Pattern Recognition Letters 70 (2016) 8–16

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Automatic graph pruning based on kernel alignment for spectral clustering☆

A.M. Alvarez-Meza[a,*], A.E. Castro-Ospina[a,b], G. Castellanos-Dominguez[a]

[a] Signal Processing and Recognition Group, Universidad Nacional de Colombia sede Manizales, Km 9 vía al aeropuerto, Campus La Nubia, Manizales, Colombia
[b] MIRP – Research Center, Instituto Tecnológico Metropolitano, Colombia

ARTICLE INFO

Article history:
Received 28 January 2015
Available online 30 November 2015

Keywords:
Spectral clustering
Kernel alignment
Graph pruning,

ABSTRACT

Detection of data structures in spectral clustering approaches becomes a difficult task when dealing with complex distributions. Moreover, there is a need of a real user prior knowledge about the influence of the free parameters when building the graph. Here, we introduce a graph pruning approach, termed Kernel Alignment based Graph Pruning (KAGP), within a spectral clustering framework that enhances both the local and global data consistencies for a given input similarity. The KAGP allows revealing hidden data structures by finding relevant pair-wise relationships among samples. So, KAGP estimates the loss of information during the pruning process in terms of a kernel alignment-based cost function. Besides, we encode the sample similarities using a compactly supported kernel function that allows obtaining a sparse data representation to support spectral clustering techniques. Attained results shows that KAGP enhances the clustering performance in most of the cases. In addition, KAGP avoids the need for a comprehensive user knowledge regarding the influence of its free parameters.

1. Introduction

In practice, the only presence of unlabeled data forces to develop unsupervised clustering techniques that search for hidden points of similar entities expressed in terms of a given pattern proximity measure. As a promising alternative in some disciplines –like data mining, pattern recognition, image processing, and machine learning– spectral clustering has been widely used for data grouping. Mostly, these algorithms, which group points using matrix eigenvectors derived from the data, get better performance on complex data sets with non-convex clusters where traditional methods (such as K-means) frequently fail [5]. In fact, spectral clustering, which has its root in graph partitioning problems, can handle the optimization problem within the standard linear algebra framework for avoiding the local optima [27]. Also, spectral clustering formulations are very closed to kernel-based clustering approaches such as Kernel K-means, SOM, and Neural Gas [10]. Indeed, the objective function of a graph partitioning problem is mathematically equivalent to the weighted extension of the kernel K-means algorithm [8,9].

As discussed in [29], spectral clustering approaches focus mainly on two issues: *i*) the extraction of an optimal partition and *ii*) choosing a suitable affinity matrix when building the graph representation. With respect to the former issue, it can be shown that the second smallest eigenvalue of the matrix estimated from the circuit netlist provides acceptable cut approximations. Nevertheless, the size of a graph subset is proportional to its number of vertices that is not always related to the within-cluster similarity. The following approaches have been proposed to cope with this drawback: the normalized cut criterion, termed NCut, measuring the total dissimilarity among groups and the overall one within clusters [23], the random-walk interpretation of spectral clustering [18], computation of the eigenvectors using matrix perturbation theory [21], among others. Nonetheless, the choice of a suitable affinity matrix for the graph building has received much less attention regardless its importance on the clustering performance [3]. To encode the pairwise relationships among samples in the affinity matrix correctly, the corresponding measure should be smooth with respect to the intrinsic data structure. Thus, samples belonging to the same group should have high similarity each other and have enough space consistency. For this purpose, two assumptions on consistency are proposed in [29]: (*i*) local consistency, meaning that nearby points in the space should have high similarity, (*ii*) global consistency, meaning that samples in the same cluster should have high similarity. In spectral clustering, similarity graphs model the local neighborhood relationships between samples. Even though the choice of any of the regularly used graph constructions (ϵ-neighborhood, k-nearest graphs, and fully connected graphs) does not influence the clustering result

☆ This paper has been recommended for acceptance by Prof. A. Petrosino.

* Corresponding author. Tel./fax: +57 6 8879400 ext 55713.
E-mail address: amalvarezme@unal.edu.co, andresmarino07@gmail.com (A.M. Alvarez-Meza).

http://dx.doi.org/10.1016/j.patrec.2015.11.006
0167-8655/

[25], the similarity graph implementation requires fixing the ruling parameters. Since this procedure is not easy to automate, the free parameters of the similarity measure are manually adjusted in practice. Then, their corresponding partition graph matrix cannot be constructed to preserve the above assumptions on consistency. In particular for the widely-known Gaussian-based similarity, commonly known as Gaussian kernel, its bandwidth parameter is commonly set as a fraction of the pairwise distance estimated from the whole data. However, multi-scale datasets cannot be properly partitioned using a unique bandwidth value because of their local variable data density.

A strategy overcoming this restriction is to adapt the scaling measure according to each k-neighbor distance related to the local sample density [22]. Due to this approach seeks just for preservation of the local consistency, it mostly fails when dealing with outliers or noisy data. Besides, automatic tuning of the k-neighbor size remains an open issue. Clustering aggregation can be improved based on probability accumulation where the co-association matrices built from K-means clustering are weighted by the average pairwise distance of each cluster [26]. Nevertheless, some imposed constraints regarding the probability distribution of the data are not always satisfied in practice. More elaborate approaches are also in use like the locally adaptive similarity measure using neighborhood density information and the incorporation of the assumption about consistency of the similarity [29]. Even though both approaches aim to reveal hidden data structures by imposing the density or structural constraints, each formulation requires a real user prior knowledge about the influence of the free parameters. This aspect becomes crucial to achieve a suitable trade-off between local and global consistency preservation.

Here, a graph pruning approach, called Kernel Alignment based Graph Pruning (KAGP), is proposed within a spectral clustering framework to enhances both the local and global data consistencies for a given input similarity matrix. Our approach aims to reveal the salient complex structure of the input data by finding relevant pairwise relationships among samples. To weak all irrelevant relationships of the input similarity matrix, KAGP quantifies the loss of information during the pruning process in terms of a kernel alignment-based function [7,16]. Moreover, we encode the sample similarities using a compactly supported kernel that allows obtaining a sparse data representation to support the graph partitioning problem. As a consequence, KAGP takes advantage of an initial guess of the relationships among points to identify all relevant connections. Testing that is carried out on synthetic and real-world datasets shows that the proposed methodology allows enhancing the graph representation of different state of the art approaches, improving the clustering performance in most of the cases. Moreover, KAGP avoids the need for a comprehensive user knowledge regarding the influence of its free parameters. The remainder is organized as follows: In Section 2 we describe the proposed approach, experiments and discussion are presented in Sections 3 and 4, and in Section 5 we conclude about the achieved results.

2. Graph pruning using kernel alignment

Let $\boldsymbol{X} \in \mathbb{R}^{N\times D}$ be an input data matrix holding N samples and D features, where each row $\{\boldsymbol{x}_i \in \mathbb{R}^D : i{=}1,\ldots,N\}$ represents a single data point. The goal of clustering is to divide the data into different clusters, where samples within the same cluster are similar to each other. To discover the main topological relationships among data points, spectral clustering-based approaches build from $\boldsymbol{X}$ a weighted graph representation $\mathcal{G}(\boldsymbol{X},\boldsymbol{K})$, where each sample point, $\boldsymbol{x}$, is a vertex or node and $\boldsymbol{K} \in \mathbb{R}^{N\times N}$ is a similarity (affinity) matrix encoding all associations between graph nodes. In turn, each element of the similarity matrix, $k_{ij} \subseteq \boldsymbol{K}$, corresponding to the edge weight between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, is commonly defined as follows [10]: $k_{ij}{=}\kappa\left(\boldsymbol{x}_i,\boldsymbol{x}_j;\sigma\right)$, $k_{ij} \in \mathbb{R}^+$, where $\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j;\sigma){=}\exp\left(-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2^2/2\sigma^2\right)$ is the Gaussian kernel, and $\sigma \in \mathbb{R}^+$ is the kernel bandwidth. Notation $\|\cdot\|_2$ stands for the L_2-norm. Among many others available kernels (like Laplacian or polynomial), the Gaussian function has the advantages of finding Hilbert spaces with universal approximating capability and its mathematical tractability [15].

