

Advancing Data Clustering via Projective Clustering Ensembles

Francesco Gullo
DEIS Dept.
University of Calabria
87036 Rende (CS), Italy
fgullo@deis.unical.it

Carlotta Domeniconi
Dept. of Computer Science
George Mason University
22030 Fairfax – VA, USA
carlotta@cs.gmu.edu

Andrea Tagarelli
DEIS Dept.
University of Calabria
87036 Rende (CS), Italy
tagarelli@deis.unical.it

ABSTRACT

Projective Clustering Ensembles (PCE) are a very recent advance in data clustering research which combines the two powerful tools of *clustering ensembles* and *projective clustering*. Specifically, PCE enables clustering ensemble methods to handle ensembles composed by projective clustering solutions. PCE has been formalized as an optimization problem with either a two-objective or a single-objective function. Two-objective PCE has shown to generally produce more accurate clustering results than its single-objective counterpart, although it can handle the object-based and feature-based cluster representations only independently of one other. Moreover, both the early formulations of PCE do not follow any of the standard approaches of clustering ensembles, namely instance-based, cluster-based, and hybrid.

In this paper, we propose an alternative formulation to the PCE problem which overcomes the above issues. We investigate the drawbacks of the early formulations of PCE and define a new single-objective formulation of the problem. This formulation is capable of treating the object- and feature-based cluster representations as a whole, essentially tying them in a distance computation between a projective clustering solution and a given ensemble. We propose two cluster-based algorithms for computing approximations to the proposed PCE formulation, which have the common merit of conforming to one of the standard approaches of clustering ensembles. Experiments on benchmark datasets have shown the significance of our PCE formulation, as both the proposed heuristics outperform existing PCE methods.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*clustering*; I.2.6 [Artificial Intelligence]: Learning; I.5.3 [Pattern Recognition]: Clustering

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGMOD'11, June 12–16, 2011, Athens, Greece.

Copyright 2011 ACM 978-1-4503-0661-4/11/06 ...\$10.00.

General Terms

Algorithms, Theory, Experimentation

Keywords

Data Mining, Clustering, Clustering Ensembles, Projective Clustering, Subspace Clustering, Dimensionality reduction, Optimization

1. INTRODUCTION

Given a set of data objects as points in a multi-dimensional space, *clustering* aims to detect a number of homogeneous, well-separated subsets (clusters) of data, in an unsupervised way [18]. After more than four decades, a considerable corpus of methods and algorithms has been developed for data clustering, focusing on different aspects such as data types, algorithmic features, and application targets [14]. In the last few years, there has been an increased interest in developing advanced tools for data clustering. In this respect, *clustering ensembles* and *projective clustering* represent two of the most important directions of research. Clustering ensemble methods [28, 13, 36, 29, 17] aim to extract a “consensus” clustering from a set (ensemble) of clustering solutions. The input ensemble is typically generated by varying one or more aspects of the clustering process, such as the clustering algorithm, the parameter setting, and the number of features, objects or clusters. The output consensus clustering is usually obtained using *instance-based*, *cluster-based*, or *hybrid* methods. Instance-based methods require a notion of distance measure to directly compare the data objects in the ensemble solutions; cluster-based methods exploit a meta-clustering approach; and hybrid methods attempt to combine the first two approaches based on hybrid bipartite graph clustering.

Projective clustering [32, 35, 30, 34] aims to discover clusters that correspond to subsets of the input data and have different (possibly overlapping) dimensional subspaces associated with them. Projected clusters tend to be less noisy—because each group of data is represented in a subspace that does not contain irrelevant dimensions—and more understandable—because the exploration of a cluster is easier when few dimensions are involved.

Clustering ensembles and projective clustering hence address two major issues in data clustering distinctly: projective clustering deals with the high-dimensionality of data, whereas clustering ensembles handle the lack of a-priori knowledge on clustering targets. The first issue arises due to the sparsity that naturally occurs in data representation.

As such, it is unlikely that all features are equally relevant to form meaningful clusters. The second issue is related to the fact that there are usually many aspects that characterize the targets of a clustering task; however, due to the algorithmic peculiarities of any particular clustering method, a single clustering solution may not be able to capture all facets of a given clustering problem.

In [16], projective clustering and clustering ensembles are treated for the first time in a unified framework. The underlying motivation of that study is that the high-dimensionality and the lack of a-priori knowledge problems usually co-exist in real-world applications. To address both issues simultaneously, [16] hence formalizes the problem of *projective clustering ensembles* (PCE): the objective is to define methods that, by exploiting the information provided by an ensemble of projective clustering solutions, are able to compute a robust *projective consensus clustering*.

PCE is formulated as an optimization problem, hence the sought projective consensus clustering is computed as a solution to that problem. Specifically, two formulations of PCE have been proposed in [16], namely *two-objective* PCE and *single-objective* PCE. The two-objective PCE formulation consists in the simultaneous optimization of two objective functions, which separately consider the data object clustering and the feature-to-cluster assignment. A well-founded heuristic developed for this formulation of PCE (called *MOEA-PCE*) has been found to be particularly accurate, although it has drawbacks concerning efficiency, parameter setting, and interpretability of results. By contrast, the single-objective PCE formulation embeds in one objective function the object-based and feature-based representations of candidate clusters. Apart from being a weaker formulation than two-objective PCE, the developed heuristic for single-objective PCE (called *EM-PCE*) is outperformed by two-objective PCE in terms of effectiveness, while showing more efficiency.

Both the early formulations of PCE have their own drawbacks and advantages, however none of them refers to any of the common approaches of clustering ensembles, i.e., the aforementioned instance-based, cluster-based, and hybrid approaches. This may limit the versatility of such early formulations of PCE and, eventually, their comparability with existing ways of solving clustering ensemble problems at least in terms of experience gained in some real-world scenarios. Besides this common shortcoming, an even more serious weakness concerns the inability of the two-objective PCE of treating the object-based and feature-based cluster representations as interrelated. This fact in principle may lead to projective consensus clustering solutions that contain conceptual flaws in their cluster composition.

In this work, we face all the above issues revisiting the PCE problem. For this purpose, we pursue a different approach to the study of PCE, focusing on the development of methods that are closer to the standard clustering ensemble methods. By providing an insight into the theoretical foundations of the early two-objective PCE formulation, we show its weaknesses and propose a new single-objective formulation of PCE. The key idea underlying our proposal is to define a function that measures the distance of any projective clustering solution from a given ensemble, where the object-based and feature-based cluster representations are considered as a whole. The new formulation enables the development of heuristic algorithms that are easy to define

and, at the same time, are well-founded as they can exploit a corpus of research results obtained by the majority of existing clustering ensemble methods. Particularly, we investigate the opportunity of adapting each of the various approaches of clustering ensembles to the new PCE problem. We define two heuristics that follow a cluster-based approach, namely *Cluster-Based Projective Clustering Ensembles* (CB-PCE) and a step-forward version called *Fast Cluster-Based Projective Clustering Ensembles* (FCB-PCE). We show not only the suitability of the proposed heuristics to the PCE context but also their advantages in terms of computational complexity w.r.t. the early formulations of PCE. Moreover, based on an extensive experimental evaluation, we assessed effectiveness and efficiency of the proposed algorithms, and found that both outperform the early PCE methods in terms of accuracy of projective consensus clustering. In addition, FCB-PCE reveals to be faster than the early two-objective PCE and comparable or even faster than the early single-objective PCE in the online phase.

The rest of the paper is organized as follows. Section 2 provides background on clustering ensembles, projective clustering, and the PCE problem. Section 3 describes our new formulation of PCE and presents the two developed heuristics along with an analysis of their computational complexities. Section 4 contains experimental evaluation and results. Finally, Section 5 concludes the paper.

2. BACKGROUND

2.1 Clustering Ensembles (CE)

Given a set \mathcal{D} of data objects, a *clustering solution* defined over \mathcal{D} is a partition of \mathcal{D} into a number of groups, i.e., *clusters*. A set of clustering solutions defined over the same set \mathcal{D} of data objects is called *ensemble*. Given an ensemble defined over \mathcal{D} , the goal of CE is to derive a *consensus clustering*, which is a (new) partition of \mathcal{D} derived by suitably exploiting the information available from the ensemble.

The earliest CE methods aim to explicitly solve the *label correspondence problem* to find a correspondence between the cluster labels across the clusterings of the ensemble [10, 11, 12]. These approaches typically suffer from efficiency issues. More refined methods fall into *instance-based*, *cluster-based*, and *hybrid* categories.

2.1.1 Instance-based CE

Instance-based CE methods perform a direct comparison between data objects. Typically, instance-based methods operate on the *co-occurrence* or *co-association* matrix W , which resembles the pairwise object similarities according to the information available from the ensemble. For each pair of objects (\bar{o}', \bar{o}'') , the matrix W stores the number of solutions of the ensemble in which \bar{o}' and \bar{o}'' are assigned to the same cluster divided by the size of the ensemble. Instance-based methods derive the final consensus clustering by applying one of the following strategies: (i) performing an additional clustering step based on W , using this matrix either as a new data matrix [20], or as a pairwise similarity matrix involved in a specific clustering algorithm [13, 22, 15]; (ii) constructing a weighted graph based on W and partitioning the graph according to well-established graph-partitioning algorithms [28, 3, 29].

2.1.2 Cluster-based CE

Cluster-based CE lies on the principle “to cluster clusters” [7, 28, 6]. The key idea is to apply a clustering algorithm to the set of clusters that belong to the clustering solutions in the ensemble, in order to compute a set of *meta-clusters* (i.e., sets of clusters). The consensus clustering is finally computed by assigning each data object to the meta-cluster that maximizes a specific criterion, such as the commonly used *majority voting*, which assigns each data object \bar{o} to the metacluster that contains the maximum number of clusters which \bar{o} belongs to.

2.1.3 Hybrid CE

Hybrid CE methods combine ideas from instance-based and cluster-based approaches. The objective is to build a *hybrid bipartite graph* whose vertices belong to the set of data objects and the set of clusters. For each object \bar{o} and cluster C , the edge (\bar{o}, C) of the bipartite graph usually assumes a unit weight, if the object \bar{o} belongs to the cluster C according to the clustering solution that includes C , and zero otherwise [36]. Some methods use weights in the range $[0, 1]$, which express the probability that object \bar{o} belongs to cluster C [29]. The consensus clustering of hybrid CE methods is obtained by partitioning the bipartite graph according to well-established methods (e.g., METIS [19]). The nodes representing clusters are filtered out from the graph partition.

2.2 Projective Clustering (PC)

Let \mathcal{D} be a set of data objects, where each $\bar{o} \in \mathcal{D}$ is defined on a feature space $\mathcal{F} = \{1, \dots, |\mathcal{F}|\}$. A *projective cluster* C defined over \mathcal{D} is a pair $\langle \Gamma_C, \Delta_C \rangle$ such that

- Γ_C denotes the *object-based* representation of C . It is a $|\mathcal{D}|$ -dimensional real-valued vector whose component $\Gamma_{C,\bar{o}} \in [0, 1]$, $\forall \bar{o} \in \mathcal{D}$, represents the *object-to-cluster* assignment of \bar{o} to C , i.e., the probability $\Pr(C|\bar{o})$ that object \bar{o} belongs to C ;
- Δ_C denotes the *feature-based* representation of C . It is a $|\mathcal{F}|$ -dimensional real-valued vector whose component $\Delta_{C,f} \in [0, 1]$, $\forall f \in \mathcal{F}$, represents the *feature-to-cluster* assignment of the feature f to C , i.e., the probability $\Pr(f|C)$ that feature f belongs to the subspace of features associated with C .

