).

of c' can help ensure good results in small sample cases. Fig. 5(a) gives a scatter plot of a data set of $N = 500$ points that can be partitioned into four CS clusters. The cardinality of each of the three small clusters is 20 and the cardinality of the single large cluster is 440. The VAT image of the full data set accurately shows the existence of a cluster with large cardinality and three other clusters of much smaller, equal cardinality.

Parts (c)–(f) of Fig. 5 give sVAT images for the sample size $n = 40$ and varying values of c'. The sVAT image of Fig. 5(c) corresponds to $c'=1$ and only indicates the presence of two clusters. The images of Figs. 5(d) and (e), for $c' = 2$

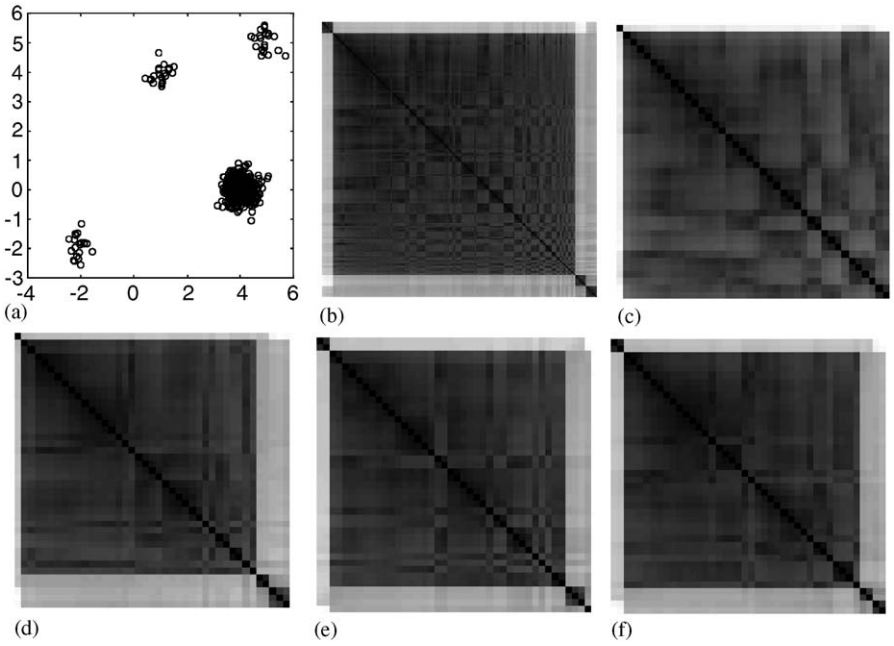

Fig. 5. Normal mixture data scatter plot and images for different values of c'. (a) Scatter plot of X. (b) VAT; $N = 500$. (c) sVAT; $n = 40$; $c' = 1$. (d) sVAT; $n = 40$; $c' = 2$. (e) sVAT; $n = 40$; $c' = 3$. (f) sVAT; $n = 40$; $c' = 4$.

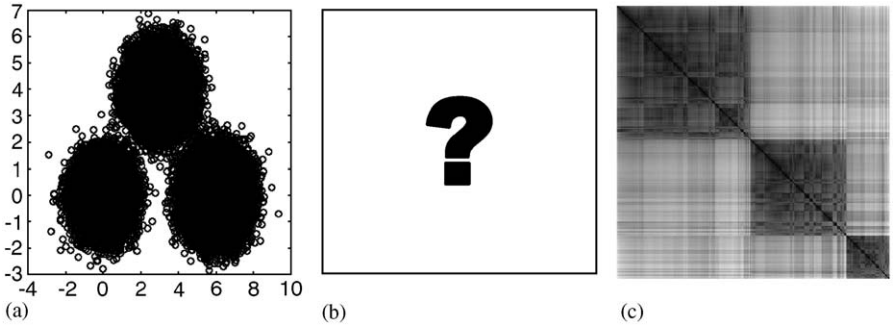

Fig. 6. Scatter plot and image for normal mixture in R^2 with $N = 100,000$, $\sigma^2 = 0.5$ and $c' = 5$. (a) Scatter plot of X. (b) VAT; $N = 100,000$ (?? s). (c) sVAT; $n = 500$ (6.81 s).

and 3, respectively, correctly indicate four clusters, but the represented cardinalities of the three smaller clusters are all different. However, using $c' = 4$ gives the sVAT image in Fig. 5(f), which most accurately represents the full data image in Fig. 5(b). Proposition 2 guarantees that each of the clusters will be represented for this example as long as $c' \geqslant 4$.

