Consensus Clustering with Robust Evidence Accumulation

André Lourenço[1,2], Samuel Rota Bulò[3],
Ana Fred[2], and Marcello Pelillo[3]

[1] Instituto Superior de Engenharia de Lisboa, Lisbon, Portugal
[2] Instituto de Telecomunicações, Instituto Superior Técnico, Lisbon, Portugal
[3] DAIS, Università Ca' Foscari Venezia, Venice, Italy

Abstract. Consensus clustering methodologies combine a set of partitions on the clustering ensemble providing a consensus partition. One of the drawbacks of the standard combination algorithms is that all the partitions of the ensemble have the same weight on the aggregation process. By making a differentiation among the partitions the quality of the consensus could be improved. In this paper we propose a novel formulation that tries to find a median-partition for the clustering ensemble process based on the evidence accumulation framework, but including a weighting mechanism that allows to differentiate the importance of the partitions of the ensemble in order to become more robust to noisy ensembles. Experiments on both synthetic and real benchmark data show the effectiveness of the proposed approach.

Keywords: Clustering Algorithm, Clustering Ensembles, Median-Partition, Evidence Accumulation Clustering, Clustering Selection, Clustering Weighting.

1 Introduction

The combination of multiple sources of information either in the supervised or unsupervised learning setting allows to obtain improvements on the classification performance. In the unsupervised paradigm, this task is difficult due to the label correspondence problem, i.e., the lack of explicit correspondences between the cluster labels produced by the different clustering algorithms. This problem is made more serious if additionally clusterings with different numbers of clusters are allowed in the ensemble. Clustering ensemble methods, also known as consensus clustering methods, propose a formalism to tackle this problem, allowing the combination of a set of base clustering algorithms into a single consensus partition.

Recent surveys present an overview on this research topic [1, 2]. One of the main approaches is known as *median-partition* [2], where the consensus solution is obtained as the partition having lowest divergence from all the partitions in the clustering ensemble. Another significant approach, known as Evidence Accumulation Clustering (EAC) [3], is based on object *co-occurences*, where the

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 307–320, 2013.

consensus is obtained through a voting process among the objects. Specifically, the consensus clustering problem is addressed by summarizing the information of the ensemble into a pairwise *co-association matrix*, where each entry holds the fraction of clusterings in the ensemble in which a given pair of objects is placed in the same cluster. By doing so, the label correspondence problem is implicitly solved. This matrix, which is regarded to as a similarity matrix, is used to feed a pairwise similarity clustering algorithm to deliver the final consensus clustering. In [3] agglomerative hierarchical algorithms are used to extract the consensus partition and in [4] a graph partitioning algorithm (METIS [5]) are used.

In [6] a more principled way of using the information in the co-association matrix has been proposed. Specifically, the problem of extracting a consensus partition is posed as a matrix factorization problem involving the co-association matrix, where the factor matrix is left-stochastic, *i.e.* nonnegative with columns summing up to one. Each column of the factor matrix can be interpreted as a multinomial distribution expressing the probabilities of each object of being assigned to each cluster. In [7], a probabilistic model for the co-association matrix has been proposed, entitled Probabilistic Evidence Accumulation for Clustering Ensembles. In this model, the entries of the co-association matrix are regarded as independent observations of binomial random variables counting the number of times two objects occur in a same cluster. These random variables are parametrized by the unknown assignments of objects to clusters, which are in turn estimated by adopting a maximum-likelihood approach. In [8] a new formulation is proposed that constitute a generalization of [6] which is solved in way which is close in spirit to [7]. This method, entitled PEACE, creates sparse co-association matrices by a simple uniform sampling criterion and exploits this sparsity to achieve sub-linear iterations in the consensus clustering algorithm.

One of the drawbacks of previous combination methodologies is that all input partitions of the ensemble have the same weight in the aggregation process, when in fact some of them are less important than other [9]. The partitions of the ensemble may come from different algorithms, or from the same algorithm with different initializations. It was shown that the diversity on the clustering ensemble leads to an enhancement on the performance [10], but extreme cases introduce to much variability leading to a significant drop on the performance. Moreover the clustering ensemble can be composed by a subset of partitions highly correlated that can decrease significantly the variability biasing the consensus solution to one of the one of the input partitions.

The problem of weighting differently each of the base clustering solutions was already studied in the literature [9, 11–16]. In Duarte et al. [11] the weighting of the partitions is obtained using internal and relative clustering validation indexes, and the combination is performed using the Evidence Accumulation Clustering algorithm. Fern and Lin [12] define two important quantities that should be considered on the selection process, namely quality calculated between each ensemble and a consensus solution, and diversity of the ensemble. They propose three different heuristics that jointly consider these criteria. Hong et al. [15] also ground their algorithm on these criteria, and use a re-sampling-based method to estimate them. Jia et al. [16] only use a quality criterion on their selection mechanism. In Li and Ding [9], the weighted consensus clustering is based on nonnegative matrix factorization framework but focus only on the quality criterion. Vega-Pons et al. [13] follow the idea of finding the median partition but weighting differently each partition, finding their relevance through an intermediate step.

In this work, we propose a consensus clustering approach based on the evidence accumulation framework, which includes a weighting mechanism that allows to differentiate the importance of the partitions of the ensemble in order to become more robust to noisy ensembles. Our approach tries to find a median-partition, *i.e.* minimizing its divergence from the other partitions in the ensemble, in a way that takes into account the co-occurrences of objects in clusters. Additionally, we jointly optimize the importance of each partition in the ensemble by means of weight variables representing a discrete probability distribution over the set of partitions in the ensemble. To overcome the occurrence of trivial weightings, *i.e.* putting full mass on a single partition, we introduce two regularization mechanisms which lead to two different formulations. In the first formulation, regularization is achieved by restricting the set of feasible probability distributions determining the partitions' importance weights. In the second formulation, a classical ℓ_2 regularization is adopted. We determine the median-partition and the weights vector by means of an alternating optimization procedure, which guarantees to find a local solution. Finally, we perform experiments on ensembles derived from synthetic and real datasets to assess the validity of our model.

