Subspace Clustering and Feature Typicality Degrees: a Prospective Study

Marie-Jeanne Lesot, Adrien Revault d'Allonnes

Subspace Clustering and Feature Typicality Degrees: a Prospective Study

Marie-Jeanne Lesot
Sorbonne Université, LIP6,
F-75005, Paris, France
Email: Marie-Jeanne.Lesot@lip6.fr

Adrien Revault d'Allonnes
Université Paris 8, EA 4383, CUBI,
FR-93526, Saint-Denis, France
Email: ara@up8.edu

***Abstract*—Subspace clustering can offer, beside a decomposition of data into homogeneous and distinct clusters, a characterisation of the subspaces in which the clusters live. This paper explores the possibility of capturing the notion of characteristic features in the framework of typicality degrees, as typical features. To that aim, it discusses the notion of typicality degrees for features and proposes an Alternating Cluster Estimation algorithm, named TbSC, to exploit these degrees within subspace clustering. It illustrates their differences experimentally using simple data sets.**

I. Introduction

Subspace clustering is an unsupervised learning task that addresses a form of refined clustering objective: as clustering, it decomposes data sets into subgroups that must be both distinct and compact. In addition to this double constraint, subspace clustering takes into account the fact that the data subgroups belong to different subspaces of the feature space. On the one hand, this assumption can help identify more relevant data subgroups and relaxes the need to use a single, global, similarity measure: subgroups need not be dense in the whole feature space, but only in local subspaces. Moreover these subspaces are automatically identified, along with the clusters themselves and they do not need to be predefined, e.g. by expert knowledge. On the other hand, this assumption can lead to enrich the data description: beyond the data summary provided by its decomposition into a reduced number of subgroups, each cluster can be further characterised and described by its associated subspace.

As detailed in Section II-A, numerous subspace clustering approaches have been proposed, that can be structured into two main families, depending on the point of view they favour among the two abovementioned advantages.

This paper proposes to explore the interpretation of subspace clustering in the framework of typicality degrees and prototypes introduced in [24] following the cognitive notions proposed in [25]: the computational definition of prototypes characterises a set of instances, called class, as opposed to other classes, through a representative instance that both does possess properties shared among the class members and does not possess properties shared with members of the other classes. As detailed in Section II-B, the definition of prototypes relies on the definition of typicality degrees, that can be applied to instances or to attribute values.

This paper proposes to examine whether the subspace that characterises a cluster in subspace clustering can be interpreted as a subspace that is typical for it. To that aim, it explores how typicality degrees can be defined for subspaces, viewed as feature combinations: it proposes a prospective study about the introduction of typicality related notions for subspace clustering. At a formal level, it interprets existing weighted approaches of subspace clustering in the typicality framework and it proposes a new typicality based subspace clustering algorithm, named TbSC. It illustrates their differences experimentally using simple 2D data sets.

The paper is organised as follows: Section II reminds the key notions used in the paper, considering in turn subspace clustering and typicality degrees. Section III motivates the proposed approach by interpreting some existing subspace clustering approaches in the framework of typicality degrees. It then discusses the notion of typicality degrees attached to attributes and presents the proposed TbSC algorithm. Section IV describes the experimental study conducted to examine the properties of the proposed approach. Section V draws the conclusion and discusses directions for future work.

II. Reminder on the Key Notions

This section provides some required basics, regarding subspace clustering first, typicality degrees and prototypes then.

A. Subspace Clustering

1) Overview: As mentioned in the introduction, subspace clustering [3], [26], [18] refines the clustering task by considering that each cluster can be associated with, or live in, its own subspace of the feature space. As illustrated in Section IV and in particular on Fig. 1, a cluster following a Gaussian distribution may for instance not be spherical, but it may live in a subspace of the whole space characterised by the small values of its covariance matrix. Among the numerous approaches that have been proposed to perform subspace clustering, two main paradigms can be distinguished : they respectively (i) exploit the hypothesis that the clusters live in different subspaces to identify them better vs (ii) explicitly identify the subspaces to extract more information from the data and characterise the clusters.

The first paradigm can be exemplified by so-called self-expressive approaches that represent the data as linear com-

bination of other data points that must thus be in the same subspace. Then they minimise the reconstruction cost with various penalisation terms (see e.g. SSC [8], [9] or LRR [21]). Deep learning methods have been proposed to extend this principle to the case of non-linear subspaces [22], [14], [27].