Next we demonstrate sVAT on a data set that is too large for conventional VAT. The data set consists of $N = 100,000$ object data generated from a mixture of normals with identical component parameters to those used in the examples of Figs. 2 and 3 except that the three covariance matrices $\Sigma_1 = \Sigma_2 = \Sigma_3 = \sigma^2 I$ for $\sigma^2 = 0.5$. Applying VAT to a problem of this size is not conveniently done on a 3.0 GHz PC using MATLAB for the following reasons: (1) the time required to execute the algorithm is estimated at about 18 h (not counting the increased overhead time caused by memory considerations); (2) the memory required to store just the top half of the relational data matrix is about 38 GB; and (3), the number of elements in a relational matrix of size $100,000 \times 100,000$ is greater than the maximum integer possible with MATLAB, so that element indexing becomes a problem.

In spite of the problems for VAT mentioned above, sVAT remains applicable and produces the reasonable image shown in Fig. 6(c) in 6.81 s. It correctly reflects the structure in the large data set whose scatter plot is included in Fig. 6(a). The representation in Fig. 6(b) depicts our inability to produce the actual VAT image of the full data set.

The final example involves a data set from the UCI Machine Learning Repository at http://www.ics.uci.edu/~mlearn/MLSummary.html and is included to provide an

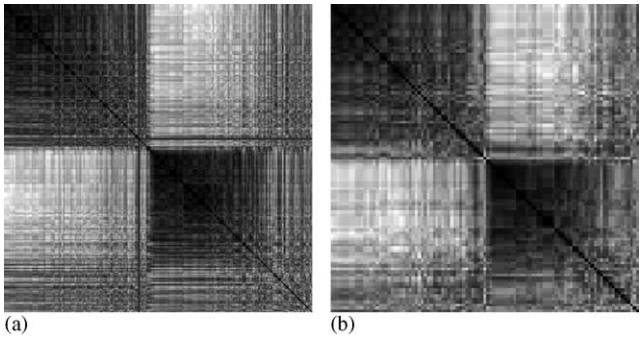

Fig. 7. VAT and sVAT images for 16×435 congressional data. (a) VAT; $N = 435$. (b) sVAT; $n = 100$; $c' = 5$.

example with collected (in contrast to artificially generated) data. Before we describe the data set of size $N = 435$, note the similarity between its VAT image and the corresponding sVAT approximation using a sample size of $n = 100$ and $c' = 5$ distinguished features (Fig. 7). Both images show two diagonal blocks indicating well defined clusters and a smaller subset of data (corresponding to the last 15% or so of the rows and columns) with no clear cluster structure. Most importantly, both images give roughly the same insight into the data set.

As it turns out, the data is generated from the Congressional Voting Records Database, and consists of the 1984 records of the 435 United States Representatives on 16 key votes. Votes were numerically encoded as 0.5 for "yea", −0.5 for "nay" and 0 for "unknown disposition", so that the voting record of each Congressman is represented using an object data vector in $\mathscr{R}^{16}$. The relational data are generated from the object data as pairwise squared Euclidean distances. The two identified classes are Republican (54.8%) and Democrat (45.2%). The VAT and sVAT images give evidence that there are two clusters in the data with similar voting records and another group of around 15% who fail to consistently vote with those in the well-defined clusters (or each other). The main point of this example is to demonstrate agreement of VAT and sVAT images for an example involving real data. Moreover, while the physical process which generates this data suggests that it should contain $c=2$ clusters, there is no way to directly corroborate this supposition for 16-D data. So, if you were unaware of the process producing the data, you would have no good way to deduce that you should look for $c = 2$ clusters. sVAT solves your problem!

4. Discussion

The sVAT (scalable visual assessment of tendency) procedure was introduced, analyzed and demonstrated using six numerical examples. sVAT produces an approximation to the VAT [4] image for a data set using a sample designed to have the same cluster representation as the full data set. Representation is obtained by first identifying distinguished objects located throughout the various clusters, and then sampling in a structured way so that some objects near each of the distinguished ones are included. The first part of the theoretical analysis tells us that the c' distinguished objects will infiltrate all c clusters in case the clusters are compact and separated and $c' \geqslant c$. The second part states that the cluster representation in the sample will match the cluster representation in the population, thus ensuring that the sVAT and VAT images will be similar. The first three numerical examples include cases of CS clusters, overlapping clusters, and no clusters, respectively; and in all cases the sVAT images closely approximate the full data VAT images. The fourth example demonstrates the importance of picking $c' \geqslant c$ in cases when the (approximate) sample size n is small and some of the clusters make up a small proportion of the overall full data set. While VAT becomes inconvenient on a PC at around $N = 5000$, the fifth example demonstrates good results (obtained in less than 7 s) using sVAT on a two-dimensional data set of $N = 100,000$ observations. The final example successfully applies sVAT to a real (i.e., collected in an actual survey) data set with good results.

