An Algorithm for Clustering Tendency Assessment

Yingkang Hu
Georgia Southern University
Department of Mathematical Sciences
Statesboro, GA 30460-8093
USA
yhu@GeorgiaSouthern.edu

Richard J. Hathaway
Georgia Southern University
Department of Mathematical Sciences
Statesboro, GA 30460-8093
USA
rhathaway@GeorgiaSouthern.edu

Abstract: The visual assessment of tendency (VAT) technique, developed by J.C. Bezdek, R.J. Hathaway and J.M. Huband, uses a visual approach to find the number of clusters in data. In this paper, we develop a new algorithm that processes the numeric output of VAT programs, other than gray level images as in VAT, and produces the *tendency curves*. Possible cluster borders will be seen as high-low patterns on the curves, which can be caught not only by human eyes but also by the computer. Our numerical results are very promising. The program caught cluster structures even in cases where the visual outputs of VAT are virtually useless.

Key–Words: Clustering, similarity measures, data visualization, clustering tendency

1 Introduction

In clustering one partitions a set of objects

$$O = \{o_1, o_2, \ldots, o_n\}$$

into c self-similar subsets (clusters) based on available data and some well-defined measure of similarity. But before using a clustering method one has to decide whether there are meaningful clusters, and if so, how many are there. This is because all clustering algorithms will find any number (up to n) of clusters, even if no meaningful clusters exist. The process of determining the number of clusters is called the *assessing of clustering tendency*. We refer the reader to Tukey [1] and Cleveland [2] for visual approaches in various data analysis problems, and to Jain and Dubes [3] and Everitt [4] for formal (statistics-based) and informal techniques for cluster tendency assessment. Other interesting articles include [13]–[17]. Recently the research on the *visual assessment of tendency* (VAT) technique has been quite active; see the original VAT paper by Bezdek and Hathaway [5], also see Bezdek, Hathaway and Huband [6], Hathaway, Bezdek and Huband [7], and Huband, Bezdek and Hathaway [8, 9].

The VAT algorithms apply to relational data, in which each *pair* of objects in O is represented by a relationship. Most likely, the relationship between o_i and o_j is given by their dissimilarity R_{ij} (a distance or some other measure; see [11] and [12]). These n^2 data items form a symmetric matrix $R = [R_{ij}]_{n\times n}$. If each object o_i is represented by a feature vector $x_i = (x_{i1}, x_{i2}, \ldots, x_{is})$, where x_{ij} are properties of o_i such as height, length, color, etc, the set

$$X = \{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^s$$

is called *object data* representation of the set O. In this case R_{ij} can be computed as the distance between x_i and x_j measured by some norm or metric in $\mathbb{R}^s$. In this paper if the data is given as object data X, we will use as R_{ij} the square root of the Euclidean norm of $x_i - x_j$, that is,

$$R_{ij} = \sqrt{\|x_i - x_j\|_2}.$$

The VAT algorithms reorder (through indexing) the points so that points that are close to one another in the feature space will generally have similar indices. Their numeric output is an *ordered dissimilarity matrix* (ODM). We will still use the letter R for the ODM. It will not cause confusion since this is the only information on the data we are going to use. The ODM satisfies

$$0 \le R_{ij} \le 1, \quad R_{ij} = R_{ji} \quad \text{and} \quad \mathrm{R_{ii}} = 0.$$

The largest element of R is 1 because the algorithms scale the elements of R.

The ODM is displayed as *ordered dissimilarity image* (ODI), which is the visual output of VAT. In ODI the gray level of pixel (i, j) is proportional to the value of R_{ij}: pure black if $R_{ij} = 0$ and pure white if $R_{ij} = 1$.

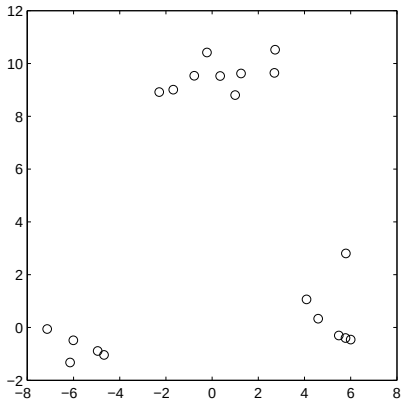

Figure 1: Scatterplot of the data set X.

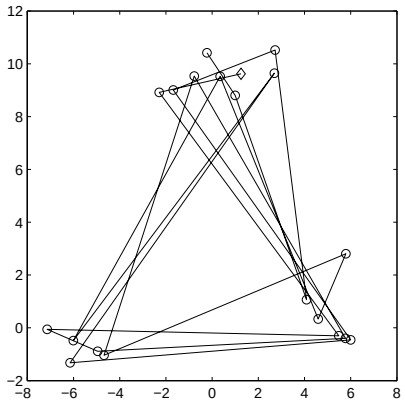

Figure 2: The original order of X.

