Supervised and Unsupervised Information Granulation: A Study in Hyperbox Design

Andrzej Bargiela [1], Witold Pedrycz [2]
[1] *The Nottingham University, Nottingham NG1 4BU, UK*
(abb@cs.nott.ac.uk) - corresponding author

[2] *University of Alberta, Edmonton, Canada*
[2] *Systems Research Institute, Polish Academy of Sciences, Poland*
(pedrycz@ee.ualberta.ca)

Abstract: In this study, we are concerned with information granulation realized both in supervised and unsupervised mode. Our focus is on the exploitation of the technology of hyperboxes and fuzzy sets as a fundamental conceptual vehicle of information granulation. In case of supervised learning (classification), each class is described by one or more fuzzy hyperboxes defined by their corresponding minimum- and maximum vertices and the corresponding hyperbox membership function. Two types of hyperboxes are formed, namely inclusion hyperboxes that contain input patterns belonging to the same class, and exclusion hyperboxes that contain patterns belonging to two or more classes, thus representing contentious areas of the pattern space. With these two types of hyperboxes each class fuzzy set is represented as a union of inclusion hyperboxes of the same class minus a union of exclusion hyperboxes. The subtraction of sets provides for efficient representation of complex topologies of pattern classes without resorting to a large number of small hyperboxes to describe each class. The proposed fuzzy hyperbox classification is compared to the original Min-Max Neural Network and the General Fuzzy Min-Max Neural Network and the origins of the improved performance of the proposed classification are identified. When it comes to the unsupervised mode of learning, we revisit a well-known method of Fuzzy C-Means (FCM) by incorporating Tchebyschev distance using which we naturally form hyperbox-like prototypes. The design of hyperbox information granules is presented and the constructs formed in this manner are evaluated with respect to their abilities to capture the structure of data.

Keywords: pattern classification, fuzzy hyperbox, min-max neural networks, information granulation, fuzzy clustering, Tchebyschev distance

1. Introduction

Fuzzy hyperbox classification derives from the original idea of Zadeh [16] of using fuzzy sets for representation of real-life data. Such data frequently is not *crisp* (has a binary inclusion relationship) but rather has a property of a *degree of membership*. In this case the use of traditional set theory introduces unrealistic constraint of forcing binary decisions where the graded response is more appropriate. An early application of fuzzy sets to the pattern classification problem [3] proves the point that fuzzy sets represent an excellent tool simplifying the representation of complex boundaries between the pattern classes while retaining the full expressive power for the representation of the *core area* for each class. By having classes represented by fuzzy set membership functions it is possible to describe the degree to which a pattern belongs to one class or another.

Bearing in mind that the purpose of classification is the enhancement of interpretability of data or, in other words, derivation of a good abstraction of such data the use of hyperbox fuzzy sets as a description of pattern classes provides clear advantages. Each hyperbox can be interpreted as a fuzzy rule. However, the use of a single hyperbox fuzzy set for each pattern class is too limiting in that the topology of the original data is frequently quite complex [1] (and incompatible with the convex topology of the hyperbox). This limitation can be overcome by using a collection (union) of hyperboxes to cover each pattern class set [14], [15], [8]. Clearly, the smaller the hyperboxes the more accurate cover of the class set can be obtained. Unfortunately, this comes at the expense of increasing the number of hyperboxes thus eroding the original objective of interpretability of the classification result. We have therefore a task of balancing the requirements of accuracy of coverage of the original data (which translates on the minimization of misclassifications) with the interpretability of class sets composed of many hyperboxes. These concerns are not unique to hyperboxes and they demonstrate themselves also in the context of other topologies of information gramules, [2].

The tradeoff originally proposed by Simpson [14] was the optimization of a single parameter defining the maximum hyperbox size as a function of misclassification rate. However, the use of a single maximum hyperbox size is somewhat restrictive. For class sets that are well separated from each other the use of large hyperboxes is quite adequate while for the closely spaced class sets, with a complex partition boundary, there is a need for small hyperboxes, so as to avoid high misclassification rates. One solution to this problem, proposed in [8], is the adaptation of the size of hyperboxes so that it is possible to generate larger hyperboxes in some areas of the pattern space while in the other areas the hyperboxes are constrained to be small to maintain low misclassification rates. The adaptation procedure requires however several presentations of data to arrive at the optimum sizes of hyperbox sizes for the individual classes.

In this paper we consider an alternative approach to achieving low misclassification rate while maintaining good interpretability of the classification results. Rather than trying to express the class sets as a union of fuzzy hyperbox sets [8], [14], we represent them as a difference of two fuzzy sets. The first set is a union of hyperboxes produced in the standard way and the second set is a union of intersections of all hyperboxes that belong to different classes. We will refer to the first type of hyperboxes as inclusion hyperboxes and the second type as exclusion hyperboxes. By subtracting the exclusion hyperboxes from the inclusion ones it is possible to express complex topologies of the class set using fewer hyperboxes. Also, the three steps of the Min-Max clustering [8], [14], namely *expansion*, *overlap test* and *contraction* can be reduced to two, namely *expansion* and *overlap* tests. Expansion step results in generating inclusion hyperboxes and the overlap test results in exclusion hyperboxes.

Clustering has been widely recognized as one of the dominant techniques of data analysis. The broad spectrum of the detailed algorithms and underlying technologies (fuzzy sets, neural networks, heuristic approaches) is impressive. In spite of this diversity, the key objective remains the same which is to understand the data. In this sense, clustering becomes an integral part of data mining [6], [17]. Data mining is aimed at making the findings that are inherently *transparent* to the end user. The transparency is accomplished through suitable knowledge representation mechanisms, namely a way in which generic data elements are formed, processed and presented to the user. The notion of information granularity becomes a cornerstone concept that needs to be discussed in this context, cf [17] [11].

The underlying idea is that in any data set we can distinguish between a core part of a structure of the data that is easily describable and interpretable in a straightforward manner and a residual part, which does not carry any evident pattern of regularity. The core part can be described in a compact manner through several information granules while the residual part does not exhibit any visible geometry and requires some formal descriptors such as membership formulas. The scheme of unsupervised learning proposed in this study dwells on the augmentation of the standard FCM method which is now equipped with a Tchebyschev distance. This form of distance promotes hyperbox geometry of the information granules (hyperboxes). Starting from the results of clustering, our objective is to develop information granules forming a core structure in the data set, provide their characterization and discuss an interaction between the granules leading to their deformation.

The study is organized in the following fashion. Starting with the mode of supervised learning, we discuss the fuzzy min-max classification in Section 2. After discussing the inherent limitations of classes built as the union of hyperboxes, in Section 3, we proceed to introduce a novel approach to min-max classification by suggesting the coverage of complex class topologies by means of a set difference operator. This is discussed in Section 4 and is illustrated by some numerical examples in Section 5. We then look at the unsupervised learning mode in Section 6 where we emphasise the role of key parameters used in the information granulation process. A detailed clustering (unsupervised learning) algorithm together with a representative set of examples illustrating the topology of the resulting information granules is given in Section 7. The conclusions from our study are presented in Section 8.

2. Fuzzy Min-Max classification

The fuzzy Min-Max classification neural networks are built using hyperbox fuzzy sets. A hyperbox defines a region in $\mathbf{R}^n$, or more specifically in $[0\ 1]^n$ (since the data is normalized to [0 1]) and all patterns contained within the hyperbox have full class membership. A hyperbox **B** is fully defined by its minimum **V** and maximum **W** vertices. So that, $\mathbf{B}=[\mathbf{V}\ ,\ \mathbf{W}] \subset [0\ 1]^n$ with $\mathbf{V}, \mathbf{W} \in [0\ 1]^n$.