Algorithm 1. Basic Spectral Clustering

Input: $\boldsymbol{X}\in\mathbb{R}^{N\times D}$, $\boldsymbol{K}\in\mathbb{R}^{N\times N}$, $C\in\mathbb{N}$.
Output: $\mathcal{V}=\{\mathcal{V}_c : c{=}1,\ldots,C\}$.

1 initialization;
2 Compute the diagonal degree matrix $\boldsymbol{W}\in\mathbb{R}^{N\times N}$ holding elements $w_{ii}{=}\sum_{j\in N} k_{ij}$.
3 Estimate the normalized Laplacian matrix: $\boldsymbol{L}{=}\boldsymbol{W}^{-\frac{1}{2}}\boldsymbol{K}\boldsymbol{W}^{-\frac{1}{2}}$.
4 Calculate the eigenvalues$\{\lambda_i\in\mathbb{R}^+\}$, and eigenvectors $\{\boldsymbol{u}_i\in\mathbb{R}^N : i{=}1,\ldots,N\}$, of $\boldsymbol{L}$ and stack the C eigenvectors corresponding to the first C largest eigenvalues into $\boldsymbol{A}\in\mathbb{R}^{N\times C}$.
5 Assuming each row of $\boldsymbol{A}$ as a point of dimension $\mathbb{R}^C$, cluster them into C clusters by using the K-means algorithm.
6 Assign the original point $\boldsymbol{x}_i$ to cluster c, if and only if, the i-th row of the matrix $\boldsymbol{A}$ is allocated to the cluster c.

Hence, the clustering task now relies on the statement of the conventional graph cut problem, where the goal is to partition the set of vertices $\mathcal{V}\in\boldsymbol{X}$ into $C\in\mathbb{N}$ disjoint subsets $\mathcal{V}_c$, so that $\mathcal{V}{=}\cup_{c=1}^{C}\mathcal{V}_c$ and $\mathcal{V}_{c'}\cap\mathcal{V}_c{=}\emptyset$, $\forall\, c'\neq c$. Since graph-cut approaches require high computational burden, relaxation of the clustering optimization problem has been developed based on the spectral graph analysis [20]. So, spectral clustering-based methods decompose the input data $\boldsymbol{X}$ into C disjoint subsets by using both spectral information and orthogonal transformations of $\boldsymbol{K}$. Algorithm 1 describes the well-known solution of the cut problem (termed NCut) that is based on the Rayleigh-Ritz theory.

To get available data partitioning, performance of the spectral clustering-based methods primarily resides in the choice of the similarity measure which should be smooth with respect to the intrinsic structure of the samples [27]. In fact, the quality of the graph representation, $\mathcal{G}(,)$ directly depends on the estimated similarity matrix $\boldsymbol{K}$ [14]. Moreover, an adequate similarity measure for clustering should hold the following two kinds of assumptions of consistency [29]: *(i)* nearby points in the input space should have high similarity (local consistency); *(ii)* points belonging to the same cluster should reach high similarity (global consistency).

Aiming to enhance the local and global consistency of the similarity matrix, we propose to employ a kernel alignment-based function that is subject to sparse constraints. As a result, a graph pruning method is developed that takes advantage of the initial guess for the relationship of the input samples, making possible to extract complex data structures. Thus, based on the properties of the Gaussian kernel, the following compactly supported kernel can be constructed [11]:

$$\kappa_\phi\left(\boldsymbol{x}_i,\boldsymbol{x}_j;\sigma\right)=\phi\left(\boldsymbol{x}_i,\boldsymbol{x}_j\right)\kappa\left(\boldsymbol{x}_i,\boldsymbol{x}_j;\sigma\right). \tag{1}$$

being $\boldsymbol{K}$ the similarity matrix encoding the graph structure, $\mathcal{G}(\boldsymbol{X},\boldsymbol{K})$, and $\phi:\mathbb{R}^D\times\mathbb{R}^D\to\mathbb{R}^+$, is a compactly supported radial basis function. To preserve positive definiteness of κ_ϕ and to enhance the local and global data consistency in $\boldsymbol{K}$, operator $\phi(,)$ is chosen as a sparseness function [12]:

$$\phi\left(\boldsymbol{x}_i,\boldsymbol{x}_j;b,\nu\right)=\left(\max\{1-(\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2)/b,0\}\right)^\nu, \tag{2}$$

being $b\in\mathbb{R}^+$. Notation $\max\{\cdot,0\}$ stands for the maximum value between the argument and zero. The power term that rules the degree of smoothness (i.e. differentiability) of $\phi(,)$ is adjusted as $\nu\geq(D+1)/2$. Therefore, the sparseness function introduces a hard threshold in Eq. (2), making all entries having distance $\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2 > b$

to be zero. Based on the operator ϕ established in Eq. (2), a compactly supported kernel-based matrix $\boldsymbol{K}^{b,\nu}\in\mathbb{R}^{N\times N}$ can be computed as: $\boldsymbol{K}^{b,\nu}=\boldsymbol{S}^{b,\nu}\circ\boldsymbol{K}$, where $\boldsymbol{S}^{b,\nu}\in\mathbb{R}^{N\times N}$ is a sparse matrix holding elements $s_{ij}^{b,\nu}=\phi\left(\boldsymbol{x}_i,\boldsymbol{x}_j;b,\nu\right)$, and $\circ$ stands for the Hadamard product. Nonetheless, the values of the b and ν parameters must be properly adjusted to reveal the main structures of $\boldsymbol{X}$ for facilitating further spectral clustering analysis. Yet, the latter parameter has a negligible effect in comparison with the former one when building the compactly supported kernel as discussed in [28]. Hence, to achieve a suitable local and global data structure representation, we just tune the value b by searching the sparse kernel encoding the most relevant node connections. To this end, we weaken all irrelevant relationships of the input similarity matrix $\boldsymbol{K}$, but taking into account the loss of information during the sparsification process in terms of a kernel alignment-based cost function [7]. Namely, the reciprocal relationship between $\boldsymbol{K}^{b,\nu}$ and $\boldsymbol{K}$ can be estimated using the correlation index, $\rho(b,\nu)\in\mathbb{R}[0,1]$ as:

$$\rho(b,\nu)=\frac{\langle\bar{\boldsymbol{K}},\bar{\boldsymbol{K}}^{b,\nu}\rangle_F}{\sqrt{\langle\bar{\boldsymbol{K}},\bar{\boldsymbol{K}}\rangle_F\langle\bar{\boldsymbol{K}}^{b,\nu},\bar{\boldsymbol{K}}^{b,\nu}\rangle_F}},\tag{3}$$

where matrices $\bar{\boldsymbol{K}}=\boldsymbol{H}\boldsymbol{K}\boldsymbol{H}$ and $\bar{\boldsymbol{K}}^{b,\nu}=\boldsymbol{H}\boldsymbol{K}^{b,\nu}\boldsymbol{H}$ are the centralized kernel versions of $\boldsymbol{K}$ and $\boldsymbol{K}^{b,\nu}$, respectively, and the centralization matrix $\boldsymbol{H}\in\mathbb{R}^{N\times N}$ is defined as $\boldsymbol{H}=\boldsymbol{I}-N^{-1}\boldsymbol{1}\boldsymbol{1}^\top$, where $\boldsymbol{1}\in\mathbb{R}^N$ is the all-ones vector and $\boldsymbol{I}\in\mathbb{R}^{N\times N}$ is the identity matrix. Notation $\langle,\rangle_F$ stands for the Frobenius norm. It is worth noting that centered alignment-based functions have been demonstrated to correlate better than the uncentered case [6]. As a result, the higher the $\rho(b)$ value the lower information loss during the sparsifying process. Additionally, an sparsity index that defines the degree of matrix sparseness is quantified as $\varrho(b,\nu)=N_0/N^2$, with $\varrho(b,\nu)\in\mathbb{R}[0,1]$, where N_0 is the number of zero entries of the matrix $\boldsymbol{K}^{b,\nu}$. Here, the higher the $\varrho(b,\nu)$ value – the higher degree of sparseness.