Note that the above definition addresses all possible types of projective clusters handled by existing PC algorithms. In fact, both *soft* and *hard* object-to-cluster assignments are taken into account—the assignment is hard when $\Gamma_{C,\bar{o}} \in \{0, 1\}$ rather than $[0, 1]$, $\forall \bar{o} \in \mathcal{D}$. Similarly, feature-to-cluster assignments may be equally-weighted, i.e., $\Delta_{C,f} = 1/R$ (where R is the number of relevant features for C), if f is recognized as relevant, $\Delta_{C,f} = 0$ otherwise. This representation is suited for dealing with the output of all those PC algorithms which only select the relevant features for each cluster, without specifying any feature-to-cluster assignment probability distribution. Such algorithms fall into *bottom-up* [34, 25], *top-down* [32, 31, 2, 37, 5], and *hybrid* approaches [24, 35, 1]. On the other hand, the methods defined in [34, 8, 30] handle projective clusters having soft object-to-cluster assignment and/or feature-to-cluster assignment unequally weighted.

The object-based (Γ_C) and the feature-based (Δ_C) representations of any projective cluster C are exploited to define

the *projective cluster representation matrix* (for brevity, *projective matrix*) X_C of C . X_C is a $|\mathcal{D}| \times |\mathcal{F}|$ matrix that stores, $\forall \bar{o} \in \mathcal{D}$, $f \in \mathcal{F}$, the probability of the intersection of the events “object \bar{o} belongs to C ” and “feature f belongs to the subspace associated with C ”. Under the assumption of independence between the two events, such a probability is equal to the product of $\Pr(C|\bar{o}) = \Gamma_{C,\bar{o}}$ with $\Pr(f|C) = \Delta_{C,f}$. Hence, given $\mathcal{D} = \{\bar{o}_1, \dots, \bar{o}_{|\mathcal{D}|}\}$ and $\mathcal{F} = \{1, \dots, |\mathcal{F}|\}$, matrix X_C can be formally defined as:

$$X_C = \begin{pmatrix} \Gamma_{C,\bar{o}_1} \times \Delta_{C,1} & \dots & \Gamma_{C,\bar{o}_1} \times \Delta_{C,|\mathcal{F}|} \\ \vdots & & \vdots \\ \Gamma_{C,\bar{o}_{|\mathcal{D}|}} \times \Delta_{C,1} & \dots & \Gamma_{C,\bar{o}_{|\mathcal{D}|}} \times \Delta_{C,|\mathcal{F}|} \end{pmatrix} \quad (1)$$

The goal of a PC method is to derive from an input set \mathcal{D} of data objects a *projective clustering solution* denoted by \mathcal{C} , which is defined as a set of projective clusters that satisfy the following conditions:

$$\sum_{C \in \mathcal{C}} \Gamma_{C,\bar{o}} = 1, \forall \bar{o} \in \mathcal{D} \quad \text{and} \quad \sum_{f \in \mathcal{F}} \Delta_{C,f} = 1, \forall C \in \mathcal{C}$$

The semantics of any projective clustering \mathcal{C} is that for each projective cluster $C \in \mathcal{C}$, the objects belonging to C are actually close to each other if (and only if) they are projected onto the subspace associated with C .

2.3 Projective Clustering Ensembles (PCE)

A *projective ensemble* \mathcal{E} is defined as a set of projective clustering solutions. No information about the ensemble generation strategy (algorithms and/or setups), nor original feature values of the objects within \mathcal{D} are provided along with \mathcal{E} . Moreover, each projective clustering solution in \mathcal{E} may contain in general a different number of clusters.

The goal of PCE is to derive a *projective consensus clustering* by exploiting information on the projective solutions within the input projective ensemble.

2.3.1 Two-objective PCE

In [16], PCE is formulated as a two-objective optimization problem, whose objectives take into account the object-based (function Ψ_o) and the feature-based (function Ψ_f) cluster representations of a given projective ensemble \mathcal{E} :

$$\mathcal{C}^* = \arg \min_{C \in \mathcal{E}} \{ \Psi_o(C, \mathcal{E}), \Psi_f(C, \mathcal{E}) \} \quad (2)$$

where

$$\Psi_o(C, \mathcal{E}) = \sum_{\hat{C} \in \mathcal{E}} \bar{\psi}_o(C, \hat{C}), \quad \Psi_f(C, \mathcal{E}) = \sum_{\hat{C} \in \mathcal{E}} \bar{\psi}_f(C, \hat{C}) \quad (3)$$

Functions $\bar{\psi}_o$ and $\bar{\psi}_f$ are defined as $\bar{\psi}_o(C', C'') = (\psi_o(C', C'') + \psi_o(C'', C'))/2$ and $\bar{\psi}_f(C', C'') = (\psi_f(C', C'') + \psi_f(C'', C'))/2$, respectively, where

$$\psi_o(C', C'') = \frac{1}{|C'|} \sum_{C'' \in C'} \left(1 - \max_{C'' \in C''} J(\Gamma_{C'}, \Gamma_{C''}) \right)$$

$$\psi_f(C', C'') = \frac{1}{|C'|} \sum_{C'' \in C'} \left(1 - \max_{C'' \in C''} J(\Delta_{C'}, \Delta_{C''}) \right)$$

$J(\vec{u}, \vec{v}) = (\vec{u} \cdot \vec{v}) / (\|\vec{u}\|_2^2 + \|\vec{v}\|_2^2 - \vec{u} \cdot \vec{v}) \in [0, 1]$ denotes the extended Jaccard similarity coefficient (also known as Tanimoto coefficient) between any two real-valued vectors \vec{u} and \vec{v} [26].

The problem defined in (2) is solved by a well-founded heuristic, in which a *Pareto-based Multi-Objective Evolutionary Algorithm*, called *MOEA-PCE*, is used to avoid combining the two objective functions into a single one.

2.3.2 Single-objective PCE

To overcome some issues of the two-objective PCE formulation (such as those concerning efficiency, parameter setting, and interpretation of the results), [16] proposes an alternative PCE formulation based on a single-objective function, which aims to consider the object-based and the feature-based cluster representations in \mathcal{E} as a whole:

$$\mathcal{C}^* = \arg \min_{\mathcal{C} \in \mathcal{E}} \sum_{\mathcal{C} \in \mathcal{C}} \sum_{\bar{\sigma} \in \mathcal{D}} \Gamma_{\mathcal{C}, \bar{\sigma}}^\alpha \sum_{\hat{\mathcal{C}} \in \mathcal{E}} \sum_{\hat{\sigma} \in \mathcal{D}} \Gamma_{\hat{\mathcal{C}}, \hat{\sigma}} \sum_{f \in \mathcal{F}} \left(\Delta_{\mathcal{C}, f} - \Delta_{\hat{\mathcal{C}}, f} \right)^2$$

where $\alpha > 1$ is a positive integer that ensures non-linearity of the objective function w.r.t. $\Gamma_{\mathcal{C}, \bar{\sigma}}$.

To solve the above problem, the *EM-based Projective Clustering Ensembles (EM-PCE)* heuristic is defined. EM-PCE iteratively looks for the optimal values of $\Gamma_{\mathcal{C}, \bar{\sigma}}$ (resp. $\Delta_{\mathcal{C}, f}$) while keeping $\Delta_{\mathcal{C}, f}$ (resp. $\Gamma_{\mathcal{C}, \bar{\sigma}}$) fixed, until convergence.

3. CLUSTER-BASED PCE

3.1 Problem Statement

Experimental results have shown that the two-objective PCE formulation is much more accurate than the single-objective counterpart [16]. Nevertheless, two-objective PCE suffers from an important conceptual issue that has not been discussed in [16], proving that the accuracy of two-objective PCE can be further improved. We unveil this issue in the following example.

Example: Let \mathcal{E} be a projective ensemble defined over a set \mathcal{D} of data objects and a set \mathcal{F} of features. Suppose that \mathcal{E} contains only one projective clustering solution \mathcal{C} and that \mathcal{C} in turn contains two projective clusters \mathcal{C}' and \mathcal{C}'' , whose object- and feature-based representations are different from one another, i.e., $\exists \bar{\sigma} \in \mathcal{D}$ s.t. $\Gamma_{\mathcal{C}', \bar{\sigma}} \neq \Gamma_{\mathcal{C}'', \bar{\sigma}}$, and $\exists f \in \mathcal{F}$ s.t. $\Delta_{\mathcal{C}', f} \neq \Delta_{\mathcal{C}'', f}$.

Let us consider two candidate projective consensus clusterings $\mathcal{C}_1 = \{\mathcal{C}'_1, \mathcal{C}''_1\}$ and $\mathcal{C}_2 = \{\mathcal{C}'_2, \mathcal{C}''_2\}$. We assume that $\mathcal{C}_1 = \mathcal{C}$, whereas \mathcal{C}_2 is defined as follows. Cluster \mathcal{C}'_2 has object- and feature-based representations given by $\Gamma_{\mathcal{C}'}$ (i.e., the object-based representation of the first cluster \mathcal{C}' within \mathcal{C}) and $\Delta_{\mathcal{C}''}$ (i.e., the feature-based representation of the second cluster \mathcal{C}'' within \mathcal{C}), respectively; cluster \mathcal{C}''_2 has object- and feature-based representations given by $\Gamma_{\mathcal{C}''}$ (i.e., the object-based representation of the second cluster \mathcal{C}'' within \mathcal{C}) and $\Delta_{\mathcal{C}'}$ (i.e., the feature-based representation of the first cluster \mathcal{C}' within \mathcal{C}), respectively. According to (3), it is easy to see that:

$$\Psi_o(\mathcal{C}_1, \mathcal{E}) = \Psi_o(\mathcal{C}_2, \mathcal{E}) = 0 \quad \text{and} \quad \Psi_f(\mathcal{C}_1, \mathcal{E}) = \Psi_f(\mathcal{C}_2, \mathcal{E}) = 0$$

Both the candidates \mathcal{C}_1 and \mathcal{C}_2 minimize the objectives of the early two-objective PCE formulation reported in (2), and hence, they are both recognized as optimal solutions. This conclusion is conceptually wrong, because only \mathcal{C}_1 should be recognized as an optimal solution, since only \mathcal{C}_1 is exactly equal to the unique solution of the ensemble. Conversely, \mathcal{C}_2 is not well-representative of the ensemble \mathcal{E} , as the object- and feature-based representations of its clusters are inversely associated to each other w.r.t. the associations present in

\mathcal{C} . Indeed, in \mathcal{C}_2 , $\mathcal{C}'_1 = \langle \Gamma_{\mathcal{C}'}, \Delta_{\mathcal{C}''} \rangle$ and $\mathcal{C}''_1 = \langle \Gamma_{\mathcal{C}''}, \Delta_{\mathcal{C}'} \rangle$, whereas, the solution $\mathcal{C} \in \mathcal{E}$ is such that $\mathcal{C}' = \langle \Gamma_{\mathcal{C}'}, \Delta_{\mathcal{C}'} \rangle$ and $\mathcal{C}'' = \langle \Gamma_{\mathcal{C}''}, \Delta_{\mathcal{C}''} \rangle$.

