Projective Clustering Ensembles

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

Carlotta Domeniconi
Department 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

Recent advances in data clustering have regarded clustering ensembles *and* projective clustering *methods, which distinctly aim to face typical issues in many clustering problems. In this paper, we address for the first time the* projective clustering ensembles *(PCE) problem, whose main goal is to derive a proper* projective consensus partition *from an ensemble of projective clustering solutions. We formalize PCE as an optimization problem which is designed to satisfy strong requirements on the independence on the specific clustering ensembles algorithm, ability to handle hard as well as soft data clustering, and different feature weightings. Specifically, we provide two formulations for PCE, namely a two-objective and a single-objective problem, in which the object-based and feature-based representations of the ensemble solutions are differently taken into account. Experiments have demonstrated the significance of the proposed methods for PCE, showing clear improvements in terms of accuracy of the output consensus partition.*

1. Introduction

Research on data clustering [8] has traditionally assumed that, given a set of input data and a clustering problem for that data, (i) such a problem is addressed by a clustering method, which is usually equipped with a certain distance/similarity measure, and (ii) all the features (dimensions) of the given data are considered in the clustering task.

The above assumptions are usually given for enabling a proposed approach to satisfy some special requirements for data clustering, such as simplicity, practical applicability, understandability of the results, and low computational cost. On the other hand, such assumptions may bring any clustering method to incur serious issues in both effectiveness and efficiency, especially when (1) the problem at hand is inherently multi-faceted as there is a number of (differently relevant) aspects according to which a clustering task is worth of being performed, and/or (2) the input data is highly dimensional. Issue 1 is related to the fact that a solution for the clustering problem is inevitably biased due to the peculiarities of the specific clustering algorithm being used. Issue 2 is instead related to the so-called *curse-of-dimensionality*, which breaks down the significance of the concept of proximity (thus, cluster) as the number of dimensions or features increases.

In relatively recent years, methodologies have been studied to distinctly address the above issues in clustering problems, orthogonally to the existing literature on clustering algorithms and data proximity measures.

Clustering ensembles [12, 13, 7, 5] has recently emerged as a powerful tool to face issue 1. Given a data collection, a set of clustering solutions, or *ensemble*, can be generated by varying one or more aspects, such as the clustering algorithm, the parameter setting, and the number of features, objects or clusters. Given an ensemble, the objective is to extract a *consensus partition*, i.e., a clustering solution that maximizes some objective function (the *consensus function*), which is defined by taking into account different information available from the ensemble.

Concerning the aforementioned issue 2, a major consequence of the high dimensionality is that not all features are relevant for all data in a cluster analysis. Due to the sparsity naturally occurring in the representation of data, it is unlikely for the data to form meaningful clusters in the full dimensional space. Traditional feature selection and extraction aim to reduce the number of dimensions, but they treat the dataset as a whole; consequently, some dimensions potentially relevant for part of the data might be filtered out. *Projective clustering* [10, 14, 1, 9] aims to discover clusters which correspond to subsets of the input data and have different (possibly overlapping) dimensional subspaces associated to them. Projected clusters tend to be less noisy—because each group of data is represented over a subspace which does not contain irrelevant dimensions—and more understandable—because the exploration of a cluster is easier when few dimensions are involved.

Projective clustering is also related to the *subspace clustering* problem, whose main goal is to find clustering structures in every possible subspace. As a result, a major difference between the two problems is that projective clustering outputs a single partition of the input set of data objects, whereas subspace clustering methods aim to find a set of clustering solutions, each one having clusters defined in a specific subspace.

In this paper, the problem of *projective clustering ensembles* (PCE) is addressed for the first time. The objective is to define methods for clustering ensembles that are able to deal with ensembles of projective clustering solutions and provide a projective consensus partition. In particular, we focus on ensembles composed by *axis-aligned* (or *axis-parallel*) projective clustering solutions, i.e., solutions in which the subspace associated to each cluster is given by a subset of the original feature space.

The projective consensus partition to be discovered is computed as a solution of an optimization problem formulated by exploiting information available from the input ensemble. Since we are interested in developing general methods for PCE, such objective functions should meet the following strong requirements: (i) to discard the original feature values of the input data; (ii) to be independent of the specific clustering algorithm and of any prior knowledge on the setup for ensemble generation; (iii) to handle hard as well as soft data clustering in a projective setting; (iv) to allow for unequally weighted feature-to-cluster assignments.

Within this view, we propose two formulations of PCE, namely a *two-objective* and a *single-objective*. The first one involves two objective functions which consider the data object clustering and feature-to-cluster assignment, respectively; the second formulation has a unique objective function which acts as an error criterion in the computation of any cluster (of a candidate clustering solution) by involving both the object-based representation and the feature-based representation of the cluster.