In order to fix the optimal value of b, for a given fixed ν value, we introduce a regularization-based criterion as to reach a trade-off between both $\rho(b,\nu)$ and $\varrho(b,\nu)$ measures as follows:

$$\begin{aligned}b^*&=\underset{b}{\operatorname{argmax}}\sqrt{(1-\gamma)(\log(\rho(b,\nu)))^2+\gamma(\log(\varrho(b,\nu)))^2}\\&\text{s.t. }\min_{i,j}\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2<b<\max_{i,j}\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2\end{aligned}\tag{4}$$

where $\gamma\in\mathbb{R}[0,1]$ rules the compromise between the local and global consistency terms. As $\gamma\to 0$, the optimization function in Eq. (4) heavily penalizes the sparsifying process, that is, the obtained matrix $\boldsymbol{K}^{b^*,\nu}$ will try to preserve, as well as possible, all data similarities hold by $\boldsymbol{K}$. Therefore, matrix $\boldsymbol{K}^{b,\nu}$ will hold mainly global consistencies. On the contrary, as $\gamma\to 1$, the obtained compactly supported matrix will favor those sparse representations preserving local data consistency. It is worth noting that the imposed constraint in Eq. (4) is derived from the definition of the value b, according to Eq. (2). Thus, if b becomes lower that the minimum value of all computed input sample distances, the sparsifying function will always be zero, making useless the derived matrix. In contrast, if b becomes higher, the sparsifying function will not affect $\boldsymbol{K}$. Therefore, the provided optimization in Eq. (4) takes advantage of the relevant input data information. Thus, the cost function allows finding the b^* value that is adequate to extract the main data structures to be encoded in $\boldsymbol{K}^{b,\nu}$. This kernel matrix is used to build a suitable input data graph representation $\mathcal{G}(\boldsymbol{X},\boldsymbol{K}^{b^*,\nu})$.

3. Experimental set-up

Evaluation of the proposed Kernel Alignment-based Graph Pruning (KAGP) approach is carried out by performing an unsupervised clustering task demanding estimation of the graph structure from the underlying data. For the sake of comparison, KAGP performance is applied on four different spectral clustering approaches requiring computation of the initial graph representation, namely: Adjusted Line Segment (ALS) [27], k-Nearest Neighbor Spectral Clustering (k-SC) [22], ϵ-Spectral Clustering (ϵ-SC) [10], and Common Nearest Neighbors (CNN) [29]. Parameter setting of these algorithms is as follows:

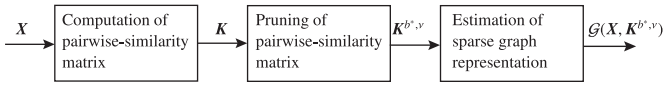

Fig. 1. KAGP block scheme.

The ALS algorithm incorporates a prior assumption about consistency of the similarity between samples, which means that nearby points and data points on the same structure are likely to share high similarities [27]. The ALS similarity measure is defined as $k_{ij}=(z_{ij}+1)^{-1}$, where $z_{ij}\mathbb{R}^+$ is an introduced pair-wise density sensitive distance defined as, $z_{ij}=\min\sum_{r=i}^{|P_{ij}|}\iota(\boldsymbol{x}_r,\boldsymbol{x}_{r+1})$. Here, $P_{ij}=\{\boldsymbol{x}_i,\boldsymbol{x}_r,\ldots,\boldsymbol{x}_j:i\le r<j\}$ is the path and denotes the set of points connecting from $\boldsymbol{x}_i$ to $\boldsymbol{x}_j$. In turn, the parameter $\iota(,)\in\mathbb{R}^+$ is an adjustable line segment length between points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ computed as follows: $\iota(\boldsymbol{x}_i,\boldsymbol{x}_j)=\left(\exp\left(\zeta\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2\right)-1\right)^{1/\zeta}$, where $\zeta>1$ is a density factor parameter that squeezes distances within high-density regions while it widens them in low-density regions. The ζ value is heuristically set as 2, providing that the initial graph is built as an ϵ-neighborhood graph fixing $\epsilon=\xi$.

In the k-SC algorithm, the similarity matrix is estimated as: $k_{ij}=\kappa\left(\boldsymbol{x}_i,\boldsymbol{x}_j;\sqrt{\sigma_i\sigma_j}\right)$, where the local scaling parameters $\{\sigma_i,\sigma_j\}\in\mathbb{R}^+$ are computed in terms of the Euclidean distance as $\sigma_i=\|\boldsymbol{x}_i-\boldsymbol{x}_K\|_2$, being $\boldsymbol{x}_K$ the Kth neighbor of each point $\boldsymbol{x}_i$, so that each specific scaling parameter allows self-tuning of the point-to-point distances according to the local statistics of the neighborhoods surrounding points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ [22]. In order to avoid overfitting, the K value is adjusted (taking into account the dataset size) as $K=\lfloor\sqrt{N}\rceil$, where $\lfloor\cdot\rceil$ is the operator that computes the rounded value to the closest integer for its argument.

Now, in the ϵ-SC approach the similarity is computed as: $k_{ij}=\kappa\left(\boldsymbol{x}_i,\boldsymbol{x}_j;\sigma_\epsilon\right)$, where the $\sigma_\epsilon\in\mathbb{R}^+$ value encodes the average neighborhood size of the input data. As suggested in [1], we choose the median operator. The pairwise similarity of the CNN algorithm becomes adaptive in dependence to the neighborhoods of the correlative points, that is, if a couple of points are located within the same cluster, both points are assumed as belonging to a high density region [29]. Therefore, the CNN local density adaptive similarity measure can be written as: $k_{ij}=\kappa\left(\boldsymbol{x}_i,\boldsymbol{x}_j;(\varsigma_{ij}+1)^{1/2}\sigma_\varsigma\right)$, where $\sigma_\varsigma=\sigma_\epsilon$, $\varsigma_{ij}\in\mathbb{N}$ is a local density parameter, which is employed to distinguish points within the same cluster from others located at different clusters as:

$$\varsigma_{ij}=\left|\{r:\|\boldsymbol{x}_i-\boldsymbol{x}_r\|<\xi\text{ and }\|\boldsymbol{x}_j-\boldsymbol{x}_r\|<\xi\}\right|,$$

being $\xi\in\mathbb{R}^+$ a neighborhood radius parameter that is adjusted experimentally as the 10-percentile of the input data Euclidean distances. Notation $|\cdot|$ denotes the cardinality operator.

Once the similarity matrix is computed for each method, the proposed KAGP is carried out to prune all irrelevant pair-wise relationship values, where the optimal value of the parameter b is computed by solving the cost function in Eq. (4) using a Particle Swarm Optimization-based solver. Besides, the regularization parameter γ is heuristically set as 0.5. Lastly, the sparse matrix $\boldsymbol{K}^{b^*,\nu}$ is estimated to perform further the well-known spectral clustering algorithm [25]. Fig. 1 outlines the main sketch of the used spectral clustering task used for validation of the proposed KAGP approach.

The KAGP method is validated as a suitable tool for pruning graph representations to improve unsupervised clustering performance. To

this end, we employ both, synthetic and real-world, databases. In the former case, we have visual insight of the performed clustering for the following well-known datasets: *Bull's eye 3 circles*, *Happy face* [22], *Half moon* [13], and *Bull's eye with outliers*. Two main reasons account for selecting these concrete databases: (*i*) they represent a challenging clustering task due to their complex structures, and (*ii*) each ground-truth is either known or can be visually inspected. Regarding the real-world data, a subset of the UCI Machine Learning Repository that is widely used for quantifying clustering performance [27,29]. The selected data collection provides a variety of real-world clustering scenarios with different conditions in terms of number of features, number of samples, number of clusters, and data distribution complexity. Besides, validation is carried out on a representative subset of 30 images chosen randomly from the free access Berkeley Segmentation dataset [17]. This data collection provisions hand-labeled segmentation of every sample, making supervised testing suitable. All testing images are rescaled at 15% and characterized by five features: RGB color space and the spatial position of each pixel. Thus, each image is represented by the input matrix $\boldsymbol{X} \in \mathbb{R}^{3577\times 5}$, where $N{=}73 \times 49$ is the obtained image resolution after resizing.