The paper is organized as follows: in Section 2 we present the notation used throughout the paper and we introduce our robust consensus clustering model and the two related formulations deriving from the use of different regularization techniques. In Section 3, we present the optimization procedure adopted to find a consensus clustering solution, together with the weights associated to each partition in the ensemble, according to the two proposed formulations. Section 4 is devoted to assessing the effectiveness of the proposed approach. Finally, we draw conclusions in Section 5.

2 Formulation

We start introducing the basic notation and definitions adopted throughout the paper. We denote *sets* with upper-case calligraphic-style letters (*e.g.*, $\mathcal{X}$), column *vectors* with bold lower-case letters (*e.g.*, $\boldsymbol{x}$), *matrices* with upper-case typewriter-style letters (*e.g.*, $\mathtt{X}$), *indices* with lower-case letters (*e.g.*, i) and *constants* with lower-case typewriter-style letters (*e.g.*, $\mathtt{n}$). We denote by $\mathbb{1}_P$ the indicator function yielding 1 if proposition P holds true, 0 otherwise. We indicate with $\mathtt{A}_{ij}$ the ijth component of matrix $\mathtt{A}$ and with $\boldsymbol{v}_i$ the ith component of vector $\boldsymbol{v}$. The vector of all 1s of size $\mathtt{k}$ is denoted by $\mathbf{1}_\mathtt{k}$, the subscript being dropped where size is unambiguous. Let $\mathtt{A}$ be a $\mathtt{k} \times \mathtt{n}$ matrix. The *transposition* of $\mathtt{A}$ is denoted by $\mathtt{A}^\top$. The *Frobenious norm* of $\mathtt{A}$ is denoted by $\|\mathtt{A}\| = \sqrt{\sum_{ij} \mathtt{A}_{ij}^2}$. The sets of real and nonnegative real numbers are denoted by $\mathbb{R}$ and $\mathbb{R}_+$ as usual. We compactly write $[\mathtt{n}]$ for the set $\{1, \dots, \mathtt{n}\}$. We denote by $\lfloor r \rfloor$ and $\lceil r \rceil$

the *floor* and *ceil* operators giving the largest integer value upper bounded by $r \in \mathbb{R}$ and the smallest integer value lower bounded by r, respectively. The set of *left-stochastic* $\mathsf{k} \times \mathsf{n}$ matrices is denoted by $\mathcal{S} = \left\{\mathtt{X} \in \mathbb{R}_+^{\mathsf{k}\times\mathsf{n}} : \mathtt{X}^\top \mathbf{1}_\mathsf{k} = \mathbf{1}_\mathsf{n}\right\}$ and we denote by $\mathcal{S}_{01} = \mathcal{S} \cap \{0,1\}^{\mathsf{k}\times\mathsf{n}}$ the set of *binary* left-stochastic matrices.

Let $\mathcal{X} = \{\boldsymbol{x}_i\}_{i\in[\mathsf{n}]}$ be a set of n data points. An *ensemble of clusterings* of $\mathcal{X}$ is a collection $\mathcal{E} = \{\mathtt{X}^{(u)}\}_{u\in[\mathsf{m}]}$ of m partitions, obtained by running different algorithms (*e.g.*, different parametrizations and/or initializations) on the data set $\mathcal{X}$. Each partition $\mathtt{X}^{(u)} \in \mathcal{S}_{01}$ can be regarded as a binary left-stochastic matrix where $\mathtt{X}^{(u)}_{ki}$ indicates whether the ith data point belongs to cluster k in the uth partition.

A *consensus clustering* (or consensus partition) for an ensemble $\mathcal{E}$ is typically defined as a partition minimizing its divergence from the other partitions in the ensemble. This definition, however, implicitly assumes that the ensemble is noiseless and, thus, that all clusterings should be given equal importance during the establishment of a consensus partition. In order to be more robust to noisy elements, we introduce a probability distribution over the set of partitions in the ensemble that allows to automatically tune the importance of each partition in the ensemble. Formally, a consensus, median partition under this setting can be found as the solution of the following optimization problem:

$$\begin{aligned} \min\ & \textstyle\sum_{u\in\mathcal{U}} \alpha_u d(\mathtt{Z}, \mathtt{X}^{(u)}) \\ \text{s.t. } & \mathtt{Z} \in \mathcal{S}_{01} \\ & \boldsymbol{\alpha} \in \Delta\,, \end{aligned} \tag{1}$$

where $d(\cdot,\cdot)$ is a function providing a distance between the partitions given as arguments, and Δ is the set of m-dimensional vectors representing a discrete probability distribution.

We select the distance function $d(\cdot,\cdot)$ by following the EAC principles. In specific, we implicitly sidestep the problem of cluster correspondences in the computation of the distance between partitions, by counting the errors in the pairwise cluster co-occurrences as follows

$$d(\mathtt{Z}, \mathtt{X}) = \left\|\mathtt{Z}^\top \mathtt{Z} - \mathtt{X}^\top \mathtt{X}\right\|^2 . \tag{2}$$

Indeed, this distance counts the number of times two data points are assigned the same cluster in $\mathtt{X}$ but different ones in $\mathtt{Z}$ and vice versa. Moreover, for convenience, instead of attacking (1) directly, we relax the troublesome integer constraints by replacing the left-stochastic binary matrix variable $\mathtt{Z}$ with a left-stochastic *real* matrix variable $\mathtt{Y} \in \mathcal{S}$ yielding the following relaxed continuous optimization problem:

$$\begin{aligned} \min\ & \textstyle\sum_{u\in\mathcal{U}} \alpha_u \left\|\mathtt{Y}^\top \mathtt{Y} - {\mathtt{X}^{(u)}}^\top \mathtt{X}^{(u)}\right\|^2 \\ \text{s.t. } & \mathtt{Y} \in \mathcal{S} \\ & \boldsymbol{\alpha} \in \Delta\,. \end{aligned} \tag{3}$$

We can finally project $\mathtt{Y}$ back on $\mathcal{S}_{01}$ by performing a maximum a posteriori choice over each column of $\mathtt{Y}$.