For each of the two proposed formulations of PCE, we developed well-founded heuristics, in which a *multi-objective evolutionary strategy* [2] and an EM-like approach are employed. Experiments conducted on ten benchmark datasets have shown that both the proposed algorithms lead to more accurate consensus partitions, in terms of internal similarity with respect to reference classifications (i.e., external classifications and clustering ensembles) and in terms of intra-cluster error-rate.

Among the existing clustering ensemble and projective clustering methods in the literature, the *Weighted Subspace Bipartite Partitioning Algorithm* (WSBPA) [5] is somehow related to the approaches proposed in this work. However, we point out that WSBPA does not represent a valid solution for the projective clustering ensembles problem, since it does not satisfy any of the aforementioned requirements. Indeed, WSPA requires to access the original features of the data objects, works only if the projective solutions are generated by running a specific projective clustering algorithm (i.e., LAC), and it does not deal with projective solutions that are soft at data clustering level.

2. Projective Clustering Ensembles

Definition 1 (projective clustering solution) *Let* $\mathcal{D} = \{\vec{o}_1, \ldots, \vec{o}_N\}$ *be a set of D-dimensional points (data objects). A* projective clustering solution C *defined over* $\mathcal{D}$ *is a triple* $\langle \mathcal{L}, \Gamma, \Delta \rangle$*:*

- $\mathcal{L} = \{\ell_1, \ldots, \ell_K\}$ *is a set of cluster labels which uniquely represent the* K *clusters*
- $\Gamma : \mathcal{L} \times \mathcal{D} \rightarrow S_\Gamma$ *is a function which stores the probability that object* $\vec{o}_n$ *belongs to the cluster labeled with* ℓ_k*,* $\forall k \in [1..K], n \in [1..N]$*, such that* $\sum_{k=1}^{K} \Gamma_{kn} = 1, \forall n \in [1..N]$*, where* Γ_{kn} *hereinafter refers to* $\Gamma(\ell_k, \vec{o}_n)$
- $\Delta : \mathcal{L} \times [1..D] \rightarrow [0, 1]$ *is a function which stores the probability that the* d*-th feature is a relevant dimension for the objects in the cluster labeled with* ℓ_k*,* $\forall k \in [1..K], d \in [1..D]$*, such that* $\sum_{d=1}^{D} \Delta_{kd} = 1, \forall k \in [1..K]$*, where* Δ_{kd} *hereinafter refers to* $\Delta(\ell_k, d)$

Definition 2 (projective ensemble) *Given a set* $\mathcal{D}$ *of data objects, a* projective ensemble *defined over* $\mathcal{D}$ *is a set* $\mathcal{E} = \{C_1, \ldots, C_M\}$*, where each* $C_m = \langle \mathcal{L}^{(m)}, \Gamma^{(m)}, \Delta^{(m)} \rangle$ *is a projective clustering solution defined over* $\mathcal{D}$*,* $\forall m \in [1..M]$*, and* $\mathcal{L}^{(i)} \cap \mathcal{L}^{(j)} = \emptyset$*,* $\forall i, j \in [1..M], i \neq j$*.*

Definition 3 (ensemble label set) *Let* $\mathcal{E} = \{C_1, \ldots, C_M\}$ *be a projective ensemble, where* $C_m = \langle \mathcal{L}^{(m)}, \Gamma^{(m)}, \Delta^{(m)} \rangle$*,* $\forall m \in [1..M]$*. The* ensemble label set *of* $\mathcal{E}$ *is defined as* $\mathbf{L} = \{\mathbf{l}_1, \ldots, \mathbf{l}_H\} = \bigcup_{m=1}^{M} \mathcal{L}^{(m)}$*.*

Definition 4 (projective cluster representation) *Let* $\mathcal{D} = \{\vec{o}_1, \ldots, \vec{o}_N\}$ *be a set of D-dimensional data objects and* $\mathcal{E}$ *be a projective ensemble defined over* $\mathcal{D}$*. The N-dimensional* object-based representation *and the D-dimensional* feature-based representation *for the cluster labeled with* $\mathbf{l}_h$*,* $\forall h \in [1..H]$*, are given by the vectors* $\vec{\gamma}_h$ *and* $\vec{\delta}_h$*, respectively, which are defined as follows:*

$$\vec{\gamma}_h = (\Gamma'_{k'1}, \ldots, \Gamma'_{k'N}) \qquad \vec{\delta}_h = (\Delta'_{k'1}, \ldots, \Delta'_{k'D})$$

where the Γ' *and* Δ' *functions are involved in the solution* $C' \in \mathcal{E}$ *such that* $C' = \langle \mathcal{L}', \Gamma', \Delta' \rangle$*,* $\mathcal{L}' = \{\ell'_1, \ldots, \ell'_{K'}\}$*,* $\mathbf{l}_h \in \mathcal{L}'$*, and* $k' \in [1..K']$ *is the index such that* $\ell'_{k'} = \mathbf{l}_h$*.*