Further, we assess the clustering performance in terms of the following commonly used indexes of quality [2,4]:

Adjusted Rand Index (ARI), $\rho_{ARI} \in \mathbb{R}[-1, 1]$: This index measures the agreement between two compared partitions, namely, the ground truth (noted as $\mathcal{U}$) and the estimated by the tested clustering approach ($\mathcal{V}$), as follows:

$$\rho_{ARI} = \frac{a_{11} - (a_{11}+a_{01})(a_{11}+a_{10})/a_{00}}{(a_{11}+a_{01}) + (a_{11}+a_{10})/2 - (a_{11}+a_{01})(a_{11}+a_{10})/a_{00}} \tag{5}$$

where $a_{11} \in \mathbb{N}$ is the number of sample pairs belonging to the same subset in $\mathcal{U}$ and in $\mathcal{V}$, a_{10} is the number of sample pairs belonging to the same subset in $\mathcal{U}$ and to different subsets in $\mathcal{V}$, $a_{01} \in \mathbb{N}$ is the number of sample pairs belonging to different subset in $\mathcal{U}$ and to the same one in $\mathcal{V}$, and $a_{00} \in \mathbb{N}$ is the number of sample pairs belonging to different subsets in $\mathcal{U}$ and in $\mathcal{V}$.

Purity Index (PUR) $\rho_{PUR} \in \mathbb{R}[0, 1]$: This measure matches the clustering partition $\mathcal{V}$ with the ground truth $\mathcal{U}$ as a weighted sum of the maximal precision values for each subset. That is:

$$\rho_{PUR} = \sum_{c=1}^{C} \frac{|\mathcal{V}_c|}{N} \max_j \rho_{PRE}\left(\mathcal{V}_c, \mathcal{U}_j\right); \tag{6}$$

where $\rho_{PRE}\left(\mathcal{V}_c, \mathcal{U}_j\right) = |\mathcal{V}_c \cap \mathcal{U}_j|/|\mathcal{V}_c|$.

Accuracy Index (ACC) $\rho_{ACC} \in \mathbb{R}[0, 1]$: This index is the total fraction of samples belonging to the same subset in $\mathcal{U}$ and $\mathcal{V}$, and is expressed as:

$$\rho_{ACC} = \frac{1}{|\mathcal{V}|} \sum_{c=1}^{C} |\mathcal{V}_c \cap \mathcal{U}_c| \tag{7}$$

where $|\cdot|$ is the cardinality operator over a given set.

Jaccard Index (JAC), $\rho_{JAC} \in \mathbb{R}[0, 1]$: This index matches the similarity among two sets, $\mathcal{U}$ and $\mathcal{V}$, as follows:

$$\rho_{JAC} = \frac{a_{11}}{a_{11} + a_{10} + a_{01}} \tag{8}$$

Probabilistic Rand Index (NPR) This measure allows comparing the performed partition under testing $\mathcal{V}$ against T available ground truths, $\Psi = \{\mathcal{U}^t : t{=}1, \ldots, T\}$, through an introduced soft nonuniform weighting of all sample pairs as function of the ground truth variability. We assume that each $\mathcal{U}^t$ is the tth partition of $\boldsymbol{X}$ according to the tth expert, so that $\mathcal{U}^t = \cup_{c=1}^{C} \mathcal{U}_c^t$, being $\mathcal{U}_c^t$ a disjoint subset in $\mathcal{U}^t$ and $\mathcal{U}_{c'}^t \cap \mathcal{U}_c^t = \emptyset$, $\forall\, c' \neq c$. Therefore, the NPR, is defined as [19,24]:

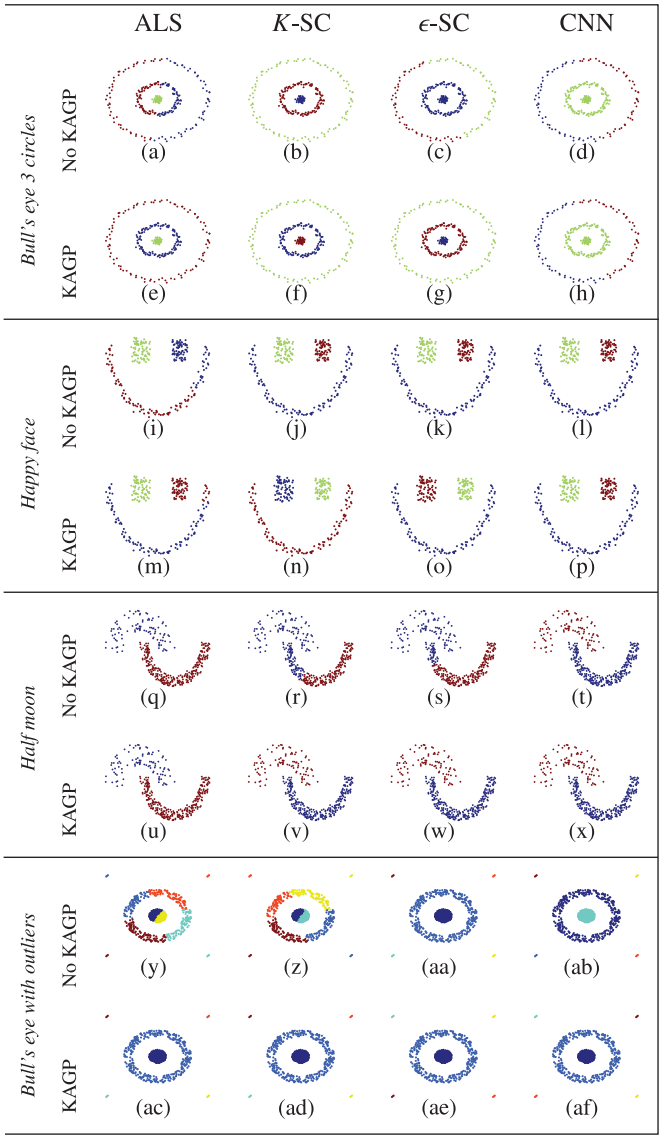

Fig. 2. Clustering results carried out on synthetic data sets.

$$\rho_{NPR} = \frac{1}{T\binom{N}{2}} \sum_{t=1}^{T} \sum_{i,j=1}^{N} \left[\delta\left(l_i^{\mathcal{V}} = l_j^{\mathcal{V}}\right)\delta\left(l_i^{\mathcal{U}^t} = l_j^{\mathcal{U}^t}\right) + \delta\left(l_i^{\mathcal{V}} \neq l_j^{\mathcal{V}}\right)\delta\left(l_i^{\mathcal{U}_t} \neq l_j^{\mathcal{U}_t}\right)\right] \tag{9}$$

where $l_i^{\mathcal{V}} = \{c : \boldsymbol{x}_i \in \mathcal{V}_c\}$, $l_i^{\mathcal{U}^t} = \{c : \boldsymbol{x}_i \in \mathcal{U}_c^t\}$, and notation $\delta(\cdot, \cdot)$ stands for the delta function.

4. Results and discussion

To evaluate the proposed pruning method, we firstly carry out the visual inspection of the performed data grouping for all tested databases. However, the indexes of quality above-explained in Section 2 are used depending on the available information. Thus, the ρ_{NPR} index is the only one estimated for the Berkeley dataset due to the hand-labeled segmentation is for every testing image. Otherwise, we calculate the ρ_{ACC}, ρ_{PUR}, and ρ_{ARI} indexes in the remaining databases. Fig. 2 shows the accomplished grouping results on the synthetic datasets before and after applying the proposed KAGP method to each compared spectral clustering technique. The first method we compare is the ALS that aims to reveal complex data structures by encoding topological and density properties of neighboring