Unfortunately, the optimization problem in (3) is ill posed, because trivial distributions, putting full mass on a single partition $u \in [\mathsf{m}]$, lead to an optimal solution by setting $\mathtt{Y} = \mathtt{X}^{(u)}$. The same issue clearly afflicts also the original formulation in (1). To overcome this problem, we need a form of regularization on the probability distribution $\boldsymbol{\alpha}$. In this paper, we propose two different regularization solutions that will be described in the next subsections, one acting on the feasible domain of (3), the other acting on its objective function.

2.1 Regularization of $\boldsymbol{\alpha}$ Using a Restricted Simplex

The first formulation forces the consensus clustering to agree with at least a share $0 < \rho \leq 1$ of the partitions in the ensemble. By doing so, noisy partitions might be excluded from the objective and thus their importance can be nullified and, at the same time, trivial weighting solutions can be excluded as a minimum number of partitions should actively be involved in the minimization. This is achieved by constraining $\boldsymbol{\alpha}$ to lie in a *ρ-restricted simplex* Δ_ρ which is defined as

$$\Delta_\rho = \left\{\boldsymbol{\alpha} \in \mathbb{R}^{\mathsf{m}}_{+} \,:\, \boldsymbol{\alpha}^\top \mathbf{1} = 1\,,\ \boldsymbol{\alpha} \leq \rho\mathbf{1}\right\}\,,$$

where $\rho \geq \mathsf{m}^{-1}$, since otherwise the set Δ_ρ would be empty. Since ρ can be regarded as the largest probability that can be taken by an element of $\boldsymbol{\alpha}$, automatically at least $\lfloor \rho^{-1} \rfloor$ entries of $\boldsymbol{\alpha}$ have to be strictly positive. Clearly, if $\rho \geq 1$ we fall back to the standard simplex, *i.e.* $\Delta_\rho = \Delta$ for all $\rho \geq 1$.

By changing the domain of $\boldsymbol{\alpha}$ in (3) to a ρ-restricted simplex, with $\mathsf{m}^{-1} \leq \rho < 1$ we get the following regularized formulation:

$$\begin{aligned} \min\ & \textstyle\sum_{u\in\mathcal{U}} \alpha_u \left\| \mathtt{Y}^\top \mathtt{Y} - {\mathtt{X}^{(u)}}^\top \mathtt{X}^{(u)} \right\|^2 \\ \text{s.t. }& \mathtt{Y} \in \mathcal{S} \\ & \boldsymbol{\alpha} \in \Delta_\rho\,. \end{aligned} \tag{4}$$

This formulation falls back to the unweighted case when $\rho = \mathsf{m}^{-1}$ and to the unregularized case (3) when $\rho \geq 1$.

2.2 Regularization of $\boldsymbol{\alpha}$ Using $\boldsymbol{\ell_2}$-Norm

Our second formulation, considers the following, classical ℓ_2 regularization parametrized by $\lambda \geq 0$:

$$\begin{aligned} \min\ & \textstyle\sum_{u\in\mathcal{U}} \alpha_u \left\| \mathtt{Y}^\top \mathtt{Y} - {\mathtt{X}^{(u)}}^\top \mathtt{X}^{(u)} \right\|^2 + \frac{\lambda}{2}\|\boldsymbol{\alpha}\|^2 \\ \text{s.t. }& \mathtt{Y} \in \mathcal{S} \\ & \boldsymbol{\alpha} \in \Delta\,. \end{aligned} \tag{5}$$

This formulation falls back to (3) by taking $\lambda = 0$ and to the unweighted case when $\lambda \to \infty$. Indeed, the probability distribution $\boldsymbol{\alpha}$ is pushed towards the uniform distribution as the regularization constant λ increases.

It is interesting to notice that (5) presents a special case of elastic net regularization [17] in which the parameter related to the ℓ_1 regularization is taken to infinity and becomes a constraint in Δ.

3 Algorithm

Solving (4) or (5) is in general a hard problem. We propose here for both an alternating, local optimization procedure which interleaves updates of the cluster assignments $\mathtt{Y}$ and updates of the weights vector $\boldsymbol{\alpha}$. The update procedure for the cluster assignments $\mathtt{Y}$ is the same for both the regularized formulations and will be addressed in the next subsection, while the update procedure of $\boldsymbol{\alpha}$ is in general different for (4) or (5) and it will be addressed in Subsection 3.2 and 3.3, respectively.

3.1 Optimization of $\mathtt{Y}$ in (4) and (5)

Assume $\boldsymbol{\alpha}$ to be fixed and consider the problem of optimizing (4) or (5) with respect to $\mathtt{Y}$ only. This yields a non-convex continuous optimization problem which can be conveniently rewritten into the following one, which shares with (4) and (5) the same local minimizers:

$$\begin{aligned} &\min \left\| \mathtt{Y}^\top \mathtt{Y} - \textstyle\sum_{u\in\mathcal{U}} \alpha_u \mathtt{X}^{(u)^\top} \mathtt{X}^{(u)} \right\|^2 \\ &\text{s.t. } \mathtt{Y} \in \mathcal{S}\,. \end{aligned} \tag{6}$$

The equivalence between (4)/(5) and (6) can be grasped by noting that the terms depending on $\mathtt{Y}$ are the same in both the optimization problems and the objectives differ simply by an additive, constant term. The matrix $\sum_{u\in\mathcal{U}} \alpha_u \mathtt{X}^{(u)^\top} \mathtt{X}^{(u)}$ can be regarded as the *weighted co-association matrix*. Indeed, when $\boldsymbol{\alpha}$ is the uniform distribution, we fall back to the classic notion of co-association matrix as originally defined in [3].

