Data summarisation by typicality-based clustering for vectorial and non vectorial data

Marie-Jeanne Lesot and Rudolf Kruse, *Fellow, IEEE*

***Abstract*— In this paper, a typicality-based clustering algorithm is proposed: it exploits typicality degrees defined in a prototype construction framework to identify a decomposition of the dataset into homogeneous and distinct clusters and to provide characteristic representatives of the obtained clusters, so as to summarise the initial dataset. The proposed algorithm can be applied both to vectorial and non vectorial data, such as trees for instance. Tests performed on artificial and real data illustrate the interest of the proposed approach.**

I. INTRODUCTION

Clustering is an unsupervised learning task that aims at decomposing a dataset into homogeneous and distinct subgroups called clusters. Through this decomposition, it offers a simplified representation of the dataset that can be summarised by the clusters, and thus it helps the user to better apprehend the dataset.

In this paper a typicality-based algorithm is proposed to perform clustering with the aim to further improving the data summarisation property. It is based on typicality degrees [1], [2], that were proposed in a prototype construction framework as a means to build characteristic representatives of data categories. Typicality degrees model the representativeness of each data point, indicating the extent to which a point is characteristic of the group it belongs to. They depend both on the resemblance of the point to the other members of its category, and on its dissimilarity from members of other categories. Prototypes derived from these typicality degrees thus take into account both the common features of the category members and their discriminative features as compared to other groups [1].

In this paper, typicality degrees and prototypes are extended to unsupervised learning and the clustering task: the fact that both common and distinctive features are modelled is related to the aim of finding clusters that are both compact and separable, i.e. homogeneous and distinct. Indeed, points assigned to the same cluster are expected to be more similar to another than to points belonging to other clusters. Another advantage of this framework is that it makes it possible to consider vectorial data as well as non vectorial data: it enables to consider data for which a structured information is available, as for instance sequences, trees or more generally graphs, and not only vectors of numerical values. Furthermore, the typicality degree framework offers means to characterise each obtained cluster, and thus to improve the data summarisation property of clustering. These properties are discussed and illustrated in the next sections.

The paper is organised as follows: section II briefly recalls the typicality degree framework. Section III describes the proposed algorithm for the dataset decomposition into clusters and the cluster characterisation through their most typical members, taking into account both vectorial and non vectorial data. Section IV compares the proposed algorithm with other clustering algorithms from a theoretical point of view. Section V presents experimental results obtained both with artificial and real datasets of vectorial and non vectorial data; section VI concludes the paper.

II. TYPICALITY DEGREES AND FUZZY PROTOTYPES

A. Principle

Typicality degrees [1], [2] were defined in the context of prototype construction, as a means to build representatives to summarise sets of data: they are numerical coefficients taking values between 0 and 1 that measure the extent to which a point is representative of the category it belongs to. According to the typicality notion defined by Rosch [3], they take into account two complementary components, respectively called internal resemblance and external dissimilarity: a point is said typical of its category depending both on its resemblance to the other members of the category (internal resemblance), and on its dissimilarity to members of other categories (external dissimilarity). This for instance models why whales and platypuses can be considered as atypical examples of the mammal category: whales are too similar to the members of the fish category and thus have a low external dissimilarity. On the other hand, platypuses are not similar enough to other mammals and have a low internal resemblance. Both have a low typicality degree with respect to the mammal category.

Typicality degrees can then be exploited to define a prototype, that takes into account both the common features of the category members, and their discriminative features as compared to other categories. This leads to a significant representative that characterises the considered category.

B. Formalisation

There exist different definitions for typicality degrees (see e.g. [4]) but most of them do not implement the previous Rosch definition. We consider here the formalisation proposed in [1] and extended in [2] that complies with the previous principles.

Marie-Jeanne Lesot is with the Knowledge Processing and Language Engineering, Otto-von-Guericke University of Magdeburg Universitätsplatz 2, D-39106 Magdeburg, Germany (email: lesot@iws.cs.uni-magdeburg.de).

Rudolf Kruse is with the Knowledge Processing and Language Engineering, Otto-von-Guericke University of Magdeburg Universitätsplatz 2, D-39106 Magdeburg, Germany (email: kruse@iws.cs.uni-magdeburg.de).

Let's denote $X = \{x_i, i = 1..n\}$ a dataset with points belonging to several categories, C a category, and x a point in X assigned to C. The computation of typicality degrees and prototypes requires the definition of comparison measures to evaluate resemblance and dissimilarity. These comparison measures are functions taking as input data point couples and respectively indicating by a value in $[0, 1]$ the extent to which the two points are similar and dissimilar [5]. We denote them ρ and δ respectively in the following.