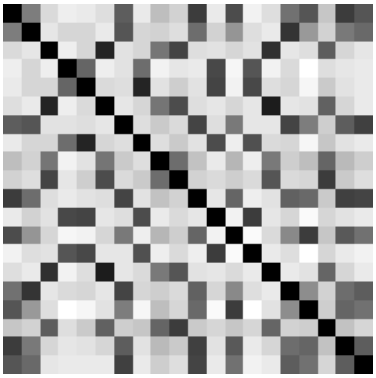

Figure 3: Dissimilarity image *before* reordering X

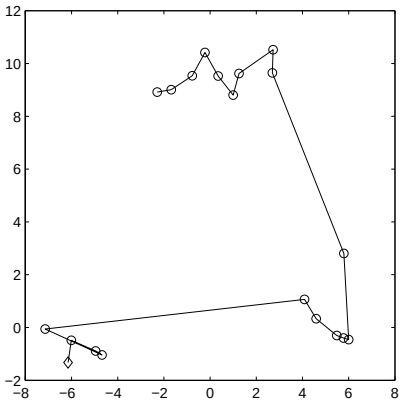

Figure 4: A data set X ordered by VAT

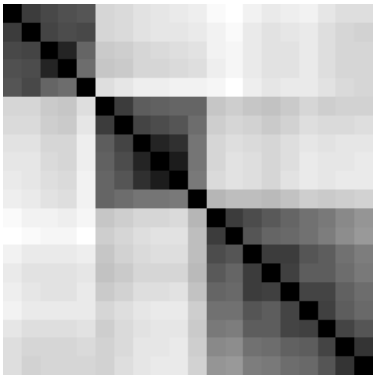

Figure 5: ODI using the order in Fig 4.

The idea of VAT is shown in Fig.1–5. Fig.1 shows a scatterplot of a data set $X = \{x_1, x_2, \ldots, x_{20}\} \subset \mathbb{R}^2$ of 20 points containing three well-defined clusters. Its original order, shown by the broken line in Fig.2 with $x_1 \approx (1.3, 9.6)$ marked by a diamond, is random, as in most applications. The corresponding dissimilarity image in Fig.3 contains no useful visual information about the cluster structure in X. Fig.4 shows the new order of the data set X, with the diamond in the lower left corner representing the new $x_1 \approx (-6.1, -1.3)$ in the ordered data set. Fig.5 gives the corresponding ODI. Now the three clusters are represented by the three well-formed black blocks.

The VAT algorithms are certainly useful, but there is room for improvements. It seems to us that our eyes are not very sensitive to structures in gray level images. One example is given in Fig.6. There are three clusters in the data as we will show later. The clusters are not well sep-

arated, and the ODI from VAT does not reveal any sign of the existence of the structure.

The approach of this paper is to focus on changes in dissimilarities in the ODM, the numeric output of VAT that underlies its visual output ODI. The results will be displayed as curves, which we call the *tendency curves*. The borders of clusters in the ODM (or blocks in the ODI) are shown as certain patterns in peaks and valleys on the tendency curves. The patterns can be caught not only by human eyes but also by the computer. It seems that the computer is more sensitive to these patterns on the curves than human eyes are to them or to the gray level patterns in the ODI. For example, the computer caught the three clusters in the data set that produced the virtually useless ODI in Fig.6.

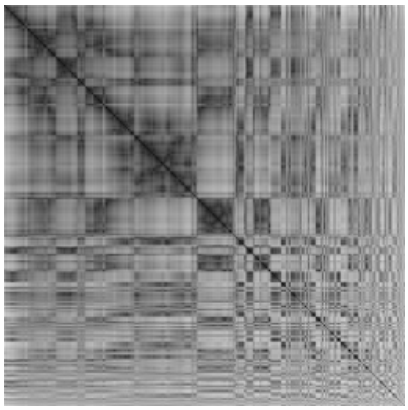

Figure 6: How many clusters are in this ODI?

2 Tendency Curves

Our approach is to catch possible diagonal blocks in the ordered dissimilarity matrix R by using various averages of distances, which are stored as vectors and displayed as curves. Let n be the number of points in the data, we define

$$m = 0.05n, \quad M = 5m, \quad w = 3m. \qquad (1)$$

We restrict ourselves to the w-subdiagonal band (excluding the diagonal) of R, as shown in Fig.7. Let $\ell_i = \max(1, i - w)$, then the i-th "*row-average*" is defined by

$$r_1 = 0, \qquad r_i = \frac{1}{i - \ell_i} \sum_{j=\ell_i}^{i-1} R_{ij}, \quad 2 \le i \le n. \quad (2)$$

In another word, each r_i is the average of the elements of row i in the w-band. The i-th m-row moving average is defined as the average of all elements in up to m rows above row i, inclusive, that fall in the w-band. This corresponds to the region between the two horizontal line segments in Fig.7. We also define the M-row moving average in almost the identical way except with m replaced by M. They will be referred to as the r-curve, the m-curve and the M-curve, respectively.