2.1. Two-objective PCE

A projective consensus partition $C^* = \langle \mathcal{L}^*, \Gamma^*, \Delta^* \rangle$ derived from an ensemble $\mathcal{E}$ should meet two different kinds of requirements: the first one is related to the data object clustering of the solutions in $\mathcal{E}$, whereas the other one regards the feature-to-cluster assignment of the solutions in $\mathcal{E}$. For this purpose, the PCE problem can be naturally formulated as a two-objective optimization problem:

$$C^* = \arg\min_{\hat{C}} \left[\Psi_o(\hat{C}, \mathcal{E}, \mathcal{D}),\ \Psi_f(\hat{C}, \mathcal{E}, \mathcal{D})\right] \qquad (1)$$

where Ψ_o and Ψ_f are two optimization functions that account for the data clustering and the feature-to-cluster assignment of the projective clusterings in $\mathcal{E}$, respectively, and are defined as follows:

$$\Psi_o(\hat{C}, \mathcal{E}, \mathcal{D}) = \sum_{C \in \mathcal{E}} \overline{\psi}_o(C, \hat{C}) \qquad (2)$$

$$\Psi_f(\hat{C}, \mathcal{E}, \mathcal{D}) = \sum_{C \in \mathcal{E}} \overline{\psi}_f(C, \hat{C}) \qquad (3)$$

where $\overline{\psi}_o(C_i, C_j)$ (resp., $\overline{\psi}_f(C_i, C_j)$) is a function that measures the distance between the projective clustering solutions $C_i = \langle \mathcal{L}^{(i)}, \Gamma^{(i)}, \Delta^{(i)} \rangle$ and $C_j = \langle \mathcal{L}^{(j)}, \Gamma^{(j)}, \Delta^{(j)} \rangle$ in terms of their corresponding object-based partitioning (resp., feature-to-cluster assignment):

$$\overline{\psi}_o(C_i, C_j) = \frac{1}{2}\Big(\psi_o(C_i, C_j) + \psi_o(C_j, C_i)\Big) \qquad (4)$$

$$\overline{\psi}_f(C_i, C_j) = \frac{1}{2}\Big(\psi_f(C_i, C_j) + \psi_f(C_j, C_i)\Big) \qquad (5)$$

where

$$\psi_o(C_i, C_j) = \frac{1}{|\mathcal{L}^{(i)}|} \sum_{k=1}^{|\mathcal{L}^{(i)}|} \Big(1 - \max_{k' \in [1..|\mathcal{L}^{(j)}|]} J\big(\vec{a}_k^{(i)}, \vec{a}_{k'}^{(j)}\big)\Big)$$

$$\psi_f(C_i, C_j) = \frac{1}{|\mathcal{L}^{(i)}|} \sum_{k=1}^{|\mathcal{L}^{(i)}|} \Big(1 - \max_{k' \in [1..|\mathcal{L}^{(j)}|]} J\big(\vec{b}_k^{(i)}, \vec{b}_{k'}^{(j)}\big)\Big)$$

with $\vec{a}_z^{(y)} = \big(\Gamma_{z1}^{(y)}, \ldots, \Gamma_{zN}^{(y)}\big)$, $\vec{b}_z^{(y)} = \big(\Delta_{z1}^{(y)}, \ldots, \Delta_{zN}^{(y)}\big)$, and $J\big(\vec{u}, \vec{v}\big) = \big(\vec{u}\,\vec{v}\big)/\big(\|\vec{u}\|^2 + \|\vec{v}\|^2 - \vec{u}\,\vec{v}\big)$ ranging within $[0, 1]$ and denoting the extended Jaccard similarity coefficient between two any real-valued vectors $\vec{u}$ and $\vec{v}$ [8].

The MOEA-PCE algorithm. The NP-hard problem P defined in Eq. (1) is a multi-objective optimization problem, in which the objectives are conflicting with each other; consequently, it is hard to solve, since traditional optimization techniques do not apply. An approach that has been recognized as particularly appropriate for this kind of problem is given by the *Multi Objective Evolutionary Algorithms (MOEAs)* [2]. These methods are able to maintain the underlined multi-objective structure, i.e., they work without requiring to combine the objectives into a single one.

Within this view, in order to provide a valuable heuristic for P, we resort to the MOEAs domain and define the proposed *MOEA-based Projective Clustering Ensembles (MOEA-PCE)* algorithm. In particular, we exploit the elitist MOEA *Nondominated Sorting Genetic Algorithm-II (NSGA-II)* [3], whose evolutionary strategy is based on the notion of *Pareto-ranking*.