Methods in this first paradigm view the subspace existence as a useful intermediary tool for the clustering aim, but their identification is not a goal in itself and they are not further used. On the other hand, methods in the second paradigm consider that the subspaces provide valuable information in themselves and they aim at identifying them explicitly.

They can be further organised into three categories: bottom-up strategies [3], [6], [5] start from atomic clusters with high density and very low dimensionality that are then iteratively fused to build more complex clusters and subspaces using A PRIORI like iterative fusion. Projected clustering approaches [1], [2] apply a top-down strategy that progressively refines subspaces initially defined as the whole feature space. Finally partitioning strategies optimise a cost function that extends the classical k-means cost function so as to integrate the desired subspaces associated with the clusters. As they are the basis for the typicality based approach to subspace clustering studied in this paper, they are described in more details in the next subsection.

2) Cost Function-based Subspace Clustering: The common point to subspace partitioning strategies is to replace the standard Euclidean distance usually applied for clustering by weighted variants thereof: the weights are associated with each cluster, leading to local definitions of the distance function. These weights then define the subspaces associated with each cluster.

The mathematical notations used in the paper are the classic following ones: n denotes the number of data points and d the number of features, so that the data are denoted, for any $i = 1..n$, $x_i = (x_{ip})_{p=1..d} \in \mathbb{R}^d$. The desired number of clusters is denoted c, the cluster centres $c_r = (c_{rp})_{p=1..d} \in \mathbb{R}^d$ for $r = 1..c$. The assignment of data x_i to cluster r is denoted u_{ri}, with $u_{ri} \in \{0, 1\}$ or $u_{ri} \in [0, 1]$ for crisp and fuzzy assignments respectively. Finally, the feature weights for cluster r are denoted $w_r = (w_{rp})_{p=1..d}$. They are usually normalised in $[0, 1]$.

A reference cost-function subspace clustering algorithm is AWFCM, the Attribute Weighted Fuzzy c-Means [16] that generalises the fuzzy c-means (*fcm*) cost function, considering locally weighted Euclidean distances:

$$J_{AWFCM}(U, C, W) = \sum_{i=1}^{n} \sum_{r=1}^{c} u_{ri}^m \sum_{p=1}^{d} w_{rp}^q (x_{ip} - c_{rp})^2 \quad (1)$$

under the constraints $\sum_{r=1}^{c} u_{ri} = 1$ for all i, $\sum_{i=1}^{n} u_{ri} > 0$ for all r and $\sum_{p=1}^{d} w_{rp} = \alpha_r$ for all r. The third constraint, new as compared to the *fcm* ones, forbids the trivial solution $\forall r, p, w_{rp} = 0$. The algorithm hyperparameters are m and q, usually set to 2, and the α_r, usually set to 1. Other values for these (α_r) hyperparameters allow to weight the relative importance of the c clusters in the final partition. This cost function is optimised using a classic alternated scheme, that iteratively updates the cluster centres, membership degrees and cluster feature weights. The update equations for the cluster centres and the membership degrees are identical to that of *fcm*, replacing the Euclidean distance by the local weighted variants when comparing a point to a cluster centre. Regarding the attribute weights, for the classical choice $q = 2$, the update equation is

$$w_{rp} = \frac{s_{rp}^{-2}}{\sum_{q=1}^{d} s_{rq}^{-2}} \text{ with } s_{rp}^2 = \sum_{i=1}^{n} u_{ri}^m (x_{ip} - c_{rp})^2 \quad (2)$$

Due to the considered feature weighted Euclidean distance, the cluster subspaces are defined as axes-parallel hyperplanes.

Many variants have later been proposed: to name a few in decreasing order of their similarity to AWFCM, the PFCM algorithm [11] proposes an inline version of the constraint with an original optimisation process, based on proximal descent. Sparsity constraints on the w_{rp} weights can be introduced [4], [12], as well as entropy terms [15] in a crisp and not fuzzy assignment setting (i.e. imposing $u_{ri} \in \{0, 1\}$). In particular, it has been proposed [4] to apply to the subspace coefficients w_{rq} the approach used to make the membership degrees u_{ri} sparse: considering w_{rq}^q with the usual choice $q = 2$ actually only leads to values in $]0, 1[$, but does not allow for 0 values. Therefore it is proposed to replace w_{rq}^q with $g(w_{rq})$ where $g(x) = \frac{1-\beta}{1+\beta} x^2 + \frac{2\beta}{1+\beta} x$: the combination of a term of degree 1 with the quadratic term guides the weight optimisation in the direction of binary, and thus sparser, degrees.