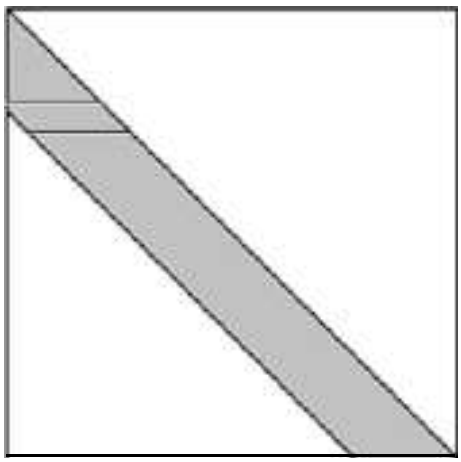

Figure 7: Sub-diagonal band of the ODM

The idea of the r-curve is simple. Just imagine a horizontal line, representing the current row in the program, moving downward in an ODI such as the one in Fig.5. When it moves out of one diagonal black block and into another, the r-curve should first show a peak because the numbers to the left of diagonal element R_{ii} will suddenly increase. It should drop back down rather quickly when the line moves well into the next black block. Therefore the border of two blocks should be represented as a peak on the r-curve if the clusters are well separated.

When the situation is less than ideal, there will be noise, which may destroy possible patterns on the r-curve. That is how the m-curve comes in, which often reveals the pattern beneath the noise. Since the VAT algorithms tend to order outliers near the end, so the m-curve tends to move up in the long run, which makes it hard for the program to identify peaks. That is why we introduce the M-curve, which shows long term trends of the r-curve. The difference of the m- and M-curves, which we call the d-curve, retains the shape of the m-curve but is more horizontal, basically lying on the horizontal axis. Furthermore, the M-curve changes more slowly than the m-curve, thus when moving from one block into another in the ODM,

it will tend to be lower than the m-curve. As a result, the d-curve will show a valley, most likely below the horizontal axis, after a peak. It is the peak-valley, or high-low, patterns that signal the existence of cluster structures. This will become clear in our examples in the section that follows.

3 Numerical Examples

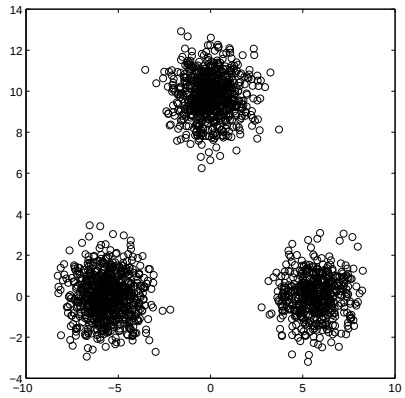

Figure 8: Scatterplot of three normally distributed clusters in $\mathbb{R}^2$, with $\alpha = 8$.

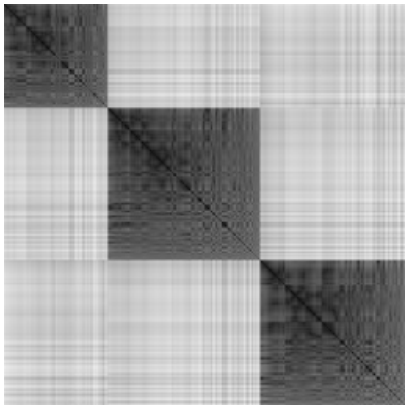

Figure 9: ODI from VAT for the data set with $\alpha = 8$

We give one group of examples in $\mathbb{R}^2$ so that we can use their scatterplots to show how well/poorly the clusters are separated. We also give the visual outputs (ODIs) of VAT for comparison. These sets are generated by choosing $\alpha = 8, 4, 3, 2, 1$ and 0 in the following set-

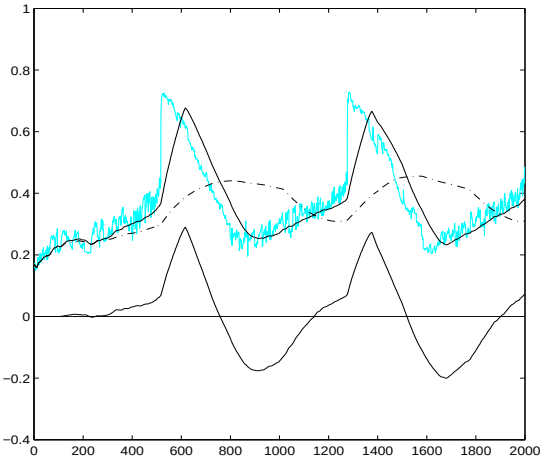

Figure 10: Tendency curves for the data set with $\alpha = 8$

tings: 2000 points (observations) are generated in three groups from multivariate normal distribution having mean vectors $\mu_1 = (0, \alpha\sqrt{6}/2)$, $\mu_2 = (-\alpha\sqrt{2}/2, 0)$ and $\mu_3 = (\alpha\sqrt{2}/2, 0)$. The probabilities for a point to fall into each of the three groups are 0.35, 0.4 and 0.25, respectively. The covariance matrices for all three groups are I_2. Note that μ_1, μ_2 and μ_3 form an equilateral triangle of side length $\alpha\sqrt{2}$.