Definition 5 (domination) *Let P be a multi-objective optimization problem of the form $\{x^* = \arg\min_{\hat{x}}[f_1(\hat{x}), \ldots, f_s(\hat{x})]\}$, and x' and x'' two candidate solutions of P. x'* dominates *x'' ($x' \prec x''$) if and only if $f_i(x') \leq f_i(x'')$, $\forall i \in [1..s]$, and* (ii) *$\exists j \in [1..s]\ :\ f_j(x') < f_j(x'')$.*

Definition 6 (Pareto-optimality) *Let P be a multi-objective optimization problem of the form $\{x^* = \arg\min_{\hat{x}}[f_1(\hat{x}), \ldots, f_s(\hat{x})]\}$, and $\mathcal{S}$ a population of individuals for P, i.e., a set of candidate solutions of P. $\mathcal{S}_P^* \subseteq \mathcal{S}$ is a* Pareto-optimal *solution set of P w.r.t. $\mathcal{S}$ if and only if $x \nprec x^*$, $\forall x \in \mathcal{S}$, $\forall x^* \in \mathcal{S}_P^*$.*

Definition 7 (Pareto-ranking) *Let P be a multi-objective optimization problem of the form $\{x^* = \arg\min_{\hat{x}}[f_1(\hat{x}), \ldots, f_s(\hat{x})]\}$, and $\mathcal{S}$ a population of individuals for P. The* Pareto-ranking *function $\rho : \mathcal{S} \to \mathbb{N}$ for P is defined as $\rho(x) = \min\{r \in \mathbb{N}, r > 0\ :\ x \in \mathcal{S}_{P,r}^*\}$, $\forall x \in \mathcal{S}$, where $\mathcal{S}_{P,z}^*$ is the Pareto-optimal solution set of P w.r.t. the population $\mathcal{S}_{P,z} = \{x' \in S\ :\ \rho(x') \geq z\}$.*

The*MOEA-PCE algorithm* (Algorithm 1)[1] starts by randomly generating the initial population $\mathcal{S}$ (Line 1), and proceeds by performing the main loop until a maximum number I of iterations has been reached (Lines 3-9). At each iteration, the Pareto-ranking function ρ, defined w.r.t. the current population $\mathcal{S}$, is computed according to Definition 7, where the problem denoted with P is the one reported in Eq. (1) (Line 4). The procedure used for computing ρ is the one described in [3]. The ρ values of each individual in $\mathcal{S}$ are then exploited for sorting $\mathcal{S}$ and partitioning it into two equal-sized subsets, i.e., $\mathcal{S}'$ and $\mathcal{S}''$, so that each individual in $\mathcal{S}'$ has a ρ value not greater than any other individual in $\mathcal{S}''$ (Line 5). The subset $\mathcal{S}'$ is involved into a crossover-and-mutation step, which is performed as described in [11] (Line 6). In particular, the mutation step consists in adding random Gaussian noise to the solutions in $\mathcal{S}'$. The result of this step is the "offspring" set $\mathcal{S}'_{CM}$ of new individuals, which, along with $\mathcal{S}'$, forms the new population (Line 7). Finally, the Pareto-optimal solution set $\mathcal{S}^*$ (i.e., the set of output projective consensus partitions) is derived from the population $\mathcal{S}$ computed at the last iteration (Line 11).

2.2. Single-objective PCE

The two-objective projective clustering ensembles formulation may incur issues concerning the parameter setting and the interpretation of the convergence criterion. Within this view, we alternatively propose a different, simpler for-

[1] The complexity of Algorithm 1is $\mathcal{O}(I\ t\ M\ K^2\ (N + D))$.

Algorithm 1 MOEA-PCE

```
Input: a projective ensemble $\mathcal{E}$ of size $M$, defined over a set $\mathcal{D}$ of $N$ $D$-dimensional objects; the number $K$ of clusters in the output projective consensus partitions; the population size $t$; the maximum number $I$ of iterations
Output: a set $\mathcal{S}^*$ of projective consensus partitions
$\mathcal{S} \leftarrow populationRandomGen(\mathcal{E}, t, K)$
$it \leftarrow 1$
repeat
    $\rho \leftarrow computeParetoRanking(\mathcal{S})$ {see Def. 7}
    $\langle \mathcal{S}', \mathcal{S}'' \rangle \leftarrow \langle \check{\mathcal{S}}' \subset \mathcal{S},\ \check{\mathcal{S}}'' \subset \mathcal{S} \rangle \ :\ |\check{\mathcal{S}}'| = |\mathcal{S}|/2,\ |\check{\mathcal{S}}''| = |\mathcal{S}|/2,\ \check{\mathcal{S}}' \cup \check{\mathcal{S}}'' = \mathcal{S},\ \rho(x') \leq \rho(x''), \forall x' \in \check{\mathcal{S}}', x'' \in \check{\mathcal{S}}''$
    $\mathcal{S}'_{CM} \leftarrow crossoverAndMutation(\mathcal{S}')$
    $\mathcal{S} \leftarrow \mathcal{S}' \cup \mathcal{S}'_{CM}$
    $it \leftarrow it + 1$
until $it = I$
$\rho \leftarrow computeParetoRanking(\mathcal{S})$
$\mathcal{S}^* \leftarrow \{x' \in \mathcal{S} \ :\ \rho(x') \leq \rho(x''), \forall x'' \in \mathcal{S}, x'' \neq x'\}$
```