The issue described in the above Example arises because the two-objective PCE formulation ignores that the object-based and feature-based representations of any projective cluster are strictly coupled to each other, and hence, need to be considered as a whole. In other words, in order to effectively evaluate the quality of a candidate projective consensus clustering, the objective functions Ψ_o and Ψ_f cannot be kept separated from each other.

We attempt to solve the above drawback by proposing the following alternative formulation of PCE, which is based on a single objective function:

$$\mathcal{C}^* = \arg \min_{\mathcal{C} \in \mathcal{E}} \Psi_{of}(\mathcal{C}, \mathcal{E}) \quad (4)$$

where Ψ_{of} is a function designed to measure the “distance” of any well-defined projective clustering solution \mathcal{C} from \mathcal{E} in terms of both data clustering and feature-to-cluster assignment. To carefully take into account efficiency, we define Ψ_{of} based on an asymmetric function, which has been derived by adapting the measure defined in [16] to our setting:

$$\Psi_{of}(\mathcal{C}, \mathcal{E}) = \sum_{\hat{\mathcal{C}} \in \mathcal{E}} \bar{\psi}_{of}(\mathcal{C}, \hat{\mathcal{C}}) \quad (5)$$

where

$$\bar{\psi}_{of}(\mathcal{C}', \mathcal{C}'') = \frac{1}{2} \left(\psi_{of}(\mathcal{C}', \mathcal{C}'') + \psi_{of}(\mathcal{C}'', \mathcal{C}') \right) \quad (6)$$

and

$$\psi_{of}(\mathcal{C}', \mathcal{C}'') = \frac{1}{|\mathcal{C}'|} \sum_{\mathcal{C}' \in \mathcal{C}'} \left(1 - \max_{\mathcal{C}'' \in \mathcal{C}''} \hat{J}(X_{\mathcal{C}'}, X_{\mathcal{C}''}) \right) \quad (7)$$

In (7), the similarity between any pair $\mathcal{C}', \mathcal{C}''$ of projective clusters is computed in terms of their corresponding projective matrices $X_{\mathcal{C}'}$ and $X_{\mathcal{C}''}$ (cf. (1), Sect. 2.2). For this purpose, the Tanimoto similarity coefficient can easily be generalized to operate on real-valued matrices (rather than vectors):

$$\hat{J}(X, \hat{X}) = \frac{\sum_{i=1}^{|\text{rows}(X)|} X_i \cdot \hat{X}_i}{\|X\|_2^2 + \|\hat{X}\|_2^2 - \sum_{i=1}^{|\text{rows}(X)|} X_i \cdot \hat{X}_i} \quad (8)$$

where $X_i \cdot \hat{X}_i$ denotes the scalar product between the i -th rows of matrices X and \hat{X} . >From a dissimilarity viewpoint, as $\hat{J} \in [0, 1]$, we adopt in this work the measure $1 - \hat{J}$. We hereinafter refer to $1 - \hat{J}$ as *Tanimoto distance*.

It can be noted that the proposed formulation based on the function Ψ_{of} fulfils the requirement of measuring the quality of a candidate consensus clustering in terms of both data clustering and feature-to-cluster assignments as a whole. In particular, we remark that the issue described in the previous Example does not arise in the proposed formulation. Indeed, considering again the two candidate projective consensus clusterings \mathcal{C}_1 and \mathcal{C}_2 of the Example, it is easy to see that:

$$\Psi_{of}(\mathcal{C}_1, \mathcal{E}) = 0 \quad \text{and} \quad \Psi_{of}(\mathcal{C}_2, \mathcal{E}) > 0$$

Thus, \mathcal{C}_1 is correctly recognized as an optimal solution, whereas \mathcal{C}_2 is not.

3.2 Heuristics

Apart from solving the critical issue of two-objective PCE previously explained, a major advantage of the proposed PCE formulation w.r.t. the early ones defined in [16] is its close relationship to the classic formulations typically employed by CE algorithms. Like standard CE, the problem defined in (4) can be straightforwardly proved to be a special version of the *median partition problem* [4], which is defined as follows: given a number of partitions (clusterings) defined over the same set of objects and a distance measure between partitions, find a (new) clustering that minimizes the distance from all the input clusterings. The only difference between (4) and any standard CE formulation is that the former deals with projective clustering solutions (and hence, it needs a new measure for comparing projective clusterings), whereas the latter involves standard clustering solutions. The closeness to CE is a key point of our work, as it enables the development of heuristic algorithms for PCE following standard approaches to CE. The advantage in this respect is twofold: heuristics for PCE can be defined by exploiting the extensive and well-established work so far given for standard CE, which enables the development of solutions that are simple and easy-to-understand, and effective at the same time.

Within this view, a reasonable choice for defining proper heuristics for PCE is to adapt the standard CE approaches, i.e., instance-based, cluster-based, and hybrid (cf. Sect. 2.1), to the PCE context. However, it is arguable if all such CE approaches are well-suited for PCE. In fact, defining an instance-based PCE method is intrinsically tricky, and this also holds for the hybrid approach, which is essentially a combination of the instance-based and cluster-based ones. We explain the issues on defining instance-based PCE in the following.

First, as the focus of any hypothetical instance-based PCE is primarily on data objects, performing the two PCE steps of data clustering and feature-to-cluster assignment altogether would be hard. Indeed, focusing on data objects may produce information about data clustering only (for instance, by exploiting a co-occurrence matrix properly re-defined for the PCE context). This would force the assignment of the features to the various clusters to be performed in a separate step, and only once the objects have been grouped in clusters. Unfortunately, performing the two PCE steps of data clustering and feature-to-cluster assignment distinctly may negatively affect accuracy of the output consensus clustering. According to the definition of projective clustering, the information about the various objects belonging to any projective cluster should not be interpreted as absolute, but always in relation to the subspace associated to that cluster and vice versa. Thus, data clustering and feature-to-cluster assignment should be interrelated, at each step of the heuristic algorithm to be defined.

A more crucial issue arises even accepting to perform data clustering and feature-to-cluster assignment separately. Given a set of data objects to be included in any projective cluster, the feature-to-cluster assignment process should take into account that the notion of subspace of any given projective cluster makes sense only if it refers to the whole set of objects belonging to that cluster. In other words, saying that any set of data objects forms a cluster C having a subset \mathcal{S} of features associated with it does not mean that each object within C is represented by \mathcal{S} , but rather that

Algorithm 1 CB-PCE

Input: a projective ensemble \mathcal{E} ; the number K of clusters in the output projective consensus clustering;
Output: the projective consensus clustering \mathcal{C}^*

```

1:  $\Phi_{\mathcal{E}} \leftarrow \bigcup_{\hat{c} \in \mathcal{E}} \hat{C}$ 
2:  $P \leftarrow \text{pairwiseClusterDistances}(\Phi_{\mathcal{E}})$ 
3:  $\mathbf{M} \leftarrow \text{metaclusters}(\Phi_{\mathcal{E}}, P, K)$ 
4:  $\mathcal{C}^* \leftarrow \emptyset$ 
5: for all  $\mathcal{M} \in \mathbf{M}$  do
6:    $\Gamma_{\mathcal{M}}^* \leftarrow \text{object-basedRepresentation}(\Phi_{\mathcal{E}}, \mathcal{M})$ 
7:    $\Delta_{\mathcal{M}}^* \leftarrow \text{feature-basedRepresentation}(\Phi_{\mathcal{E}}, \mathcal{M})$ 
8:    $\mathcal{C}^* \leftarrow \mathcal{C}^* \cup \{(\Gamma_{\mathcal{M}}^*, \Delta_{\mathcal{M}}^*)\}$ 
9: end for

```

the entire set \mathcal{C} is represented by \mathcal{S} . Unfortunately, performing feature-to-cluster assignment apart from data clustering contrasts with the semantics of a subspace associated to a set of objects in a projective cluster. Indeed, the various features could be assigned to any given cluster C only by considering the objects within C independently of one another. Let us consider, for example, the case where the assignment is performed by averaging over the objects within C and over the feature-based representations of all the clusters within the ensemble \mathcal{E} , i.e., $\Delta_{C,f} = \text{avg}_{\sigma \in C, \hat{c} \in \mathcal{E}} \{\Gamma_{\hat{c},\sigma} \times \Delta_{\hat{c},f}\}$, $\forall f \in \mathcal{F}$. This case clearly shows that each feature f is assigned to C by considering each object within C independently from the other ones belonging to C .

Within this view, we discard instance-based and hybrid approaches to embrace a cluster-based approach. In the following, we describe our cluster-based proposal in detail and also show how this is particularly appropriate to the PCE context.

3.2.1 The CB-PCE algorithm

The *Cluster-Based Projective Clustering Ensembles (CB-PCE)* algorithm is proposed as a heuristic approach to the PCE formulation given in (4). In addition to the notation provided in Sect. 2, CB-PCE employs the following symbols: \mathbf{M} denotes a set of metaclusters (i.e., a set of sets of clusters), $\mathcal{M} \in \mathbf{M}$ denotes a metacluster (i.e., a set of clusters), and $M \in \mathcal{M}$ denotes a cluster (i.e., a set of data objects).

The outline of CB-PCE is reported in Alg. 1. Similarly to standard cluster-based CE, the first step of CB-PCE aims to group the set $\Phi_{\mathcal{E}}$ of clusters from each solution within the input ensemble \mathcal{E} into metaclusters (Lines 1-2). A clustering step over the set $\Phi_{\mathcal{E}}$ is performed by the function *metaclusters*. This step exploits the matrix P of pairwise distances between the clusters within $\Phi_{\mathcal{E}}$ (Line 1). The distance between any pair of clusters is computed by resorting to the Tanimoto similarity coefficient reported in (8). The set \mathbf{M} of metaclusters is finally exploited to derive the object- and feature-based representations of each projective cluster to be included into the output consensus clustering \mathcal{C}^* (Lines 3-8). Such representations are denoted by $\Gamma_{\mathcal{M}}^*$ and $\Delta_{\mathcal{M}}^*$, $\forall \mathcal{M} \in \mathbf{M}$, respectively; more precisely, $\Gamma_{\mathcal{M}}^*$ (resp. $\Delta_{\mathcal{M}}^*$) denotes the object-based (resp. feature-based) representation of the projective cluster within \mathcal{C}^* corresponding to the metacluster \mathcal{M} .