Fuzzy hyperbox ***B*** is described by a membership function (in addition to its minimum and maximum vertices), which maps the universe of discourse (**X**) into a unit interval

$$\boldsymbol{B} : \mathbf{X} \rightarrow [0,\, 1] \tag{1}$$

Formally, ***B***(x) denotes a degree of membership that describes an extent to which x belongs to ***B***. If ***B***(x) =1 then we say that x fully belongs to ***B***. If ***B***(x) is equal to zero, x is fully excluded from ***B***. The values of the membership function that are in-between 0 and 1 represent a partial membership of x to ***B***. The higher the membership grade, the stronger is the association of the given element to ***B***. In this paper we will use an alternative notation for the hyperbox membership function *b*(***X***, **V**, **W**) which gives an explicit indication of the min- and max- points of the hyperbox. The hyperbox fuzzy set will then be denoted as *B*={***X***, **V**, **W**, *b*(***X***, **V**, **W**)}. Note that ***X*** is an input pattern that in general represents a class-labelled hyperbox in $[0\ 1]^n$. To put it formally

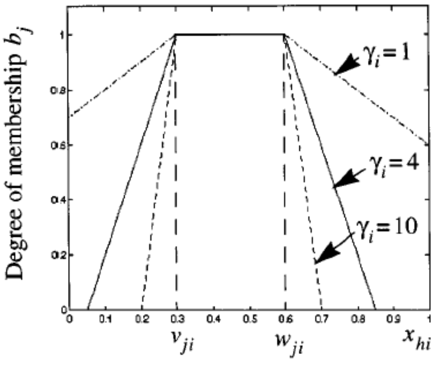
$$\boldsymbol{X}\mathbf{=\{}[\boldsymbol{X}^{l}\ \boldsymbol{X}^{u}],\, \mathrm{d}\} \tag{2}$$

where $\boldsymbol{X}^l$ and $\boldsymbol{X}^u$ represent min and max points of the input hyperbox ***X*** and $\mathrm{d} \in \{1,\ldots,\mathrm{p}\}$ is the index of the classes that are present in the data set.

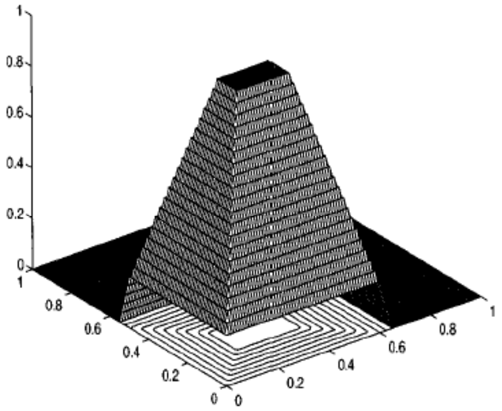
While it is possible to define various hyperbox membership functions that satisfy the boundary conditions with regard to full inclusion and full exclusion, it is quite intuitive to adopt a function that ensures monotonic (linear) change in-between these extremes. Following the suggestion in [8] we adopt here

$$b_j(X_h) = \min_{i=1,\ldots,n}(\min([1 - f(x_{hi}^u - w_{ji}, \gamma_i)],\ [1 - f(v_{ji} - x_{hi}^l, \gamma_i)])) \tag{3}$$

where $f(r,\gamma) = \begin{cases} 1 & if \quad r\gamma > 1 \\ r\gamma & if \quad 0 \le r\gamma \le 1 \\ 0 & if \quad r\gamma < 0 \end{cases}$ is a two parameter function in which *r* represents the distance of the test pattern X_h from the hyperbox [***V W***] and $\gamma = [\gamma_1, \gamma_2, \ldots, \gamma_n]$ represents the gradient of change of the fuzzy membership function. This is illustrated in Figure 1.



(a)



(b)

Figure 1. One-dimensional (a) and two-dimensional (b) fuzzy membership function evaluated for a point input pattern $\boldsymbol{X}_h$.

The fuzzy Min-Max algorithm is initiated with a single point hyperbox [$\boldsymbol{V}_j$ $\boldsymbol{W}_j$]=[$\boldsymbol{0}$ $\boldsymbol{0}$]. However, this hyperbox does not persist in the final solution. As the first input pattern $\boldsymbol{X}_h$=**{**[$\boldsymbol{X}_h^l$ $\boldsymbol{X}_h^u$], d} is presented the initial hyperbox becomes [$\boldsymbol{V}_j$ $\boldsymbol{W}_j$]= [$\boldsymbol{X}_h^l$ $\boldsymbol{X}_h^u$]. Presentation of subsequent input patterns has an effect of creating new hyperboxes or modifying the size of the existing ones. A special case occurs when a new pattern falls inside an existing hyperbox in which case no modification to the hyperbox is needed.

Hyperbox expansion: When the input pattern $\boldsymbol{X}_h$ is presented the fuzzy membership function for each hyperbox is evaluated. This creates a preference order for the inclusion of $\boldsymbol{X}_h$ in the existing hyperboxes. However the inclusion of the pattern is subject to two conditions: (a) the new pattern can only be included in the hyperbox if the class label of the pattern and the hyperbox are the same and (b) the size of the expanded hyperbox that includes the new pattern must not be greater in any dimension than the maximum permitted size. To put it formally the expansion procedure involves the following

$$if\ class(B_j) = \begin{cases} d_h \ \Rightarrow & test\ if\ B_j\ satisfies\ the\ \max imum\ size\ constra\text{int} \\ else\ \Rightarrow & take\ another\ B_j \end{cases} \tag{4}$$

with the size constraint in (4) defined as

$$\underset{i=1,\dots,n}{\forall} (\max(w_{ji}, x_{hi}^u) - \min(v_{ji}, x_{hi}^l)) \le \Theta \tag{5}$$

If expansion can be accomplished then the hyperbox min and max points are updated as

$$v_{ji} = \min(v_{ji}, x_{hi}^l), \quad for\ each\ i = 1,\dots,n$$
$$w_{ji} = \max(w_{ji}, x_{hi}^u), \quad for\ each\ i = 1,\dots,n$$

The parameter Θ can either be a scalar, as suggested in [14], or a vector defining different maximum hyperbox sizes in different dimensions [8]. It can be shown that the latter can result in fewer hyperboxes defining each pattern class but requires some a-priori knowledge about the topology of individual class sets or multiple presentations of data to facilitate adaptation.

Overlap test: The expansion of the hyperboxes can produce hyperbox overlap. The overlap of hyperboxes that have the same class labels does not present any problem but the overlap of hyperboxes with different class labels must be prevented since it would create ambiguous classification. The test adopted in [14] and [8] adopts the principle of minimal adjustment, where only the smallest overlap for one dimension is adjusted to resolve the overlap. This involves consideration of four cases for each dimension

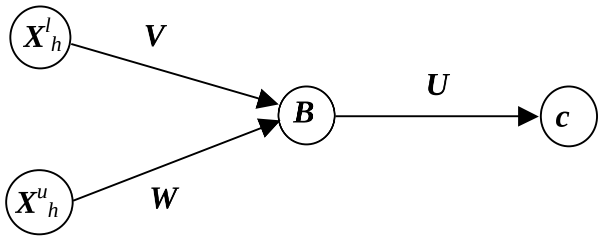
$$\begin{aligned}
&\text{Case 1: } v_{ji} < v_{ki} < w_{ji} < w_{ki}\\
&\text{Case 2: } v_{ki} < v_{ji} < w_{ki} < w_{ji}\\
&\text{Case 3: } v_{ji} < v_{ki} < w_{ki} < w_{ji}\\
&\text{Case 4: } v_{ki} < v_{ji} < w_{ji} < w_{ki}
\end{aligned}$$

The minimum value of overlap is remembered together with the index *i* of the dimension, which is stored as variable Δ. The procedure continues until no overlap is found for one of the dimensions (in which case there is no need for subsequent hyperbox contraction) or all dimensions have been tested.