Fig.8–10 for the case $\alpha = 8$ show what we should look for on the curves. The clusters are very well separated, the ODI has three black blocks on the diagonal with sharp borders. Our r-curve (the one with "noise") has two vertical rises and the m-curve (the solid curve going through the r-curve where it is relatively flat) has two peaks, corresponding to the two block borders in the ODI. The M-curve, the smoother, dash-dotted curve, is only interesting in its relative position with respect to the m-curve. That is, it is only useful in generating the d-curve, the difference of these two curves. The d-curve looks almost identical to the m-curve, also having two peaks and two valleys. The major difference is that it is in the lower part of the figure, around the horizontal axis.

Fig.11–13 show the case $\alpha = 4$. The clusters are less separated than the case $\alpha = 8$, and the slopes of the tendency curves are smaller. There are still two vertical rises on the r-curve, and two peaks followed by two valleys on all other curves where the block borders are in the ODI in Fig.12. What is really different here from the case $\alpha = 8$ is the wild oscillations near the end of the r-curve,

bringing up all other three curves. This corresponds to the small region in the lower-right corner of the ODI, where there lacks pattern. Note that no valley follows from the third rise or peak. This is understandable because a valley appears when the curve index (the horizontal variable of the graphs) runs into a cluster, shown as a block in ODI.

Now we know what we should look for: vertical rises or peaks followed by valleys, or high-low patterns, on all the tendency curves maybe except the M-curve.

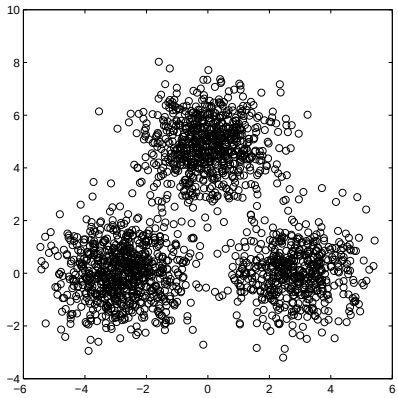

Figure 11: Three normally distributed clusters in $\mathbb{R}^2$ with $\alpha = 4$

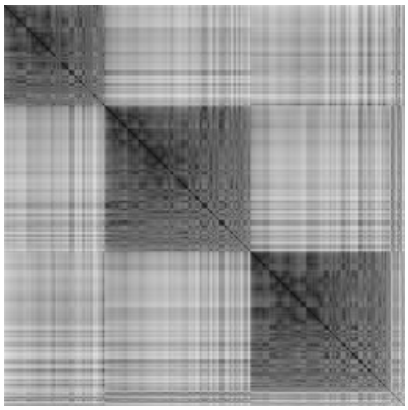

Figure 12: ODI from VAT, $\alpha = 4$

The case $\alpha = 3$ is given in Fig.14–16. One can still easily make out the three clusters in the scatterplot, but it is harder to tell to which cluster many points in the middle belong, (the memberships are very fuzzy). It is expected that every

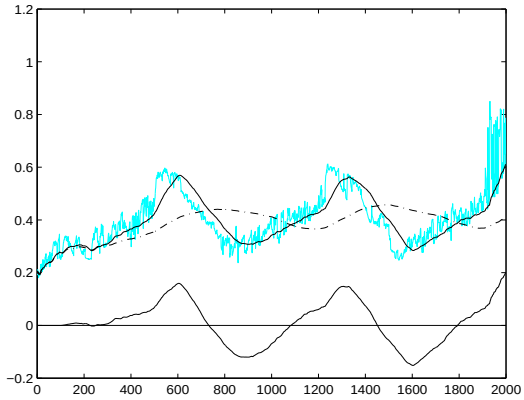

Figure 13: Tendency curves for $\alpha = 4$

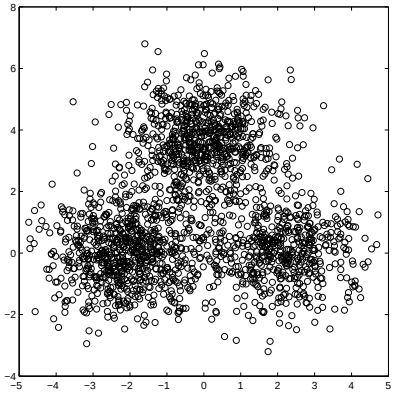

Figure 14: Three normally distributed clusters in $\mathbb{R}^2$ with $\alpha = 3$