$\Gamma_{\mathcal{M}}^*$ and $\Delta_{\mathcal{M}}^*$ are derived by focusing on the optimization of a criterion easy to solve, which enables the finding of reasonable and effective approximations at the same time. In particular, we adapt the widely used *majority voting* to the context at hand. Let us first consider $\Gamma_{\mathcal{M}}^*$ values. If

the projective clustering solutions within the ensemble are all hard at a clustering level, the majority voting criterion leads to the definition of the following optimization problem:

$$\begin{aligned} \{\Gamma_{\mathcal{M}}^* \mid \mathcal{M} \in \mathbf{M}\} &= \underset{\{\Gamma_{\mathcal{M}} \mid \mathcal{M} \in \mathbf{M}\}}{\operatorname{argmin}} \sum_{\mathcal{M} \in \mathbf{M}} \sum_{\bar{\sigma} \in \mathcal{D}} \frac{\Gamma_{\mathcal{M},\bar{\sigma}}}{|\mathcal{M}|} \sum_{M \in \mathcal{M}} 1 - \Gamma_{M,\bar{\sigma}} \\ &s.t. \\ &\sum_{\mathcal{M} \in \mathbf{M}} \Gamma_{\mathcal{M},\bar{\sigma}} = 1, \quad \forall \bar{\sigma} \in \mathcal{D} \\ &\Gamma_{\mathcal{M},\bar{\sigma}} \in \{0, 1\}, \quad \forall \mathcal{M} \in \mathbf{M}, \forall \bar{\sigma} \in \mathcal{D} \end{aligned}$$

whose solution can be easily proved to be as follows ($\forall \mathcal{M}, \forall \bar{\sigma}$):

$$\Gamma_{\mathcal{M},\bar{\sigma}}^* = \begin{cases} 1 & \text{if } \mathcal{M} = \operatorname{arg} \min_{\mathcal{M}' \in \mathbf{M}} \frac{1}{|\mathcal{M}'|} \sum_{M \in \mathcal{M}'} 1 - \Gamma_{M,\bar{\sigma}} \\ 0 & \text{otherwise} \end{cases}$$

that is, each object $\bar{\sigma}$ is assigned to the metacluster \mathcal{M} containing the maximum number of clusters to which $\bar{\sigma}$ belongs (i.e., such that $\Gamma_{\mathcal{M},\bar{\sigma}} = 1$).

If the ensemble contains projective clusterings that are soft at clustering level, the following problem can be defined:

$$\{\Gamma_{\mathcal{M}}^* \mid \mathcal{M} \in \mathbf{M}\} = \underset{\{\Gamma_{\mathcal{M}} \mid \mathcal{M} \in \mathbf{M}\}}{\operatorname{argmin}} Q \quad (9)$$

s.t.

$$\sum_{\mathcal{M} \in \mathbf{M}} \Gamma_{\mathcal{M},\bar{\sigma}} = 1, \quad \forall \bar{\sigma} \in \mathcal{D} \quad (10)$$

$$\Gamma_{\mathcal{M},\bar{\sigma}} \geq 0, \quad \forall \mathcal{M} \in \mathbf{M}, \forall \bar{\sigma} \in \mathcal{D} \quad (11)$$

where

$$Q = \sum_{\mathcal{M} \in \mathbf{M}} \sum_{\bar{\sigma} \in \mathcal{D}} \Gamma_{\mathcal{M},\bar{\sigma}}^\alpha A_{\mathcal{M},\bar{\sigma}}, \quad A_{\mathcal{M},\bar{\sigma}} = \frac{1}{|\mathcal{M}|} \sum_{M \in \mathcal{M}} 1 - \Gamma_{M,\bar{\sigma}}$$

and $\alpha > 1$ is an integer that guarantees the non-linearity of the objective function Q w.r.t. $\Gamma_{\mathcal{M},\bar{\sigma}}$, needed to ensure $\Gamma_{\mathcal{M},\bar{\sigma}}^* \in [0, 1]$ (rather than $\{0, 1\}$).¹ The solution for such a problem however is not as straightforward as that of the traditional case (i.e., hard data clustering). We derive the solution in the following.

THEOREM 1. *The optimal solution of problem P defined in (9)-(11) is given by ($\forall \mathcal{M}, \forall \bar{\sigma}$):*

$$\Gamma_{\mathcal{M},\bar{\sigma}}^* = \left[\sum_{\mathcal{M}' \in \mathbf{M}} \left(\frac{A_{\mathcal{M},\bar{\sigma}}}{A_{\mathcal{M}',\bar{\sigma}}} \right)^{\frac{1}{\alpha-1}} \right]^{-1} \quad (12)$$

PROOF. The optimal $\Gamma_{\mathcal{M},\bar{\sigma}}^*$ can be found by means of the conventional *Lagrange multipliers* method. To this end, we first consider the relaxed problem P' of P obtained by temporarily discarding the inequality constraints from the constraint set of P (i.e., the constraints defined in (11)).

We define the new (unconstrained) objective function Q' for P' as follows:

$$Q' = Q + \sum_{\bar{\sigma} \in \mathcal{D}} \lambda_{\bar{\sigma}} \left(\sum_{\mathcal{M}' \in \mathbf{M}} \Gamma_{\mathcal{M}',\bar{\sigma}} - 1 \right) \quad (13)$$

The optimal $\Gamma_{\mathcal{M},\bar{\sigma}}^*$ are computed by first retrieving the stationary points of Q' , i.e., the points for which

$$\nabla Q' = \left(\frac{\partial Q'}{\partial \Gamma_{\mathcal{M},\bar{\sigma}}}, \frac{\partial Q'}{\partial \lambda_{\bar{\sigma}}} \right) = 0$$

¹Alternatively, to obtain $\Gamma_{\mathcal{M},\bar{\sigma}}^* \in [0, 1]$, properly defined regularization terms can be introduced (see, e.g., [21]).

Thus, we solve the following system of equations:

$$\frac{\partial Q'}{\partial \Gamma_{\mathcal{M},\bar{\sigma}}} = \alpha A_{\mathcal{M},\bar{\sigma}} (\Gamma_{\mathcal{M},\bar{\sigma}})^{\alpha-1} + \lambda_{\bar{\sigma}} = 0 \quad (14)$$

$$\frac{\partial Q'}{\partial \lambda_{\bar{\sigma}}} = \sum_{\mathcal{M}' \in \mathbf{M}} \Gamma_{\mathcal{M}',\bar{\sigma}} - 1 = 0 \quad (15)$$

Solving (14) w.r.t. $\Gamma_{\mathcal{M},\bar{\sigma}}$ and substituting such a solution in (15), we obtain:

$$\sum_{\mathcal{M}' \in \mathbf{M}} \left(\frac{-\lambda_{\bar{\sigma}}}{\alpha A_{\mathcal{M}',\bar{\sigma}}} \right)^{\frac{1}{\alpha-1}} = 1 \quad (16)$$

Solving (16) w.r.t. $\lambda_{\bar{\sigma}}$ and substituting such a solution in (14), we obtain:

$$\alpha A_{\mathcal{M},\bar{\sigma}} (\Gamma_{\mathcal{M},\bar{\sigma}})^{\alpha-1} - \left[\sum_{\mathcal{M}' \in \mathbf{M}} \left(\frac{1}{\alpha A_{\mathcal{M}',\bar{\sigma}}} \right)^{\frac{1}{\alpha-1}} \right]^{-(\alpha-1)} = 0 \quad (17)$$

Finally, solving (17) w.r.t. $\Gamma_{\mathcal{M},\bar{\sigma}}$, we obtain a stationary point whose expression is exactly equal to that in (12):

$$\Gamma_{\mathcal{M},\bar{\sigma}}^* = \left[\sum_{\mathcal{M}' \in \mathbf{M}} \left(\frac{A_{\mathcal{M},\bar{\sigma}}}{A_{\mathcal{M}',\bar{\sigma}}} \right)^{\frac{1}{\alpha-1}} \right]^{-1} \quad (18)$$

As it holds that (i) the stationary points of the Lagrangian function Q' are also stationary points of the original objective function Q , (ii) the feasible region of P' is a convex set, and (iii) Q is convex w.r.t. $\Gamma_{\mathcal{M},\bar{\sigma}}$, it follows that such a stationary point represents a global minimum of Q , and, accordingly, the optimal solution of P' . Moreover, as $A_{\mathcal{M},\bar{\sigma}} \geq 0, \forall \mathcal{M}, \forall \bar{\sigma}$, it is trivial to observe that $\Gamma_{\mathcal{M},\bar{\sigma}}^* \geq 0, \forall \mathcal{M}, \forall \bar{\sigma}$. Therefore, the solution in (18) satisfies the inequality constraints that were temporarily discarded in order to define the relaxed problem P' (cf. (11)); thus, it represents the optimal solution of the original problem P , which proves the theorem. \square

An analogous reasoning can be carried out for $\Delta_{\mathcal{M},f}^*$. In this case, the problem to be solved is the following:

$$\{\Delta_{\mathcal{M}}^* \mid \mathcal{M} \in \mathbf{M}\} = \underset{\{\Delta_{\mathcal{M}} \mid \mathcal{M} \in \mathbf{M}\}}{\operatorname{argmin}} \sum_{\mathcal{M} \in \mathbf{M}} \sum_{f \in \mathcal{F}} \Delta_{\mathcal{M},f}^\beta B_{\mathcal{M},f} \quad (19)$$

s.t.

$$\sum_{f \in \mathcal{F}} \Delta_{\mathcal{M},f} = 1, \quad \forall \mathcal{M} \in \mathbf{M} \quad (20)$$

$$\Delta_{\mathcal{M},f} \geq 0, \quad \forall \mathcal{M} \in \mathbf{M}, \forall f \in \mathcal{F} \quad (21)$$

where $B_{\mathcal{M},f} = |\mathcal{M}|^{-1} \sum_{M \in \mathcal{M}} 1 - \Delta_{M,f}$ and β plays the same role as α in function Q . The solution of such a problem is similar to that derived for $\Gamma_{\mathcal{M},\bar{\sigma}}^*$:

THEOREM 2. *The optimal solution of the problem defined in (19)-(21) is given by the following ($\forall \mathcal{M}, \forall f$):*

$$\Delta_{\mathcal{M},f}^* = \left[\sum_{f' \in \mathcal{F}} \left(\frac{B_{\mathcal{M},f}}{B_{\mathcal{M},f'}} \right)^{\frac{1}{\beta-1}} \right]^{-1} \quad (22)$$

PROOF. Analogous to Theorem 1. \square

Rationale of CB-PCE.

Let us now informally show that CB-PCE is well-suited for PCE, thus supporting one of the claim of this work, i.e., cluster-based approaches are particularly appropriate to the PCE context (unlike instance-based and hybrid ones).

Looking at the PCE formulation reported in (4), it is easy to see that function Ψ_{of} retrieves the consensus clustering \mathcal{C}^* so that each cluster within \mathcal{C}^* is ideally “assigned” to exactly one cluster of each projective clustering solution in the input ensemble \mathcal{E} , where the “assignments” are performed by minimizing the Tanimoto distance $1 - \hat{J}$ (cf. (8)). Thus, considering all the solutions in the ensemble, any cluster $C \in \mathcal{C}^*$ is assigned to a set of clusters (metacluster) \mathcal{M} that contains exactly one cluster of each solution in the ensemble, that is $|\mathcal{M}| = |\mathcal{E}|$, and $M' \in \mathcal{C} \wedge M'' \in \mathcal{C} \Leftrightarrow M' = M''$, $\forall M', M'' \in \mathcal{M}, \forall C \in \mathcal{E}$.