Hyperbox contraction: The minimum overlap identified in the previous step provides basis for the implementation of the contraction procedure. Depending on which case has been identified the contraction is implemented as follows:

$$\begin{aligned}
&\text{Case 1: } v_{k\Delta}^{new} = w_{j\Delta}^{new} = \frac{v_{k\Delta}^{old} + w_{j\Delta}^{old}}{2} \quad \textit{or alternatively } (w_{j\Delta}^{new} = v_{k\Delta}^{old})\\
&\text{Case 2: } v_{j\Delta}^{new} = w_{k\Delta}^{new} = \frac{v_{j\Delta}^{old} + w_{k\Delta}^{old}}{2} \quad \textit{or alternatively } (v_{j\Delta}^{new} = w_{k\Delta}^{old})\\
&\text{Case 3: } \textit{if } \; w_{k\Delta} - v_{j\Delta} \le w_{j\Delta} - v_{k\Delta} \textit{ then } v_{j\Delta}^{new} = w_{k\Delta}^{old} \textit{ otherwise } w_{j\Delta}^{new} = v_{k\Delta}^{old}\\
&\text{Case 4: } \textit{if } \; w_{k\Delta} - v_{j\Delta} \le w_{j\Delta} - v_{k\Delta} \textit{ then } w_{k\Delta}^{new} = v_{j\Delta}^{old} \textit{ otherwise } v_{k\Delta}^{new} = w_{j\Delta}^{old}
\end{aligned}$$

The above three steps of the fuzzy Min-Max classification can be expressed as training of a three-layer neural network. The network, represented in Figure 2, has a simple feed-forward structure and grows adaptively according to the demands of the classification problem. The input layer has *2*n* processing elements, the first *n* elements deal with the min point of the input hyperbox and the second *n* elements deal with the max point of the input hyperbox $X_h = [X_h^l \; X_h^u]$. Each second-layer node represents a hyperbox fuzzy set where the connections of the first and second layers are the min-max points of the hyperbox including the given pattern and the transfer function is the hyperbox membership function. The connections are adjusted using the expansion, overlap test, contraction sequence described above. Note that the min points matrix **V** is modified only by the vector of lower bounds X_h^l of the input pattern and the max points matrix **W** is adjusted in response to the vector of upper bounds X_h^u .



$$\boldsymbol{X}^l_h = [x^l_{h1}, \ldots, x^l_{hn}], \boldsymbol{X}^u_h = [x^u_{h1}, \ldots, x^u_{hn}]$$
$$\boldsymbol{V}=[V_1,\ldots,V_m], \boldsymbol{W}=[W_1,\ldots,W_m]$$
$$\boldsymbol{B}=[B_1,\ldots,B_m]$$
$$\boldsymbol{U}=[u_{11},\ldots,u_{1p};\ldots; u_{m1},\ldots,u_{mp}]$$
$$\boldsymbol{c}=[c_1,\ldots,c_p]$$

Figure 2. The three-layer neural network implementation of the GFMM algorithm.

The connections between the second- and third-layer nodes are binary values. They are stored in matrix **U**. The elements of **U** are defined as follows:

$$u_{jk} = \begin{cases} 1 & if\ B_j\ is\ a\ hyperbox\ for\ class\ \ c_k \\ 0 & \qquad otherwise \end{cases} \tag{6}$$

where B_j is the jth second-layer node and c_k is the kth third-layer node. Each third-layer node represents a class. The output of the third-layer node represents the degree to which the input pattern X_h fits within the class k. The transfer function for each of the third-layer nodes is defined as

$$c_k = \max_{j=1}^{m} B_j u_{jk} \tag{7}$$

for each of the p third-layer nodes. The outputs of the class layer nodes can be fuzzy when calculated using expression (7), or crisp when a value of one is assigned to the node with the largest c_k and zero to the other nodes.

3. Inherent limitations of the fuzzy Min-Max classification

Training of the Min-Max neural network involves adaptive construction of hyperboxes guided by the class labels. The input patterns are presented in a sequential manner and are checked for a possible inclusion in the existing hyperboxes. If the pattern is fully included in one of the hyperboxes no adjustment of the min- and max-point of the hyperbox is necessary, otherwise a hyperbox *expansion* is initiated. However, after expansion is accomplished it is necessary to perform an *overlap test* since it is possible that the expansion resulted in some areas of the pattern space belonging simultaneously to two distinct classes, thus contradicting the classification itself. If the overlap test is negative, the expanded hyperbox does not require any further adjustment and the next input pattern is being considered. If, on the other hand, the overlap test is positive the hyperbox *contraction* procedure is initiated. This involves subdivision of the hyperboxes along one or several overlapping coordinates and the consequent adjustment of the min- and max-points of the overlapping hyperboxes. However, the contraction procedure has an inherent weakness in that it inadvertently eliminates from the two hyperboxes some part of the pattern space that was unambiguous while in the same time retaining some of the contentious part of the pattern space in each of the hyperboxes. This is illustrated in Figure 3.

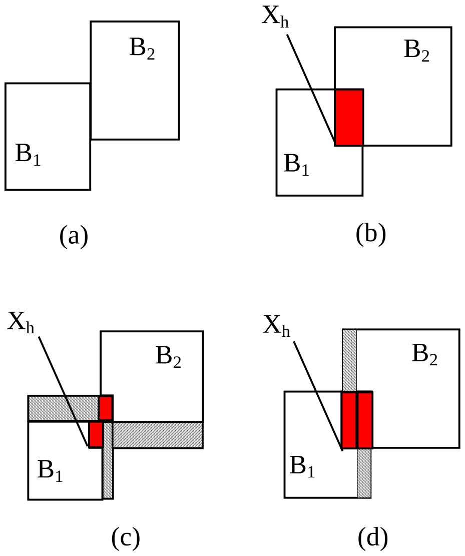


Figure 3. Training of the fuzzy Min-Max neural network.

(a) Hyperboxes belonging to two different classes *class*(B_1) ≠ *class*(B_2);
(b) Inclusion of pattern {X_h, *class*(B_2)} in B_2 implying overlap with B_1;
(c) Contraction of B_1 and B_2 with adjustment along two coordinates;
(d) Contraction of B_1 and B_2 with adjustment along one coordinate.

By inspecting Figure 3 it is clear that the contraction step of the fuzzy Min-Max network training resolves only part of the problem created by the expansion of the hyperbox B_2. Although the hyperboxes B_1 and B_2 no longer overlap after the contraction has been completed (Figure 3(c) and 3(d)), some part of the original hyperbox B_1 remains included in B_2 and similarly some part of the hyperbox B_2 remains included in the contracted B_1. The degree of this residual inclusion depends on the contraction method that is chosen but it is never completely eliminated. Incidentally, it is worth noting that the intuitive approach proposed in [14] of subdividing overlapping hyperboxes along a single coordinate with the smallest overlap does produce worse residual inclusion problem than the alternative subdivision along all overlapping coordinates (compare Figure 3(c) and 3(d)).

Another problem inherent to the contraction procedure is that it unnecessarily eliminates parts of the original hyperboxes. These eliminated portions are marked in Figure 3 with diagonal pattern lines. The elimination of these parts of hyperboxes implies that the contribution to the training of the Min-Max neural network of the data contained in these areas is nullified. If the neural network training involves only one pass through the data, then this is an irreversible loss that demonstrates itself in a degraded classification performance. The problem can be somewhat alleviated by allowing multiple presentations of data in the training process, as in [8], or reducing the maximum size of hyperboxes. In either case the result is that additional hyperboxes are created to cover the eliminated portions of the original hyperboxes. Unfortunately, the increased number of hyperboxes reduces the interpretability of classification so that there is a limit as to how far this problem can be resolved in the context of the standard Min-Max expansion/contraction procedure.

Finally, it is worth noting that the training pattern $\{X_h, class(B_2)\}$ continues to be misclassified in spite of the contraction of the hyperboxes. This means that a 100% correct classification rate is not always possible even with the multiple-pass Min-Max neural network training.