The internal resemblance of point x with respect to category C, $R(x, C)$, is defined as its average resemblance to the other members of the category; likewise, its external dissimilarity, $D(x, C)$, is computed as its average dissimilarity to points in other categories:

$$R(x, C) = avg(\rho(x, y), y \in C) \qquad (1)$$

$$D(x, C) = avg(\delta(x, y), y \notin C) \qquad (2)$$

The typicality degree is then derived as the aggregation of these two quantities, as

$$T(x, C) = \varphi(R(x, C), D(x, C)) \qquad (3)$$

φ is an aggregation operator that expresses how the typicality degree depends on $R(x, C)$ and $D(x, C)$: a conjunctive operator e.g. requires that a point have both high internal resemblance and external dissimilarity to be considered as typical. Tradeoff operators, such as the weighted mean, relax this definition and can model a compensation property: a high internal resemblance may compensate for a low external dissimilarity. The choice of the aggregation operator determines the semantics of the typicality degrees [2].

Lastly a prototype can be deduced from these typicality degrees, to provide a summarised representation of the category and underline its most characteristic features. The simplest method consists in representing the category by extracting its most typical members: the definition of the typicality degrees guarantees that these representatives will provide a relevant summary of the category. Another possibility consists in aggregating these most typical points, to build a prototype, in the form [1]

$$p_C = \psi(\{x_i / T(x_i, C) > \tau\}) \qquad (4)$$

where τ is a typicality threshold and ψ an aggregation operator. Rifqi [1] considers the case of fuzzy data, i.e. data whose attributes take as values fuzzy sets; ψ is then a fuzzy set aggregation operator. In the case of numerical crisp data, ψ can be the weighted mean, using typicality degrees as weights. ψ can also be an operator that builds a fuzzy set from the typicality degrees [2], so as to model the imprecision of the prototype and not reduce it to a single point.

III. Typicality-based clustering algorithm

A. Principle

In this paper, the previous typicality degrees are extended to unsupervised learning, so as to define a clustering algorithm that improves the data summarisation property. The underlying idea is that a good partition is such that each point is typical of the group it is assigned to. Thus the algorithm aims at maximising the typicality degrees: it considers a hypothetical partition of the data, computes the typicality degrees with respect to this partition, and corrects the partition so that each point becomes more typical of the group it is assigned to.

Consider for instance the illustrative example represented on figure 1: clustering aims at identifying the two groups A and B indicated on figure 1a by $\circ$ and $*$ respectively. Figure 1b represents a random initialisation, that assigns points to 2 clusters, a and b, respectively depicted with $+$ and $\triangleleft$; figure 1c indicates the typicality degrees of all points with respect to these 2 clusters (full line for cluster a, dashed line for cluster b) as a function of their identification number. The two points in B assigned to cluster a (points 12 and 13) have a low typicality degree for cluster a, because they are on average not similar enough to the other points in a and not dissimilar enough from points in b; they have a higher typicality degree to cluster b than to cluster a. More generally, assigning each point to the cluster it is most typical of leads to the partition represented on figure 1d, which is the expected one. The associated typicality degrees are illustrated on figure 1e, they are higher than those on graph (c). It is to be noted that other initialisations may require slightly more steps to converge to the desired partition.

Thus the proposed algorithm consists in alternatively computing typicality degrees given a data partition and updating the data assignment according to typicality degrees. After obtaining the clusters, the final typicality degrees are exploited to define characteristic representatives for the clusters and summarise them. The algorithm is summarised in table I and described in details in the following.

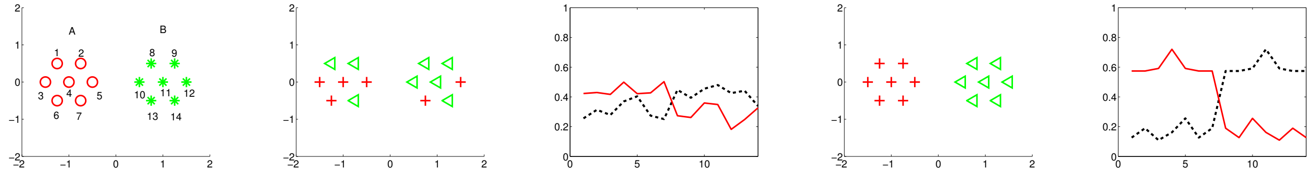
It is to be noted that the typicality framework offers a means to detect outliers and points located in overlapping regions between the clusters: both have low typicality degrees, respectively because of low internal resemblance and low external dissimilarity. Their detection makes it possible to apply special handling to them, improving the quality of the clusters: clusters are expected to be compact and distinct, which means they must be robust against outliers and not concentrated in areas where subgroups overlap.

Moreover, as can be seen from the previous section, the typicality degrees computation does not depend on the data nature: it does not handle data points as such but only through the values of their comparison, resemblance and dissimilarity. Thus it can be applied to vectorial data as well as non vectorial (structured) data.

B. Assignment Step

As indicated above, the proposed algorithm consists in alternatively computing typicality degree and data assignments. As regards the derivation of a data partition from typicality degrees, it seems natural to assign points to the cluster they are most typical of, i.e. to define

$$x_i \in C_r \iff r = \arg\max_s T(x_i, C_s) \qquad (5)$$



(a) (b) (c) (d) (e)

Fig. 1. Principle of the proposed algorithm. (a) Considered dataset, expected clusters, and data point numbering. (b) Random initialisation of the data partition, into 2 clusters, a (depicted with $+$) and b (depicted with $\triangleleft$). (c) Typicality degrees with respect to the clusters a and b, for all data point represented by their identification number; the plain line indicates the typicality degree with respect to cluster a, the dashed one for cluster b. (d) Updated partition according to the typicality degrees. (e) Resulting typicality degrees.