Clearly, if one would know in advance the optimal set of metaclusters to be assigned to the clusters within \mathcal{C}^* , the problem in (4) would be optimally solved by computing, for each metacluster \mathcal{M} , the cluster C^* that minimizes the Tanimoto distance from all the clusters within \mathcal{M} , that is:

$$C^* = \arg \min_C \sum_{M \in \mathcal{M}} 1 - \hat{J}(X_C, X_M) \quad (23)$$

However, it holds that: (i) the metaclusters are not known in advance, as their computation is part of the optimization process; (ii) the problem in (23) is hard to solve: it falls into the class of median problems in which the distance to be minimized is the Tanimoto distance; this kind of problems has been recently proved to be NP-hard [9].

The validity of CB-PCE as a heuristic approach to the PCE formulation proposed in (4) lies in that it exactly follows the scheme reported above (i.e., it first recognizes metaclusters and then assigns objects and features to metaclusters), following some approximations. These approximations are needed for solving two critical points:

1. a sub-optimal set of metaclusters is computed by clustering the overall set of projective clusters within the ensemble, where the distance measure used for comparing clusters is the Tanimoto distance, which is the measure employed by the proposed formulation in (4);
2. $\Gamma_{\mathcal{M}}^*$ and $\Delta_{\mathcal{M}}^*$ values (for each metacluster \mathcal{M}) are computed by optimizing an easy-to-solve criterion that effectively approximates the problem in (23).

3.2.2 Speeding-up CB-PCE: FCB-PCE

Given a set \mathcal{D} of data objects and a set \mathcal{F} of features, the computational complexity of the measure \hat{J} reported in (8) (used for computing the similarity between two projective clusters) is $\mathcal{O}(|\mathcal{D}| |\mathcal{F}|)$, as it involves a comparison between two $|\mathcal{D}| \times |\mathcal{F}|$ matrices. For efficiency purposes, we can lower the complexity by defining an alternative measure working in $\mathcal{O}(|\mathcal{D}| + |\mathcal{F}|)$. Given any two projective clusters C' and C'' , such a measure, called \hat{J}_{fast} , exploits the object-based ($\Gamma_{C'}$ and $\Gamma_{C''}$) and the feature-based ($\Delta_{C'}$ and $\Delta_{C''}$) representation vectors of C' and C'' , respectively, rather than their corresponding projective matrices. Formally:

$$\hat{J}_{fast}(C', C'') = \frac{1}{2} \left(\hat{J}(\Gamma_{C'}, \Gamma_{C''}) + \hat{J}(\Delta_{C'}, \Delta_{C''}) \right) \quad (24)$$

where $\hat{J}(\cdot, \cdot)$ denotes again the Tanimoto similarity coefficient defined in (8), which is in this case applied to real-

valued vectors rather than matrices. It is easy to observe that, like \hat{J} , $\hat{J}_{fast} \in [0, 1]$.

Taking into account \hat{J}_{fast} , we define a version of the CB-PCE algorithm which is similar to that defined in Sect. 3.2.1, except for the measure involved for comparing the projective clusters, which is, in this case, based on \hat{J}_{fast} . We hereinafter refer to this alternative version of the algorithm as *Fast Cluster-Based Projective Clustering Ensembles* (FCB-PCE) algorithm.

Although clearly advantageous in terms of efficiency, a major drawback of FCB-PCE concerns accuracy. In fact, a major weakness of the measure \hat{J}_{fast} exploited by FCB-PCE is that it is less accurate than its slow counterpart \hat{J} exploited by CB-PCE. This essentially depends on the fact that comparing any two projective clusters C' and C'' by involving their projective matrices $X_{C'}$ and $X_{C''}$, respectively, is generally more effective than involving their object- and feature-based representation vectors $\Gamma_{C'}$, $\Gamma_{C''}$, $\Delta_{C'}$, and $\Delta_{C''}$ [23].² Indeed, although it can be trivially proved that $X_{C'} = X_{C''} \Leftrightarrow \Gamma_{C'} = \Gamma_{C''} \wedge \Delta_{C'} = \Delta_{C''}$, the vectors $\Gamma_{C'}$, $\Delta_{C'}$, and $\Gamma_{C''}$, $\Delta_{C''}$ are in general a factorization of the matrices $X_{C'}$ and $X_{C''}$, respectively (i.e., $X_{C'} = \Gamma_{C'}^T \Delta_{C'}$ and $X_{C''} = \Gamma_{C''}^T \Delta_{C''}$). Thus, only matrices $X_{C'}$ and $X_{C''}$ provide the whole information about the representation of the corresponding projective clusters.

Although \hat{J}_{fast} is less accurate than \hat{J} , it still allows the comparison of projective clusters by taking into account their object- and feature-based representations altogether. Hence, the proposed FCB-PCE heuristic based on \hat{J}_{fast} still represents a valuable heuristic to the PCE formulation proposed in this work, as it overcomes the main issue of two-objective PCE explained in Sect. 3.1.

3.2.3 Computational Analysis

Here we discuss the computational complexity of the proposed CB-PCE and FCB-PCE algorithms. We are given: a set \mathcal{D} of data objects, each one defined over a feature space \mathcal{F} , a projective ensemble \mathcal{E} defined over \mathcal{D} and \mathcal{F} , and a positive integer K representing the number of clusters in the output projective consensus clustering. We also assume that the size $|\mathcal{C}|$ of each solution \mathcal{C} in \mathcal{E} is $\mathcal{O}(K)$. For both the algorithms, we may distinguish three steps:

1. *pre-processing*: it concerns the computation of the pairwise distances between clusters, by involving measures \hat{J} (cf. (8)) for CB-PCE and \hat{J}_{fast} (cf. (24)) for FCB-PCE; this step takes $\mathcal{O}(K^2 |\mathcal{E}|^2 |\mathcal{D}| |\mathcal{F}|)$ and $\mathcal{O}(K^2 |\mathcal{E}|^2 (|\mathcal{D}| + |\mathcal{F}|))$ for CB-PCE and FCB-PCE, respectively, because computing \hat{J} (resp. \hat{J}_{fast}) is $\mathcal{O}(|\mathcal{D}| |\mathcal{F}|)$ (resp. $\mathcal{O}(|\mathcal{D}| + |\mathcal{F}|)$) (cf. Sect. 3.2.2), and the clusters to be compared to each other are $\mathcal{O}(K |\mathcal{E}|)$;
2. *meta-clustering*: it concerns the clustering of the $\mathcal{O}(K |\mathcal{E}|)$ clusters of all the solutions in the ensemble; assuming to employ a clustering algorithm which is at most quadratic w.r.t. the size of the dataset to be partitioned, this step takes $\mathcal{O}(K^2 |\mathcal{E}|^2)$ for both CB-PCE and FCB-PCE;
3. *post-processing*: it concerns the assignment of objects and features to the metaclusters, and is exactly the

²[23] deals with hard projective clusters; however, the reasoning therein involved can be easily extended to a soft case.

Table 1: Computational complexities

	<i>total</i>	<i>online</i>	<i>offline</i>
MOEA-PCE	$\mathcal{O}(ItK^2 \mathcal{E} (\mathcal{D} + \mathcal{F}))$	$\mathcal{O}(ItK^2 \mathcal{E} (\mathcal{D} + \mathcal{F}))$	—
EM-PCE	$\mathcal{O}(K \mathcal{E} \mathcal{D} \mathcal{F})$	$\mathcal{O}(IK \mathcal{D} \mathcal{F})$	$\mathcal{O}(K \mathcal{E} \mathcal{D} \mathcal{F})$
CB-PCE	$\mathcal{O}(K^2 \mathcal{E} ^2 \mathcal{D} \mathcal{F})$	$\mathcal{O}(K \mathcal{E} (K \mathcal{E} + \mathcal{D} + \mathcal{F}))$	$\mathcal{O}(K^2 \mathcal{E} ^2 \mathcal{D} \mathcal{F})$
FCB-PCE	$\mathcal{O}(K^2 \mathcal{E} ^2(\mathcal{D} + \mathcal{F}))$	$\mathcal{O}(K \mathcal{E} (K \mathcal{E} + \mathcal{D} + \mathcal{F}))$	$\mathcal{O}(K^2 \mathcal{E} ^2(\mathcal{D} + \mathcal{F}))$

same for both CB-PCE and FCB-PCE. According to (12) and (22), both the object and the feature assignments need to look up all the clusters in each meta-cluster only once; thus, for each object and for each feature, the needed step costs $\mathcal{O}(K|\mathcal{E}|)$. Accordingly, performing this step for all objects and features leads to a total cost of $\mathcal{O}(K|\mathcal{E}|(|\mathcal{D}| + |\mathcal{F}|))$ for the entire post-processing step.

It can be noted that the first step is an *offline* phase, i.e., a phase to be performed only once in case of a multi-run execution, whereas the second and third are *online* steps. Thus, as summarized in Table 1 (where we also report the complexities of the earlier MOEA-PCE and EM-PCE methods defined in [16]³), we can finally state that:

- the *offline*, *online*, and *total* (i.e., offline + online) complexities of CB-PCE are $\mathcal{O}(K^2|\mathcal{E}|^2|\mathcal{D}||\mathcal{F}|)$, $\mathcal{O}(K|\mathcal{E}|(K|\mathcal{E}| + |\mathcal{D}| + |\mathcal{F}|))$, and $\mathcal{O}(K^2|\mathcal{E}|^2|\mathcal{D}||\mathcal{F}|)$, respectively;
- the *offline*, *online*, and *total* (i.e., offline + online) complexities of FCB-PCE are $\mathcal{O}(K^2|\mathcal{E}|^2(|\mathcal{D}| + |\mathcal{F}|))$, $\mathcal{O}(K|\mathcal{E}|(K|\mathcal{E}| + |\mathcal{D}| + |\mathcal{F}|))$, and $\mathcal{O}(K^2|\mathcal{E}|^2(|\mathcal{D}| + |\mathcal{F}|))$, respectively.

Interpretation of the complexity results.

Let us now provide an insight for the comparison between the (total) complexities derived above. For the sake of readability, we hereinafter omit the suffix “-PCE” from the names of the various PCE algorithms. We denote with $r(a_1, a_2)$ the ratio between the complexities of the PCE algorithms a_1 and a_2 . Clearly, a ratio smaller (resp. greater) than 1 means that the complexity of a_1 is smaller (resp. greater) than that of a_2 . Our main observations are summarized in the following.

- As expected, FCB-PCE is always faster than CB-PCE, as it holds that $r(\text{FCB}, \text{CB}) = (|\mathcal{D}| + |\mathcal{F}|) / (|\mathcal{D}||\mathcal{F}|) \leq 1$, $\forall |\mathcal{D}|, |\mathcal{F}| > 1$.
- CB-PCE:
 - it holds that $r(\text{CB}, \text{EM}) = K|\mathcal{E}| > 1$; thus, CB-PCE is always slower than EM-PCE;
 - the ratio $r(\text{CB}, \text{MOEA})$ is equal to $(|\mathcal{E}||\mathcal{D}||\mathcal{F}|) / (I t (|\mathcal{D}| + |\mathcal{F}|))$. This implies that $r(\text{CB}, \text{MOEA}) < 1$ if $(2|\mathcal{D}||\mathcal{F}|) / (|\mathcal{D}| + |\mathcal{F}|) < 2 I t / |\mathcal{E}|$, i.e., as $(|\mathcal{D}| + |\mathcal{F}|) / 2 \geq (2|\mathcal{D}||\mathcal{F}|) / (|\mathcal{D}| + |\mathcal{F}|)$, that $r(\text{CB}, \text{MOEA}) < 1$ if $|\mathcal{D}| + |\mathcal{F}| < 4 I t / |\mathcal{E}|$. The latter condition is true only in a small number of real cases; as an example, considering the numerical values for I , t and $|\mathcal{E}|$ suggested in [16]

³In Table 1, I denotes the number of iterations to convergence (for MOEA-PCE and EM-PCE), whereas t is the population size (for MOEA-PCE only) [16].