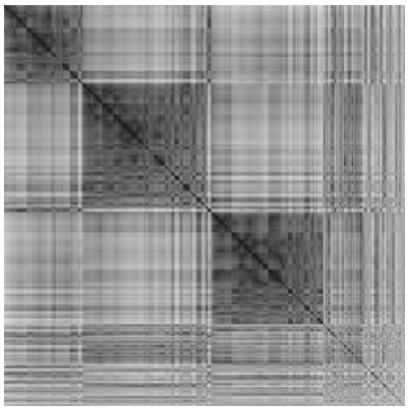

Figure 15: ODI from VAT, $\alpha = 3$

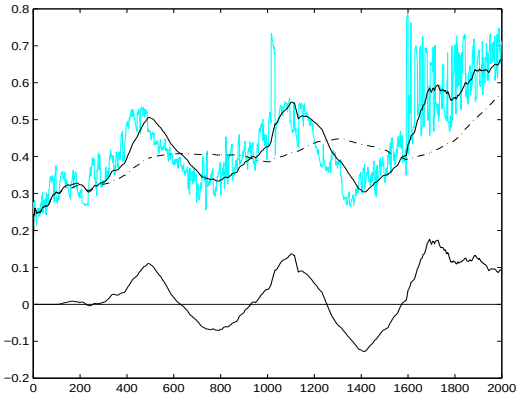

Figure 16: Tendency curves for $\alpha = 3$

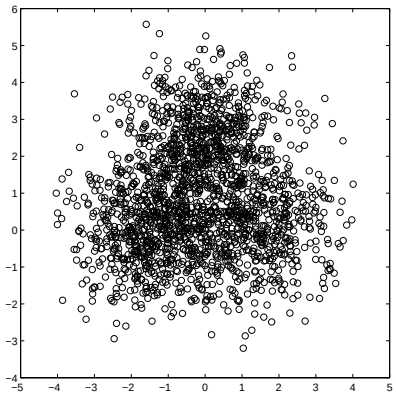

Figure 17: Three normally distributed clusters in $\mathbb{R}^2$ with $\alpha = 2$

visual method will have difficulties with them, as evidenced by the lower right corner of the ODI, and the oscillations on the last one fifth of the r-curve. The oscillations bring up the r- and m-curves, but not the d-curve. The d-curve remains almost the same as those in the two previous cases, except the third peak becomes larger and decreases moderately without forming a valley. The two high-low patterns on the m- and d-curves show the existence of three clusters. As we have said earlier that it is a valley on the m-curve and, especially, the d-curve that signals the beginning of a new cluster.

We hope by now the reader can see the purpose of the d-curve. Both the m- and M-curves in Fig.16 go up with wild oscillations, but the d-curve always stays low, lying near the horizontal axis. Unlike the other three curves, its values never get too high or too low. This enables us to detect the beginnings of new blocks in an ODM by catching the high-lows on the d-curve. When the d-curve hits a ceiling, set as 0.04, and then a floor, set as 0, the program reports one new cluster. The ceiling and floor values are satisfied by all cases in our numerical experiments where the clusters are reasonably, sometimes only barely, separated. If we lower the ceiling and raise the floor, we would be able to catch some of the less separated clusters we know we have missed, but it would also increase the chance of "catching" false clusters. We do not like the idea of tuning parameters to particular examples. We will stick to the same ceiling and floor values throughout this paper. In fact, *we do not recommend changing the suggested values of the parameters in our program*, that is, the values for the ceiling and floor set here, and those for m, M and w given in (1).

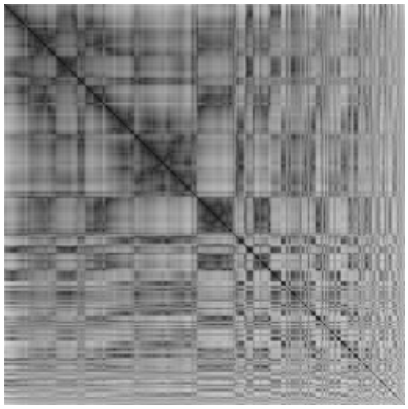

Figure 18: ODI from VAT, $\alpha = 2$

The situation in the case $\alpha = 2$, shown in Fig.17–19, really deteriorates. One can barely make out the three clusters in Fig.17 that are supposed to be there; the ODI in Fig.18 is a mess. In fact, this is the same ODI as the one in Fig.6, put here again for side-by-side comparison with the scatterplot and the tendency curves. The tendency curves in Fig.19, however, pick up cluster structure from the ODM. The d-curve has several high-lows, with two of them large enough to hit both the ceiling and floor, whose peaks are near 600 and 1000 marks on the horizontal axis, respectively. This example clearly shows that our tendency curves generated from the ODM are more sensitive than the raw block structure in the graphical display (ODI) of the same ODM. The largest advantage of the tendency curves is proba-

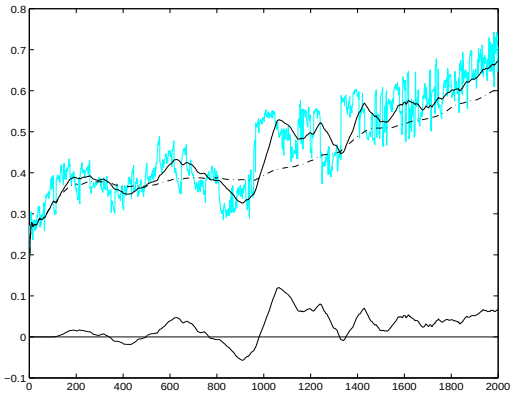

Figure 19: Tendency curves for $\alpha = 2$

bly the quantization which enables the computer, not only human eyes, to catch possible patterns.