A local solution of the optimization problem in (6) can be efficiently computed using the approach proposed in [8], which is a primal, line search approach that iteratively improves the objective by optimizing one column of $\mathtt{Y}$ at time. Each update has a linear complexity both in time and space. Another advantage of this optimization approach is that it can handle sparsified versions of the optimization problem in (6), where the Frobenious norm runs over a sparse subset of the entries of the matrix given as argument, which is useful in the case of large datasets. In the sparisified scenario, the time complexity of every line search can be reduced to sub-linear.

3.2 Optimization of $\boldsymbol{\alpha}$ in (4)

Assume $\mathtt{Y}$ to be fixed in (4) and let us focus on the optimization of the vector-valued variable $\boldsymbol{\alpha}$ only. By letting $\boldsymbol{d} \in \mathbb{R}^{\mathtt{m}}_+$ be a vector with entries

$$d_u = \left\| \mathtt{Y}^\top \mathtt{Y} - \mathtt{X}^{(u)^\top} \mathtt{X}^{(u)} \right\|^2 , \tag{7}$$

we can rewrite the optimization problem restricted to $\boldsymbol{\alpha}$ as follows:

$$\begin{aligned} \min\ & \boldsymbol{\alpha}^\top \boldsymbol{d} \\ \text{s.t. } & \boldsymbol{\alpha}^\top \mathbf{1} = 1 \\ & \boldsymbol{\alpha} \leq \rho \mathbf{1} \\ & \boldsymbol{\alpha} \in \mathbb{R}^{\mathsf{m}}_{+} . \end{aligned} \tag{8}$$

This is a linear programming problem, whose solution can be readily computed as stated by the following proposition.

Proposition 1. *Assume without loss of generality that* $\boldsymbol{d}$ *satisfies the relation* $d_u \leq d_v$ *for all* $0 \leq u \leq v \leq \mathsf{m}$. *Let* $r = 1 - \rho\lfloor \rho^{-1} \rfloor$. *A solution of* (8) *is given by* $\alpha_u = \rho \mathbb{1}_{u \leq \lfloor \rho^{-1} \rfloor} + r \mathbb{1}_{u = \lceil \rho^{-1} \rceil}$, *for all* $u \in [\mathsf{m}]$.

Proof. We proceed with a proof by contradiction. Assume that $\boldsymbol{\alpha}$ is not a solution of (8) and let $\boldsymbol{\beta} \in \Delta_\rho$ be a solution of (8), which exists since the feasible set is compact. Clearly, $\boldsymbol{\beta} \neq \boldsymbol{\alpha}$. The solution $\boldsymbol{\beta}$ must satisfy the property that $\beta_u \geq \beta_v$ for all $0 \leq u \leq v \leq \mathsf{m}$. Otherwise, by swapping the elements in $\boldsymbol{\beta}$ indexed by a pair of indices violating the condition, we would yield a better solution, contradicting the optimality of $\boldsymbol{\beta}$.

Now, let $p \in [\mathsf{m}]$ be the smallest index satisfying $\beta_p > \alpha_p$. Hence, $\alpha_u \geq \beta_u$ holds for all $u < p$. Necessarily, $p > \lfloor \rho^{-1} \rfloor$ because by construction $\alpha_u = \rho$ for all $0 \leq u \leq \lfloor \rho^{-1} \rfloor$ and therefore it cannot be exceeded by β_u. Moreover, by construction, $\alpha_u = 0$ for all $u > \lceil \rho^{-1} \rceil$, which implies that $\alpha_u \leq \beta_u$ for all $u \geq p$. By exploiting these relations and by the non-increasing ordering on $\boldsymbol{d}$, we derive that

$$\begin{aligned} \boldsymbol{d}^\top(\boldsymbol{\alpha} - \boldsymbol{\beta}) &= \left[\sum_{i=1}^{p-1} d_i(\alpha_i - \beta_i)\right] - \left[\sum_{i=p}^{\mathsf{m}} d_i(\beta_i - \alpha_i)\right] \\ &\leq d_p \left[\sum_{i=1}^{p-1} \alpha_i - \beta_i\right] - d_p \left[\sum_{i=p}^{\mathsf{m}} \beta_i - \alpha_i\right] = d_p \left[\sum_{i \in [\mathsf{m}]} \alpha_i - \beta_i\right] = 0 , \end{aligned}$$

which contradicts the non-optimality of $\boldsymbol{\alpha}$. □

The condition required on $\boldsymbol{d}$ can be met by sorting the vector $\boldsymbol{d}$ in ascending order and by keeping track of the induced permutation. The latter can be used at the end to reorder the solution $\boldsymbol{\alpha}$ using the inverse mapping.

3.3 Optimization of $\boldsymbol{\alpha}$ in (5)

Assume $\mathtt{Y}$ to be fixed in (5) and let us focus on the optimization of the vector-valued variable $\boldsymbol{\alpha}$ only. By taking $\boldsymbol{d} \in \mathbb{R}^{\mathsf{m}}_{+}$ as defined in (7), we reduce (5) to following convex, quadratic optimization problem:

$$\begin{aligned} \min\ & \boldsymbol{\alpha}^\top \boldsymbol{d} + \tfrac{\lambda}{2} \boldsymbol{\alpha}^\top \boldsymbol{\alpha} \\ \text{s.t. } & \boldsymbol{\alpha}^\top \mathbf{1} = 1 \\ & \boldsymbol{\alpha} \in \mathbb{R}^{\mathsf{m}}_{+} . \end{aligned} \tag{9}$$

Fortunately, also this optimization problem can be readily computed in linear time. The solution procedure is detailed in the following proposition.