(i.e., 200, 30 and 200, respectively), CB-PCE is faster than MOEA-PCE if $|\mathcal{D}| + |\mathcal{F}| < 120$, i.e., when the input dataset is very small and/or low-dimensional. For this purpose, CB-PCE can be recognized as in practice always slower than MOEA-PCE.

- FCB-PCE:

- it holds that the ratio $r(\text{FCB}, \text{EM}) = (K|\mathcal{E}|(|\mathcal{D}| + |\mathcal{F}|)) / (|\mathcal{D}||\mathcal{F}|)$ is greater than 1 if $(2|\mathcal{D}||\mathcal{F}|) / (|\mathcal{D}| + |\mathcal{F}|) < 2 K|\mathcal{E}|$, which essentially means that FCB-PCE is slower than EM-PCE if $|\mathcal{D}| + |\mathcal{F}| < 4 K|\mathcal{E}|$, as $(|\mathcal{D}| + |\mathcal{F}|) / 2 \geq (2|\mathcal{D}||\mathcal{F}|) / (|\mathcal{D}| + |\mathcal{F}|)$. Thus, for large and/or high-dimensional datasets (i.e., for datasets having $|\mathcal{D}|$ and $|\mathcal{F}|$ such that $|\mathcal{D}| + |\mathcal{F}| > 4 K|\mathcal{E}|$) FCB-PCE may be faster than EM-PCE, whereas for small and/or low-dimensional datasets may not;
- $r(\text{FCB}, \text{MOEA}) = |\mathcal{E}| / (I t)$; assuming to set t equal to 15% of the ensemble size $|\mathcal{E}|$ as suggested in [16], it holds that $r(\text{FCB}, \text{MOEA}) = 20 / (3 I)$. Thus, as it typically holds that $I \gg 7$ (e.g., in [16] $I = 200$), $r(\text{FCB}, \text{MOEA})$ is always smaller than 1 and, therefore, FCB-PCE is always faster than MOEA-PCE.

To summarize, we can state that CB-PCE is the slowest method. FCB-PCE is faster than MOEA-PCE, whereas, compared to EM-PCE, it is faster (resp. slower) for large (resp. small) and/or high-dimensional (resp. low-dimensional) datasets.

4. EXPERIMENTAL EVALUATION

We conducted an experimental evaluation to assess the accuracy and efficiency of the consensus clusterings obtained by the proposed CB-PCE and FCB-PCE. The comparison also involved the previous existing PCE algorithms (i.e., MOEA-PCE and EM-PCE) [16] as baseline methods.⁴

4.1 Evaluation methodology

Following [16], we used eight benchmark datasets from the UCI Machine Learning Repository [27], namely Iris, Wine, Glass, Ecoli, Yeast, Segmentation, Abalone and Letter, and two time-series datasets from the UCR Time Series Classification/Clustering Page [33], namely Tracedata and ControlChart. Table 2 reports the main characteristics of the datasets; the interested reader is referred to [27, 33] for a description of the datasets.

⁴Experiments were conducted on a quad-core platform Intel Pentium IV 3GHz with 4GB memory and running Microsoft WinXP Pro.

Table 2: Datasets used in the experiments

dataset	objects	attributes	classes
Iris	150	4	3
Wine	178	13	3
Glass	214	10	6
Ecoli	327	7	5
Yeast	1,484	8	10
Segmentation	2,310	19	7
Abalone	4,124	7	17
Letter	7,648	16	10
Tracedata	200	275	4
ControlChart	600	60	6

4.1.1 Ensemble generation

We generated ensembles as suggested in [16]. In particular, for each set of experiments and dataset we considered 20 different ensembles; all results we present in the following refer to averages over these ensembles. Ensemble generation was carried out by running the LAC projective clustering algorithm [30], in which the diversity of the solutions was ensured by randomly choosing the initial centroids and varying the parameter h ; here we recall that this parameter controls the incentive for clustering on more features depending on the strength of the local correlation of data. To test the ability of the proposed algorithms to deal with soft clustering solutions and with solutions having equally weighted feature-to-cluster assignments, we generated each ensemble \mathcal{E} as a composition of four equal-sized subsets, denoted as \mathcal{E}_1 (hard data clustering, feature-to-cluster assignments unequally weighted), \mathcal{E}_2 (hard data clustering, feature-to-cluster assignments equally weighted), \mathcal{E}_3 (soft data clustering, feature-to-cluster assignments unequally weighted), and \mathcal{E}_4 (soft data clustering, feature-to-cluster assignments equally weighted).

4.1.2 Setting of the PCE algorithms

We set the parameters of MOEA-PCE and EM-PCE as reported in [16]. In particular, as far as MOEA-PCE, the population size (t) was set equal to 15% of the ensemble size and the number I of maximum iterations equal to 200. The random noise needed for the mutation step was obtained via *Monte Carlo* sampling on a standard Gaussian distribution. Regarding EM-PCE, the parameter α was set equal to 2; this value also represented the optimal value for the parameters α and β of our CB-PCE and FCB-PCE.

4.1.3 Assessment criteria

We assessed the quality of a consensus clustering \mathcal{C} using both an external and an internal validity approach; specifically, we carried out two evaluation stages, the first based on the similarity of \mathcal{C} w.r.t. a reference classification and the second based on the average similarity w.r.t. the solutions in the input ensemble \mathcal{E} .

Similarity w.r.t. the reference classification.

We denote with $\tilde{\mathcal{C}}$ a reference classification, where the object-based representations $\Gamma_{\tilde{\mathcal{C}}}$ of each projective cluster $\tilde{\mathcal{C}}$ within $\tilde{\mathcal{C}}$ are provided along with \mathcal{D} (the selected datasets are all available with a reference classification), whereas the feature-based representations $\Delta_{\tilde{\mathcal{C}},f}$, $\forall \tilde{\mathcal{C}} \in \tilde{\mathcal{C}}, \forall f \in \mathcal{F}$, are

computed as suggested in [30]:

$$\Delta_{\tilde{\mathcal{C}},f} = \frac{\exp(-U(\tilde{\mathcal{C}}, f)/h)}{\sum_{f' \in \mathcal{F}} \exp(-U(\tilde{\mathcal{C}}, f')/h)}$$

where the LAC’s parameter h was set equal to 0.2 and:

$$U(\hat{\mathcal{C}}, \hat{f}) = \left(\sum_{\tilde{\sigma} \in \mathcal{D}} \Gamma_{\hat{\mathcal{C}}, \tilde{\sigma}} \right)^{-1} \sum_{\tilde{\sigma} \in \mathcal{D}} \Gamma_{\hat{\mathcal{C}}, \tilde{\sigma}} (c(\hat{\mathcal{C}}, \hat{f}) - o_{\hat{f}})^2$$

$$c(\hat{\mathcal{C}}, \hat{f}) = \left(\sum_{\tilde{\sigma} \in \mathcal{D}} \Gamma_{\hat{\mathcal{C}}, \tilde{\sigma}} \right)^{-1} \sum_{\tilde{\sigma} \in \mathcal{D}} \Gamma_{\hat{\mathcal{C}}, \tilde{\sigma}} \times o_{\hat{f}}$$

with $o_{\hat{f}}$ denoting the \hat{f} -th feature value of object $\tilde{\sigma}$.

Similarity between \mathcal{C} and $\tilde{\mathcal{C}}$ was computed in terms of the *Normalized Mutual Information*, by taking into account their object-based (NMI_o) representations, feature-based representations (NMI_f), or both (NMI_{of}), and by adapting the original definition given in [28] to handle soft solutions. Here we report the formal definition of NMI_{of} , NMI_o and NMI_f can be derived in a similar way:

$$NMI_{of}(\mathcal{C}, \tilde{\mathcal{C}}) = \frac{\sum_{C \in \mathcal{C}} \sum_{\tilde{C} \in \tilde{\mathcal{C}}} \frac{a(C, \tilde{C})}{T(C, \tilde{C})} \times \log \left(\frac{|\mathcal{D}|^2 \times a(C, \tilde{C})}{T(C, \tilde{C}) \times b(C) \times b(\tilde{C})} \right)}{\sqrt{H(\mathcal{C}) \times H(\tilde{\mathcal{C}})}}$$

where

$$a(C', C'') = \sum_{\tilde{\sigma} \in \mathcal{D}} \sum_{f \in \mathcal{F}} \Gamma_{C', \tilde{\sigma}} \Delta_{C', f} \Gamma_{C'', \tilde{\sigma}} \Delta_{C'', f}$$

$$b(\hat{\mathcal{C}}) = \sum_{\tilde{\sigma} \in \mathcal{D}} \sum_{f \in \mathcal{F}} \Gamma_{\hat{\mathcal{C}}, \tilde{\sigma}} \Delta_{\hat{\mathcal{C}}, f} \quad H(\hat{\mathcal{C}}) = - \sum_{\hat{\mathcal{C}} \in \hat{\mathcal{C}}} \frac{b(\hat{\mathcal{C}})}{|\mathcal{D}|} \log \frac{b(\hat{\mathcal{C}})}{|\mathcal{D}|}$$

$$T(C', C'') = \sum_{\tilde{\sigma} \in \mathcal{D}} \sum_{f \in \mathcal{F}} \left(\sum_{C' \in \mathcal{C}'} \Gamma_{C', \tilde{\sigma}} \Delta_{C', f} \right) \left(\sum_{C'' \in \mathcal{C}''} \Gamma_{C'', \tilde{\sigma}} \Delta_{C'', f} \right)$$

We now explain the rationale of this evaluation stage. Let us consider NMI_{of} , where analogous considerations hold for NMI_o and NMI_f . Since no additional information is provided along with any given input projective ensemble \mathcal{E} —the reference classifications associated to the benchmark datasets are indeed exploited only for testing purposes—randomly extracting a projective solution from \mathcal{E} is the only fair way to proceed in case no PCE method is used. Within this view, in order to establish the validity of a projective consensus \mathcal{C} computed by any PCE algorithm, we compare the results achieved by \mathcal{C} w.r.t. those obtained by any projective clustering randomly chosen from \mathcal{E} . Such a comparison can be performed according to the following expression, which aims to compute the “expected difference” between the results by \mathcal{C} and those by \mathcal{E} :

$$\Theta_{of}(\mathcal{C}, \mathcal{E}, \tilde{\mathcal{C}}) = \sum_{\hat{\mathcal{C}} \in \mathcal{E}} \left(NMI_{of}(\mathcal{C}, \tilde{\mathcal{C}}) - NMI_{of}(\hat{\mathcal{C}}, \tilde{\mathcal{C}}) \right) \Pr(\hat{\mathcal{C}})$$

where $\Pr(\hat{\mathcal{C}})$ is the probability of randomly choosing $\hat{\mathcal{C}}$ from \mathcal{E} . Since no prior knowledge is provided along with \mathcal{E} , we can assume a uniform distribution for the probabilities $\Pr(\hat{\mathcal{C}})$, i.e., $\Pr(\hat{\mathcal{C}}) = |\mathcal{E}|^{-1}$, $\forall \hat{\mathcal{C}} \in \mathcal{E}$. Computing Θ_{of} hence becomes equal to computing the similarity between \mathcal{C} and $\tilde{\mathcal{C}}$ minus