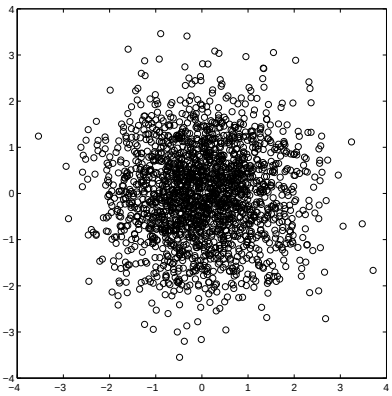

Figure 20: Three normally distributed clusters in $\mathbb{R}^2$ with $\alpha = 0$ (combining into one)

When α goes down to zero, the cluster structure disappears. The scatterplots for $\alpha = 0$ (Fig.20) and $\alpha = 1$ (not shown) are almost identical, showing a single cluster in the center. The tendency curves for both cases (Fig.21 and 22) have no high-lows large enough to hit the ceiling then the floor, which is the way they should be. Note that while all other three curves go up when moving to the right, the d-curve, the difference of the m- and M-curves, stays horizontal, which is, again, the reason we introduced it.

We also tested our program on many other data sets, including small data sets containing 150 points in $\mathbb{R}^4$. It worked equally well. We also tested two examples in Figures 12 and 13 of Bezdek and Hathaway [5], where the points are regularly arranged, on a rectangular grid, and

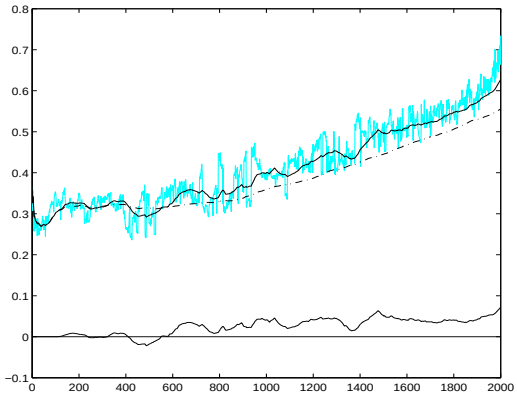

Figure 21: Tendency curves for $\alpha = 0$

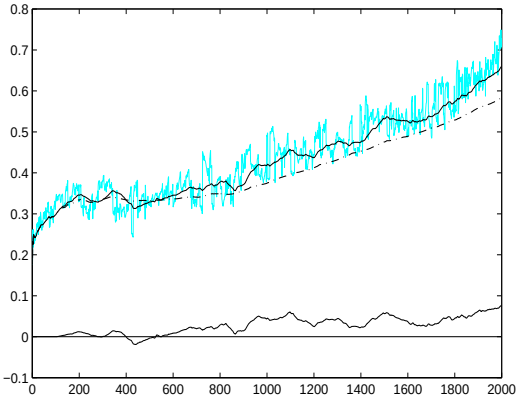

Figure 22: Tendency curves for $\alpha = 1$

along a pair of concentric circles, respectively. Because we could only have speculated from ODI images produced from the sets before applying our program, it was to our great, and pleasant, surprise that the program worked seamlessly on them, accurately reporting the number of clusters that exist. What we want to emphasize is that *we did all this without ever having to modify any suggested parameter values*! These tests will be reported in a forthcoming paper.

It is almost a sacred ritual that everybody tries the Iris data in a paper on clustering, so we conclude ours by trying our program on it. The ODI from VAT is given by Fig.23, and the tendency curves are given in Fig.24. The computer caught the large high-low on the left and ignored the small one on the right, and correctly reporting two clusters.

Remark 1 *There is more than one version of the Iris data, we are using the "real" version as dis-*

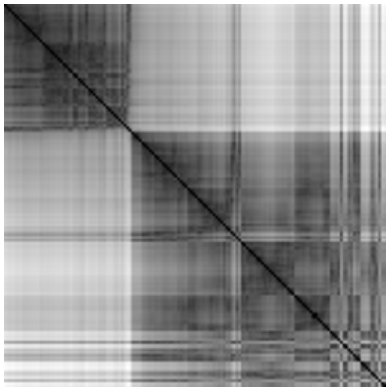

Figure 23: ODI for the Iris data

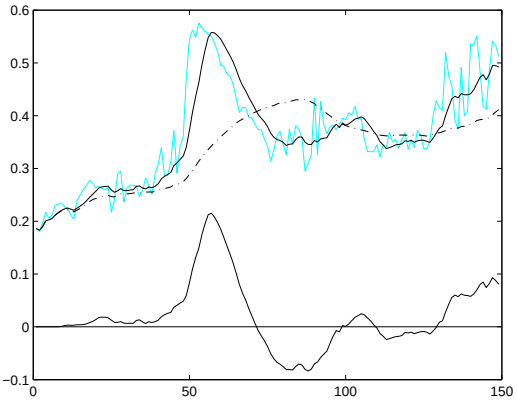

Figure 24: Tendency curves for the Iris data

cussed in [10]. The original object data consists of measurements of four features of each of 150 irises, which are of three, not two, different physical types. Fifty of each type were used to generate the data. Thus it seems there should be three clusters in the data. The reason both the ODI and the tendency curves show only two clusters is that two of the three iris types generate data that overlap in $\mathbb{R}^4$*, so many argue that the unlabeled data naturally form two clusters; see [5].*