Proposition 2. *Assume without loss of generality that* $\boldsymbol{d}$ *satisfies the relation* $d_u \leq d_v$ *for all* $0 \leq u \leq v \leq \mathsf{m}$. *Let* $\boldsymbol{y} \in \mathbb{R}^{\mathsf{m}}$ *be defined as*

$$y_u = \frac{1}{u}\left(1 + \sum_{v=1}^{u} \frac{d_v}{\lambda}\right)$$

and let w *be the largest element in* $[\mathsf{m}]$ *satisfying* $y_u > d_u/\lambda$. *A solution of* (5) *is given by* $\alpha_u = \mathbb{1}_{u \leq w}(y_w - d_u/\lambda)$, *for all* $u \in [\mathsf{m}]$.

Proof. We start showing that $y_u \leq y_{u+1}$ for all $u > w$. To this end, note that the relation $y_u \leq d_u/\lambda$ holds for all $u > w$ by definition of w. Then

$$y_u = \frac{u+1}{u}\left(y_{u+1} - \frac{1}{u+1}\frac{d_{u+1}}{\lambda}\right) \leq \frac{u+1}{u}\left(y_{u+1} - \frac{1}{u+1}y_{u+1}\right) = y_{u+1}\,. \quad (10)$$

By repeated application of this relation, we have that $y_u \leq y_v$ for all $v \geq u$ and $u > w$.

Since the optimization problem in (9) is convex, the Karush-Kuhn-Tucker (KKT) necessary conditions for optimality are also sufficient. Hence, a solution $\boldsymbol{\alpha}$ satisfying the following KKT conditions, for some value of the Lagrangian multipliers $\gamma \in \mathbb{R}$ and $\boldsymbol{\mu} \in \mathbb{R}^{\mathsf{m}}_{+}$, is a solution of (9):

$$d_u + \lambda\alpha_u - \gamma - \mu_u = 0\,, \qquad \forall u \in [\mathsf{m}] \quad (11)$$

$$\boldsymbol{\alpha}^\top \mathbf{1} = 1\,, \quad (12)$$

$$\boldsymbol{\alpha}^\top \boldsymbol{\mu} = 0\,. \quad (13)$$

We proceed by showing that the solution computed as detailed in the proposition satisfies the KKT conditions, *i.e.* we have to show that Equations (11)-(13) are satisfied for a specific choice of the Lagrangian multipliers and that our choice of $\boldsymbol{\mu}$ also satisfies the non-negativity constraint. We start noting that $\alpha_u > 0$ for all $u \leq w$, for we have that $y_w > d_w/\lambda \geq d_u/\lambda$ for all $u \leq w$, and $\alpha_u = 0$ for all $u > w$ by construction. Set $\gamma = \lambda y_w$. For all $u \leq w$, Equation (11) is satisfied by taking $\mu_u = 0$, while for all $u > w$ it is satisfied by taking $\mu_u = d_u - \lambda y_w$. This choice of the elements of $\boldsymbol{\mu}$ clearly satisfies Equation (13). Moreover, $\mu_u \geq 0$ is clearly satisfied for all $u \leq w$ with equality and it is also satisfied for all $u > w$ because $y_w \leq y_u \leq d_u/\lambda$ holds by the relation proven at the beginning of this proof and by definition of w. We conclude by showing that also Equation (12) holds. Indeed,

$$\sum_{u=1}^{\mathsf{m}} \alpha_u = \sum_{u=1}^{w} \alpha_u = w y_w - \sum_{u=1}^{w} \frac{d_u}{\lambda} = 1\,.$$

□

As in the case of the previous formulation, the condition required for $\boldsymbol{d}$ can be met by sorting the vector in ascending order and by keeping track of the induced permutation, which will be used to recover the original ordering on the solution vector $\boldsymbol{\alpha}$.

3.4 Summary of the Algorithm

As anticipated at the beginning of this section, the algorithm used to optimize (4) and (5) alternates between the optimization of the cluster assignment probabilities $\mathtt{Y}$ and the optimization of the weights vector $\boldsymbol{\alpha}$.

The pseudo-code of the algorithm is shown in Algorithm 1. The input consists of the ensemble of clusterings $\mathcal{E} = \{\mathtt{X}^{(u)}\}_{u\in[\mathsf{m}]}$ the maximum number of desired clusters k and the regularization parameter, namely ρ for the formulation in (4) and λ for the formulation in (5). At line 1, we initialize the matrix of probabilistic cluster assignments by randomly sampling an element of $\mathtt{S}$, or by considering a uniform distribution for all objects. At line 2, we initialize the weights to the uniform distribution. Lines 3-6 represent the alternating optimization loop, which is iterated until a stopping criterion is met, *e.g.* maximum number of iterations has been reached, or the difference of either $\mathtt{Y}$ or $\boldsymbol{\alpha}$ between two consecutive iterations is below a given threshold. At line 4, we optimize (6), which is equivalent to optimizing either regularized formulation with respect to $\mathtt{Y}$. The solution is obtained by using the algorithm described in [8]. At the next line, we focus on optimizing the weights vector-valued variable $\boldsymbol{\alpha}$. Based on the chosen formulation, we optimize (8) or (9) following the procedure described in Proposition 1 or Proposition 2, respectively. Both solutions can be obtained efficiently in linear time. Finally, once we exit the optimization loop, we project $\mathtt{Y}$ on the set of left-stochastic binary matrices obtaining matrix $\mathtt{X} \in \mathcal{S}_{01}$, and we return $\mathtt{X}$ and $\boldsymbol{\alpha}$.