Two specific cases are considered separately: the case where the maximal typicality degree is small and the case where the maximum is not clearly defined.

When the maximal typicality degree is small (lower than 0.1 in our experiments), a point assignment would not be significant: this case corresponds to points for which all typicality degrees are small, i.e. points that are typical for no cluster. They are to be interpreted as outliers and should not be assigned to any cluster: if they were assigned to “real” clusters, they would for instance distort the computation of the average in eq. (1), leading to non significant and low internal resemblances for the other cluster members. Therefore, they are assigned to a fictitious cluster for which no typicality degree is computed, that can be interpreted as a noise cluster.

Another special case corresponds to points for which the maximal typicality degree is not clearly defined, i.e. the second biggest typicality value is close to the biggest one (in our experiments, when their difference is smaller than 0.02). A tie-breaking strategy is needed, to avoid arbitrary assignments. These points are considered as equally typical of two clusters and usually correspond to points located in overlapping areas between clusters; their typicality degrees are then rather low for both clusters. Such points are also assigned to the fictitious cluster.

C. *Typicality Degree Step*

Given a partition of the data, typicality degrees are then computed. In the supervised learning framework, typicality is only considered for the category a point belongs to, and equals 0 for the other categories. In the unsupervised framework, clusters are to be questioned; therefore, typicality is computed with respect to all clusters: for each point, its assignment to each cluster is successively considered. The assumed partition is only used to determine, for each considered assignment, which points belong to the same cluster and to other clusters in order to compute the internal resemblance and external dissimilarity. Typicality degrees are also computed for points assigned to the fictitious cluster: this assignment only means these points are not taken into account in the typicality degree computation for other points.

The typicality degree computation step requires the computation of internal resemblance, external dissimilarity and their aggregation. In the following, the choices of comparison measures and aggregation operator are discussed, taking into account the cases of both vectorial and non vectorial data.

1) Choice of the Resemblance Measure: Resemblance measures are functions that take as arguments two data points and return a value in the interval $[0,1]$ indicating the extent to which the two points are similar [5]. Their choice, together with the dissimilarity measure choice, is obviously essential for the clustering result and has a major influence.

In the case of vectorial data, by analogy with the possibilistic clustering algorithm (PCM) [6], we propose to use

$$\rho(x,y) = \frac{1}{1+\left(\frac{d(x,y)}{\gamma_R}\right)^2} \tag{6}$$

where d is a distance, as the Euclidian distance and γ_R a user-defined parameter. γ_R corresponds to a reference distance: it is a normalising coefficient that rules the distance from which the resemblance between two points will be lower than 0.5.

Now in the typicality framework, the resemblance measure is only used to compare points assigned to the same cluster, so as to compute internal resemblance. Therefore it seems natural to define a normalisation for each cluster independently, leading to one resemblance measure pro cluster. We propose to define γ_{Rr} as half the cluster r diameter: this implies that each cluster contains point couples that are considered as totally resemblant, and other couples having a 0 resemblance. Thus the resemblance measure values indeed cover the range $[0,1]$ in each cluster.

As clusters are searched for, their diameters is not known at the beginning of the process, and γ_{Rr} cannot be defined. Therefore, we propose to define a two-step algorithm, following the PCM methodology: first initialisation is performed using a few steps of fuzzy c-means (FCM) (10 iterations in our experiments). Then for all $r = 1..c$, $\gamma_{Rr} = (\sum_i u_{ri}^m d(x_i,w_r))/(\sum_i u_{ri}^m)$ where (u_{ri}) and (w_r) are the membership degrees and centre positions provided by FCM. Second, after having converged using these values, the obtained data partition is used to update the estimation of the cluster diameters and γ_{Rr} values.

In the case of non vectorial data, the resemblance measure must also be defined, and depends on the considered data nature. One possibility is to use kernel functions [7]: the latter are functions applied to data couples that have the properties of scalar products and can be interpreted as resemblance measures. Still they require to be normalised;

this normalisation step can be difficult to design from an interpretation point of view. Therefore, we propose to define a resemblance measure deduced from the distance associated to the scalar product: denoting k the kernel function, one can consider the distance defined by $d_K^2(x, y) = k(x, x) - 2k(x, y) + k(y, y)$ and use it in eq. (6). Indeed, one can then use the same normalisation process as indicated above: the reference distance determination is also justified with this other distance measure.