4. Exclusion/Inclusion Fuzzy Classification network (EFC)

The solution proposed here is the explicit representation of the contentious areas of the pattern space as *exclusion hyperboxes*. This is illustrated in Figure 4. The original hyperbox *b1* and the expanded hyperbox *b2* do not lose any of the undisputed area of the pattern space but the patterns contained in the exclusion hyperbox are eliminated from the relevant classes in the $\{c_1,\ldots,c_p\}$ set and are instead assigned to class c_{p+1} (contentious area of the pattern space class). This overruling implements in effect the subtraction of hyperbox sets which allows for the representation of non-convex topologies with a relatively few hyperboxes.

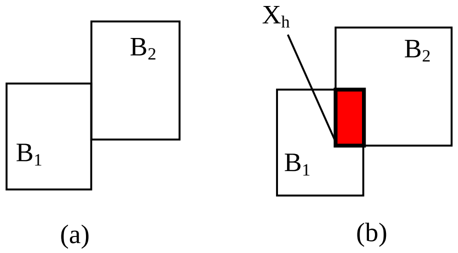
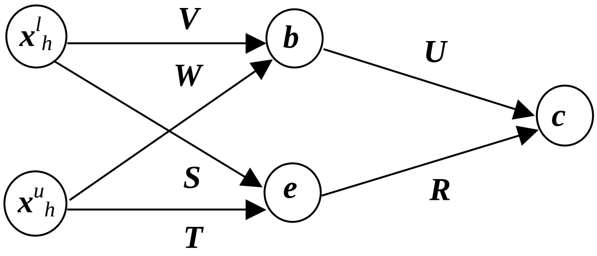


Figure 4. The concept of the exclusion/inclusion fuzzy hyperboxes.
(a) Hyperboxes belonging to two different classes $class(B_1) \neq class(B_2)$;
(b) Inclusion of pattern $\{X_h, class(B_2)\}$ in B_2 implying overlap with B_1 and consequent identification of the exclusion hyperbox.



$$\boldsymbol{x}^l_h = [x^l_{h1}, \ldots, x^l_{hn}],\ \boldsymbol{x}^u_h = [x^u_{h1}, \ldots, x^u_{hn}]$$
$$\boldsymbol{V}=[v_1,\ldots,v_m],\ \boldsymbol{W}=[w_1,\ldots,w_m],\ \boldsymbol{S}=[s_1,\ldots,s_m],\ \boldsymbol{T}=[t_1,\ldots,t_m]$$
$$\boldsymbol{b}=[b_1,\ldots,b_m],\ \boldsymbol{e}=[e_1,\ldots,e_q]$$
$$\boldsymbol{U}=[u_{10},\ldots,\ u_{1p},\ u_{1(p+1)};\ldots;\ u_{m0},\ldots,u_{mp},\ u_{m(p+1)}]$$
$$\boldsymbol{R}=[r_{10},\ldots,\ r_{1p},\ r_{1(p+1)};\ldots;\ r_{q0},\ldots,r_{qp},\ r_{q(p+1)}]$$
$$\boldsymbol{c}=[\ c_1,\ldots,c_p,c_{p+1}]$$

Figure 5. Exclusion/Inclusion Fuzzy Classification Network.

The additional second-layer nodes ***e*** are formed adaptively in a similar fashion as for nodes ***b***. The min-point and the max-point of the exclusion hyperbox are identified when the overlap test is positive for two hyperboxes representing different classes. These values are stored as new entries in matrix ***S*** and matrix ***T*** respectively. If the new exclusion hyperbox contains any of the previously identified exclusion hyperboxes, the included hyperboxes are eliminated from the set ***e***. The connections between the nodes ***e*** and nodes ***c*** are binary values stored in matrix ***R***. The elements of ***R*** are defined as follows:

$$r_{lk} = \begin{cases} 1 & if\ e_l\ overlapped\ hyperbox\ of\ class\ \ c_k\ and\ 1 < k < p \\ 1 & if \quad k = p+1 \\ 0 & otherwise \end{cases} \tag{8}$$

Note that the third layer has $p+1$ nodes [$c_1,\ldots,\ c_p,\ c_{p+1}$] with the node c_{p+1} representing the new exclusion hyperbox class. The output of the third-layer is now moderated by the output from the exclusion hyperbox nodes ***e*** and the values of matrix ***R.*** The transfer function for the third-layer nodes is defined as:

$$c_k = \max_{k=1}^{p+1}(\max_{j=1}^{m} b_j u_{jk} - \max_{i=1}^{q} e_i r_{ik}\ ) \tag{9}$$

The second component in (9) cancels out the contribution from the overlapping hyperboxes that belonged to different classes..

5. Numerical example

The EFC was applied to a number of synthetic data sets and demonstrated improvement over the GFMM and the original FMM [14]. As a representative example, we illustrate the performance of the network using the IRIS data-set from the Machine Learning Repository [18]. It is important to emphasise however that the use of a single specific data set does not detract from the essence of the topological argument that we are making in this paper; that is, that the pattern classes are covered more efficiently by the difference of fuzzy sets compared to the usual covering with the union of fuzzy sets.

Using the IRIS data set we have trained the network on the first 75 patterns and the EFC performance was checked using the remaining 75 patterns. The results for FMM have been obtained using our implementation of the FMM algorithm, which produced results consistent with those reported in [14]. The results are summarized in Table 1.

Table 1. Comparison of performance of FMM, GFMM and EFC

Performance criterion	**FMM [14]**	**GFMM [8]**	**EFC**
Correct classification rate (range)	97.33-92%	100-92%	100-97%
Number of hyperboxes (max. size 0.03)	56	49	34
Number of hyperboxes (max. size 0.06)	32	29	18
Number of hyperboxes (max. size 0.20)	16	12	7
Number of hyperboxes (max. size 0.40)	16	12	4*

* the smallest number of classes; the number is not affected by the increase of the maximum size of hyperbox Θ

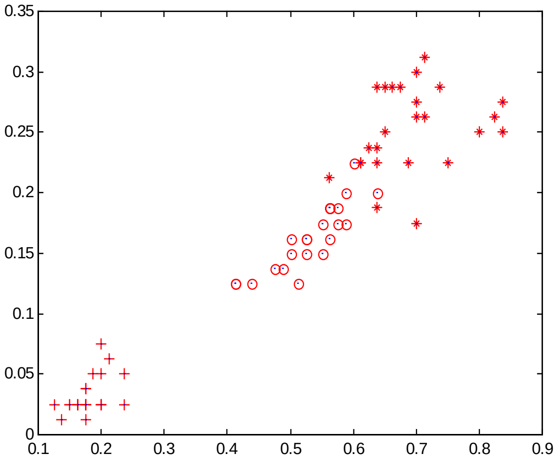


Figure 6. IRIS data projected onto petal-length/petal-width two-dimensional space.