Algorithm 1. Algorithm description

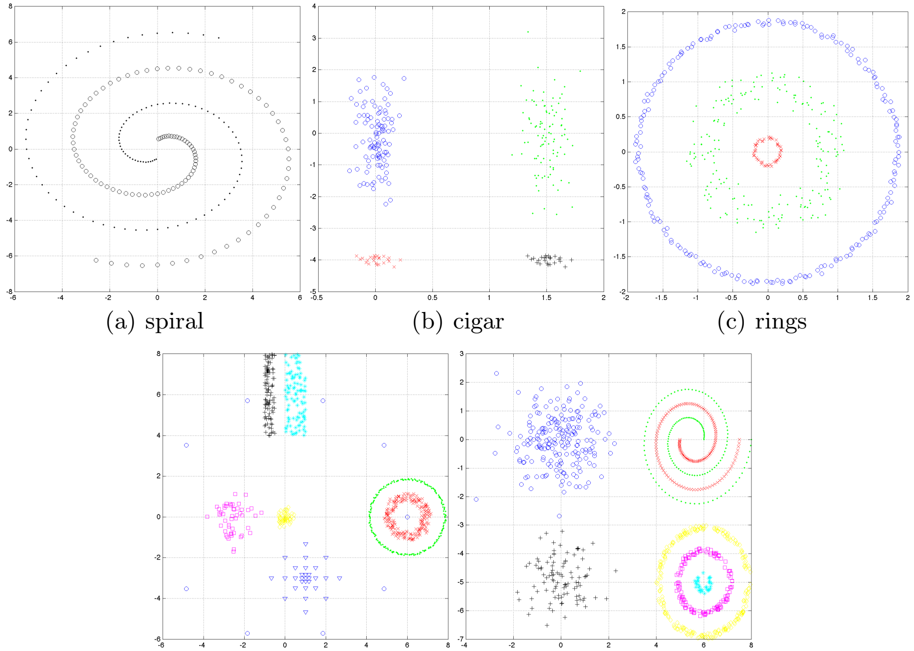
```
Require: $\mathcal{E} = \{\mathtt{X}^{(u)}\}_{u\in[\mathsf{m}]}$: ensemble of clusterings
Require: $\mathsf{k}$: maximum number of desired clusters
Require: $\mathsf{m}^{-1} \leq \rho \leq 1$, regularization parameter for formulation (4), or $\lambda > 0$,
    regularization parameter for formulation (5)
1: Initialize $\mathtt{Y} \in \mathcal{S}$
2: $\boldsymbol{\alpha} \leftarrow \mathsf{m}^{-1}\mathbf{1}$
3: repeat
4:     $\mathtt{Y} \leftarrow$ solve (6) using the approach in [8]
5:     $\boldsymbol{\alpha} \leftarrow$ solve (4) or (5) based on the desired formulation
6: until termination condition is met
7: $\mathtt{X} \leftarrow$ project $\mathtt{Y}$ on $\mathcal{S}_{01}$
8: return $\mathtt{X}$, $\boldsymbol{\alpha}$
```

4 Experimental Evaluation

We evaluate our formulation using synthetic and real-world datasets from the UCI Machine Learning Repository. We compare the performance of our algorithm against the algorithm in [8], which we refer to as UN-WEIGHTED. We call our two algorithms WEIGHTED+$\varDelta_\rho$ and WEIGHTED+ℓ_2 corresponding to the variants described in Section 3.2 and 3.3, respectively. For all the experiments,

Table 1. Benchmark datasets - synthetic: (s-1) spiral, (s-2) cigar, (s-3) rings, (s-4) image-c, (s-5) image-1; real: (r-1) iris, (r-2) wine, (r-3) house-votes, (r-4) ionsphere, (r-5) std-yeast-cell, (r-6) breast-cancer, (r-7) optdigits.

Data-Sets	s-1	s-2	s-3	s-4	s-5	r-1	r-2	r-3	r-4	r-5	r-6	r-7
k	2	4	3	7	8	3	3	2	2	5	2	10
n	200	250	450	739	1000	150	178	232	351	384	683	1000
$\mathsf{k}_{\min} - \mathsf{k}_{\max}$	2-10	2-10	2-10	7-40	8-40	3-20	4-20	2-20	4-20	5-20	2-20	10-50



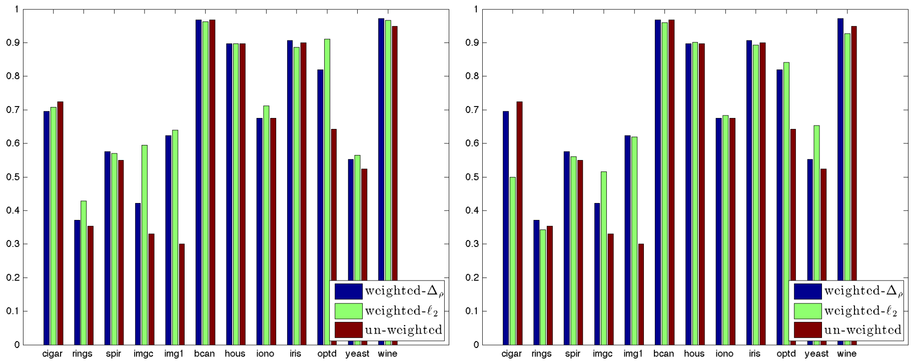
(a) spiral (b) cigar (c) rings

(d) image-c (e) image-1

Fig. 1. Sketch of the Synthetic Datasets

we used the following setting for the regularization parameters of our algorithms: $\rho = (0.8\,\mathsf{m})^{-1}$ and $\lambda = 0.5\,\mathsf{n}^2$.

We performed different series of experiments to compare the performance of our approaches on several types of ensembles: i) *k-means ensemble* - consisting of $\mathsf{m} = 150$ partitions generated running the classical k-means algorithm [18] with different number of clusters, and different initializations; ii) *mixed ensemble* - consisting of $\mathsf{m} = 56$ partitions generated by multiple algorithms (agglomerative hierarchical algorithms: single, average, ward, centroid link; k-means[18]; spectral clustering [19]) with different number of clusters iii) *noisy ensemble* - an ensemble with noisy partitions obtained from previous ensembles, changing a percentage of the partitions of the ensemble to random partitions.