2) Choice of the Dissimilarity Measure: A dissimilarity measure is a function that takes as arguments two points and returns a value in the interval $[0, 1]$ that indicates the extent to which the two points are dissimilar [5]. We propose to use

$$\delta(x, y) = 1 - \frac{1}{1 + \left(\frac{d(x,y)}{\gamma_D}\right)^2} \qquad (7)$$

that is the complement to one of the previous resemblance measure. Yet, the normalisation parameter γ_D is chosen independently of γ_R, to adapt to the different distance scale: dissimilarity is used to compare points belonging to different clusters. Thus the considered distances are on average bigger than those involved in the resemblance computation: using $\gamma_D = \gamma_R$ would lead to dissimilarity values that are all very high, and non informative, the ulterior aggregation of R and D would not be informative either. Therefore the normalisation parameter must refer to a different distance, we propose to base it on the data diameter $diam(X)$. More precisely, we choose γ_D so that the dissimilarity is 0.9 for points at distance $diam(X)/2$. This implies that there exist point couples that are totally dissimilar, as well as points that are to be considered as having a zero dissimilarity, i.e. δ indeed covers the $[0, 1]$ range.

In the case of non vectorial data, as for resemblance measures, the kernel-derived distance d_K can be used in eq. (7), without requiring to modify the normalisation process.

3) Choice of the Aggregation Operator: Lastly typicality degrees are defined by the aggregation of internal resemblance and external dissimilarity. This choice also has a major influence on the obtained results and determines the semantics of typicality [2]. It cannot be chosen as freely as in the supervised framework: in the latter, one can be interested in discriminative prototypes (in classification tasks e.g.), thus a high importance may be given to the external dissimilarity.

In the clustering case, both internal resemblance and external dissimilarity must be simultaneously influential: otherwise outliers may be considered as highly typical of any cluster, and may not be excluded, distorting the clustering results. This means that the operator should be a conjunctive operator, at least at the beginning of the process to exclude outliers; we use $\varphi = \min$. In a second step, one can be more tolerant, we recommend to use a variable behaviour operator offering a full reinforcement property [8] such as the MICA operator, defined as $\varphi(a, b) = \max(0, \min(1, a + b - t))$ where t is a user-defined parameter we set at 0.6.

TABLE I

PROPOSED TYPICALITY-BASED CLUSTERING ALGORITHM

```
Notations: $X = \{x_i, i = 1..n\}$, the considered dataset, $c$ the desired number of clusters, $\rho$ and $\delta$ resemblance and dissimilarity measures, $\varphi$ an aggregation operator (see text for their recommended choice).
Initialisation: apply a few steps of FCM and assign points according to their maximal membership degree. Compute the normalising coefficients used in the comparison measures.
Loop: while assignment evolves, alternate
    1) Typicality step: for each point $x$ and each cluster $C_r$, $r = 1..c$
        a) Compute the internal resemblance
           $R(x, C_r) = avg(\rho(x, y), y \in C_r)$
        b) Compute the external dissimilarity
           $D(x, C_r) = avg(\delta(x, y), y \notin C_r)$
        c) Compute the typicality degree
           $T(x, C_r) = \varphi(R(x, C_r), D(x, C_r))$
    2) Assignment step: for each point $x$
        a) If $x$ is typical for no cluster, i.e $\max_r T(x, C_r) < 0.1$,
           assign $x$ to a fictitious cluster $C_0$
        b) If the maximum is not clear, i.e. $T_1(x) - T_2(x) < 0.02$
           where $T_i(x)$ is the $i$th biggest typicality value, assign
           $x$ to the fictitious cluster $C_0$
        c) Else assign $x$ to the cluster maximising the typicality,
           i.e. to $C_r$ where $r = \arg\max_s T(x, C_s)$
After convergence of the loop, update the values of the cluster diameters based on the new data assignment and apply the loop a second time.
```

D. Cluster Representative

1) Principle: The clustering algorithm, summarised in table I, alternatively computes typicality degrees and data partition until stability of data assignments. It provides a decomposition of the initial dataset into subgroups that offers a summarisation of the dataset, through the representation as a reduced number of clusters instead of all individual data points. Exploiting the obtained typicality degrees, this data summarisation can be improved by characterising each cluster to help the user to better apprehend it.

As indicated in section II, different definitions of prototypes can be deduced from typicality degrees.

The fuzzy set definition [2] makes it possible to model the imprecision of group representatives, that are not reduced to a single precise point. In the case of vectorial data, when fuzzy prototypes are built attribute by attribute, they can be directly interpreted in the linguistic variable framework. Thus they provide an interpretable representation of the data. Yet, when built globally and not attribute by attribute, these fuzzy sets may be more difficult to interpret, although they can model correlation between attributes. Likewise, in the case of non vectorial data such fuzzy sets are difficult to interpret.

As mentioned in section II, other solutions include the computation of weighted average, but this approach can only be applied in the case of vectorial data. Therefore, a simpler representation method is applied here, consisting in representing the cluster by its most typical members. This corresponds to an intuitive process: in the case of complex objects, a description is often performed through a set of typical examples that illustrate and characterise the underlying notion. Typicality degrees offer the means to determine representative points, that indeed lead to a relevant cluster summarisation, due to their definition as detailed in the previous sections.

It is to be noticed that the produced representation can only be interpreted if the user has a means to interpret data points individually, be it based on his knowledge of the data, or on visualisation for individual data points.

Note that there exist other methods for providing interpretable description for clusters, as for example the method proposed by [9]. The aim here is simply to exploit further the rich information provided by the typicality framework.