Algorithm 2 EM-PCE

```
Input: a projective ensemble $\mathcal{E}$ of size $M$, defined over a set $\mathcal{D}$ of $N$ $D$-dimensional data objects; the number $K$ of clusters in the output projective consensus partition;
Output: the projective consensus partition $C^*$
$\mathcal{L}^* \leftarrow \{1, \ldots, K\}$
$\langle \Gamma^*, \Delta^* \rangle \leftarrow randomGen(\mathcal{E}, K)$
repeat
    compute $\Gamma^*$ according to Eq. (10)
    compute $\Delta^*$ according to Eq. (11)
until $convergence$
$C^* = \langle \mathcal{L}^*, \Gamma^*, \Delta^* \rangle$
```

mulation that is based on a single objective function :

$$C^* \quad = \quad \arg\min_{\hat{C}} Q(\hat{C}, \mathcal{E}) \tag{6}$$

$$s.t.$$

$$\sum_{k=1}^{K} \hat{\Gamma}_{kn} = 1, \quad \forall n \in [1..N] \tag{7}$$

$$\sum_{d=1}^{D} \hat{\Delta}_{kd} = 1, \quad \forall k \in [1..K] \tag{8}$$

$$\hat{\Gamma}_{kn} \geq 0,\ \hat{\Delta}_{kd} \geq 0, \quad \forall k \in [1..K], n \in [1..N], d \in [1..D] \tag{9}$$

where $Q(\hat{C}, \mathcal{E}) = \sum_{k=1}^{K} \sum_{n=1}^{N} \hat{\Gamma}_{kn}^{\alpha} \sum_{h=1}^{H} \gamma_{hn} \sum_{d=1}^{D} \big(\hat{\Delta}_{kd} - \delta_{hd}\big)^2$ and $\alpha > 1$ is an integer that guarantees the nonlinearity of Q w.r.t. $\hat{\Gamma}_{kn}$, which is needed for ensuring that the values of $\hat{\Gamma}_{kn}$ range within $[0, 1]$ (instead of $\{0, 1\}$).

The EM-PCE algorithm. In order to provide a heuristic solution for the NP-hard problem defined in Eq. (6)-(9), we define a novel procedure that is inspired to the popular *Expectation Maximization (EM)* algorithm [4].

The proposed algorithm, i.e., *EM-based Projective Clustering Ensembles (EM-PCE)* (Algorithm 2),[2] consists of two main EM-like steps, which are iteratively repeated until a convergence criterion is met. Such steps exploit the function Q and aim to find an optimal solution for $\hat{\Gamma}_{kn}$ (resp., $\hat{\Delta}_{kd}$) values, while maintaining fixed $\hat{\Delta}_{kd}$ (resp., $\hat{\Gamma}_{kn}$) values. The basic equations for the two steps are:

$$\Gamma^*_{kn} = \left[\sum_{k'=1}^{K} \left(\frac{X_{kn}}{X_{k'n}} \right)^{\frac{1}{\alpha-1}} \right]^{-1} \tag{10}$$

$$\Delta^*_{kd} = \frac{Z_{kd}}{Y_k} \tag{11}$$

where $X_{kn} = \sum_{h=1}^{H} \gamma_{hn} \sum_{d=1}^{D} \big(\hat{\Delta}_{kd} - \delta_{hd}\big)^2$, $Y_k = \sum_{n=1}^{N} \hat{\Gamma}_{kn}^{\alpha} \sum_{h=1}^{H} \gamma_{hn}$, and $Z_{kd} = \sum_{n=1}^{N} \hat{\Gamma}_{kn}^{\alpha} \sum_{h=1}^{H} \gamma_{hn}\, \delta_{hd}$.