Table 3: Evaluation w.r.t. the reference classification

data	Θ_{of}				Θ_o				Θ_f			
	MOEA-PCE	EM-PCE	CB-PCE	FCB-PCE	MOEA-PCE	EM-PCE	CB-PCE	FCB-PCE	MOEA-PCE	EM-PCE	CB-PCE	FCB-PCE
Iris	+0.146	+0.168	+0.218	+0.185	+0.319	+0.228	+0.309	+0.297	+0.198	-0.095	+0.139	+0.117
Wine	+0.136	+0.083	+0.275	+0.224	+0.201	+0.130	+0.272	+0.253	+0.152	+0.030	+0.211	+0.206
Glass	+0.105	+0.162	+0.158	+0.157	+0.092	+0.134	+0.180	+0.167	+0.048	+0.060	+0.001	+0.009
Ecoli	+0.164	+0.086	+0.211	+0.232	+0.245	+0.125	+0.223	+0.213	+0.042	+0.042	+0.023	+0.017
Yeast	+0.049	+0.021	+0.092	+0.095	+0.090	+0.066	+0.113	+0.110	+0.006	+0.090	+0.102	+0.010
Segmentation	+0.137	+0.144	+0.148	+0.141	+0.102	+0.206	+0.194	+0.185	+0.075	+0.079	+0.098	+0.150
Abalone	+0.116	+0.111	+0.134	+0.130	+0.141	+0.116	+0.185	+0.182	+0.093	+0.092	+0.123	+0.120
Letter	+0.111	+0.107	+0.141	+0.134	+0.146	+0.122	+0.188	+0.185	+0.092	+0.097	+0.131	+0.124
Trace	+0.097	+0.019	+0.125	+0.140	+0.032	+0.026	+0.154	+0.132	-0.007	+0.114	+0.112	+0.115
ControlChart	+0.091	+0.204	+0.345	+0.276	+0.050	+0.011	+0.027	+0.051	+0.233	+0.416	+0.287	+0.283
min	+0.049	+0.019	+0.092	+0.095	+0.032	+0.011	+0.027	+0.051	-0.007	-0.095	+0.001	+0.009
max	+0.164	+0.204	+0.345	+0.276	+0.319	+0.228	+0.309	+0.297	+0.233	+0.416	+0.287	+0.283
avg	+0.115	+0.110	+0.185	+0.171	+0.142	+0.116	+0.185	+0.178	+0.093	+0.093	+0.123	+0.122

the average similarity between $\tilde{\mathcal{C}}$ and the solutions within \mathcal{E} , as proved by the following:

$$\begin{aligned}
 \Theta_{of}(\mathcal{C}, \mathcal{E}, \tilde{\mathcal{C}}) &= \sum_{\hat{\mathcal{C}} \in \mathcal{E}} \left(NMI_{of}(\mathcal{C}, \tilde{\mathcal{C}}) - NMI_{of}(\hat{\mathcal{C}}, \tilde{\mathcal{C}}) \right) \Pr(\hat{\mathcal{C}}) = \\
 &= NMI_{of}(\mathcal{C}, \tilde{\mathcal{C}}) - \sum_{\hat{\mathcal{C}} \in \mathcal{E}} NMI_{of}(\hat{\mathcal{C}}, \tilde{\mathcal{C}}) \times |\mathcal{E}|^{-1} = \\
 &= NMI_{of}(\mathcal{C}, \tilde{\mathcal{C}}) - \text{avg}_{\hat{\mathcal{C}} \in \mathcal{E}} NMI_{of}(\hat{\mathcal{C}}, \tilde{\mathcal{C}}) \quad (25)
 \end{aligned}$$

Θ_o and Θ_f can be defined analogously. The larger Θ_{of} , Θ_o and Θ_f , the better the quality of \mathcal{C} .

Similarity w.r.t. the ensemble solutions.

The goal of this evaluation stage was to assess how well a consensus clustering complies with the solutions in the input ensemble. For this purpose, we evaluated the average similarity $\overline{NMI}_{of}(\mathcal{C}, \mathcal{E}) = \text{avg}_{\mathcal{C}' \in \mathcal{E}} NMI_{of}(\mathcal{C}, \mathcal{C}')$ between the consensus clustering \mathcal{C} and the solutions in the ensemble \mathcal{E} (\overline{NMI}_o and \overline{NMI}_f are defined analogously). To improve the readability of the results, we normalize \overline{NMI}_{of} , \overline{NMI}_o and \overline{NMI}_f by dividing them by the average pairwise similarity of the solutions in the ensemble. Formally, we define the ratios (coefficients of variation) Υ_{of} , Υ_o , and Υ_f :

$$\Upsilon_{of}(\mathcal{C}, \mathcal{E}) = \overline{NMI}_{of}(\mathcal{C}, \mathcal{E}) / \text{avg}_{\mathcal{C}', \mathcal{C}'' \in \mathcal{E}} NMI_{of}(\mathcal{C}', \mathcal{C}'') \quad (26)$$

Υ_o and Υ_f are defined similarly. The larger these quantities are, the better the quality of \mathcal{C} is.

4.2 Results

4.2.1 Accuracy

For each algorithm, dataset and ensemble, we performed 50 different runs. We reported average clustering results obtained by CB-PCE and FCB-PCE, as well as by the early MOEA-PCE and EM-PCE in Tables 3 and 4.

Evaluation w.r.t. the reference classification.

Both CB-PCE and FCB-PCE achieved higher Θ_{of} results (first 4-column groups in Table 3) than MOEA-PCE on all datasets. In particular, CB-PCE obtained an average improvement of 0.070, with a maximum gain of 0.254 (ControlChart), whereas FCB-PCE obtained an average improvement of 0.056, with a maximum of 0.185 (ControlChart again). EM-PCE was on average less accurate than MOEA-PCE; thus, the average gains of CB-PCE and FCB-PCE w.r.t. EM-PCE were higher than those achieved w.r.t.

MOEA-PCE (0.075 and 0.061, respectively). Comparing the two proposed CB-PCE and FCB-PCE, the former achieved higher quality on nearly all datasets (all but Ecoli, Yeast and Trace), with an average gain of about 0.014 and peaks on ControlChart (0.069) and Wine (0.051). The higher performance of CB-PCE vs. FCB-PCE confirms one of the major claims of this work (cf. Sect. 3.2.2).

The superior performance of CB-PCE and FCB-PCE w.r.t. the early MOEA-PCE and EM-PCE was also confirmed in terms of object-based (Θ_o) and feature-based (Θ_f) representations. In particular, CB-PCE achieved average Θ_o equal to 0.185 and average improvements w.r.t. MOEA-PCE and EM-PCE of 0.043 and 0.069, respectively. Also, CB-PCE outperformed MOEA-PCE (resp. EM-PCE) on seven (resp. eight) out of ten datasets. As far as FCB-PCE, the average Θ_o was 0.178, with average gains w.r.t. MOEA-PCE and EM-PCE equal to 0.036 and 0.062, respectively. FCB-PCE performed better than MOEA-PCE and EM-PCE on eight and nine out of ten datasets, respectively.

In terms of Θ_f , both CB-PCE and FCB-PCE were on average comparable to each other; in fact, they achieved average Θ_f equal to 0.123 and 0.122, respectively. The average improvements obtained by CB-PCE (resp. FCB-PCE) w.r.t. both MOEA-PCE and EM-PCE were equal to 0.030 (resp. 0.029). Like Θ_{of} and Θ_o , both the proposed CB-PCE and FCB-PCE performed better than MOEA-PCE and EM-PCE on the majority of the datasets also in terms of Θ_f .

Evaluation w.r.t. the ensemble solutions.

Concerning the coefficients of variation due to the consensus clustering w.r.t. the average pairwise similarity of the input ensemble (Table 4), CB-PCE and FCB-PCE led to average values respectively equal to 1.110 and 1.108 (Υ_{of}), 1.318 and 1.316 (Υ_o), 1.049 and 1.030 (Υ_f). Particularly, in the case Υ_{of} , CB-PCE improved MOEA-PCE and EM-PCE by 0.062 and 0.114 on average, respectively, whereas the average improvements obtained by FCB-PCE w.r.t. MOEA-PCE and EM-PCE were equal to 0.060 and 0.112, respectively. Also, CB-PCE was able to obtain peaks of improvement up to 0.297 (w.r.t. MOEA-PCE) and 0.454 (w.r.t. EM-PCE). The maximum gains of FCB-PCE were instead equal to 0.3 and 0.457 w.r.t. MOEA-PCE and EM-PCE, respectively. Both CB-PCE and FCB-PCE outperformed MOEA-PCE and EM-PCE on nearly all datasets. CB-PCE results were better than those of MOEA-PCE and EM-PCE on seven and nine out of ten datasets, respectively. As far

Table 4: Evaluation w.r.t. the ensemble solutions

data	Υ_{of}				Υ_o				Υ_f			
	MOEA-PCE	EM-PCE	CB-PCE	FCB-PCE	MOEA-PCE	EM-PCE	CB-PCE	FCB-PCE	MOEA-PCE	EM-PCE	CB-PCE	FCB-PCE
Iris	1.019	.914	.984	.989	1.025	1.004	1.044	1.039	.953	.906	.986	.977
Wine	.993	.960	1.074	1.072	1.060	.991	1.057	1.056	1.018	.952	1.001	1.001
Glass	1.023	.918	1	1.003	1.114	.971	1.064	1.066	.979	.915	1.004	1.004
Ecoli	1.074	1.052	1.058	1.015	1.034	1.023	1.027	1.028	.975	.924	.986	.992
Yeast	1.074	1.050	1.217	1.189	1.189	1.182	1.310	1.297	.960	1.021	1.036	1.037
Segmentation	1.008	.851	1.305	1.308	1.367	1.304	1.788	1.786	.971	.969	1.032	1.013
Abalone	1.044	1.001	1.068	1.071	1.121	1.102	1.208	1.208	.982	.902	.980	.986
Letter	1.040	1.001	1.045	1.088	1.118	1.099	1.277	1.274	.981	.891	1.169	.998
Trace	1.170	1.207	1.196	1.196	1.325	1.501	1.503	1.503	.949	.927	1.062	1.062
ControlChart	1.034	1.006	1.152	1.152	1.162	1.237	1.903	1.903	1.085	.577	1.234	1.234
min	.993	.851	.98	.989	1.025	.971	1.027	1.028	.949	.577	.980	.977
max	1.170	1.207	1.305	1.308	1.367	1.501	1.903	1.903	1.085	1.021	1.234	1.234
avg	1.048	.996	1.110	1.108	1.152	1.141	1.318	1.316	.985	.898	1.049	1.030

Table 5: Execution times (milliseconds)

data	TOTAL				ONLINE				OFFLINE			
	MOEA-PCE	EM-PCE	CB-PCE	FCB-PCE	MOEA-PCE	EM-PCE	CB-PCE	FCB-PCE	MOEA-PCE	EM-PCE	CB-PCE	FCB-PCE
Iris	17,223	55	13,235	906	17,223	53	343	372	-	2	12,892	534
Wine	21,098	184	50,672	993	21,098	153	306	323	-	31	50,366	670
Glass	61,700	281	110,583	3,847	61,700	239	1,713	1,713	-	42	108,870	2,134
Ecoli	94,762	488	137,270	4,911	94,762	427	1,643	1,689	-	61	135,627	3,222
Yeast	1,310,263	1,477	2,218,128	56,704	1,310,263	477	12,159	12,157	-	1,000	2,205,969	44,547
Segmentation	1,250,732	11,465	6,692,111	47,095	1,250,732	8,496	6,095	5,126	-	2,969	6,686,016	41,969
Abalone	13,245,313	34,000	19,870,218	527,406	13,245,313	12,922	107,547	90,078	-	21,078	19,762,671	437,328
Letter	7,765,750	54,641	26,934,327	271,064	7,765,750	28,766	15,593	15,610	-	25,875	26,918,734	255,454
Trace	86,179	4,880	2,589,899	3,731	86,179	3,224	836	840	-	1,656	2,589,063	2,891
ControlChart	291,856	2,313	3,383,936	12,439	291,856	735	2,717	2,783	-	1,578	3,381,219	9,656

as FCB-PCE, it was superior to MOEA-PCE and EM-PCE on seven and eight out of ten datasets, respectively. Υ_o and Υ_f results followed similar trends as Υ_{of} .