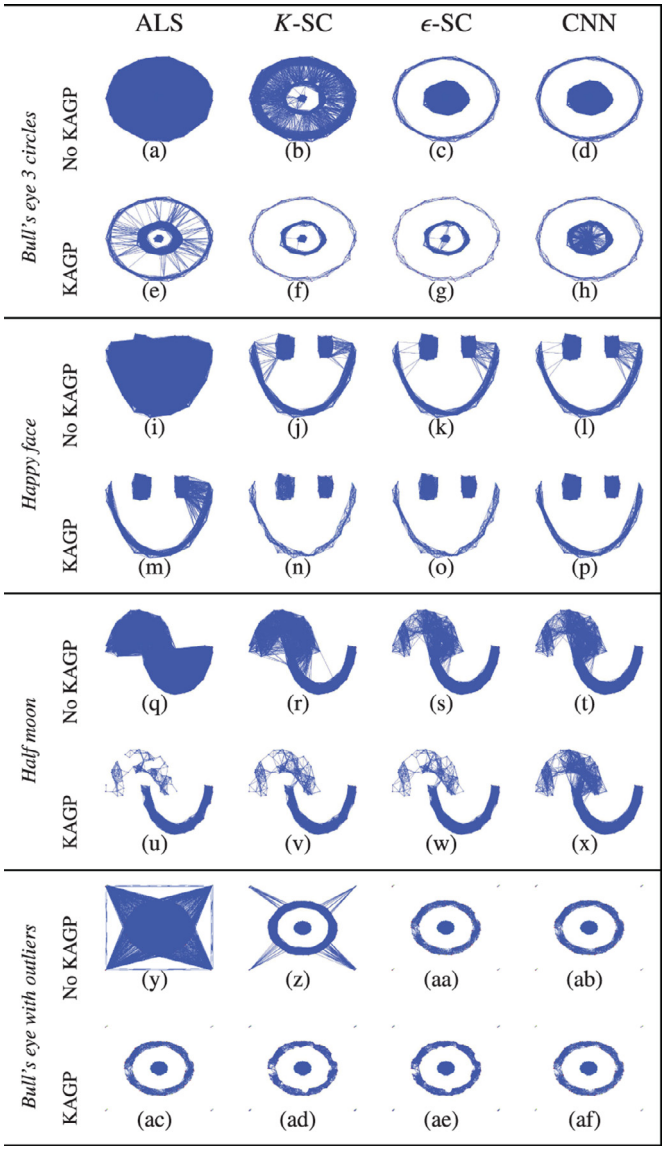

Fig. 3. Graph representation for synthetic data sets.

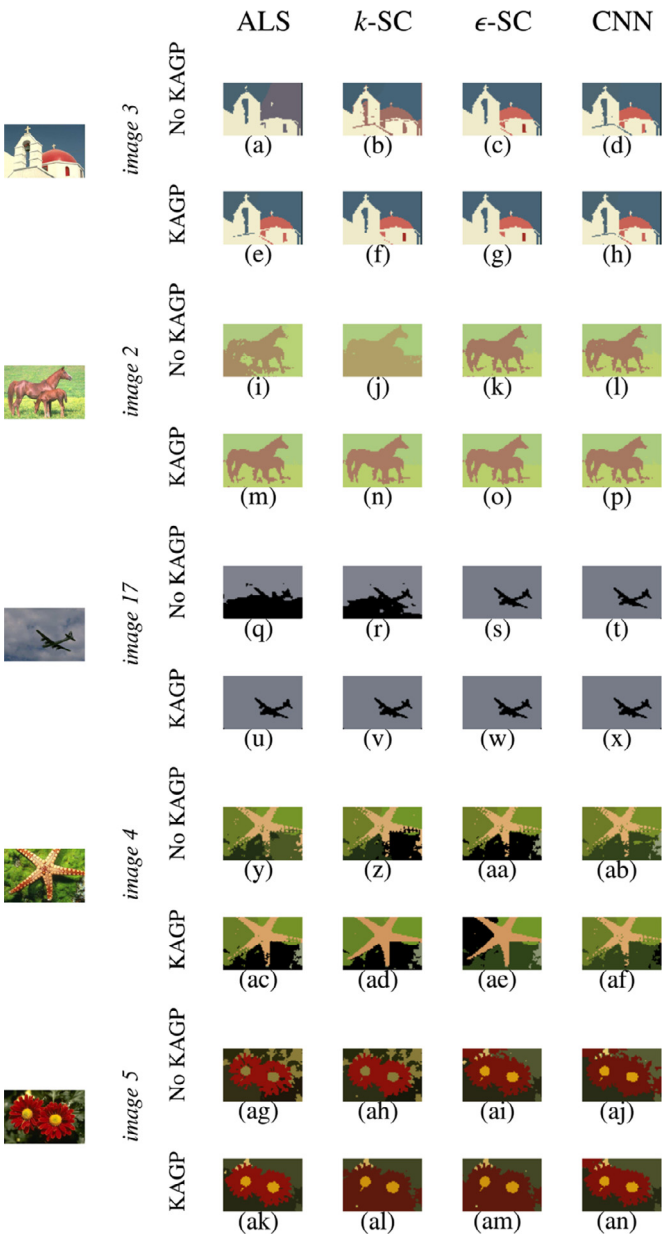

Fig. 4. Exemplary of obtained clustering results for the Berkeley Segmentation dataset.

samples. And yet, the use of a unique density parameter does not allow squeezing properly the distances among points. Moreover, the plain ALS (i.e. without the proposed graph pruning) gets the worst connectivity graphs on every single dataset (see Fig. 3(a), (i), (q), and (y)), giving rise to poor splitting data. Still, the use of the proposed KAGP method remarkably improves clustering performance on all databases (Fig. 2(e), (m), and (ac)) except for the *Half moon* dataset that is the one having the simpler structure (Fig. 2(u)).

The following clustering method, *k*-SC technique, handles correctly on *Bull's eye 3 circles* and *Happy face* data collections as seen in Fig. 2(b) (j), respectively. However, the clusters are wrongly split in the *Half moon* (Fig. 2(r)) and the *Bull's eye with outliers* (Fig. 2(z)) datasets. This drawback is explained since the *k*-SC method mainly seeks just for local consistency preservation, making some global connections supply wrong clusters mostly at handling complex circumstances, e.g., a multi-scale data set [22]. In fact, the estimated connectivity graphs show that the plain *k*-SC graphs incorrectly assign some pairwise connections, namely, when both moons get close enough to each other (see Fig. 3(r)) or when dealing with outliers (Fig. 3(z)) that are mistakenly assigned to the structure of the inner rings. Conversely, the proposed KAGP algorithm performs better clustering since it produces graphs (Fig. 3(v) and (ad)) that evidently prune irrelevant connections from the *k*-SC similarity matrix, but jointly preserving both the local and global consistencies.

The improved version, ϵ-SC, is assumed to better describe local relationships through an introduced ϵ value that rules the size of neighboring vicinities within the connectivity graph estimation framework. Nonetheless, this strategy may be not enough to tackle the problem inasmuch as the ruling ϵ parameter barely handles multi-scale, noisy or complex structures, yielding connection graphs (see Fig. 3(c)–(ae)) that are quite similar to the ones estimated by the *k*-SC method. As a result, the ϵ-SC reaches the same clustering performance as the *k*-SC does, that is, the method does not benefit from the KAGP algorithm in the cases of *Bull's eye 3 circles* (see Fig. 2(c) and (g)) and *Happy face* (Fig. 2(k) and (o)) while the pruning method improves performance on the *Half moon* (Fig. 2(s) and (w)) and the *Bull's eye with outliers* (Fig. 2(aa) and (ae)) datasets.

Lastly, the compared CNN method provides a more elaborate strategy to improve characterization of local neighborhoods by

Table 1
Clustering quality assessment results (synthetic datasets).