The expressions reported in Eq. (10) and (11), i.e., the solutions for the problem P defined in Eq. (6)-(9), have been derived by means of the conventional Lagrange multipliers method, considering the relaxed problem P' obtained by temporarily discarding the inequality constraints from the constraint set of P. In particular, we defined the new (unconstrained) objective function Q_λ for P' as $Q_\lambda(\hat{C}, \mathcal{E}) = Q(\hat{C}, \mathcal{E}) + \sum_{n=1}^{N} \lambda'_n \ \big(\sum_{k'=1}^{K} \hat{\Gamma}_{k'n} - 1\big) + \sum_{k=1}^{K} \lambda''_k \ \big(\sum_{d'=1}^{D} \hat{\Delta}_{kd'} - 1\big)$, and, for a fixed assignment of $\hat{\Delta}_{kd}$, we computed the optimal Γ^*_{kn} by solving the system of equations given by $\partial\, Q_\lambda / \partial\, \hat{\Gamma}_{kn} = \alpha\, (\hat{\Gamma}_{kn})^{\alpha-1}\, X_{kn} + \lambda'_n = 0$ and $\partial\, Q_\lambda / \partial\, \lambda'_n = \sum_{k'=1}^{K} \hat{\Gamma}_{k'n} - 1 = 0$, whose solution is given by Eq. (10). Analogously, for a fixed assignment of $\hat{\Gamma}_{kn}$, we compute the optimal Δ^*_{kd} by solving the equations $\partial\, Q_\lambda / \partial\, \hat{\Delta}_{kd} = \sum_{n=1}^{N} \hat{\Gamma}_{kn}^{\alpha} \sum_{h=1}^{H} 2\, \gamma_{hn} \Big(\hat{\Delta}_{kd} - \delta_{hd}\Big) + \lambda''_k = 0$ $\partial\, Q_\lambda / \partial\, \lambda''_k = \sum_{d'=1}^{D} \hat{\Delta}_{kd'} - 1 = 0$ which are solved by Eq. (10). Since, according to the solutions for P' reported in Eq. (10) and (11), it holds that $\Gamma^*_{kn} \geq 0$, $\Delta^*_{kd} \geq 0$, $\forall k \in [1..K], n \in [1..N], d \in [1..D]$, then such solutions satisfy the inequality constraints that were temporarily discarded in order to define the relaxed problem P'; thus, they represent the optimal solutions of the original problem P.

3. Experimental evaluation

3.1. Evaluation methodology

Datasets. We used eight benchmark datasets from the UCI Machine Learning Repository,[3] namely Iris, Wine, Glass, Ecoli, Yeast, Segmentation, Abalone and Letter, and two time-series datasets from the UCR Time Series Classification/Clustering Page,[4] namely Tracedata and ControlChart. Table 1 reports on the main characteristics of the selected datasets.

[2] Algorithm 2 works in $\mathcal{O}(I\ M\ K^2\ N\ D)$, where I is the number of iterations needed for the convergence.

[3] http://archive.ics.uci.edu/ml/

[4] http://www.cs.ucr.edu/∼eamonn/time_series_data/

Table 1. 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

Ensemble generation. For each set of experiments and dataset we generated 20 different ensembles; all the reported results were averaged over the results obtained on each such ensembles. Ensembles for each dataset were generated by running the LAC algorithm [6], where the diversity of the solutions was guaranteed by randomly choosing the initial centroids and varying the parameter $\mathbf{h}$ in LAC.[5] LAC yields projective clusterings that are hard at data clustering level and have feature-to-cluster assignments unequally weighted; consequently, in order to test the ability of the proposed algorithms to deal also with soft clustering solutions and with solutions having feature-to-cluster assignments equally weighted, we generated each ensemble $\mathcal{E}$ as a composition of four equal-sized subsets, namely $\mathcal{E}_1$, $\mathcal{E}_2$, $\mathcal{E}_3$, and $\mathcal{E}_4$ such that:

- $\mathcal{E}_1$ contains solutions hard at data clustering level and having feature-to-cluster assignments unequally weighted, i.e., solutions obtained by standard LAC;
- $\mathcal{E}_2$ contains solutions that are hard at data clustering level and have feature-to-cluster assignments equally weighted. Starting from a LAC solution $C = \langle \mathcal{L}, \Gamma, \Delta \rangle$ defined over a set of N D-dimensional objects, where $\mathcal{L} = \{\ell_1, \ldots, \ell_K\}$, we derive the corresponding projective clustering C', having feature-to-cluster assignments equally weighted, as follows: $C' = \langle \mathcal{L}, \Gamma, \Delta' \rangle$, where $\Delta'_{kd} = \lfloor \Delta_{kd} + 1/D \rfloor$, $\forall k \in [1..K], d \in [1..D]$;
- $\mathcal{E}_3$ contains solutions that are soft at data clustering level and have feature-to-cluster assignments unequally weighted. Starting from a LAC solution $C = \langle \mathcal{L}, \Gamma, \Delta \rangle$ defined over a set of N D-dimensional objects, where $\mathcal{L} = \{\ell_1, \ldots, \ell_K\}$, we derive the corresponding soft projective clustering C'' as follows: $C'' = \langle \mathcal{L}, \Gamma'', \Delta \rangle$, where $\Gamma''_{kn} = \Pr(k|n)$, $\forall k \in [1..K], n \in [1..N]$. $\Pr(k|n)$ is the probability of the cluster labeled with ℓ_k given the observation of the object $\vec{o}_n$, which is computed as described in [5].
- $\mathcal{E}_4$ contains solutions that are soft at data clustering level and have feature-to-cluster assignments equally weighted, which are derived from the standard LAC solutions according to the methods employed for generating $\mathcal{E}_2$ and $\mathcal{E}_3$, respectively.