It should be noted that, although proposed in a different framework, the Gaussian Mixture Model (GMM) can be interpreted as subspace clustering insofar as it introduces, for each cluster, a local Mahalanobis distance: the covariance matrix attached to each cluster defines an ellipsoidal subspace. GMM optimises the data log-likelihood, in a probabilistic approach to clustering. The Gustafson-Kessel algorithm (see e.g. [13]) also relies on local Mahalanobis distance, but is more similar to the algorithms mentioned above insofar as it also optimises the quantisation error cost, and not the log-likelihood. Both GMM and Gustafson-Kessel provide rich subspaces, beyond axes-parallel hyperplaces, that can adapt to the data complexity, at the expense of a high computational cost.

B. Prototypes and Typicality Degrees

Cognitive psychology [25] has shown that category representative members are instances that share both many features with the other members of the category and few features with the members of other categories. This well-studied principle has been transposed to a computational definition [24], [20] so as to make it possible to characterise and summarise data sets, classes or clusters, taking into account both common and discriminative features: using these two principles simultaneously makes it possible to build context-dependent representatives that are more relevant than classic representatives, such as the

average or the median for instance, that actually only depend on common features.

In the machine learning view, the construction of a prototype consists in three steps [24]: first internal resemblance and external dissimilarity are computed, respectively comparing an instance with the other members of its category and with the members of the other categories. These two quantities are then aggregated into the typicality degree of the considered instance. Finally, the most typical instances are in turn aggregated into the prototype. These definitions have been implemented to compute prototypes of individual values or of data instances [24], [20].

The definitions of internal resemblance and external dissimilarity obviously depend on the definition of resemblance and dissimilarity measures, respectively denoted r and d to compare the instances. They are formally defined as

$$\begin{aligned} R(x, C) &= avg(r(x, y), y \in C) \\ D(x, C) &= avg(d(x, y), y \notin C) \end{aligned}$$

The typicality degree is then defined as the aggregation of these two quantities, the choice of the aggregation operator determining the prototype semantics and in particular the extent to which they should be central or discriminative elements. Common choices include conjunctive or trade-off operators offering a compensation property as well as operators with variable behaviours, such as the symmetric sum that is conjunctive, disjunctive or trade-off depending on the values to be aggregated and offers a full reinforcement property (see e.g. [7]). The last step builds the prototype itself, aggregating the most typical objects for instance in a representative fuzzy prototype [24], [20].

The notions of typicality degrees and prototypes have been exploited for machine learning, both in supervised [23] and unsupervised settings [17]. In the latter case, the underlying principle consists in assigning points to data subsets so that they have high typicality degrees for the subset they are assigned to. Typicality degrees are iteratively computed for each data point with respect to the tentative clusters.

III. Proposed Approach

This section explores, at a theoretical level, the relevance of applying the principles of typicality degrees to the subspace clustering task: it relies on the intuition that characteristic features, as looked for when performing subspace clustering, can be interpreted as typical attributes. Nevertheless, typicality degrees are defined for feature values or instances, but not for features themselves: the section discusses a possible definition of the latter and then proposes a heuristic algorithm, called TbSC, to perform typicality-based subspace clustering.

A. Typicality Analysis of Subspace Weights

This section proposes to interpret the weights used in cost-based subspace clustering approaches, recalled in Section II-A, and in particular AWFCM, in the framework of typicality degrees, recalled in Section II-B: the AWFCM cluster weight update equation (see Eq. (2)) can be interpreted as a feature internal resemblance, computed relatively to the other features.

Indeed, the s_{rp}^2 component computes the weighted average squared distance to the cluster centre across all data, using membership degrees as weights. It thus defines a weighted cluster dispersion measured on feature p and, as a consequence, its inverse captures a notion of internal resemblance of cluster r on feature p. Its being based on the cluster centre reduces the computational complexity as compared to a measure that would rely on the distances between all pairs of points. The latter would, in addition, raise the question of the membership degree aggregation, as two memberships should be taken into account and combined.

The last step for computing w_{rp} normalises the $1/s_{sp}^2$ coefficients in $[0, 1]$, so as to satisfy the constraint that the sum of w_{rp} weights equals 1 for each cluster across all features: it compares the dispersion inverse over feature p to the dispersion inverse over the other features.

It can be shown that other subspace clustering algorithms based on cost function rely on similar principles: even if they may take into account additional components, for instance to favour sparse weights, the definition of the latter corresponds to an internal resemblance measure, defined relatively to the other features.