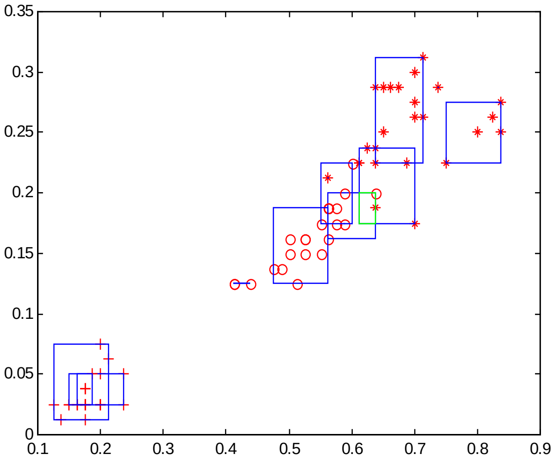


Figure 7. Exclusion/inclusion hyperboxes evaluated for θ=0.10

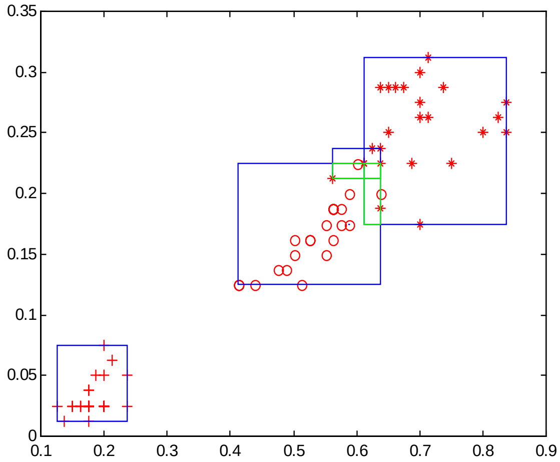


Figure 8. Exclusion/inclusion hyperboxes evaluated for θ=0.25

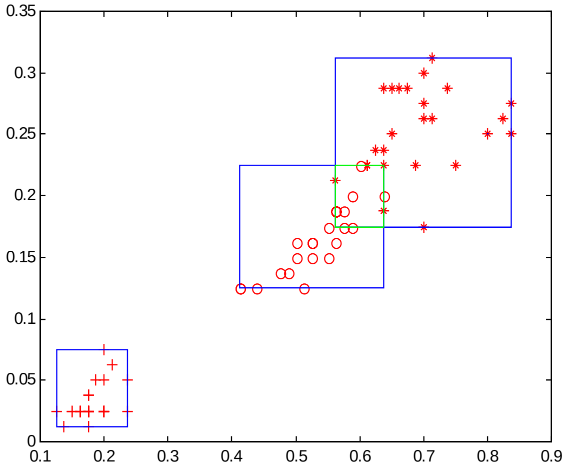


Figure 9. Exclusion/inclusion hyperboxes evaluated for θ=0.35

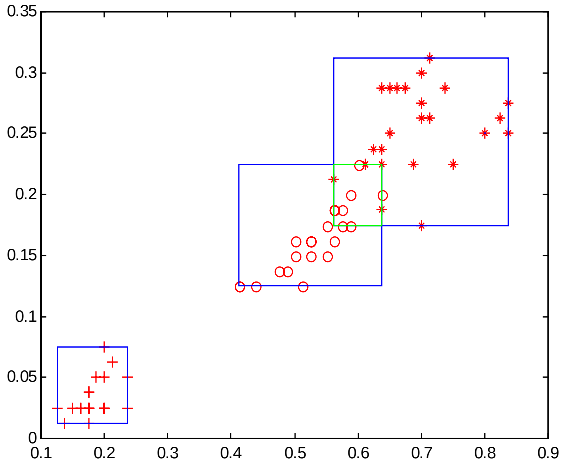


Figure 10. Exclusion/inclusion hyperboxes evaluated for θ=0.45

The results reported in Table 1 deserve some additional commentary. First we need to point out that the EFC algorithm would normally start without any constraint on the hyperbox size, i.e. Θ=1. This is in contrast to the two other algorithms that do require precise control of the maximum hyperbox size. So, in the interest of comparability of the results we have run the EFC algorithm with Θ equal to 0.03, 0.06, 0.2 and 0.4.

The detailed results obtained with EFC for other values of the parameter Θ (0.1, 0.25, 0.35 and 0.45) are illustrated in Figures 6-10. Figure 6 shows the projection of the IRIS data onto a two-dimensional space of petal-length/petal-width. Subsequent figures show the effect of the gradual increase of the value of the maximum hyperbox size parameter Θ. Although it is clear that for Θ=0.10 (Figure 7) the covering of the data with hyperboxes is more accurate than for Θ=0.45 (Figure 10), we argue that this is achieved at a too great expense of reduced interpretability of the classification. The large number of rules, implied by the individual hyperboxes, is clearly counterproductive. From the viewpoint of the interpretability of classification the result illustrated in Figure 10, is much preferred. Also, by comparing Figures 9 and 10, we note that for large Θ the result of classification is no longer dependent on the value of the parameter but is exclusively defined by the data itself. This in itself is a very desirable feature of the proposed algorithm.

Another point worth emphasizing is that the number of classes identified by the EFC is p+1 where p is the number of classes identified by the FMM and GFMM. This implies that the calculation of the “classification rate” is not identical in all three cases. We have taken the view that the exclusion hyperbox(es) offer a positive identification of the patterns that are ambiguous. In this sense the fact of having some test data fall into the exclusion hyperbox is not considered a misclassification. Clearly, this has an effect of improving the classification rate of the EFC with respect to the other two methods. However, to do otherwise and to report all data falling into the exclusion hyperboxes as misclassified would be also misleading since we already have a knowledge about the nature of the exclusion hyperbox and it would effectively make no use of the “p+1st” pattern class.

Of course we do need to balance the assessment of the EFC algorithm by highlighting the importance of the ratio of the volumes of the exclusion and inclusion hyperbox sets. If this ratio is small (e.g. 1/35 in the case of the IRIS dataset) the classification results are very good. However, if the ratio increases significantly, the classification is likely to return a large proportion of patterns as belonging to the “exclusion” class. This in itself offers a constructive advice on the reduction of the maximum size of hyperboxes.

6. From fuzzy clustering to hyperbox information granules

Let us briefly recall the basic notions and terminology of unsupervised learning. As before the set of data (patterns) is denoted by X, where X= {$\mathbf{x}_1$, $\mathbf{x}_2$, …, $\mathbf{x}_N$} while each pattern is an element in the n-dimensional unit hypercube, that is $[0,1]^n$. The objective is to cluster X into “c” clusters and the problem is cast as an optimization task (objective function based optimization)

$$Q = \sum_{i=1}^{c} \sum_{k=1}^{N} u_{ik}^{2} d_{ik} \tag{10}$$

where U = [u_{ik}], $u_{ik} \geq 0$, i=1,2, ..,c, k=1, 2, …,N is a partition matrix describing clusters in data. The distance function (metric) between the k-th pattern and i-th prototype is denoted by d_{ik}. d_{ik} = dist($\mathbf{x}_k$, $\mathbf{v}_i$) while $\mathbf{v}_1$, $\mathbf{v}_2$, …, $\mathbf{v}_c$ are the prototypes characterizing the clusters. The type of the distance implies certain geometry of the clusters one is interested in exploiting when analyzing the data. For instance, it is well known that a commonly used Euclidean distance promotes an ellipsoidal shape of the clusters.

Emphasizing the role of the parameters to be optimized, the above objective function reads now as

$$\text{Min Q with respect to } \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_c \text{ and U} \tag{11}$$

Its minimization carried out for the partition matrix as well as the prototypes. With regard to the prototypes (centroids), we end up with a constraint-free optimization while the other one calls for the constrained optimization. The constraints assure that U is a partition matrix meaning that the following well-known conditions are met

$$\sum_{i=1}^{c} u_{ik} = 1 \quad \text{for all } k = 1,2,.., N \tag{12}$$

$$0 < \sum_{k=1}^{N} u_{ik} < N \quad \text{for all } i = 1,2,.., c \tag{13}$$

The choice of the distance function is critical to our primary objective of achieving the transparency of the findings. We are interested in such distances whose equidistant contours are “boxes” with the sides parallel to the coordinates. The Tchebyschev distance (l_∞ distance) is a distance satisfying this property. The boxes are decomposable that is the region within a given equidistant contour of the distance can be treated as a decomposable relation R in the feature space, viz.

$$R = A \times B \tag{14}$$

where A and B are sets (or more generally information granules) in the corresponding feature spaces. It is worth noting that the Euclidean distance does not lead to the decomposable relations in the above sense (as the equidistant regions in such construct are spheres or ellipsoides). The illustration of the decomposability property is illustrated in Figure 11.