Setting of the proposed algorithms. We experimentally observed that our methods were scarcely influenced by any specific setting, which allowed us to easily detect setup values well-suited to each of the evaluation datasets. Precisely, in case of the MOEA-PCE algorithm, the population size (t) was set equal to 15% of the ensemble size and the number I of maximum iterations equal to 200; also, the random Gaussian noise needed for the mutation step was obtained by performing a *Monte Carlo* sampling on a Gaussian probability density function with a null mean value and variance equal to one. In case of the EM-PCE algorithm, parameter α of the objective function Q was set equal to 2.

Evaluation criteria. For each dataset $\mathcal{D} = \{\vec{o}_1, \ldots, \vec{o}_N\}$, where $\vec{o}_n = (o_{n1}, \ldots, o_{nD}), \forall n \in [i..N]$, accuracy of the results by the proposed algorithms, i.e., accuracy of the consensus partition $\check{C} = \langle \check{\mathcal{L}}, \check{\Gamma}, \check{\Delta} \rangle$, $|\check{\mathcal{L}}| = \check{K}$, was evaluated in terms of:

1. *similarity w.r.t. the (hard) reference classification* $\widetilde{C}$, which is defined as follows. $\widetilde{C} = \langle \widetilde{\mathcal{L}}, \widetilde{\Gamma}, \widetilde{\Delta} \rangle$, where $\widetilde{\mathcal{L}} = \{\widetilde{\ell}_1, \ldots, \widetilde{\ell}_{\widetilde{K}}\}$ and $\widetilde{\Gamma}$ are directly available from $\mathcal{D}$, whereas $\widetilde{\Delta}$ is computed according to the following formula [6]: $\widetilde{\Delta}_{kd} = \big(\exp(-X_{kd}/\mathbf{h})\big) / \big(\sum_{d'=1}^{D} \exp(-X_{kd'}/\mathbf{h})\big)$, $\forall k \in [1..\widetilde{K}], d \in [1..D]$, where $X_{kd} = \big(\sum_{n=1}^{N} \widetilde{\Gamma}_{kn}\big)^{-1} \sum_{n=1}^{N} \widetilde{\Gamma}_{kn} \, \big(\bar{c}_{kd} - o_{nd}\big)^2$, $\bar{c}_{kd} = \big(\sum_{n=1}^{N} \widetilde{\Gamma}_{kn}\big)^{-1} \sum_{n=1}^{N} \widetilde{\Gamma}_{kn} \, o_{nd}$; also, parameter $\mathbf{h}$, in our experiments, was set equal to 0.2. The evaluation between $\check{C}$ and $\widetilde{C}$ was performed according to both object- and feature-based representations, by using $1 - \overline{\psi}_o$ (Eq. (4)) and $1 - \overline{\psi}_f$ (Eq. (5)), respectively;
2. *error-rate* (E) [6], which is an internal criterion and measures the intra-cluster compactness: $E(\check{C}) = \sum_{k=1}^{\check{K}} \sum_{d=1}^{D} (\check{\Delta}_{kd} / (\sum_{n=1}^{N} \check{\Gamma}_{kn}) \sum_{n=1}^{N} \check{\Gamma}_{kn} \, (\bar{c}_{kd} - o_{nd})^2$.

3.2. Results

For each algorithm, dataset and ensemble, we performed 50 different runs and reported average results, and maximum (best) results with relative standard deviation.

Evaluation w.r.t. reference classification. Table 2 and Table 3 show the performance on the various datasets in terms of similarity w.r.t. the reference classifications, by considering the object-based representation and the feature-based representation, respectively.

In both cases, the performances of the proposed algorithms lead to an average similarity of the consensus partition(s) that are comparable or far better than the average

[5] This parameter controls the incentive for clustering on more features depending on the strength of the correlation of data along the features

intra-ensemble similarity. According to the object-based representation (Table 2), the average improvements (gains) by MOEA-PCE and EM-PCE over all datasets are 13.6% and 4.3%, respectively, with peaks above 16% on five out of ten datasets by MOEA-PCE (up to 29% on Iris), and peaks above 10% on three datasets by EM-PCE (up to 13% on Iris). According to the feature-based representation (Table 3), the average improvements by MOEA-PCE and EM-PCE over all datasets are 13.3% and 7.3%, respectively.