2) Method: The summarisation of each cluster thus simply consists in characterising it through the most typical examples, i.e. defining as prototype

$$p_C = \{x_i / T(x_i, C) > \tau\} \tag{8}$$

without aggregating these most typical data points. This is equivalent to producing as representative an α-cut of the fuzzy prototype, with high α values.

The difficulty is then to choose the threshold τ. It must be a compromise between the quantity of data a user can interpret, and the precision of the proposed representation. It must be noted that the threshold can vary between clusters as it can be the case that some clusters are more easily characterised than others: for instance well separated clusters contain points with on average higher typicality degrees than other clusters. Thus an absolute threshold does not seem appropriate.

One way to determine it can be to define the proportion of cluster members one wants to consider as typical examples, in some cases, a single representative can be desired, in other cases more details can be more useful.

IV. Comparison with other clustering algorithms

Before illustrating the results obtained with the proposed algorithm, we compare it to other classic clustering methods, namely fuzzy c-means and some of their variants, that follow the same approach. Indeed, they are also based on iteratively alternating two steps, consisting in computing weighting coefficients and updating cluster parameters based on these coefficients. In the proposed algorithm, no cluster parameters are considered, i.e. clusters are directly described by their members, thus the cluster parameter update reduces to a data partition update. Still, the underlying principle is similar, and in the following we compare the used weighting coefficients and the typicality degrees the proposed method relies on.

In the following, we mention the basic algorithms, defined for vectorial data. There exist other variants to handle non vectorial data, such as the relational clustering approaches [10], [11] or the kernel-based methods (see [12] e.g). The following comments also apply for these cases. It is to be noted that in the kernel approach, no cluster center in the input space can be defined, they remain implicit; a cluster characterisation similar to the one described in section III-D could be used, representing a cluster by the points having the maximal weighting coefficients. Yet, as discussed below, the semantics of these coefficients do not justify such a method.

A. Fuzzy c-means

The fuzzy c-means (FCM) algorithm is based on the definition of membership coefficients, that indicate the extent to which a point belongs to a cluster. More precisely, they indicate the extent to which a point is shared between the clusters: the quantities involved in their definition are relative distances, that compare the distance to a cluster centre to the distance to other cluster centres.

Due to this relative definition, the influence of a point does not decrease with its absolute distance to the centres (see e.g. [13]). This is not compatible with an interpretation in terms of typicality degrees, where such a decrease is expected: FCM do not associate outliers, i.e. points located far away from all data, with a small degree as expected, but consider them as equally shared between all clusters. Their membership degrees equal the reciprocal of the number of clusters and they influence the cluster centre positions.

B. Possibilistic c-means

The possibilistic c-means (PCM) [6] relax the constraint that causes the relative definition of membership degrees in FCM, so as to be more robust. The coefficients they are based on then measure the absolute resemblance between data points and cluster centres, and not a relative resemblance. Outliers are thus associated to small coefficients as expected.

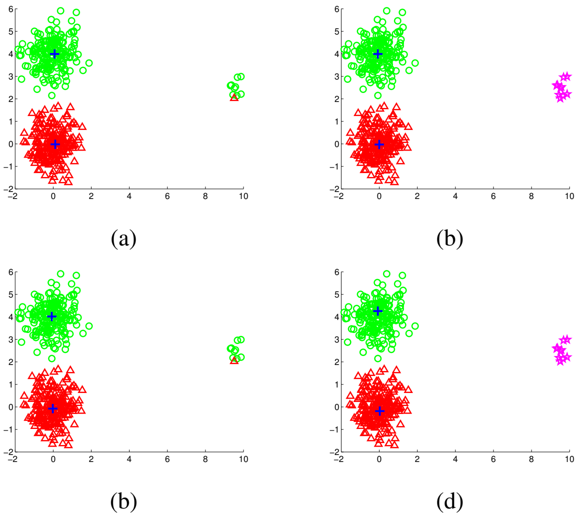
Now the resemblance between a data point and a cluster centre can be interpreted as an internal resemblance: it replaces the average resemblance to the group members (see eq. (1)) by the resemblance to some average of the group members. Thus PCM can be seen as based on internal resemblance: the possibilistic coefficients correspond to typicality degrees in the specific case where the aggregation operator φ does not depend on external dissimilarity.

Partially due to this fact, PCM suffer from a coincident cluster problem (see e.g. [14]): they sometimes lead to confounded clusters, whereas natural subgroups of the data are overlooked. Timm et al. [15] propose to solve this problem by modifying the cost function to impose cluster repulsion and lead clusters apart from each other. The cluster centre expressions are then modified, the weighting coefficients keep the same definition as the PCM coefficients.

The proposed typicality-based algorithm also provides a cluster repulsion property, but the latter is incorporated in the coefficient definition itself: taking into account external dissimilarity, typicality degrees consider the condition that clusters must be distinct one from another and have a repelling effect.

C. Possibilistic Fuzzy c-means

Pal et al. [13] propose another solution to the PCM cluster merging problem: they argue that both possibilistic and membership degrees are necessary to perform clustering and propose to combine the two approaches: in the Possibilistic Fuzzy c-means algorithm (PFCM), they consider a weighted sum of FCM and PCM coefficients, each one being raised to a parameter power. The obtained coefficients are thus combination of relative and absolute resemblance.