Dataset	Method	ALS		k-SC		ϵ-SC		CNN	
	Measure	No KAGP	KAGP	No KAGP	KAGP	No KAGP	KAGP	No KAGP	KAGP
be3	ARI	0.31	1.00	1.00	1.00	0.51	1.00	0.51	0.51
$N = 299$	Purity	0.61	1.00	1.00	1.00	0.84	1.00	0.84	0.84
$D = 2$	Accuracy	0.61	1.00	1.00	1.00	0.64	1.00	0.63	0.63
$C = 3$	Jaccard	0.39	1.00	1.00	1.00	0.56	1.00	0.56	0.56
happy	ARI	1.00	1.00	1.00	1.00	0.85	1.00	1.00	1.00
$N = 266$	Purity	1.00	1.00	1.00	1.00	0.95	1.00	1.00	1.00
$D = 2$	Accuracy	1.00	1.00	1.00	1.00	0.95	1.00	1.00	1.00
$C = 3$	Jaccard	1.00	1.00	1.00	1.00	0.83	1.00	1.00	1.00
hm2	ARI	-0.08	0.01	0.37	1.00	0.59	1.00	0.97	0.98
$N = 373$	Purity	0.80	0.66	0.81	1.00	0.89	1.00	0.99	0.99
$D = 2$	Accuracy	0.54	0.60	0.81	1.00	0.89	1.00	0.99	0.99
$C = 2$	Jaccard	0.44	0.41	0.56	1.00	0.70	1.00	0.97	0.98
tar	ARI	0.38	1.00	0.38	1.00	1.00	1.00	1.00	1.00
$N = 770$	Purity	0.40	1.00	0.40	1.00	1.00	1.00	1.00	1.00
$D = 2$	Accuracy	0.40	1.00	0.40	1.00	1.00	1.00	1.00	1.00
$C = 6$	Jaccard	0.38	1.00	0.38	1.00	1.00	1.00	1.00	1.00
Average	ARI	0.40	**0.75**	0.69	**1.00**	0.74	**1.00**	0.87	0.87
	Purity	0.70	**0.92**	0.80	**1.00**	0.92	**1.00**	0.96	0.96
	Accuracy	0.64	**0.90**	0.80	**1.00**	0.87	**1.00**	0.91	0.91
	Jaccard	0.55	**0.85**	0.74	**1.00**	0.77	**1.00**	0.88	**0.89**

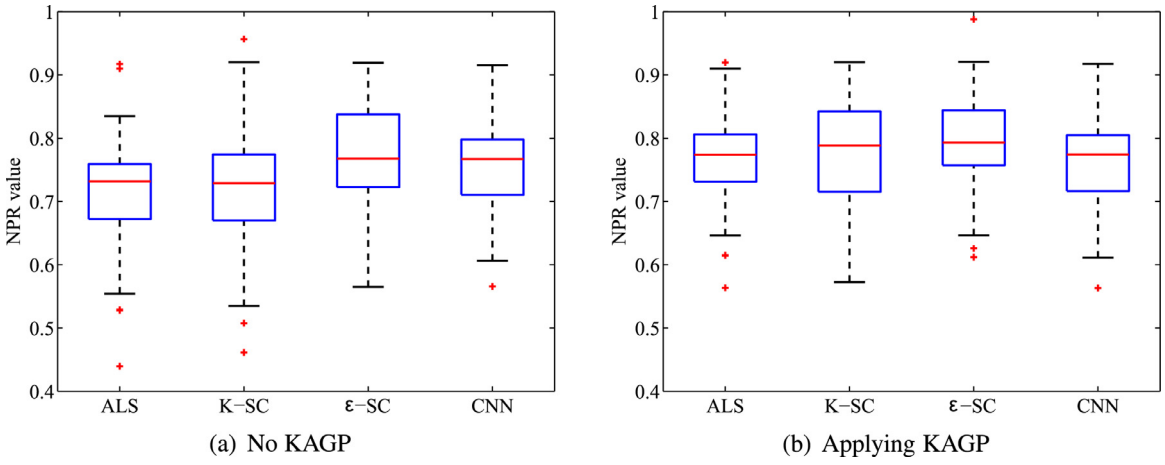

Fig. 5. Berkeley Segmentation dataset results (statistical analysis of the NPR).

adapting the pairwise edge weight between points, allowing to handle more information about multi-scale data. Detailed inspection of the CNN graphs shows that the use of the KAGP method removes more accurately irrelevant connections (see Fig. 3(l) vs (p) and Fig. 3(t) vs (x)). Thus, the CNN method reaches almost the same performance in either case of preprocessing consideration as seen in Fig. 2(d)–(af).

Table 1 displays the clustering quality assessments estimated on the examined synthetic datasets. As perceived, the proposed KAGP approach allows improving all considered indices of clustering quality. In particular, KAGP-based graph pruning remarkably enhances the clustering quality of the ALS, k-SC and ϵ-SC techniques in all tested synthetic data. Yet, the KAGP does not exhibit a notable quality enhancement of the CNN algorithm, though it does not decrease the system performance.

As seen in Table 2 showing the quality assessments calculated for the UCI repository real-world datasets, the proposed KAGP enhances (at least, does not degrade) the performed clustering. In particular, KAGP remarkably enhances the purity index for the k-SC, the ϵ-SC, and the CNN algorithms. Thus, pruning of irrelevant connections using KAGP favors the clustering robustness against noisy and/or complex data distributions. Concerning the ARI, the Accuracy, and the Jaccard indexes, KAGP achieves comparable results in comparison to the benchmark approaches.

The image segmentation results are shown in Fig. 4, where it is clear how the proposed KAGP enhances the segmentation performance. Particularly, as seen in Fig. 4(b)–(al), it is possible to notice how after applying KAGP over the k-SC similarity matrix facilitates the discrimination among objects into the scene. Alike, KAGP is able to enhance the ϵ-SC-based graph representation for facilitating the image clustering, e.g., see Fig. 4(c)–(am). Regarding this, only considering local consistencies for computing the relationships among pixels, e.g., k-SC and ϵ-SC algorithms, leads to noisy segmentation results. Moreover, it is remarkably how the image segmentation results for CNN and ALS approaches are improved after carrying out the proposed KAGP (see Fig. 4(d)(an), and Fig. 4(a)–(ak)). Even though CNN and ALS approaches estimate pair-wise similarities by considering local neighborhood properties complex data structures related to object textures and shapes are not suitable highlighted. In terms of clustering assessment, Fig. 5 shows the corresponding boxplots obtained for the tested images. As seen, the use of the KAPG makes more stable the estimated clustering performance since it makes the results be less variable with less outliers. Consequently, the KAGP is able to find a trade-off between local and global consistency

Table 2
Clustering quality assessment results (UCI repository datasets).