Table 2. Similarity results w.r.t. reference classification (object-based representation)

	ensemble	*MOEA-PCE*			*EM-PCE*		
data	*avg-max*	*avg*	*max-std*	*gain w.r.t. ens. (avg)*	*avg*	*max-std*	*gain w.r.t. ens. (avg)*
Iris	.632 .925	.919	.925 .015	+.287	.762	.767 .040	+.130
Wine	.738 .910	.913	.928 .105	+.175	.782	.840 .028	+.044
Glass	.565 .775	.683	.768 .046	+.118	.639	.644 .002	+.074
Ecoli	.421 .689	.603	.686 .054	+.182	.329	.419 .040	-.092
Yeast	.675 .750	.723	.745 .015	+.048	.638	.641 .001	-.037
Segm.	.590 .821	.755	.835 .049	+.165	.653	.663 .004	+.063
Abal.	.509 .520	.518	.558 .043	+.009	.512	.542 .002	+.003
Letter	.522 .640	.597	.612 .031	+.075	.554	.562 .006	+.032
Trace	.772 .868	.862	.998 .059	+.090	.875	.935 .030	+.103
Contr.	.681 .981	.895	.965 .049	+.214	.790	.806 .007	+.109

Evaluation in terms of error rate. We also compared the performance of MOEA-PCE and EM-PCE to both the reference classification and the ensemble, for each dataset, in terms of error rate. Due to the limited space available, we do not report all the detailed results.

However, similarly to the previously discussed evaluations, this evaluation shows that MOEA-PCE outperforms the standard ensemble, obtaining an average improvement (gain) over all the datasets of +0.6 w.r.t. the reference classification and +0.358 w.r.t. the ensemble. EM-PCE also improves upon the error rate of the reference classification (+0.51) and of the ensemble (+0.27).

4. Conclusion

In this paper we addressed for the first time the *projective clustering ensembles problem* (PCE). Given an ensemble of projective clustering solutions, PCE aims to find a proper projective consensus partition, i.e., a new projective clustering computed by optimizing one or more criteria properly defined by exploiting the information from the ensemble. We proposed two different formulations of PCE, according to which the problem at hand was defined as a two- and single-objective optimization problem, respectively, and provided heuristic algorithms for solving both PCE problems. Experimental results have shown the validity of the proposed algorithms, showing improvements in terms of accuracy of the output projective consensus partition, in terms of both external and internal evaluation criteria⟹***meglio qui***⟸.

Table 3. Similarity results w.r.t. reference classification (feature-based representation)

	ensemble	*MOEA-PCE*			*EM-PCE*		
data	*avg-max*	*avg*	*max-std*	*gain w.r.t. ens. (avg)*	*avg*	*max-std*	*gain w.r.t. ens. (avg)*
Iris	.662 .998	.988	1 .029	+.326	.845	.895 .043	+.183
Wine	.822 .989	.955	.997 .027	+.133	.869	.899 .080	+.047
Glass	.731 .891	.851	.900 .027	+.120	.817	.877 .041	+.086
Ecoli	.763 .879	.858	.884 .016	+.095	.903	.953 .052	+.140
Yeast	.720 .805	.790	.804 .009	+.070	.684	.690 .003	-.036
Segm.	.618 .720	.729	.737 .049	+.111	.625	.632 .008	+.007
Abal.	.716 .754	.759	.849 .023	+.043	.726	.748 .013	+.010
Letter	.646 .693	.767	.818 .012	+.121	.780	.786 .007	+.134
Trace	.661 .818	.755	.811 .0.25	+.094	.753	.773 .021	+.092
Contr.	.663 .894	.880	.910 .016	+.217	.734	.774 .022	+.071

References

[1] E. Achtert, C. Böhm, H. P. Kriegel, P. Kröger, I. Müller-Gorman, and A. Zimek. Finding Hierarchies of Subspace Clusters. In *Proc. PKDD Conf.*, pages 446–453, 2006.

[2] K. Deb. *Multi-objective Optimization Using Evolutionary Algorithms*. Wiley, 2001.

[3] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Trans. on Evolutionary Computation*, 6(2):182–197, 2002.

[4] A. P. Dempster, N. M. Laird, and D. B. Rdin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[5] C. Domeniconi and M. Al-Razgan. Weighted Cluster Ensembles: Methods and Analysis. *TKDD*, 2(4), 2009.

[6] 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.

[7] X. Fern and C. Brodley. Solving Cluster Ensemble Problems by Bipartite Graph Partitioning. In *Proc. ICML Conf.*, pages 281–288, 2004.

[8] A. Jain and R. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, 1988.

[9] G. Moise, J. Sander, and M. Ester. Robust projected clustering. *Knowl. Inf. Syst.*, 14(3):273–298, 2008.

[10] E. K. K. Ng, A. W.-C. Fu, and R. C.-W. Wong. Projective Clustering by Histograms. *TKDE*, 17(3):369–383, 2005.

[11] N. Srinivas and K. Deb. Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation*, 2(3):221–248, 1994.

[12] A. Strehl and J. Ghosh. Cluster Ensembles — A Knowledge Reuse Framework for Combining Multiple Partitions. *Journal of Machine Learning Research*, 3:583–617, 2002.

[13] A. Topchy, A. Jain, and W. Punch. Clustering Ensembles: Models of Consensus and Weak Partitions. *TPAMI*, 27(12):1866–1881, 2005.

[14] M. L. Yiu and N. Mamoulis. Iterative Projected Clustering by Subspace Mining. *TKDE*, 17(2):176–189, 2005.