(a)

(b)

(b)

(d)

Fig. 2. Clustering results using (a) FCM, (b) PCM, (c) PFCM, (d) proposed typicality-based algorithm. Each symbol depicts a different cluster, stars represent points that are not assigned to any cluster, plus the cluster centres (the centre is replaced by the most typical cluster member for the proposed algorithm, on graph (d)).

TABLE II

CLUSTER CENTRE POSITION

Position of the cluster centres for the dataset represented on figure 2 for several clustering algorithms. In the case of the proposed typicality-based algorithm (TB), the most typical point for each cluster is indicated.

Algorithm	Centre 1		Centre 2	
FCM	0.07	3.99	0.11	-0.02
PCM	-0.07	3.99	-0.00	-0.03
PFCM	-0.08	4.02	-0.02	-0.07
TB	-0.08	4.26	0.03	-0.18

The proposed algorithm also adds to the absolute resemblance a complementary term, but the latter is the external dissimilarity; the aggregation scheme is more flexible than the weighted sum. Internal resemblance and external dissimilarity can be considered as more clearly complementary of each other than PCM and FCM coefficients. The values obtained after combination have a higher interpretative power: their meaning can be directly understood, whereas the semantics of the PFCM coefficients are less clear. The definition of cluster representatives as points maximising these coefficients does not seem justified.

V. EXPERIMENTAL RESULTS

In this section we illustrate the results obtained using the typicality-based algorithm and the associated cluster characterisation through most typical examples. Tests are performed using vector and tree data, that are respectively artificial and real data.

A. *Artificial Data with Euclidian Distance*

We first consider a simple 2 dimensional dataset represented on figure 2 to illustrate the previously mentioned differences between the proposed typicality-based algorithm, FCM, PCM and PFCM. The dataset is constituted of two spherical Gaussian clusters centred around $(0, 4)$ and $(0, 0)$ respectively, and a small outlying group. All algorithms are applied to find 2 clusters, the FCM and PCM fuzzifier is $m = 2$, for PFCM, the parameters are $a = 1$, $b = 1$, $m = 7$ $\eta = 1.5$ (see [13] for the notations). Data assignment is performed according to the maximal weighting coefficient values (in the PFCM case, membership degrees are used, as indicated in [13]) and is illustrated on figure 2: each symbol depicts a different cluster, for PCM and PFCM, the stars represent points for which no assignment is relevant, because all coefficients values are smaller than 0.1. The plus sign represents the cluster centres (replaced by the most typical example for each cluster for the proposed algorithm). Table II gives the numerical values of the centre coordinates.

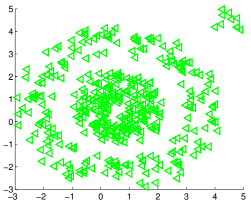
As can be observed on figure 2a, the FCM results are influenced by the outliers that attract the centres to the right side, centres do not have a 0 abscissa. PCM results are more robust; on this dataset, no cluster merging occurs. As compared to FCM, PFCM provides better cluster centres that are not attracted by the outliers: this is due to the combination with PCM that reduces the outlier influences. The assignment based on membership degrees does not recognise the outlying group as such and assigns its points to the two main clusters. Yet it is to be noted that the possibilistic coefficients are almost zero for these points, indicating their specificity. Still the combined coefficients, obtained as weighted sum of the FCM and PCM coefficients do not have an interpretable semantic.

The proposed typicality-based algorithm detects the two groups and identifies the outliers that do not influence the abscissa of the clusters centres. These results illustrate the repulsing effect of the proposed algorithm: the most typical points in each cluster are clearly forced apart on the y-axis. Indeed, they aim at characterising the groups, underlying the common points of their members but also their distinctive features. They do not correspond to the group average, that only takes into account the internal resemblance (as is the case for PCM), but characterise the groups one as opposed to the other, so as to provide a more interpretable summarisation of the dataset. Note that if this effect is judged too important, other aggregation operators can be used instead of the MICA aggregator.

B. *Ring Data*

The proposed algorithm also makes it possible to handle data for which the Euclidian distance is not appropriate: we consider for instance the ring data illustrated on figure 3, constituted of a noisy ring that surrounds a cluster whose distribution follows a Gaussian law. An outlying subgroup is present in the upper right corner.

As the Euclidian distance favours spherical convex clusters, it is not adapted to handle this dataset. Therefore, we propose to use a Gaussian kernel $k(x, y) = \exp(-\frac{d^2(x,y)}{2\sigma^2})$, with $\sigma^2 = 0.8$. Figure 3b represents the obtained clusters (as before, stars represent points assigned to the fictitious cluster). The obtained partition corresponds to the expected



(a) (b)

Fig. 3. Ring dataset and clustering result obtained using the proposed typicality-based algorithm.