B. Motivation for TbSC

The consequence is that existing subspace weights do not take into account a notion of external dissimilarity: cluster subspaces are defined without considering the other clusters. Now it can be argued that it would be relevant to reward features that are specific of a cluster and exclude others, making features all the more important to describe a cluster as they do not apply to other ones.

It should be noted that such a typicality principle differs from the one applied in typicality based clustering [17]: the latter relies on a classical notion of typicality degrees, computing the extent to which an instance is typical of a (tentative) cluster, then assigning the instance to the most appropriate cluster and iterating the steps of typicality and assignment computations. Applying the notion of typicality degrees to subspace clustering requires to get down to the features and thus to define typicality degrees associated with features.

C. Feature Typicality Degrees

The aim of the feature typicality degree is to assess the extent to which the considered feature is characteristic of a cluster. One approach could thus be to compute the typicality degree of each encountered value for this attribute and to aggregate them using a relevant aggregation operators. Yet it appears that it is difficult to specify the desired behaviour of such an aggregation operator, depending on the multiple situations that can arise.

As existing subspace weight definitions, we propose to base the feature typicality on the value the cluster centre takes for this feature, so as to reduce the computational cost. This

section discusses in turn all decision levels for the definition of the feature typicality degrees.

1) Comparison Measures: The first step is to set the considered comparison measures, both for resemblance and dissimilarity. For feature typicality degrees, they simply apply to numerical values and not data instances, however numerous possibilities can still be considered, as sketched below.

First, regarding dissimilarity, it seems natural to define it as an increasing function of the absolute value of the difference of the compared values. We propose to consider a linear function, only normalising the difference so as to get an intuition to the scale of this difference value. As normalisation factor, we propose to consider the maximal distance observed among all data for attribute p, ie to define $Z_p = \max_i(x_{ip}) - \min_i(x_{ip})$ and

$$diss_p(x, y) = \frac{|x_p - y_p|}{Z_p} \tag{3}$$

This definition implies that a single couple is considered having a dissimilarity degree equal to 1, the one providing the observed maximum and minimum. Other possibilities would for instance be to decide of a fraction of this value from which two values can be considered as totally dissimilar, e.g. $Z_p/2$. In case a data expert can provide knowledge regarding a desired saturation effect (the difference value from which the dissimilarity must be considered 1), other values can be given to Z_p. Another line of variants would be to define the dissimilarity measure with a non-linear dependence on the difference, e.g. through a quadratic function, making values with high differences have an even higher dissimilarity.

Regarding the resemblance measure, we propose to define it as the complement to 1 of dissimilarity:

$$ress_p(x, y) = 1 - diss_p(x, y) \tag{4}$$

Here also, numerous other possibilities could be considered, most of them being derived from the choice of the dissimilarity, e.g. replacing the linear function by an exponential or a Cauchy function [19].

2) Internal Resemblance and External Dissimilarity: The comparison measures must then be aggregated for the whole feature, so as to compute the internal resemblance and external dissimilarity of the considered feature p for the considered cluster r, as opposed to the other clusters. Moreover, the fact that data assignment to the cluster is usually fuzzy must be taken into account.

As a consequence, we propose to define

$$R(r, p) = \frac{\sum_{i=1}^{n} u_{ri} ress_p(x_i, c_r)}{\sum_{i=1}^{n} u_{ri}} \tag{5}$$

$$D(r, p) = \frac{\sum_{i=1}^{n} (1 - u_{ri}) diss_p(x_i, c_r)}{\sum_{i=1}^{n} (1 - u_{ri})} \tag{6}$$

Internal resemblance is a weighted average of the resemblance between the data instances and the cluster centre, for the considered feature, defining weights as membership degrees: the more an instance belongs to the cluster, the more influence it has on the cluster internal resemblance.

As a comparison, it can be observed that the proposed interpretation of AWFCM weights in the typicality framework relies on a definition that differs in its principle: the cluster internal dispersion is first computed over all data instances, and then the inverse is computed to turn dispersion to a similarity semantics. In the proposed approach, individual similarity degrees are aggregated.

Regarding external dissimilarity, the question is to assess the extent to which the feature also applies to other clusters, and thus is not discriminative enough of the considered cluster. The above definition proposes to compare data instances mostly assigned to other clusters (as expressed by the complement to 1 of the membership degree) to the considered cluster centre.