Dataset	Method	ALS		k-SC		ϵ-SC		CNN	
	Measure	No KAGP	KAGP	No KAGP	KAGP	No KAGP	KAGP	No KAGP	KAGP
iris	ARI	0.47	0.47	0.44	0.55	0.63	0.55	0.63	0.56
$N = 150$	Purity	0.91	0.91	0.87	0.97	0.84	0.97	0.84	0.98
$D = 4$	Accuracy	0.58	0.57	0.53	0.68	0.84	0.68	0.84	0.69
$C = 3$	Jaccard	0.52	0.51	0.50	0.58	0.60	0.58	0.60	0.59
wine	ARI	0.03	0.10	0.93	0.43	0.93	0.43	0.90	0.43
$N = 178$	Purity	0.57	0.54	0.98	0.94	0.98	0.94	0.97	0.94
$D = 13$	Accuracy	0.44	0.46	0.98	0.62	0.98	0.62	0.97	0.62
$C = 3$	Jaccard	0.24	0.27	0.91	0.50	0.91	0.50	0.87	0.50
sonar	ARI	−0.00	0.03	0.02	0.00	0.00	0.01	−0.00	0.01
$N = 208$	Purity	0.97	0.92	0.57	0.86	0.55	0.84	0.57	0.84
$D = 60$	Accuracy	0.50	0.60	0.57	0.55	0.54	0.56	0.52	0.56
$C = 2$	Jaccard	0.48	0.47	0.34	0.43	0.34	0.42	0.34	0.43
biomed	ARI	0.04	−0.00	0.47	0.11	0.09	0.10	0.09	0.09
$N = 194$	Purity	0.60	0.53	0.85	0.93	0.95	0.94	0.95	0.94
$D = 5$	Accuracy	0.60	0.51	0.85	0.71	0.70	0.71	0.70	0.70
$C = 2$	Jaccard	0.37	0.35	0.60	0.55	0.55	0.55	0.55	0.55
diabetes	ARI	−0.00	−0.00	0.16	0.01	0.16	0.00	0.01	−0.00
$N = 768$	Purity	0.99	0.99	0.70	0.99	0.70	0.99	0.95	0.99
$D = 8$	Accuracy	0.64	0.64	0.70	0.65	0.70	0.65	0.65	0.65
$C = 2$	Jaccard	0.54	0.54	0.43	0.54	0.43	0.54	0.52	0.54
glass	ARI	0.08	0.02	0.17	0.16	0.16	0.16	0.16	0.16
$N = 214$	Purity	0.42	0.41	0.51	0.89	0.87	0.89	0.87	0.89
$D = 9$	Accuracy	0.42	0.38	0.46	0.49	0.50	0.50	0.50	0.50
$C = 4$	Jaccard	0.20	0.18	0.25	0.34	0.33	0.34	0.34	0.34
x80	ARI	0.00	0.00	0.63	0.01	0.36	0.01	0.31	0.01
$N = 45$	Purity	0.96	0.96	0.87	0.87	0.76	0.89	0.71	0.89
$D = 8$	Accuracy	0.36	0.36	0.87	0.42	0.58	0.40	0.58	0.40
$C = 3$	Jaccard	0.31	0.31	0.60	0.29	0.42	0.30	0.38	0.30
ecoli	ARI	0.40	0.69	0.37	0.57	0.48	0.72	0.57	0.71
$N = 336$	Purity	0.63	0.75	0.56	0.72	0.66	0.80	0.71	0.77
$D = 7$	Accuracy	0.54	0.75	0.55	0.72	0.66	0.79	0.71	0.77
$C = 8$	Jaccard	0.36	0.62	0.32	0.51	0.42	0.66	0.50	0.65
heart	ARI	−0.00	0.00	0.31	0.02	0.37	0.41	0.39	0.02
$N = 297$	Purity	1.00	1.00	0.78	0.93	0.80	0.82	0.81	0.95
$D = 13$	Accuracy	0.54	0.54	0.78	0.58	0.80	0.82	0.81	0.59
$C = 2$	Jaccard	0.50	0.50	0.49	0.47	0.52	0.55	0.54	0.48
liver	ARI	0.03	0.02	−0.00	−0.01	−0.01	−0.01	−0.01	−0.01
$N = 345$	Purity	0.59	0.59	0.65	0.94	0.98	0.95	0.97	0.97
$D = 6$	Accuracy	0.59	0.58	0.52	0.56	0.57	0.56	0.57	0.56
$C = 2$	Jaccard	0.35	0.35	0.36	0.48	0.50	0.48	0.50	0.49
ionosphere	ARI	−0.00	−0.00	0.15	0.35	0.13	0.28	0.12	0.27
$N = 351$	Purity	1.00	1.00	0.70	0.83	0.68	0.86	0.68	0.87
$D = 34$	Accuracy	0.64	0.64	0.70	0.81	0.68	0.78	0.68	0.77
$C = 2$	Jaccard	0.54	0.54	0.44	0.60	0.41	0.58	0.41	0.58
soybean2	ARI	0.28	0.27	0.56	0.28	0.29	0.26	0.30	0.28
$N = 136$	Purity	0.77	0.75	0.82	0.76	0.78	0.76	0.79	0.77
$D = 35$	Accuracy	0.58	0.57	0.82	0.57	0.59	0.57	0.60	0.58
$C = 4$	Jaccard	0.35	0.34	0.52	0.35	0.36	0.34	0.36	0.35
Average	ARI	0.11	**0.13**	**0.35**	0.21	**0.30**	0.24	**0.29**	0.21
	Purity	0.78	0.78	0.74	**0.89**	0.80	**0.89**	0.82	**0.90**
	Accuracy	0.54	**0.55**	**0.69**	0.61	**0.68**	0.64	**0.68**	0.61
	Jaccard	0.40	**0.42**	**0.48**	0.47	0.48	**0.49**	**0.49**	0.48

preservation to build a graph representation without irrelevant pairwise connections, favoring the clustering robustness against outliers, noisy data, and overlapped groups.

Sensitivity analysis of free parameters. Due to all compared spectral clustering approaches have free parameters to be specified manually, we evaluate their influence on the estimation of the initial pairwise-similarity matrix, $\boldsymbol{K}$. Thus, the following free parameters are studied: *(i)* The kth neighboring value for k-SC, *(ii)* the average neighborhood size value σ_ϵ for the ϵ-SC approach, *(iii)* the neighborhood radius parameter ξ for the CNN approach, and *(iv)* the density factor parameter ζ for ALS. Thereby, the synthetic and the Berkeley image segmentation datasets are studied. Thus, the ARI and the NPR indexes are analyzed, respectively. Fig. 6 shows the results of evaluation that is carried out in terms of the ARI measure before and after the KAGP operation. As seen in Figs. 6(a)–(c) for ALS method, the lack of the proposed pruning makes worse the achieved validation ARI measure regardless the fixed value of the ruling ζ parameter, except, again, for the *Half moon*. In the cases of associated k-SC and ϵ-SC methods (see Fig. 6(e)–(g),Fig. 6(i)–(k), Fig. 7(a)–(c))), their plain versions perform clustering that rapidly worsens as the corresponding free parameter slightly varies. In contrast, the use of the KAPG-based pruning makes both techniques improve the validation outcomes over all tested data sets.

With respect to the CNN method, its plain version is the one that holds clustering performance within the longest interval of the ξ parameter variation in comparison with the other plain versions. Yet, its provided clustering sharply decreases, at the moment of parameter imbalance, and becomes the worst as seen in Fig. 6(m)–(o) and Fig. 7(d) and (f). By contrast, the insertion of the pruning operation significantly compensates for this negative effect, extending the range within the free parameter can change.

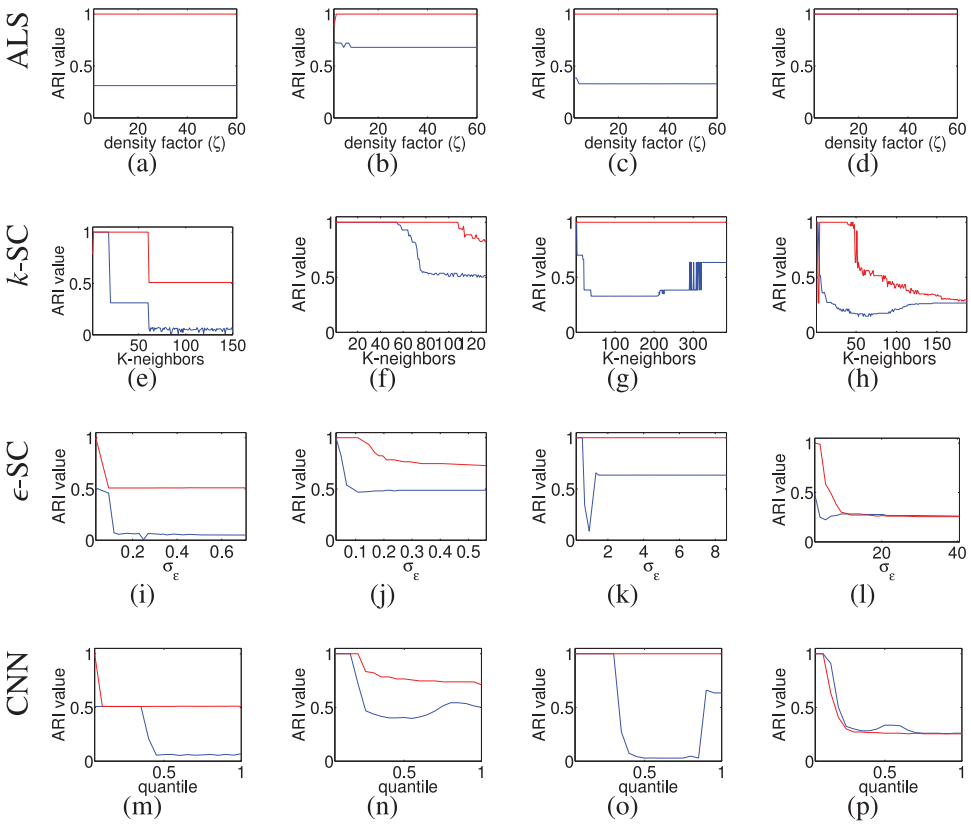

Fig. 6. Free parameter analysis over synthetic datsets based on the ARI measure. — No KAGP — after applying KAGP.