The type of data structure found in the examples done here is essentially restricted to ellipsoidal clusters. A more complete library of VAT images for other types of structure is found in Ref. [4]. Current research related to sVAT includes: (1) the use of VAT and sVAT images to help determine good choices of kernel parameters in kernelized forms of c-means clustering algorithms, and (2) application of the procedure for selecting distinguished objects in Step 1 of sVAT to the problem of generating good (and cheap) initializations for c-means clustering.

References

[1] N.R. Pal, J.C. Bezdek, Complexity reduction for "large image" processing, IEEE Trans. Syst. Man Cybern. 32 (2002) 598–611.
[2] R.J. Hathaway, J.C. Bezdek, Extending fuzzy and probabilistic clustering to very large data sets, Comput. Statist. Data Analy. (2006) in press.
[3] J.C. Bezdek, R.J. Hathaway, J.M. Huband, C. Leckie, R. Kotagiri, Approximate clustering in very large relational data sets, Int. J. Intell. Syst. (2006) in press.
[4] J.C. Bezdek, R.J. Hathaway, VAT: a tool for visual assessment of (cluster) tendency, in: Proceedings of the International Joint Conference of Neural Networks, IEEE Press, Piscataway, NJ, 2002, pp. 2225–2230.
[5] A.K. Jain, R.C. Dubes, Algorithms for Clustering Data, Prentice-Hall, Englewood Cliffs, NJ, 1988.
[6] B.S. Everitt, Graphical Techniques for Multivariate Data, North-Holland, New York, 1978.
[7] J.W. Tukey, Exploratory Data Analysis, Addison-Wesley, Reading, MA, 1977.
[8] W.S. Cleveland, Visualizing Data, Hobart Press, Summit, NJ, 1993.
[9] I. Borg, J. Lingoes, Multidimensional Similarity Structure Analysis, Springer, New York, 1987.
[10] M. Kendall, J.D. Gibbons, Rank Correlation Methods, Oxford University Press, New York, 1990.
[11] K.H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill, New York, NY, 1999.

[12] R.F. Ling, A computer generated aid for cluster analysis, Commun. ACM 16 (1973) 355–361.
[13] R.J. Hathaway, J.C. Bezdek, Visual cluster validity (VCV) for prototype generator clustering models, Pattern Recognition Lett. 24 (2003) 1563–1569.
[14] R.A. Johnson, D.A. Wichern, Applied Multivariate Statistical Analysis, third ed., Prentice-Hall, Englewood Cliffs, NJ, 1992.
[15] M. Girolami, Mercer kernel-based clustering in feature space, IEEE Trans. Neural Networks 13 (2002) 780–784.
[16] D.-Q. Zhang, S.-C. Chen, Clustering incomplete data using kernel-based fuzzy c-means algorithm, Neural Process. Lett. 18 (2003) 155–162.
[17] J.M. Huband, J.C. Bezdek, R.J. Hathaway, Revised visual assessment of (cluster) tendency (reVAT), in: Proceedings of the North American Fuzzy Information Processing Society (NAFIPS), IEEE, Banff, Canada, 2004, pp. 101–104.
[18] J.M. Huband, J.C. Bezdek, R.J. Hathaway, bigVAT: visual assessment of cluster tendency for large data sets, Pattern Recognition 38 (2005) 1875–1886.
[19] V. Ganti, J. Gehrke, R. Ramakrishnan, Mining very large databases, IEEE Computer 32 (1999) 38–45.
[20] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, J. Cybern. 3 (1973) 32–57.

About the Author—RICHARD J. HATHAWAY received the B.S. degree in Applied Math from the University of Georgia in 1979 and the Ph.D. in Mathematical Sciences from Rice University in 1983. He is currently a Professor in the Department of Mathematical Sciences at Georgia Southern University. His research interests include pattern recognition and numerical optimization.

About the Author—JAMES C. BEZDEK received the BSCE from the University of Nevada (Reno) in 1969, and the Ph.D. in Applied Math from Cornell in 1973. He is currently a Professor in the Computer Science Department at the University of West Florida. Jim's interests include woodworking, optimization, motorcycles, pattern recognition, gardening, fishing, image processing, computational neural networks, blues music, and computational medicine. Jim is the founding editor of the International Journal of Approximate Reasoning and the IEEE Transactions on Fuzzy Systems, a fellow of the IEEE and IFSA, and recipient of the IEEE 3rd Millenium and Fuzzy Systems Pioneer medals.

About the Author—JACALYN M. HUBAND received the B.A. degree in Mathematics from the University of Virginia in 1982 and the Ph.D. in Computational Mathematics from Old Dominion University in 1997. She is a faculty member in the Computer Science Department at University of West Florida. Her research interests include inverse problems, mathematical modeling, and pattern recognition.