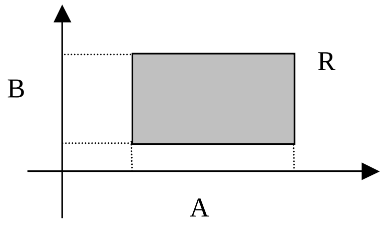


Figure 11. Decomposability property provided by the Tchebyschev distance; the region of equidistant points is represented as a Cartesian product of two sets in the corresponding feature spaces

The above clustering problem known in the literature as an l_∞ FCM was introduced and discussed by Bobrowski and Bezdek [5] more than 15 years ago. Some recent generalizations can be found in [10]. This motivation behind the introduction of this type of distance was the one about handling data structures with “sharp” boundaries (clearly the Tchebyschev distance is more suitable with this regard than the Euclidean distance). The solution proposed in [5] was obtained by applying a basis exchange algorithm.

In this study, as already highlighted, the motivation behind the use of the Tchebyschev distance is different. We are after the description of data structure and the related interpretability of the results of clustering so that the clusters can be viewed as basic models of associations existing in the data. Here, we derive a gradient-based FCM technique enhanced with some additional convergence mechanism.

7. The clustering algorithm- detailed considerations

The FCM optimization procedure is standard to a high extent [4] and consists of two steps: a determination of the partition matrix and calculations of the prototypes. The use of the Lagrange multipliers converts the constrained problem into its constraint-free version. The original objective function (10) is transformed to the form

$$\mathrm{V} = \sum_{i=1}^{c} \sum_{k=1}^{N} u_{ik}^2 d_{ik} + \sum_{k=1}^{N} \lambda_k \left(\sum_{i=1}^{c} u_{ik} - 1\right) \tag{15}$$

with $\lambda=[\lambda_1, \lambda_2, ...,\lambda_N]$ being a vector of Lagrange multipliers. The problem is then solved with respect to each pattern separately, that is, we consider the relationship below for each data point (t=1, 2, …,N)

$$\frac{\partial \mathrm{V}}{\partial \mathrm{u}_{\mathrm{st}}} = 0 \qquad \text{and} \qquad \frac{\partial V}{\partial \lambda_t} = 0 \tag{16}$$

s=1,2, …, c, t=1,2, …, N. Straightforward calculations lead to the expression

$$\mathrm{u}_{\mathrm{st}} = \frac{1}{\sum_{j=1}^{c} \frac{\mathrm{d}_{\mathrm{st}}}{\mathrm{d}_{\mathrm{jt}}}} \tag{17}$$

The determination of the prototypes is more complicated as the Tchebyschev distance does not lead to a closed-type expression (unlike the standard FCM with the Euclidean distance). Let us start with the objective in which the distance function is spelled out in an explicit manner

$$\mathrm{Q} = \sum_{\mathrm{i}=1}^{\mathrm{c}} \sum_{\mathrm{k}=1}^{\mathrm{N}} \mathrm{u}_{\mathrm{ik}}^2 \max_{\mathrm{j}=1,2,\ldots,\mathrm{n}} | \mathrm{x}_{\mathrm{kj}} - \mathrm{v}_{\mathrm{ij}} | \tag{18}$$

The minimization of Q carried out with respect to the prototype (more specifically its t-th coordinate) follows a gradient-based scheme

$$\mathrm{v}_{\mathrm{st}}(\mathrm{iter}+1) = \mathrm{v}_{\mathrm{st}}(\mathrm{iter}) - \alpha \frac{\partial \mathrm{Q}}{\partial \mathrm{v}_{\mathrm{st}}} \tag{19}$$

where α is an adjustment rate (learning rate) assuming positive values. This update expression is iterative; we start from some initial values of the prototypes and keep modifying them following the gradient of the objective function. The detailed calculations of the gradient lead to the expression

$$\frac{\partial Q}{\partial v_{st}} = \sum_{k=1}^{N} u_{sk}^{2} \frac{\partial}{\partial v_{st}} \{\max_{j=1,2,\ldots,n} | x_{kj} - v_{sj} |\} \tag{20}$$

Let us introduce the following shorthand notation

$$A_{kst} = \max_{\substack{j=1,2,\ldots,n \\ j\neq t}} | x_{kj} - v_{sj} | \tag{21}$$

Evidently, A_{kst} does not depend on v_{st}. This allows us to concentrate on the term that affects the gradient. We rewrite the above expression for the gradient as follows

$$\frac{\partial Q}{\partial v_{st}} = \sum_{k=1}^{N} u_{sk}^{2} \frac{\partial}{\partial v_{st}} \{\max(A_{kst}, | x_{kt} - v_{st} |)\} \tag{22}$$

The derivative is nonzero if A_{kst} is less or equal to the second term standing in the expression,

$$A_{kst} \leq | x_{kt} - v_{st} | \tag{23}$$

Next, if this condition holds we infer that the derivative is equal to either 1 or –1 depending on the relationship between x_{kt} and v_{st}, that is –1 if $x_{kt} > v_{st}$ and 1 otherwise. Putting these conditions together, we get

$$\frac{\partial Q}{\partial v_{st}} = \sum_{k=1}^{N} u_{sk}^{2} \begin{cases} -1 \text{ if } A_{kst} \leq | x_{kt} - v_{st} | \text{and } x_{kt} > v_{st} \\ +1 \text{ if } A_{kst} \leq | x_{kt} - v_{st} | \text{and } x_{kt} \leq v_{st} \\ \qquad 0 \text{ otherwise} \end{cases} \tag{24}$$

The primary concern that arises about this learning scheme is not the one about a piecewise character of the modulus (absolute value) function (a concern that can be raised from the formal standpoint but that is easily remedied by the appropriate selection of α in (19)) but a fact that the derivative zeroes for a significant number of situations. This may result in a poor performance of the optimization method as it could be trapped when the overall gradient becomes equal to zero. To enhance the method, we relax the binary character of the predicates (less or greater than) standing in (24). These predicates are Boolean (two-valued) as they return values equal to 0 or 1 (which translates into an expression “predicate is satisfied or it does not hold). The modification comes in the form of a degree of satisfaction of this predicate, meaning that we compute a multivalued predicate

$$\text{degree (a is included in b)} \tag{25}$$

that returns 1 if a is less or equal to b. Lower values of the degree arise when this predicate is not fully satisfied. This form of augmentation of the basic concept was

introduced in [7, 9, 12, 13] in conjunction to studies in fuzzy neural networks and relational structures (fuzzy relational equations).
The degree of satisfaction of the inclusion relation is equal to

$$\text{Degree}(\text{a is included in b}) = a \rightarrow b \tag{26}$$

where a and b are in the unit interval. The implication operation → is a residuation operation, cf. [12, 13]. Here we consider a certain implementation of such operation where the implication is implied by the product t-norm, namely

$$a \rightarrow b = \begin{cases} 1 \text{ if } a \le b \\ b/a \text{ otherwise} \end{cases} \tag{27}$$

Using this construct, we rewrite (24) as follows

$$\frac{\partial Q}{\partial v_{st}} = \sum_{k=1}^{N} u_{sk}^{2} \begin{cases} -(A_{kst} \rightarrow | x_{kt} - v_{st} |) & \text{if } x_{kt} > v_{st} \\ (A_{kst} \rightarrow | x_{kt} - v_{st} |) & \text{if } x_{kt} \le v_{st} \end{cases} \tag{28}$$

In the overall scheme, this expression will be used to update the prototypes of the clusters (28).

Summarizing, the clustering algorithm arises as a sequence of the following steps

```
repeat
        - compute partition matrix using (17);
        - compute prototypes using the partition matrix obtained in the first
          phase. (It should be noted that the partition matrix does not change at this
          stage and all updates of the prototypes work with this matrix. This phase
          is more time consuming in comparison with the FCM method equipped
          with the Euclidean distance)
until a termination criterion satisfied
```

Both the termination criterion and the initialization of the method are standard. The termination takes into account changes in the partition matrices at two successive iterations that should not exceed a certain threshold level. The initialization of the partition matrix is random.