(a) k-means ensemble (b) mixed ensemble

Fig. 2. Performance Evaluation for k-means and mixed ensembles in terms of Accuracy. The weighted approaches WEIGHTED+Δ_ρ and WEIGHTED+ℓ_2 are compared against the un-weighted one.

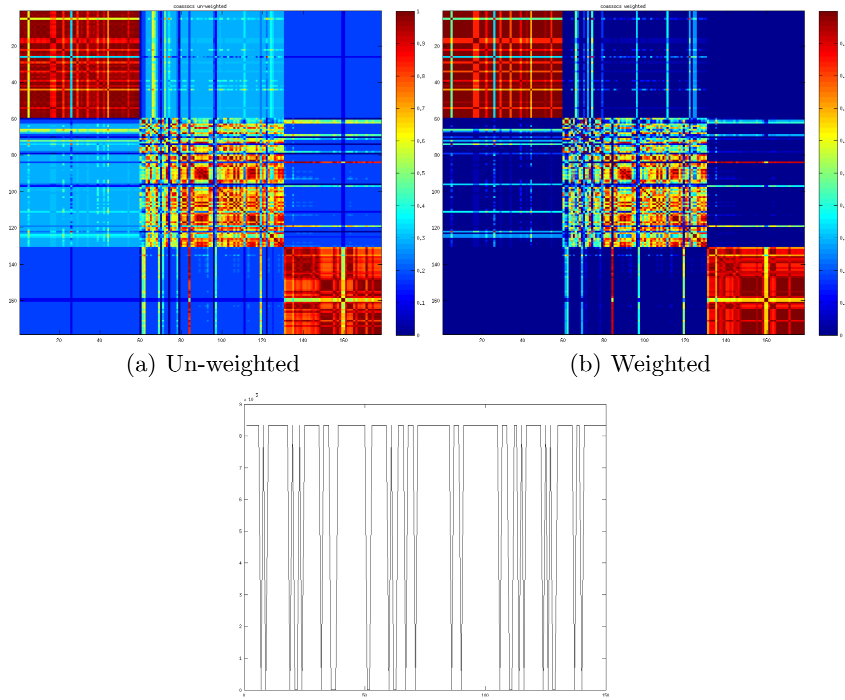
We assess the quality of a consensus partition by comparing it against the ground truth partition. In order to compare two hard clusterings we adopt the $\mathcal{H}$ criterion introduced in [20], which computes the fraction of correct cluster assignments considering the best cluster matching between the consensus partition and the ground-truth partition.

Table 1 summarizes the main characteristics of the UCI and synthetic datasets used in the evaluation (number of ground truth clusters k and number of samples n) and reports also the range of number of clusters used during the ensemble generation $\{\mathsf{k}_{min} - \mathsf{k}_{max}\}$. Figure 1 illustrates the synthetic datasets used in the evaluation: (a) spiral; (b) cigar; (c) rings; (d) image-c (e) image-1 .

4.1 Non-Noisy Ensembles

Figure 2 presents the performance results in terms of $\mathcal{H}$ criterion for the k-means and mixed ensembles. In both types of ensembles we can see that the proposed weighted consensus clustering approaches perform on average better than the unweighted one, as expected, even though we have a fixed parameterization for the regularization on all datasets. If we compare the two types of regularization on the weighted algorithms, there is no clear winner, the ℓ_2-regularized being slightly better on average. Overall, we have that in the k-means ensemble the weighted algorithms obtain better result in 8 out of 12 datasets, while in the mixed ensemble they obtain better result in 5 out of 12 datasets. On the remaining datasets the performance of weighted and unweighted formulations perform comparably well.

In Figure 3 we present the co-association matrices of the weighted and unweighted situation and the weights that were obtained by the weighted version (in this case obtained by the WEIGHTED+Δ_ρ approach). The colour scheme on the co-association matrices goes from blue (zero similarity), to red (highest similarity). The un-weighted co-association was transformed into the weighted



(a) Un-weighted (b) Weighted

(c) Weights

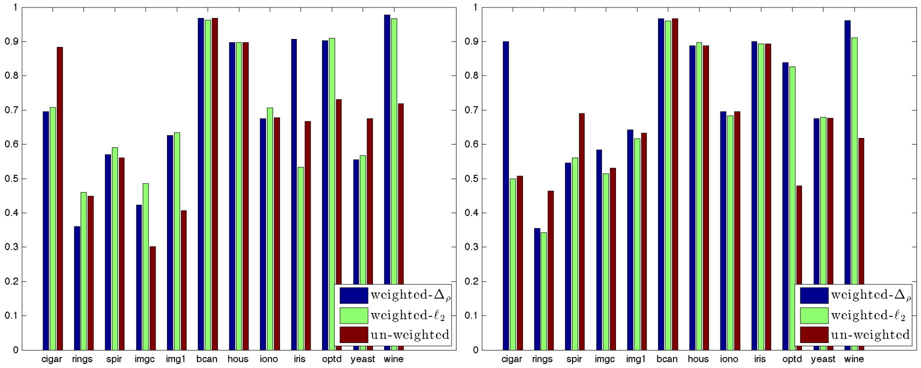
Fig. 3. Example of Co-association matrices. On the left the unweighted version of the co-association, and on the right the weighted co-association, obtained after weighting the partitions.

co-association using the weights presented below, which selectively tune the importance of each partition, turning the matrix into a more structured one.

4.2 Noisy Ensembles

Figure 4 presents the performance obtained on the noisy ensembles, which have been obtained from the k-means and mixed ensembles of the previous section by substituting 20% of the partitions with randomly generated ones. Our purpose is to assess the robustness of the approach to outliers in the ensembles. The results are evaluated in terms of the $\mathcal{H}$ criterion.