3) Aggregation: The typicality degrees are then defined as the aggregation of internal resemblance and external dissimilarity, whose operator must be defined. We propose to use a variable behaviour one, namely the symmetric sum, because of the reinforcement property it proposes: in the case when both internal resemblance and external dissimilarity are high, their aggregation is even higher:

$$T(r, p) = \frac{R(r, p) D(r, p)}{R(r, p) D(r, p) + (1 - R(r, p))(1 - D(r, p))} \tag{7}$$

An experimental study on the effect of the chosen aggregation operator for the task of subspace clustering constitutes a perspective for future work.

4) Normalisation: To the best of our knowledge, all existing subspace clustering approaches impose that, for each cluster, the sum of the feature weights across all features equals 1. We thus propose to apply the same normalisation step and to define the weights of the Euclidean distance that will be used to assign the data instances to the clusters as relative typicality degrees. This choice seems in agreement with an intuition that features can be compared one with another to identify the significant ones, leading to relative feature importance:

$$w_{rp} = \frac{T(r, p)}{\sum_{q=1}^{d} T(r, q)} \tag{8}$$

As a consequence of these choices, the weight attached to a feature for a given cluster depends both on other clusters as well as on other features.

5) Comment: It can be observed that the previous decisions, that are commented and justified in each case, actually lead to define the typicality degree for feature p as the typicality degree of the c_{rp} value, in the typicality framework reminded in Section II-B: the proposed feature typicality equals the typicality degree of the cluster centre value for the considered feature, with the specificity that its takes into account fuzzy assignments, beyond crisp ones. This is not explicitely a predefined choice, it emerges from the choices made for each of the successive steps.

D. Proposed TbSC algorithm

The proposed typicality based subspace clustering algorithm instanciates the Alternating Cluster Estimation heuristic (see

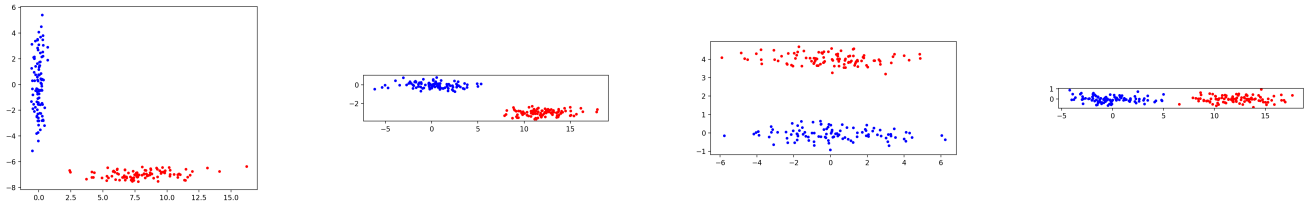

Fig. 1. Considered illustrative 2D data sets, from left to right $\mathcal{D}_1$, $\mathcal{D}_2$, $\mathcal{D}_3$, $\mathcal{D}_4$,

e.g. [13]) to identify the clusters and the subspaces in which they live: it applies the centre and assignment update equations of the AWFCM algorithm [16] and replaces its weight update equation given in Eq. (2) by the expression given in Eq. (8).

It must be underlined that the proposed approach does not offer convergence guarantee as it is based on a heuristic and it does not optimise a cost function.

IV. Preliminary Experimental Study and Discussion

This section describes some preliminary experiments conducted to examine the behaviour of the proposed TbSC heuristic.

A. Experimental Protocol

One of the difficulties of subspace clustering is experimental evaluation: as any unsupervised learning task, it does not dispose of consensual quality criteria that can be applied to compare methods one with another. Moreover, subspace clustering requires to assess both the identified clusters and their associated subspaces, including possible position and distorsion errors. In this exploratory paper, we perform visual inspection of the obtained results, both regarding the data assignment and the subspace weights.

As a consequence, the considered datasets are very simple 2D data, drawn following two Gaussian distributions that are axes parallel, in agreement with the principles of the approaches considered in the paper. Figure 1 shows the four considered configurations that correspond to different cases of relative positions of the Gaussians, respectively named $\mathcal{D}_1$, $\mathcal{D}_2$, $\mathcal{D}_3$ and $\mathcal{D}_4$ from left to right: $\mathcal{D}_1$ illustrates the principle of subspace clustering, $\mathcal{D}_2$, $\mathcal{D}_3$ and $\mathcal{D}_4$ correspond to three cases of two distinct clusters that however share the same subspace.

The experiments compare the results provided by the proposed TbSC heuristic, with the reference fuzzy subspace clustering algorithm AWFCM [16] whose cost function is reminded in Section II-A, and with the algorithm we denote BOR proposed by [4]. As recalled in Section II-A, BOR is an AWFCM variant that aims at identifying sparse representations of the cluster associated subspaces.