CB-PCE was still predominant on FCB-PCE, even if the difference between the two methods is less evident than the evaluation w.r.t. the reference classification. The average gains of CB-PCE w.r.t. FCB-PCE were 0.002 (Υ_{of}), 0.002 (Υ_o), and 0.019 (Υ_f).

4.2.2 Efficiency

Table 5 reports on the runtimes of the proposed algorithms CB-PCE and FCB-PCE, along with those of the early MOEA-PCE and EM-PCE. The reported times (expressed in milliseconds) are organized to distinguish between the online and offline phases.

The total runtimes confirm the theoretical considerations made in Sect. 3.2.3. Indeed, FCB-PCE is always faster than CB-PCE (from 2 to 3 orders of magnitude) and MOEA-PCE (1-2 orders), as well as CB-PCE is always slower than EM-PCE (2-3 orders) and slower than MOEA-PCE (up to 2 orders) on all datasets but Iris. The latter observation fully complies with the analysis of the relative performance between CB-PCE and MOEA-PCE: CB-PCE is generally outperformed by MOEA-PCE, except for the datasets having small size and/or low dimensionality, like Iris.

FCB-PCE would appear generally slower than EM-PCE. However, as stated in Sect. 3.2.3, the relative performance of the two methods mostly depends on the size $|\mathcal{D}|$ of the dataset and the dimensionality $|\mathcal{F}|$ of the data objects within \mathcal{D} and the number K of clusters; in particular, the larger $|\mathcal{D}| + |\mathcal{F}|$ and/or the smaller K , the better relative performance of FCB-PCE w.r.t. EM-PCE.

As a final remark, we note that the runtimes of the proposed CB-PCE and FCB-PCE were roughly similar to each other in the online phase. As expected, the difference between the two methods depends only on their offline phases, which are influenced by the adoption of the measures \hat{J} and \hat{J}_{fast} (cf. (8) and (24)).

5. CONCLUSION

Recent advance in data clustering resulted in the introduction of a new problem, called projective clustering ensembles (PCE), whose goal is to derive a robust projective consensus clustering from an ensemble of projective clustering solutions. PCE has been originally formulated as a two-objective or a single-objective optimization problem, and related heuristics have been developed focusing either on effectiveness or efficiency aspects. In this paper we addressed the main issues in existing PCE methods: none of them exploits approaches commonly adopted for solving the clustering ensemble problem, thus missing a wealth of experience gained by the majority of clustering ensemble methods. More importantly, the two-objective PCE is not capable of treating the object-to-cluster and the feature-to-cluster assignments as interrelated. We defined an alternative formulation of PCE as a new single-objective problem in which the objective function is able to take into account the object- and feature-based cluster representations as a whole in a notion of distance for projective clustering solutions. We developed two heuristics of such a new formulation, namely CB-PCE and FCB-PCE, which follow the cluster-based approach to the clustering ensembles problem. Experiments on benchmark datasets have shown that the proposed algorithms out-

perform in accuracy the early PCE methods, and FCB-PCE is faster than the early two-objective PCE.

6. REFERENCES

- [1] E. Achtert, C. Böhm, H. Kriegel, P. Kröger, I. Müller-Gorman, and A. Zimek. Detection and Visualization of Subspace Cluster Hierarchies. In *Proc. DASFAA Conf.*, pages 152–163, 2007.
- [2] C. C. Aggarwal, C. M. Procopiuc, J. L. Wolf, P. S. Yu, and J. S. Park. Fast Algorithms for Projected Clustering. In *Proc. SIGMOD Conf.*, pages 61–72, 1999.
- [3] H. Ayad and M. S. Kamel. Finding Natural Clusters Using Multi-Clusterer Combiner Based on Shared Nearest Neighbors. In *Proc. Int. Workshop on Multiple Classifier Systems (MCS)*, pages 166–175, 2003.
- [4] J. P. Barthélemy and B. Leclerc. The Median Procedure for Partitions. *Partitioning Data Sets*, 19:3–33, 1995.
- [5] C. Böhm, K. Kailing, H. P. Kriegel, and P. Kröger. Density Connected Clustering with Local Subspace Preferences. In *Proc. ICDM Conf.*, pages 27–34, 2004.
- [6] C. Boulis and M. Ostendorf. Combining Multiple Clustering Systems. In *Proc. PKDD Conf.*, pages 63–74, 2004.
- [7] P. S. Bradley and U. M. Fayyad. Refining Initial Points for K-Means Clustering. In *Proc. ICML Conf.*, pages 91–99, 1998.
- [8] L. Chen, Q. Jiang, and S. Wang. A Probability Model for Projective Clustering on High Dimensional Data. In *Proc. ICDM Conf.*, pages 755–760, 2008.
- [9] F. Chierichetti, R. Kumar, S. Pandey, and S. Vassilvitskii. Finding the Jaccard Median. In *Proc. SODA Conf.*, pages 293–311, 2010.
- [10] E. Dimitriadou, A. Weingesse, and K. Hornik. Voting-Merging: An Ensemble Method for Clustering. In *Proc. ICANN Conf.*, pages 217–224, 2001.
- [11] S. Dudoit and J. Fridlyand. Bagging to Improve the Accuracy of a Clustering Procedure. *Bioinformatics*, 19(9):1090–1099, 2003.
- [12] B. Fischer and J. M. Buhmann. Bagging for Path-Based Clustering. *TPAMI*, 25(11):1411–1415, 2003.
- [13] A. L. N. Fred. Finding Consistent Clusters in Data Partitions. In *Proc. Int. Workshop on Multiple Classifier Systems (MCS)*, pages 309–318, 2001.
- [14] G. Gan, C. Ma, and J. Wu. *Data Clustering: Theory, Algorithms, and Applications*. ASA-SIAM Series on Statistics and Applied Probability, 2007.
- [15] A. Gionis, H. Mannila, and P. Tsaparas. Clustering Aggregation. *TKDD*, 1(1), 2007.
- [16] F. Gullo, C. Domeniconi, and A. Tagarelli. Projective Clustering Ensembles. In *Proc. ICDM Conf.*, pages 794–799, 2009.
- [17] F. Gullo, A. Tagarelli, and S. Greco. Diversity-Based Weighting Schemes for Clustering Ensembles. In *Proc. SDM Conf.*, pages 437–448, 2009.
- [18] A. K. Jain and R. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, 1988.
- [19] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comp.*, 20(1):359–392, 1998.
- [20] L. I. Kuncheva, S. T. Hadjitodorov, and L. P. Todorova. Experimental Comparison of Cluster Ensemble Methods. In *Proc. Int. Conf. on Information Fusion*, pages 1–7, 2006.
- [21] R. P. Li and M. Mukaidono. Gaussian clustering method based on maximum-fuzzy-entropy interpretation. *Fuzzy Sets and Systems*, 102(2):253–258, 1999.
- [22] N. Nguyen and R. Caruana. Consensus Clustering. In *Proc. ICDM Conf.*, pages 607–612, 2007.
- [23] A. Patrikainen and M. Meila. Comparing subspace clusterings. *TKDE*, 18(7):902–916, 2006.
- [24] C. M. Procopiuc, M. Jones, P. K. Agarwal, and T. M. Murali. A Monte Carlo algorithm for fast projective clustering. In *Proc. SIGMOD Conf.*, pages 418–427, 2002.
- [25] K. Sequeira and M. Zaki. SCHISM: A New Approach for Interesting Subspace Mining. In *Proc. ICDM Conf.*, pages 186–193, 2004.
- [26] A. Strehl, J. Ghosh, and R. Mooney. Impact of Similarity Measures on Web-Page Clustering. In *Proc. of AAAI Workshop on AI for Web Search*, pages 58–64, 2000.
- [27] A. Asuncion and D. Newman. UCI Machine Learning Repository, <http://archive.ics.uci.edu/ml/>.
- [28] A. Strehl and J. Ghosh. Cluster Ensembles — A Knowledge Reuse Framework for Combining Multiple Partitions. *J. Mach. Learn. Res.*, 3:583–617, 2002.
- [29] C. Domeniconi and M. Al-Razgan. Weighted Cluster Ensembles: Methods and Analysis. *TKDD*, 2(4), 2009.
- [30] C. Domeniconi, D. Gunopulos, S. Ma, B. Yan, M. Al-Razgan, and D. Papadopoulos. Locally Adaptive Metrics for Clustering High Dimensional Data. *Data Mining and Knowledge Discovery*, 14(1):63–97, 2007.
- [31] E. Achtert, C. Böhm, H. Kriegel, P. Kröger, I. Müller-Gorman, and A. Zimek. Finding Hierarchies of Subspace Clusters. In *Proc. PKDD Conf.*, pages 446–453, 2006.
- [32] E. Ka Ka Ng, A. W.-C. Fu, and R. C.-W. Wong. Projective Clustering by Histograms. *TKDE*, 17(3):369–383, 2005.
- [33] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana. The UCR Time Series Classification/Clustering Page, http://www.cs.ucr.edu/~eamonn/time_series_data/.
- [34] G. Moise, J. Sander, and M. Ester. Robust projected clustering. *KAIS*, 14(3):273–298, 2008.
- [35] M. L. Yiu and N. Mamoulis. Iterative Projected Clustering by Subspace Mining. *TKDE*, 17(2):176–189, 2005.
- [36] X. Z. Fern and C. Brodley. Solving Cluster Ensemble Problems by Bipartite Graph Partitioning. In *Proc. ICML Conf.*, pages 281–288, 2004.
- [37] K. Y. Yip, D. W. Cheung, and M. K. Ng. On Discovery of Extremely Low-Dimensional Clusters using Semi-Supervised Projected Clustering. In *Proc. ICDE Conf.*, pages 329–340, 2005.