4 Discussion

The original VAT algorithm introduced in 2002 provides a useful visual display of well-separated cluster structure. Two significant weaknesses of the original approach are: (1) there is no straightforward way to automate the procedure so that cluster assessment can be done without the aid of a human interpreter; and (2) the quality of the information in the ODI–to a human observer– deteriorates badly in some cases where there is still a significant separation between clusters. Our proposed use of "tendency curves" derived from the VAT reordering output successfully addresses both of these problems. Regarding (1), while the tendency curves provide much information for detailed human interpretation of cluster structure, they also concentrate and organize the information in a way that allows the use of a simple automated procedure for detecting the number of clusters, involving a simple counting of the number of high-low patterns. And very importantly, good choices for the "high" and "low" thresholds are suggested elsewhere in the paper; the parameters defining the automated procedure do not appear to depend sensitively on the particular problem being solved. Regarding weakness (2), the success of the procedure on data sets such as that shown in Fig. 17 demonstrates a great improvement over results obtained from the raw ODIs produced by VAT.

For clearer exposition in this note, we chose to restrict the main discussion to the original VAT procedure. Actually, the tendency curve modification is completely applicable, and easily transferable, to several other cases involving more recent relatives of VAT. For example, the sVAT procedure in Hathaway, Bezdek and Huband [7] is a two step procedure for assessing cluster tendency in very large scale data sets. First a representative sample of the original data set is chosen , and then original VAT is applied to the selected sample. Clearly, the tendency curves could be added as a third step of a cluster assessment procedure. Significantly, this means the approach described in this note is scalable and applicable to very large data sets.

Another opportunity to extend the tendency curve assessment approach is in the case of the visual cluster validity (VCV) scheme suggested in Hathaway and Bezdek [18]. Whereas VAT involves a pre-clustering activity that uses regular Euclidean distances between points, the VCV scheme is done post-clustering. The pair wise distances used by VCV are defined indirectly on the basis of how closely pairs of data points fit into the same cluster. In other words, two points that may be far apart in a Euclidean sense, but both fitting closely to a single cluster prototype, which could be an ellipsoid, plane, etc., would be assigned a small pair wise distance. An example of a linear cluster structure and its corresponding VCV ODI

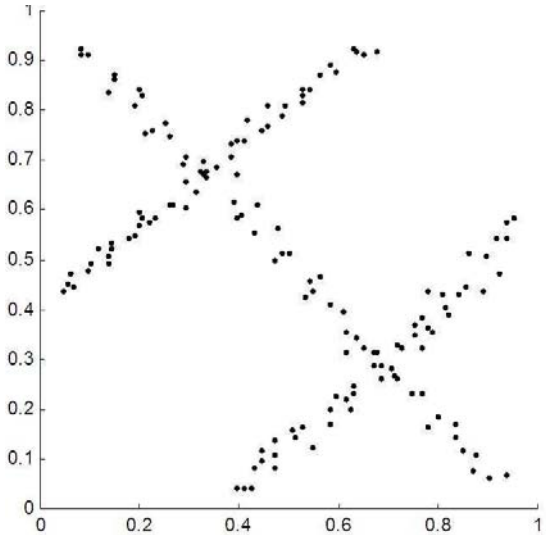

Figure 25: Scatterplot of three linear clusters

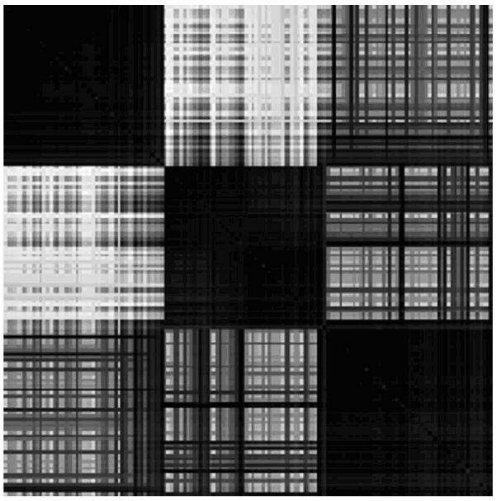

Figure 26: VCV ODI for Fig. 25 data

from [18] is given in Figs. 25-26. VCV can be used as one tool to visually validate how well the data points fit the assumed cluster structure. Even though cluster validity (done post-clustering) is a fundamentally different activity than cluster tendency assessment (done pre-clustering), the VCV approach again involves applying VAT to a matrix of pair wise distances, and is therefore completely amenable to the application of a tendency curve approach.