In all experiments, the desired number of clusters is set to the expected number of clusters, i.e. 2.

B. Obtained Results

For datasets $\mathcal{D}_1$, $\mathcal{D}_2$ and $\mathcal{D}_3$, all three algorithms output the expected clusters, the obtained partitions are thus not shown.

AWFCM	BOR	TbSC
$\begin{bmatrix} 0.016 & 0.984 \\ 0.986 & 0.014 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.475 & 0.525 \\ 0.529 & 0.471 \end{bmatrix}$
R	D	T
$\begin{bmatrix} 0.884 & 0.964 \\ 0.972 & 0.846 \end{bmatrix}$	$\begin{bmatrix} 0.475 & 0.522 \\ 0.466 & 0.533 \end{bmatrix}$	$\begin{bmatrix} 0.873 & 0.967 \\ 0.968 & 0.862 \end{bmatrix}$

TABLE I
Obtained weight matrices for dataset $\mathcal{D}_1$

For dataset $\mathcal{D}_4$, Fig. 2 shows the only unexpected result, obtained both with AWFCM and BOR. The weights obtained by the three compared approaches are given in Tables I to IV. For all tables, the first row shows the obtained w_{rp} matrices for the three algorithms, the second row gives the matrices corresponding to the internal ressemblance, external dissimilarity and typicality degrees for TbSC. For all matrices, rows correspond to clusters (first row: red cluster, second row: blue cluster) and columns to features (first column: x-dimension, second column: y-dimension).

For dataset $\mathcal{D}_1$, the three algorithms output, as expected, a value for weight w_{12} greater than the one for w_{11}, i.e. for the y-feature for the red cluster (that indeed corresponds to the horizontal one) and reciprocally a value for w_{21} greater than that for w_{22}. Indeed, the horizontal cluster lives in a subspace characterised by the y-feature values. However, differences can be observed: as expected, BOR achieves sparser weight values than AWFCM and TbSC, it outputs binary weights. TbSC has the least dissimilar weights, even if the same ranking as AWFCM and BOR is obtained for each cluster. This results comes from the fact that, with the chosen instantiation of typicality computation, and in particular the normalisation multiplicative coefficient used in the dissimilarity measure (see Eq. 3), the differences in internal similarity and external dissimilarity between the two features remain small and are further attenuated in the aggregation and normalisation of the typicality degree. This data set thus illustrates the importance of the definition of the typicality instantiation for subspace clustering.

For data set $\mathcal{D}_2$, a similar configuration is observed, except that the two clusters have the same characteristic subspace. The difference between the two features for TbSC appears to be even smaller than for $\mathcal{D}_1$. This can be explained by the global normalisation factor applied when defining internal similarity: by default, the reference value Z_p is set to the maximal observed distance, which is this case is smaller for the

AWFCM	BOR	TbSC
$\begin{bmatrix} 0.019 & 0.981 \\ 0.022 & 0.978 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.488 & 0.511 \\ 0.489 & 0.511 \end{bmatrix}$
R	D	T
$\begin{bmatrix} 0.922 & 0.927 \\ 0.921 & 0.923 \end{bmatrix}$	$\begin{bmatrix} 0.476 & 0.637 \\ 0.476 & 0.638 \end{bmatrix}$	$\begin{bmatrix} 0.915 & 0.957 \\ 0.914 & 0.954 \end{bmatrix}$

TABLE II
OBTAINED WEIGHT MATRICES FOR DATASET $\mathcal{D}_2$

AWFCM	BOR	TbSC
$\begin{bmatrix} 0.017 & 0.983 \\ 0.017 & 0.982 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.353 & 0.646 \\ 0.332 & 0.668 \end{bmatrix}$
R	D	T
$\begin{bmatrix} 0.859 & 0.914 \\ 0.847 & 0.914 \end{bmatrix}$	$\begin{bmatrix} 0.152 & 0.676 \\ 0.141 & 0.673 \end{bmatrix}$	$\begin{bmatrix} 0.523 & 0.957 \\ 0.476 & 0.956 \end{bmatrix}$

TABLE III
OBTAINED WEIGHT MATRICES FOR DATASET $\mathcal{D}_3$

AWFCM	BOR	TbSC
$\begin{bmatrix} 0.001 & 0.999 \\ 0.001 & 0.999 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.648 & 0.352 \\ 0.650 & 0.350 \end{bmatrix}$
R	D	T
$\begin{bmatrix} 0.912 & 0.868 \\ 0.903 & 0.865 \end{bmatrix}$	$\begin{bmatrix} 0.561 & 0.135 \\ 0.559 & 0.133 \end{bmatrix}$	$\begin{bmatrix} 0.930 & 0.505 \\ 0.922 & 0.497 \end{bmatrix}$