As an illustrative example, we consider a synthetic data involving 4 clusters, see Figure 12. The two larger data groupings consist of 100 data-points and the two smaller ones have 20 and 10 data-points respectively.

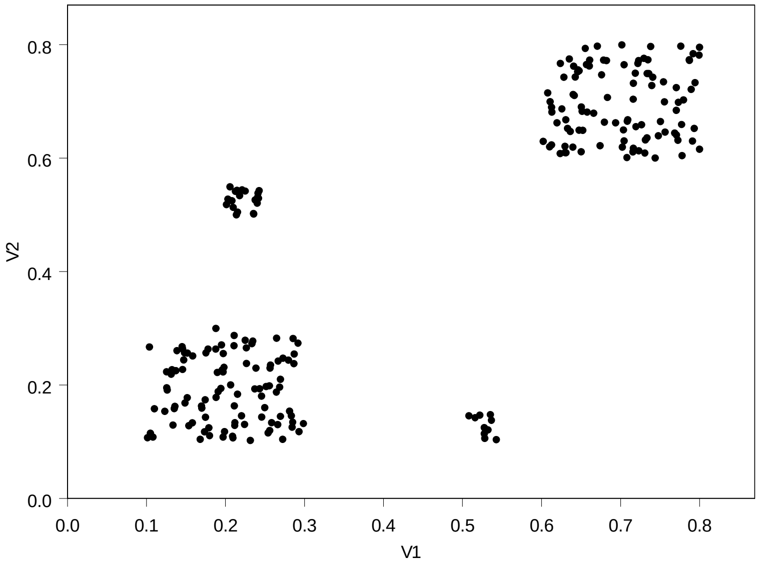


Figure 12. Two-dimensional synthetic data with four visible clusters of unequal size

Table 2. Prototypes identified by two FCM algorithms, with Euclidean and Tchebyschev distance measure respectively, for the varying number of clusters (the underlined prototypes correspond to the smaller data groupings)

Number of clusters	Prototypes for FCM with Euclidean distance	Prototypes for FCM with Tchebyschev distance
2	0.6707 0.6706 0.2240 0.2236	0.2088 0.1998 0.6924 0.6831

3	0.2700 0.3011
	0.6875 0.6841
	0.2302 0.2127
		0.7000 0.6847
		0.2440 0.4914
		0.2124 0.1852

4	0.2255 0.2035
	0.2323 0.2479
	0.6872 0.6814
	0.6533 0.6588
		0.7261 0.7377
		0.2278 0.5178
		0.2092 0.1846
		0.6523 0.6498

5	0.2525 0.2784
	0.2282 0.2014
	0.6721 0.6757
	0.2343 0.2389
	0.6919 0.6841
		0.2189 0.1451
		0.2272 0.5188
		0.1960 0.2258
		0.6568 0.6868
		0.7268 0.6593

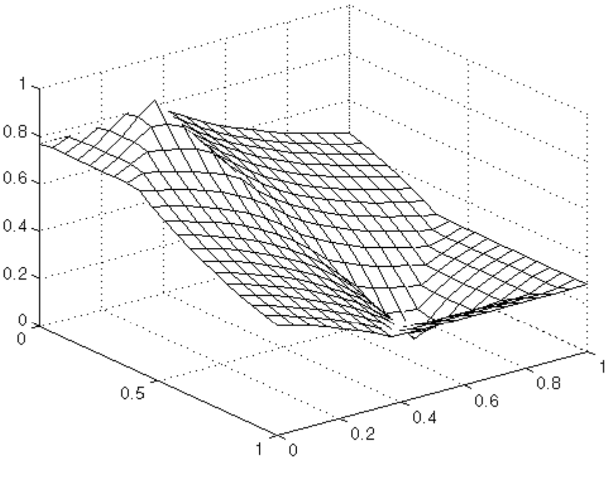
6	0.2329 0.2562
	0.6809 0.6777
	0.6857 0.6830
	0.2272 0.2206
	0.2261 0.2008
	0.6447 0.6500
		0.7469 0.6650
		0.2151 0.1364
		0.2278 0.5208
		0.6570 0.6840
		0.2619 0.2648
		0.1945 0.2239

7	0.6646 0.6697
	0.7036 0.6619
	0.6993 0.7100
	0.2395 0.5019
		0.1967 0.2255
		0.2200 0.1450
		0.7278 0.6594
		0.2277 0.5183

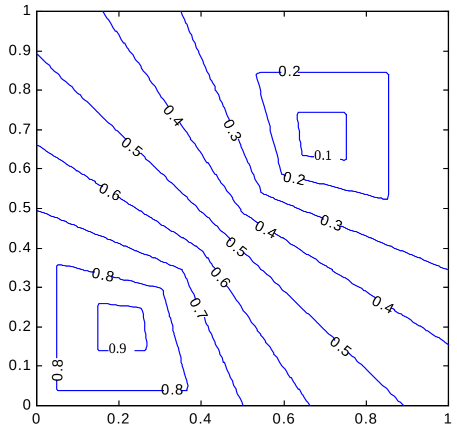
	0.2382 0.1935	0.3976 0.4051
	0.2164 0.1955	0.6099 0.6117
	0.2271 0.2018	0.6588 0.6923
8	0.6962 0.6892	0.6607 0.7615
	0.2398 0.5088	0.2122 0.1327
	0.2360 0.1980	0.3209 0.3097
	0.2441 0.2203	0.6565 0.6830
	0.6962 0.6882	0.7267 0.6590
	0.6850 0.6756	0.6460 0.6492
	0.2385 0.1942	0.2277 0.5191
	0.2166 0.1965	0.2108 0.2249

Table 2 gives a representative set of clustering results for 2 to 8 clusters. As expected, the two larger data groupings exercise dominant influence on the outcome of the FCM algorithms. Both Euclidean and Tchebyschev distance based FCM exhibit robust performance in that they find approximately the same clusters in their successive runs (within the limits of the optimization convergence criterion). While most of the identified prototypes fall within the large data groupings, the Tchebyschev distance based FCM consistently manages to associate a prototype with one of the smaller data grouping (underlined in the table). This is clearly a very advantageous feature of our modified FCM algorithm and confirms our assertion that the objective of enhancing the interpretability of data through the identification of decomposable relations is enhanced with Tchebyschev distance based FCM.

The above results are better understood if we examine the cluster membership function over the entire pattern space. The visualization of the membership function for one of the two clusters, positioned in the vicinity of (0.2, 0.2), (c=2) is given in Figure 13.



(a)

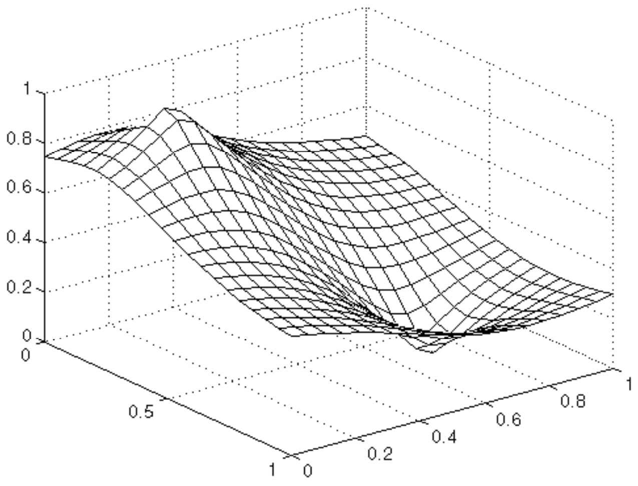


(b)

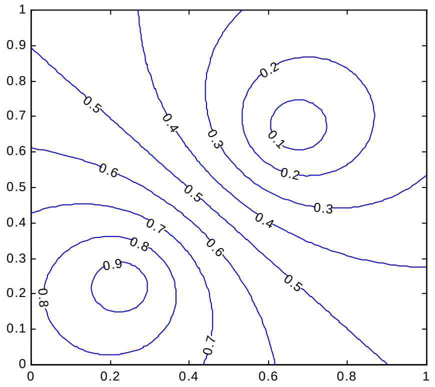
Figure 13. Visualization of the first cluster (membership function) centered around (0.2088 0.1998) : (a) 3D space and (b) contour plots.