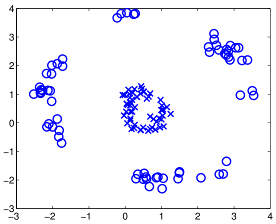


Fig. 4. Most typical data points for the two clusters of the ring dataset.

one: the two major clusters are detected and the outlying group is identified as such. Some other points are assigned to the fictitious cluster: they correspond to points located between the two clusters, whose assignments is not clear.

In order to characterise the obtained clusters, we apply the method proposed in section III-D and retrieve only the points whose typicality degree is higher than a fixed threshold. The latter is defined for each cluster so as to get around 30% of the cluster as representatives. The obtained points are represented on figure 4. This visual exploitation gives a quite reliable representation, where one can recognise the structure of the dataset.

C. Structured Data

Lastly we consider structured data, in the form of XML data representing student results to several exams, as trees identical to the one showed on figure 5: each student is described by his marks for several subjects, the XML structure indicates relationships between these subjects. For instance, it distinguishes between courses and internships, and opposes theoretical courses to practical ones. It is to be noted that the structure is the same for all students, only the leave content varies from one student to the other. Students could be represented as vectors of their results, but the available structure provides useful information that should be exploited.

Indeed, it indicates that attributes are not independent, suggesting to enrich the classic Euclidian distance that performs an attribute per attribute comparison: two attributes belonging to the same branch of the tree convey a similar meaning and their values should also be compared. Therefore, in [16], we proposed to consider the kernel defined as

$$k(x, y) = \sum_{i=1}^{d} \sum_{j=1}^{d} \lambda_{ij} x_i y_j$$

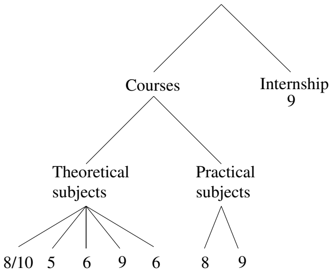


Fig. 5. Example of considered XML data.

$$\text{with } \lambda_{ij} = \delta_{ij} + \frac{l}{P + 1 - p(i, j)} (1 - \delta_{ij})$$

where d denotes the number of fields in the XML structure, $l < 1$ is a user-defined parameter, P the depth of the XML tree and $p(i, j)$ the depth of the deepest common node for fields i and j. This corresponds to a weighted Euclidian scalar product, where the weights are derived from the structure. When the values for the same attribute are compared, i.e. $i = j$, $\lambda_{ij} = 1$ as in the standard case. If the values of two different attributes are compared, $i \neq j$, the weight of this comparison is $\lambda_{ij} \neq 0$, taking a value that depends on the XML structure: if the 2 fields i and j have nothing in common, their deepest common node is the root, $p(i, j) = 1$, which leads to $\lambda_{ij} = l/P$. If they belong to the same branch of the tree, $\lambda_{ij} = l$ because $p(i, j) = P - 1$. Thus they have a higher influence in the comparison of the corresponding values in the kernel definition.

This kernel corresponds to a linear transformation of the vectorial representation of the students. The kernel advantage in this framework is that it is much easier to encode the available additional information about the relationships between attributes in the kernel, as weights, rather than make explicit the associated transformation [16].

We applied the proposed clustering algorithm to a real dataset describing 42 students using the XML structure indicated on figure 5, choosing to look for 3 clusters, and setting $l = 0.6$. Figure 6 presents the obtained clusters, representing students as vectors of their field values with parallel coordinates, although this representation leads a visual bias as it suggests an attribute by attribute interpretation.

The comparison and interpretation of this graphical representation is not so easy. Therefore, we retrieve the most typical student of each group, and represent the three of them on the same graph (see fig. 7). Knowing that typicality takes into account both the common and distinct features, we can compare them and thus characterise the three obtained groups: the first cluster, represented by the typical example in plain line, corresponds to the best students, that have in particular good results for the theoretical part. The second group, represented by the dashed line, corresponds on the contrary to the students having more difficulties for theoretical subjects as well as for the second practical exam. The third group corresponds to middle students, who are only slightly less good than those in the first group. These representative most typical members of the groups thus help

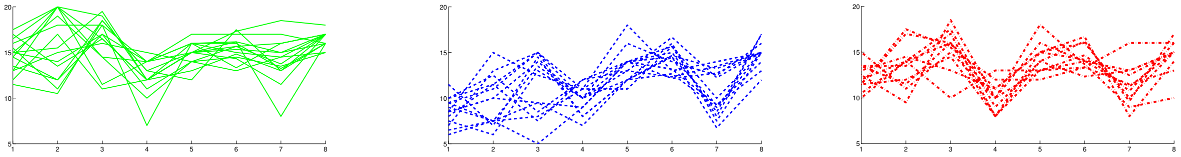


Fig. 6. Clustering results for XML data represented as the sequences of their field values (parallel coordinates).

interpreting the clustering results and summarising the initial dataset in an interpretable way.

Looking back to the whole data, and exploiting the previous interpretation, one can notice that the proposed kernel indeed leads to some correlation between the attributes: it can be seen in the first group that the second and third exam compensate one for another. Students in this group do not all have excellent result on the second exam, some compensate a lower result by the third exam.