TABLE IV
OBTAINED WEIGHT MATRICES FOR DATASET $\mathcal{D}_4$

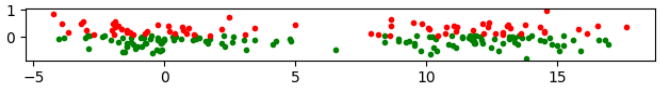

Fig. 2. Partition output by AWFCM and BOR for $\mathcal{D}_4$,

y-axis than for the x-one. As a consequence, the difference that can be observed for the two features is less marked for TbSC that actually rescales the axes, questioning the normalisation choice. The definition of the latter appears to be a crucial and difficult question.

Regarding data set $\mathcal{D}_3$, it must be underlined that all three algorithms appear to be sensitive to the random initialisation and sometimes converge to an undesired data assignment, where two left-right clusters are distinguished, instead of the expected top-bottom ones. We here focus on the case where they provide the desired partition. $\mathcal{D}_3$, is then handled similarly to data set $\mathcal{D}_2$ by AWFCM and BOR, whereas TbSC makes a difference between them: for $\mathcal{D}_3$, TbSC identifies the y-axis as being significantly more typical than the x-axis, for both clusters, whereas the difference between the two axes is smaller for data set $\mathcal{D}_2$. This is due to the low external dissimilarity of the x-axis, as the clusters approximately have the same distribution when projected on it. $\mathcal{D}_3$ also illustrates that the characterisation captured by typicality degrees differs from the one performed by AWFCM and BOR and allows to make fine distinction between cases where the clusters live in the same subspaces.

Dataset $\mathcal{D}_4$ illustrates a case where AWFCM and BOR do not identify the expected clusters, as shown on Figure 2, in a configuration similar to the one pointed out in [10]: AWFCM and BOR can be considered as over-enthusiastic in identifying subspaces, to the point that they oversee the gap between the two expected clusters and interpret them as being prolongation one of another. Consistently, they identify the y-axis as being totally characteristic for both clusters.

TbSC is the only algorithms among the three considered ones that successfully identifies the expected clusters. Regarding the subspace weights, it indicates that the most typical dimension for both clusters is the x-axis: this characterisation is consistent with the absence of external dissimilarity of the y-axis, which prevents it from being typical of the cluster data. Regarding internal similarity, due to the implemented normalisation, as discussed above, no big difference between the two axes is observed, and it does not counterbalance the external dissimilarity effect when both criteria are aggregated. These results illustrate that the considered definition of typicality degrees differs from the classical description of the cluster associated subspaces, i.e. the information captured by typicality degrees in the proposed implementation differs from the principle of characteristic subspaces in the sense of subspace clustering.

V. CONCLUSION AND FUTURE WORKS

This paper proposes a prospective study of the possibility opened by the notion of feature typicality degrees, as tools to characterise these features, for the task of subspace clustering. It proposes an implementation of the typicality principles for this specific case and experimentally examines the results it leads to on simple data sets. Although preliminary, the obtained results allow to conclude that, as can be expected, the concepts of subspaces characteristic-of vs typical-for clusters differ and possess different semantics. These differences require to be further studied, in particular to identify their respective relevance from possible application points of view.

Ongoing works aim at performing such thorough and systematic experiments to test more diverse cluster configurations and examine the effect of the parameters, in particular when varying the relative positions of the Gaussian distributions. Other instantiations of typicality degree for the case of feature typicality also constitute a perspective of these works: taking into account a notion of external dissimilarity in the weight computation for subspace clustering can be considered as a relevant principle, but its appropriate implementation remains a research question.

REFERENCES

[1] C. C. Aggarwal, J. L. Wolf, P. S. Yu, C. Procopiuc, and J. S. Park. Fast algorithms for projected clustering. In *Proc. of the Int. Conf. on Management of Data, SIGMOD*, pages 61–72. ACM, 1999.

[2] C. C. Aggarwal and P. S. Yu. Finding generalized projected clusters in high dimensional spaces. In *Proc. of the Int. Conf. on Management of Data*, SIGMOD'00, pages 70–81. ACM, 2000.

[3] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, pages 94–105, 1998.