As we can see, the performance of weighted approaches tends to be more stable than the un-weighted version. There are isolated dataset where the un-weighted version improved the performance (when compared to non-noisy ensembles), but this situation is not generalizable to the other datasets. The opposed situation was also observed, with the weighted approaches improving the performance (when compared with non-noisy ensembles), but the general trend was to conserve the previous result. if we compare the two types of regularization, we see that Weighted+Δ_ρ apparently was more stable, preserving in more situations the previous result.



(a) k-means ensemble

(b) mixed ensemble

Fig. 4. Performance Evaluation for the noisy k-means and mixed ensembles in terms of Accuracy. The weighted approaches Weighted+Δ_ρ and Weighted+ℓ_2 are compared against the unweighted one.

5 Conclusions and Future Work

One of the drawbacks of the classical clustering combination methodologies is that all the partitions of the ensemble have the same weight in the combination process. In this paper we propose a consensus clustering approach with a weighting mechanism that allows to select a subset of the ensemble becoming more robust to noisy ensembles. Our approach tries to find a median-partition based on co-occurences of objects in clusters. We follow an alternating optimization procedure, which allows the determination of the median-partition and the weights vector. Experiments on syntethic and real-world datasets show that the proposed approach outperforms state-of-the-art approaches delivering more robust results. Future work will focus on the application of this framework to large-scale problems.

Acknowledgments. This work was partially funded by FCT under grants SFRH/PROTEC/49512/2009, PTDC/EEI-SII/2312/2012 (LearningS project), and by the ADEETC from Instituto Superior de Engenharia de Lisboa, whose support the authors gratefully acknowledge.

References

1. Ghosh, J., Acharya, A.: Cluster ensembles. WIREs Data Mining and Knowledge Discovery 1(4), 305–315 (2011)
2. Vega-Pons, S., Ruiz-Shulcloper, J.: A survey of clustering ensemble algorithms. IJPRAI 25(3), 337–372 (2011)
3. Fred, A., Jain, A.: Combining multiple clustering using evidence accumulation. IEEE Trans Pattern Analysis and Machine Intelligence 27(6), 835–850 (2005)
4. Strehl, A., Ghosh, J.: Cluster ensembles - a knowledge reuse framework for combining multiple partitions. J. of Machine Learning Research 3 (2002)

5. Karypis, G., Kumar, V.: A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM J. Sci. Comput. 20(1), 359–392 (1998)
6. Rota Bulò, S., Lourenço, A., Fred, A., Pelillo, M.: Pairwise probabilistic clustering using evidence accumulation. In: Hancock, E.R., Wilson, R.C., Windeatt, T., Ulusoy, I., Escolano, F. (eds.) SSPR&SPR 2010. LNCS, vol. 6218, pp. 395–404. Springer, Heidelberg (2010)
7. Lourenço, A., Rota Bulò, S., Rebagliati, N., Fred, A., Figueiredo, M., Pelillo, M.: Probabilistic evidence accumulation for clustering ensembles. In: 2nd Int. Conf. on Pattern Recognition Applications and Methods, ICPRAM 2013 (2013)
8. Lourenço, A., Rota Bulò, S., Rebagliati, N., Fred, A., Figueiredo, M., Pelillo, M.: Consensus clustering using partial evidence accumulation. In: Sanches, J.M., Micó, L., Cardoso, J.S. (eds.) IbPRIA 2013. LNCS, vol. 7887, pp. 69–78. Springer, Heidelberg (2013)
9. Li, T., Ding, C.: Weighted Consensus Clustering. In: Proceedings of 2008 SIAM International Conference on Data Mining (SDM 2008) (2008)
10. Hadjitodorov, S.T., Kuncheva, L.I., Todorova, L.P.: Moderate diversity for better cluster ensembles. Inf. Fusion 7(3), 264–275 (2006)
11. Duarte, F.J.F., Fred, A.L.N., Rodrigues, F., Duarte, J.M.M., Lourenço, A.: Weighted evidence accumulation clustering using subsampling. In: Proceedings of the 6th International Workshop on Pattern Recognition in Information Systems, PRIS 2006, In conjunction with ICEIS, pp. 104–116 (2006)
12. Fern, X.Z., Lin, W.: Cluster ensemble selection. Stat. Anal. Data Min. 1(3), 128–141 (2008)
13. Vega-Pons, S., Correa-Morris, J., Ruiz-Shulcloper, J.: Weighted cluster ensemble using a kernel consensus function. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) CIARP 2008. LNCS, vol. 5197, pp. 195–202. Springer, Heidelberg (2008)
14. Azimi, J., Fern, X.: Adaptive cluster ensemble selection. In: Proceedings of the 21st International Jont Conference on Artifical Intelligence, IJCAI 2009, pp. 992–997. Morgan Kaufmann Publishers Inc., San Francisco (2009)
15. Hong, Y., Kwong, S., Wang, H., Ren, Q.: Resampling-based selective clustering ensembles. Pattern Recognition Letters 30(3), 298–305 (2009)
16. Jia, J., Xiao, X., Liu, B., Jiao, L.: Bagging-based spectral clustering ensemble selection. Pattern Recognition Letters 32(10), 1456–1467 (2011)
17. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. J. of the Royal Stat. Society, Series B, 301–320 (2005)
18. Jain, A.K., Dubes, R.: Algorithms for Clustering Data. Prentice Hall (1988)
19. Ng, A.Y., Jordan, M.I., Weiss, Y.: On spectral clustering: Analysis and an algorithm. In: NIPS, pp. 849–856. MIT Press (2001)
20. Meilă, M.: Comparing clusterings by the variation of information. In: Schölkopf, B., Warmuth, M.K. (eds.) COLT/Kernel 2003. LNCS (LNAI), vol. 2777, pp. 173–187. Springer, Heidelberg (2003)