It is easily noticed that for higher values of the membership grades (e.g. 0.9), the shape of contours is rectangular. This changes for lower values of the membership grades when we witness a gradual departure from this geometry of the clusters. This is an effect of interaction between the clusters that manifests itself in a deformation of the original rectangles. The deformation depends on the distribution of the clusters, their number and a specific threshold β being selected. The lower is the value of this threshold, the more profound departure from the rectangular shape. For higher values of β such deformation is quite limited. This suggests that when using high values of the threshold level the rectangular (or hyperbox) form of the core part of the clusters is fully legitimate.

Let us contrast these results with the geometry of the clusters constructed when using a Euclidean distance. Again, we consider two prototypes, as identified by the Euclidean distance based FCM, see Figure 14. The results are significantly different: the clusters are close to the Gaussian-like form and do not approximate well by rectangular shapes.



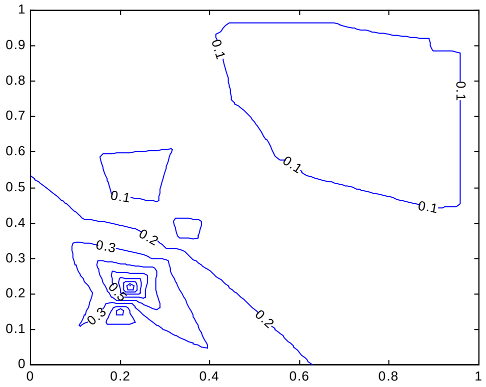
(a)



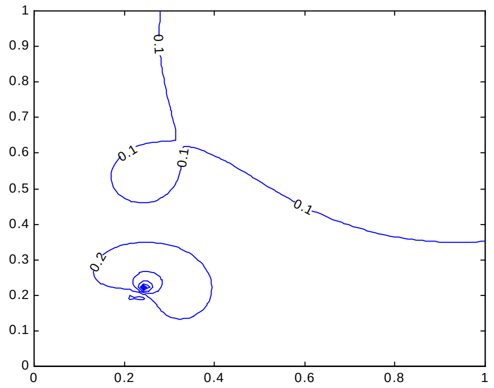
(b)

Figure 14. Visualization of the first cluster (membership function) centered around (0.2240 0.2236) : (a) 3D space and (b) contour plots. The Euclidean distance function was used in the clustering algorithm.

The above effect is even more pronounced when there are more clusters interacting with each other. We consider 8 prototypes identified by the two FCM algorithms, see Figure 15. In the case of Chebyschev FCM, it is clear that despite strong interactions between the clusters, the rectangular shape of the cluster membership function is preserved for a range of values of this function. These undistorted rectangles cover a good proportion of the original data, which is represented by the selected prototype. On the other hand, the Euclidean FCM results in contours of the membership function that are undistorted circles only in the very close proximity of the prototype itself. Thus the task of linking the original data with the prototype representing an association existing in the data is quite difficult for most of the data points.



(a)



(b)

Figure 15. Contour plots for one of the 8 clusters (membership function) centered around (0.2108 0.2248) for the Tchebyschev distance (a); and (0.2441 0.2203) for the Euclidean distance (b).

8. Conclusions

In this chapter we discuss a new algorithm for pattern classification that is based on novel representation of class sets as a difference of two types of fuzzy sets (the union of hyperboxes belonging to the given class and the union of hyperboxes belonging to different classes). It has been shown that, compared to the standard hyperbox paving approaches, the proposed algorithm results in a more efficient generation of complex topologies that are necessary to describe the pattern classes. The consequence of the adoption of the exclusion/inclusion framework is a greater interpretability of the classification results (smaller number of hyperboxes needed to cover the data). It has been shown that in the proposed approach the size of the hyperboxes does not need to be pre-determined and is indeed defined by the data itself. This is a very beneficial feature as it frees the analyst from making arbitrary choices with regard to the parameters of the algorithm. The low misclassification rate and good interpretability of the results of the proposed algorithm is achieved at the expense of rejecting a proportion of patterns that fall into the exclusion hyperbox set. If this proportion is small the algorithm provides an optimum mix of good classifier features. However, if the exclusion set becomes comparable in size to the inclusion set the maximum size of hyperboxes needs to be reduced. This is analogous to the standard hyperbox paving approaches but unlike in the standard approaches we do not use the misclassifications rate (that is dependent on the test data set) but instead use the ratio of exclusion to inclusion hyperbox sets (evaluated with training data only) as an indicator of how small hyperboxes need to be.

A general point raised by this investigation is that of a benefit of a richer vocabulary of topological constructs in describing data sets in multi-dimensional pattern spaces.

Acknowledgments

Support from the Engineering and Physical Sciences Research Council (UK) and the Natural Sciences and Engineering Research Council of Canada (NSERC) is gratefully acknowledged.

References

[1] Bargiela A., Pedrycz W., Granular clustering with partial supervision, *Proc. European Simulation Multiconference, ESM2001*, Prague, June 2001, 113-120.

[2] Bargiela, A., *Interval and Ellipsoidal Uncertainty Models* In: Granular Computing, Pedrycz, W. (ed.), Springer Verlag, 2001

[3] Bellman, R.E., Kalaba, R., Zadeh, L., 1966, Abstraction and pattern classification, *J. Math. Anal. Appl.*, 13, 1-7

[4] Bezdek, J.C., *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, N. York, 1981.

[5] Bobrowski, L., Bezdek, J.C., C-Means clustering with the l_1 and l_∞ norms, *IEEE Trans. on Systems Man and Cybernetics*, 21, 1991, 545-554.

[6] Cios, K., Pedrycz, W., Swiniarski, R., *Data Mining Techniques*, Kluwer Academic Publishers, Boston, 1998.

[7] Dubois, D., Prade, H., Fuzzy relation equations and causal reasoning, *Fuzzy Sets and Systems*, 75, 1995, pp. 119-134

[8] Gabrys B., Bargiela A., General fuzzy min-max neural network for clustering and classification, *IEEE Trans. Neural Networks*, vol.11, 2000, 769-783.

[9] Gottwald, S., Approximate solutions of fuzzy relational equations and a characterization of t-norms that define metrics for fuzzy sets, *Fuzzy Sets and Systems*, 75, 1995, pp. 189-201

[10] Groenen, P. J.F. Jajuga, K., Fuzzy clustering with squared Minkowski distances, *Fuzzy Sets and Systems*, 120, 2001, 227-237.

[11] Maimon, O., Kandel, A., Last, M., Information-theoretic fuzzy approach to data reliability and data mining, *Fuzzy Sets and Systems*, 117, 2001 pp. 183-194

[12] Pedrycz, W., *Fuzzy Control and Fuzzy Systems*, RSP/Wiley, 1989.

[13] Pedrycz, W., Gomide, F., *An Introduction to Fuzzy Sets*, Cambridge, MIT Press, Cambridge, MA, 1998.

[14] Simpson PK., Fuzzy min-max neural networks - Part1: Classification, *IEEE Trans. Neural Networks*, vol.3, 5, 1992, 776-786.

[15] Simpson PK., Fuzzy min-max neural networks – Part 2: Clustering, *IEEE Trans. Neural Networks*, vol.4, 1, 1993, 32-45.

[16] Zadeh, L.A., Fuzzy sets, *Inform. And Control*, 8, 1965, 189-200.

[17] Zadeh, L.A., Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, *Fuzzy Sets and Systems (FSS)*, 90, 1997, 111-117.

[18] Machine Learning Repository, http://www.ics.uci.edu/~mlearn/MLRepository.html