[4] C. Borgelt. Fuzzy subspace clustering. In *Advances in Data Analysis, Data Handling and Business Intelligence*, Studies in Classification, Data Analysis, and Knowledge Organization, pages 93–103. Springer, 2010.

[5] D. Burdick, M. Calimlim, and J. Gehrke. MAFIA : A maximal frequent itemset algorithm for transactional databases. In *Proc. of the 17th Int. Conf. on Data Engineering*, pages 443–452, 2001.

[6] C.-H. Cheng, A. W. Fu, and Y. Zhang. Entropy-based subspace clustering for mining numerical data. In *Proc. of the 5th ACM Int. Conf. on Knowledge Discovery and Data Mining*, pages 84–93, 1999.

[7] M. Detyniecki. *Mathematical Aggregation Operators and their Application to Video Querying*. PhD thesis, University of Pierre and Marie Curie, 2000.

[8] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR*, pages 2790–2797, 2009.

[9] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35(11):2765–2781, 2013.

[10] A. Guillon, M.-J. Lesot, and C. Marsala. Laplacian regularization for fuzzy subspace clustering. In *Proc. of the IEEE Int. Conf. on Fuzzy Systems, FuzzIEEE17*, 2017.

[11] A. Guillon, M.-J. Lesot, and C. Marsala. A proximal framework for fuzzy subspace clustering. *Fuzzy Sets and Systems*, 366:24–45, 2019.

[12] A. Guillon, M.-J. Lesot, and C. Marsala. Sparsity-inducing fuzzy subspace clustering. *Archives of Data Science, series B*, in press, 2019.

[13] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis: Methods for Classification, Data Analysis and Image Recognition*. Wiley-Blackwell, 1999.

[14] P. Ji, T. Zhang, H. Li, M. Salzmann, and I. Reid. Deep subspace clustering networks. In *Proc. of the 31st Int. Conf. on Neural Information Processing Systems, NIPS*, 2017.

[15] L. Jing, M. K. Ng, and J. Z. Huang. An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data. *IEEE Trans. on Knowledge and Data Engineering*, 19(8), 2007.

[16] A. Keller and F. Klawonn. Fuzzy clustering with weighting of data variables. *Int. Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 8(6):735–746, 2000.

[17] M.-J. Lesot. Typicality-based clustering. *Int. Journ. of Information Technology and Intelligent Computing*, 51(1), 2006.

[18] M.-J. Lesot. Subspace clustering and some soft variants. In *Proc. of the Int. Conf. on Scalable Uncertainty Management, SUM'19*. Springer, 2019.

[19] M.-J. Lesot, M. Rifqi, and H. Benhadda. Similarity measures for binary and numerical data: a survey. *Int. J. of Knowledge Engineering and Soft Data Paradigms (KESDP)*, 1(1):63–84, 2009.

[20] M.-J. Lesot, M. Rifqi, and B. Bouchon-Meunier. Fuzzy prototypes: From a cognitive view to a machine learning principle. In H. Bustince, F. Herrera, and J. Montero, editors, *Fuzzy Sets and Their Extensions: Representation, Aggregation and Models*, pages 431–452. Springer, 2007.

[21] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35(1):171–184, 2013.

[22] X. Peng, S. Xiao, J. Feng, W.-Y. Yau, and Z. Yi. Deep subspace clustering with sparsity prior. In *Proc. of the 25th Int. Joint Conf. on Artificial Intelligence IJCAI*, pages 1925–1931, 2016.

[23] A. Rick, S. Bothorel, B. Bouchon-Meunier, S. Muller, and M. Rifqi. Fuzzy techniques in mammographic image processing. In E. Kerre and M. Nachtegael, editors, *Fuzzy Techniques in Image Processing*, Studies in Fuzziness and Soft Computing, pages 308–336. Springer, 2000.

[24] M. Rifqi. Constructing prototypes from large databases. In *Proc. of IPMU'96*, pages 301–306, 1996.

[25] E. Rosch. Principles of categorization. In E. Rosch and B. Lloyd, editors, *Cognition and categorization*, pages 27–48. Lawrence Erlbaum associates, 1978.

[26] R. Vidal. A tutorial on subspace clustering. *IEEE Signal Processing Magazine*, 28(2):52–68, 2010.

[27] L. Zhou, X. Bai, D. Wang, X. Liu, J. Zhou, and E. Hancock. Latent distribution preserving deep subspace clustering. In *Proc. of of the 28th Int. Joint Conf. on Artificial Intelligence IJCAI*, pages 4440–4446, 2019.