A more challenging extension of the use of tendency curves is in the case of non-square dissimilarity matrices considered in [6]. For rectangular dissimilarity matrices we can calculate row-based tendency curves, as done here, and also (different) column-based tendency curves. We are currently studying exactly how to make the best use of the two sets of curves to assess the underlying co-cluster tendency.

In closing we mention a final thought that may bear no fruit, but at least for now seems intriguing. An often used tactic in computational intelligence is to find something done efficiently in nature, and then attempt to mimic it computationally in a way that solves some problem of interest. The "high-low" rule for identifying a change in clusters reminds us of technical analysis schemes used in analyzing market or individual stock pricing charts. (The bible of such graph-based rules is [19].) Is there any "intelligence" in these rules that can be adapted to clustering or other types of data analysis. Moreover, how generally applicable is the fundamental approach taken in this paper, which we see as: (1) converting a large amount of multidimensional data into a univariate data stream; and (2) then extracting some important part of the information in the data stream using some graph-based rules? What is the most useful way to represent the stream information so that it can be understood by humans? Visually? Sonically? Might it be possible to "hear" clusters? This will be researched in the near future.

References:

[1] J.W. Tukey, Exploratory Data Analysis. Reading, MA: Addison-Wesley, 1977.

[2] W.S. Cleveland, Visualizing Data. Summit, NJ: Hobart Press, 1993.

[3] A.K. Jain and R.C. Dubes, Algorithms for Clustering Data. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[4] B.S. Everitt, Graphical Techniques for Multivariate Data. New York, NY: North Holland, 1978.

[5] J.C. Bezdek and R.J. Hathaway, *VAT: A tool for visual assessment of (cluster) tendency.* Proc. IJCNN 2002. IEEE Press, Piscataway, NJ, 2002, pp.2225-2230.

[6] J.C. Bezdek, R.J. Hathaway and J.M. Huband, *Visual Assessment of Clustering Tendency for Rectangular Dissimilarity Matrices*, IEEE Trans. on Fuzzy Systems, **15** (2007) 890-903

[7] R. J. Hathaway, J. C. Bezdek and J. M. Huband, *Scalable visual assessment of cluster tendency for large data sets*, Pattern Recognition, **39** (2006) 1315-1324.

[8] J. M. Huband, J.C. Bezdek and R.J. Hathaway, Revised visual assessment of (cluster) tendency (reVAT). Proc. North American Fuzzy Information Processing Society (NAFIPS), IEEE, Banff, Canada, 2004, pp.101-104.

[9] J. M. Huband, J.C. Bezdek and R.J. Hathaway, *bigVAT: Visual assessment of cluster tendency for large data set*. PATTERN RECOGNITION, **38** (2005) 1875-1886.

[10] J. C. Bezdek, J. M. Keller, R. Krishnapuram, L. I. Kuncheva and N. R. Pal, *Will the real Iris data please stand up?* IEEE Trans. Fuzzy Systems, **7**, 368–369 (1999).

[11] I. Borg and J. Lingoes, Multidimensional Similarity Structure Analysis. Springer-Verlag, New York, 1987.

[12] M. Kendall and J.D. Gibbons, Rank Correlation Methods. Oxford University Press, New York, 1990.

[13] Nawara Chansiri, Siriporn Supratid and Chom Kimpan, *Image Retrieval Improvement using Fuzzy C-Means Initialized by Fixed Threshold Clustering: a Case Study Relating to a Color Histogram*, WSEAS Transactions on Mathematics, **5.7** (2006), 926–931.

[14] Hung-Yueh Lin and Jun-Zone Chen, *Applying Neural Fuzzy Method to an Urban Development Criterion for Landfill Siting*, WSEAS Transactions on Mathematics, **5.9** (2006), 1053–1059.

[15] Nancy P. Lin, Hung-Jen Chen, Hao-En Chueh, Wei-Hua Hao and Chung-I Chang, *A Fuzzy Statistics based Method for Mining Fuzzy Correlation Rules*, WSEAS Transactions on Mathematics, **6.11** (2007), 852–858.

[16] Miin-Shen Yang, Karen Chia-Ren Lin, Hsiu-Chih Liu and Jiing-Feng Lirng, *A Fuzzy-Soft Competitive Learning Algorithm For Ophthalmological MRI Segmentation*, to appear in WSEAS Transactions on Mathematics.

[17] Gita Sastria, Choong Yeun Liong and Ishak Hashim, *Application of Fuzzy Subtractive Clustering for Enzymes Classification*, to appear in WSEAS Transactions on Mathematics.

[18] R.J. Hathaway and J.C. Bezdek, *Visual Cluster Assessment for Prototype Generator Clustering Models*, Pattern Recognition Letters, **24** (2003), 1563-1569.

[19] J.J. Murphy, Technical Analysis of the Financial Markets, New York Institute of Finance, New York, 1999.