VI. Conclusion

This paper presents a clustering algorithm based on typicality degrees to take into account both the common and discriminative features of the clusters to be identified. It provides relevant clusters that are indeed compact and separable and offers a means to characterise the obtained clusters and facilitate their interpretation, leading to a summarisation of the initial dataset. The proposed method can be applied independently of the data nature and the considered distance, it only requires the definition of normalised resemblance and dissimilarity measures.

The performed experiments confirm the expected properties of the proposed algorithm and justify the proposed approach. A more comprehensive study of the algorithm is necessary to validate it. In particular, the result quality was here visually judged, a more objective criterion is necessary.

The proposed method makes the assumption that the number of clusters is known, which is a limiting hypothesis. The combination with an objective quality criterion would make it possible for instance to test several values and select the most appropriate one.

Another limitation of the proposed algorithm is the fact that it requires a crisp partition of the data. It would be interesting to render the assignment step more flexible, and to allow gradual membership degrees. It must be noted that this step requires a precise study: membership degrees are related to resemblance to cluster centres, i.e. to internal resemblance. Thus the role of the membership degree on the computation of the internal resemblance involved in the typicality degree must be examined thoroughly, to clearly determine the influence of each of them.

Acknowledgement

We want to thank Christian Döring for fruitful discussions. Marie-Jeanne Lesot was supported by a Lavoisier grant from the French Ministère des Affaires Etrangères.

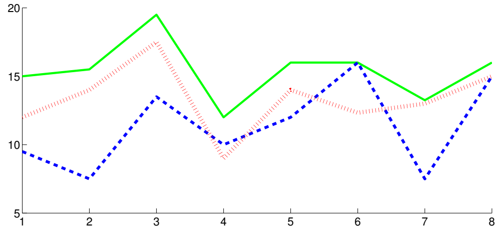


Fig. 7. Most typical example for each cluster obtained for the clusters represented on figure 6 (parallel coordinates).

References

[1] M. Rifqi, “Constructing prototypes from large databases,” in *Proc. of IPMU’96*, 1996.

[2] M.-J. Lesot, L. Mouillet, and B. Bouchon-Meunier, “Fuzzy prototypes based on typicality degrees,” in *Proc. of the 8th Fuzzy Days’04*, ser. Advances in Soft Computing. Springer, 2005.

[3] E. Rosch, “Principles of categorization,” in *Cognition and categorization*, E. Rosch and B. Lloyd, Eds. Lawrence Erlbaum associates, 1978, pp. 27–48.

[4] M.-J. Lesot, “Similarity, typicality and fuzzy prototypes for numerical data,” in *6th European Congress on Systems Science, Workshop ”Similarity and resemblance”*, 2005.

[5] B. Bouchon-Meunier, M. Rifqi, and S. Bothorel, “Towards general measures of comparison of objects,” *Fuzzy sets and systems*, vol. 84, no. 2, pp. 143–153, 1996.

[6] R. Krishnapuram and J. Keller, “A possibilistic approach to clustering,” *IEEE Transactions on fuzzy systems*, vol. 1, pp. 98–110, 1993.

[7] B. Schölkopf and A. Smola, *Learning with kernels*. MIT Press, 2002.

[8] M. Detyniecki, “Mathematical aggregation operators and their application to video querying,” Ph.D. dissertation, Université de Paris VI, 2000.

[9] M. Drobics, U. Bodenhofer, and E. P. Klement, “FS-FOIL: An inductive learning method for extracting interpretable fuzzy descriptions,” *Internat. J. Approx. Reason.*, vol. 32, no. 2–3, pp. 131–152, 2003.

[10] R. Hathaway, J. Davenport, and J. Bezdek, “Relational duals of the c-means clustering algorithm,” *Pattern Recognition*, vol. 22, no. 2, pp. 205–212, 1989.

[11] R. J. Hathaway and J. C. Bezdek, “Nerf c-means: Non-euclidean relational fuzzy clustering.” *Pattern Recognition*, vol. 27, no. 3, pp. 429–437, 1994.

[12] Z. Wu, W. Xie, and J. Yu, “Fuzzy c-means clustering algorithm based on kernel method,” in *Proc. of ICCIMA’03*, 2003, pp. 1–6.

[13] N. Pal, K. Pal, J. Keller, and J. Bezdek, “A new hybrid c-means clustering model,” in *Proc. of the IEEE Int. Conf. on Fuzzy Systems, FUZZ-IEEE’04*, I. Press, Ed., 2004, pp. 179–184.

[14] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler, *Fuzzy Cluster Analysis, Methods for classification, data analysis and image recognition*. Wiley, 2000.

[15] H. Timm, C. Borgelt, C. Döring, and R. Kruse, “An extension to possibilistic fuzzy cluster analysis,” *Fuzzy Sets and Systems*, vol. 147, pp. 3–16, 2004.

[16] M.-J. Lesot, “Kernel-based outlier preserving clustering with representativity coefficients,” in *Modern Information Processing: From Theory to Applications*, B. Bouchon-Meunier, G. Coletti, and R. Yager, Eds. Elsevier, in press, pp. 183–194.