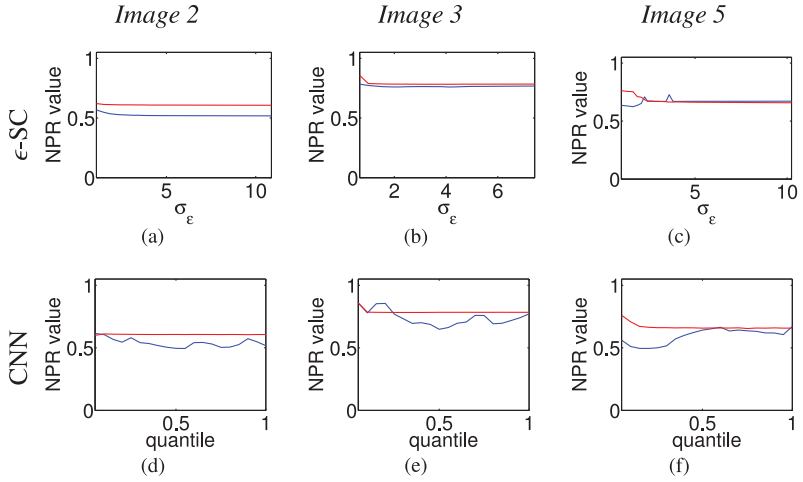

Fig. 7. ϵ-SC and CNN free parameter analysis over Berkeley image dataset based on the NPR measure. — No KAGP — after applying KAGP.

5. Conclusions

We propose a graph pruning approach, termed KAGP, that makes use of a kernel function to support grouping tasks based on spectral clustering. Here, the kernel matrix learning is based on an introduced alignment function to measure the similarity between two kernel matrices, enhancing their local and global consistencies. So, our approach takes advantage of an initial guess of the relationships between points to identify relevant connections by encoding then by means of a compactly supported kernel function. Besides, a regularization-based criterion is introduced as to reach a trade-off between the local and the global consistency preservation during the graph pruning process. For the sake of comparison, KAGP is validated on synthetic and real-world datasets using visual inspection and clustering quality measures. Performance is contrasted with four competitive graph-based spectral clustering approaches, namely: k-SC, ϵ-SC, CNN, and ALS. So, once the initial graph representation is computed for each method, the proposed KAGP is carried out before clustering the data to prune irrelevant pair-wise relationships. Obtained results of quality show that KAGP can handle complex data structures, yielding better clustering performance in comparison to the baselines. Moreover, the KAGP promotes clustering performance less sensitive to outliers, noisy data, and overlapped groups.

Due to all compared spectral clustering approaches have free parameters that commonly are to be specified manually, we also evaluate experimentally their influence during the graph pruning pro-

cess. Attained results demonstrate that the insertion of KAGP significantly compensates adverse effects when the corresponding free parameter is not fixed correctly for each clustering approach. Moreover, in most of the cases, KAGP allows extending the range within the free parameter can change. Therefore, proposed approach is a suitable alternative to support clustering tasks related to graph representations, achieving appropriate performances while avoiding the need for a comprehensive user knowledge regarding the influence of its free parameters. As future work, the authors plan to validate KAGP on other different machine learning tasks as dimensionality reduction, classification, and regression. Furthermore, an extension of KAGP to support kernel-based clustering approaches will be studied. Finally, it would be of benefit to include alternatives to measure the local and global consistency preservation, i.e., information theory.

Acknowledgments

This work is supported by the project 16882 funded by Universidad Nacional de Colombia sede Manizales and Universidad de Caldas, and by Programa Nacional de Formación de Investigadores “Generación del Bicentenario”, 2011/2012 funded by COLCIENCIAS.

References

[1] A. Álvarez-Meza, A. Castro-Ospina, G. Castellanos-Dominguez, Spectral Clustering Using Compactly Supported Graph Building, CIARP, Springer, 2014, pp. 327–334.
[2] E. Amigo, J. Gonzalo, J. Artiles, F. Verdejo, A comparison of extrinsic clustering evaluation metrics based on formal constraints, Inf. Retr. 12 (2009) 461–486.
[3] M. Beauchemin, A density-based similarity matrix construction for spectral clustering, Neurocomputing 151 (2015) 835–844.
[4] G. Chen, S.A. Jaradat, N. Banerjee, T.S. Tanaka, M.S. Ko, M.Q. Zhang, Evaluation and comparison of clustering algorithms in analyzing es cell gene expression data, Stat. Sin. 12 (2002) 241–262.
[5] W. Chen, G. Feng, Spectral clustering with discriminant cuts, Knowl. Based Syst. 28 (2012) 27–37.
[6] C. Cortes, M. Mohri, A. Rostamizadeh, Algorithms for learning kernels based on centered alignment, J. Mach. Learn. Res. 13 (2012) 795–828.
[7] N. Cristianini, J. Kandola, A. Elisseeff, J. Shawe-Taylor, On kernel target alignment, Innovations in Machine Learning, Springer, 2006, pp. 205–256.
[8] I.S. Dhillon, Y. Guan, B. Kulis, Kernel k-means: spectral clustering and normalized cuts, in: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2004, pp. 551–556.
[9] I.S. Dhillon, Y. Guan, B. Kulis, A Unified View of Kernel k-Means, Spectral Clustering and Graph Cuts, Citeseer, 2004b.
[10] M. Filippone, F. Camastra, F. Masulli, S. Rovetta, A survey of kernel and spectral methods for clustering, Patt. Recognit. 41 (2008) 176–190.
[11] M.G. Genton, Classes of kernels for machine learning: a statistics perspective, J. Mach. Learn. Res. 2 (2002) 299–312.
[12] T. Gneiting, Compactly supported correlation functions, J. Multivar. Anal. 83 (2002) 493–508.
[13] A.K. Jain, M.H. Law, Data clustering: A user’s dilemma, in: Pattern Recognition and Machine Intelligence, Springer, 2005, pp. 1–10.
[14] F. Jordan, M. Bach, F. Bach, Learning spectral clustering, Adv. Neural Inf. Process. Syst. 16 (2004) 305–312.
[15] W. Liu, J.C. Principe, S. Haykin, Kernel Adaptive Filtering: A Comprehensive Introduction, 57, John Wiley & Sons, 2011.
[16] Y. Lu, L. Wang, J. Lu, J. Yang, C. Shen, Multiple kernel clustering based on centered kernel alignment, Patt. Recognit. (2014).
[17] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: Proceedings of the ICCV, 2001, pp. 416–423.
[18] M. Meila, J. Shi, A random walks view of spectral segmentation, in: Proceedings of the AI and STATISTICS, 2001.
[19] S. Molina-Giraldo, A. Álvarez-Meza, D. Peluffo-Ordó nez, G. Castellanos-Domínguez, Image segmentation based on multi-kernel learning and feature relevance analysis, Proceedings of the IBERAMIA 2012, Springer, 2012, pp. 501–510.
[20] M. Nascimento, A. De Carvalho, Spectral methods for graph clustering–a survey, Eur. J. Oper. Res. 211 (2011) 221–231.
[21] A. Ng, M. Jordan, Y. Weiss, et al., On spectral clustering: Analysis and an algorithm, Adv. Neural Inf. Process. Syst. 2 (2002) 849–856.
[22] P. Perona, L. Zelnik-Manor, Self-tuning spectral clustering, Adv. Neural Inf. Process. Syst. 17 (2004) 1601–1608.
[23] J. Shi, J. Malik, Normalized cuts and image segmentation, Patt. Anal. Mach. Intell. IEEE Trans. 22 (2000) 888–905.
[24] R. Unnikrishnan, C. Pantofaru, M. Hebert, A measure for objective evaluation of image segmentation algorithms, in: Proceedings of the CVPR Workshops. IEEE Computer Society Conference on, IEEE, 2005, p. 34.
[25] U. Von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (2007) 395–416.
[26] X. Wang, C. Yang, J. Zhou, Clustering aggregation by probability accumulation, Patt. Recognit. 42 (2009) 668–675.
[27] P. Yang, Q. Zhu, B. Huang, Spectral clustering with density sensitive similarity function, Knowl. Based Syst. 24 (2011) 621–628.
[28] M.D. Buhmann, Radial basis functions: theory and implementations, Vol. 12, Cambridge university press, 2003.
[29] X. Zhang, J. Li, H. Yu, Local density adaptive similarity measurement for spectral clustering, Patt. Recognit. Lett. 32 (2011) 352–358.