Soft Kernel Spectral Clustering

Rocco Langone, Raghvendra Mall and Johan A. K. Suykens
Department of Electrical Engineering ESAT-SCD-SISTA
Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
Email:{rocco.langone, raghvendra.mall, johan.suykens}@esat.kuleuven.be

***Abstract*—In this paper we propose an algorithm for soft (or fuzzy) clustering. In soft clustering each point is not assigned to a single cluster (like in hard clustering), but it can belong to every cluster with a different degree of membership. Generally speaking, this property is desirable in order to improve the interpretability of the results. Our starting point is a state-of-the art technique called kernel spectral clustering (KSC). Instead of using the hard assignment method present therein, we suggest a fuzzy assignment based on the cosine distance from the cluster prototypes. We then call the new method soft kernel spectral clustering (SKSC). We also introduce a related model selection technique, called average membership strength criterion, which solves the drawbacks of the previously proposed method (namely balanced linefit). We apply the new algorithm to synthetic and real datasets, for image segmentation and community detection on networks. We show that in many cases SKSC outperforms KSC.**

I. Introduction

Clustering is an exploratory technique used for finding structure in data and consists in organizing objects into meaningful groupings. In the network literature it is also known as community detection and the clusters are usually referred as communities or modules. Many clustering (and community detection) algorithms exist, based on different principles (see [1] and [2] for an exhaustive review on the topic). Most of these methods performs only hard clustering, where each item is assigned to only one group. However, this works fine when the clusters are compact and well separated, while the performance can decrease dramatically when they overlap. Since this is the case in many real-world scenarios, soft or fuzzy clustering is becoming popular in many fields (see [3], [4] and references therein). In soft clustering each object belongs to several groups at the same time, with a different degree of membership. In data clustering it is desirable not only to cope in a more effective way with overlapping clusters, but the uncertainties on data-to-clusters assignments help also to improve the overall interpretability of the results. In community detection, nodes can naturally belong to different communities at the same time, like in a social network where somebody's cousin is also his co-worker, mate in a football team etc. Other possible applications are related to document clustering for web search engines (a document can occur in several topics), analysis of protein interaction networks where several functions can be assigned to a protein, characterization of users in mobile communications (the clusters in phone call mining do not necessarily have crisp boundaries) etc. In this work we introduce a novel algorithm for fuzzy clustering that we call soft kernel spectral clustering (SKSC). This method is based on kernel spectral clustering (KSC) [5], a state-of-the-art clustering technique formulated in the least squares support vector machines (LS-SVMs) framework [6], which is able to perform out-of-sample extension in a model-based manner. Basically, in SKSC we keep the same core model as in KSC, but we change the assignment rule allowing soft cluster memberships. By doing so, we show on toy problems how SKSC outperforms KSC where a certain amount of overlap between clusters is present. This method is different from what it has been proposed in classical spectral clustering [7], [8], where k-means is applied on the points in the eigenspace. The difference is that we do not apply k-means but we only calculate the centroids, and this is done not in the eigenspace but in the projections space (see Sec. II-C and III-A). Also, given the line structure of the clusters in the projections space, we do not use the euclidean distance but the cosine distance, as explained in Sec. III-A. Moreover, we describe a new method to tune the number of clusters and the kernel parameters based on the soft assignment. We name the new model selection method as average membership strength (AMS) criterion. The latter can solve the issues arising with the previously proposed criterion called balanced linefit (BLF). In fact, unlike BLF, AMS is not biased toward two clusters, does not have any parameter to tune and can be used in an effective way also with overlapping clusters. Moreover two alternative model selection criteria for KSC have already been proposed in [9] and [10]. The former is based on Modularity and it is more suited for analyzing network data, while the latter, called Fisher criterion, has been observed to work well only for toy data and in simple image segmentation problems. For these reasons AMS can be a valid alternative to BLF for model selection.

The rest of this paper is organized as follows: Section II summarizes the KSC model and the BLF criterion. Section III introduces the SKSC algorithm and the AMS model selection method. In section IV we describe some experiments carried out on two toy problems, namely three overlapping gaussian clouds and an artificial network with overlapping communities. Section V illustrates and compares the performance of SKSC and KSC on two real-life applications, that is image segmentation and community detection. Finally Section VI presents some conclusions and perspectives for future work.

II. KSC

A. Primal-dual framework

The kernel spectral clustering model is described by a primal-dual formulation. The dual problem is an eigenvalue problem and it is related to classical spectral clustering [11],

978-1-4673-6129-3/13/$31.00 ©2013 IEEE

[12], [7]. So far KSC has been applied in a successful way in image segmentation, analysis of gene expression data, community detection, power load time-series clustering, fault detection of industrial machineries, text mining etc.

Given N_{tr} training data-points (or nodes in case of community detection) $\mathcal{D} = \{x_i\}_{i=1}^{N_{\text{tr}}}, x_i \in \mathbb{R}^N$ and the number of clusters/communities k, the KSC problem in its primal formulation can be expressed as [5]:

$$\min_{w^{(l)},e^{(l)},b_l} \frac{1}{2}\sum_{l=1}^{k-1} w^{(l)^T} w^{(l)} - \frac{1}{2N}\sum_{l=1}^{k-1} \gamma_l e^{(l)^T} D_\Omega^{-1} e^{(l)} \quad (1)$$

$$\text{such that } e^{(l)} = \Phi w^{(l)} + b_l 1_{N_{\text{tr}}} \quad (2)$$

where $e^{(l)} = [e_1^{(l)}, \ldots, e_{N_{\text{tr}}}^{(l)}]^T$ are the projections, $l = 1, \ldots, k-1$ indicates the number of score variables needed to encode the k clusters to find, $D_\Omega^{-1} \in \mathbb{R}^{N_{\text{tr}} \times N_{\text{tr}}}$ is the inverse of the degree matrix associated to the kernel matrix Ω (as explained later), Φ is the $N_{\text{tr}} \times d_h$ feature matrix $\Phi = [\varphi(x_1)^T; \ldots; \varphi(x_{N_{\text{tr}}})^T]$ and $\gamma_l \in \mathbb{R}^+$ are regularization constants. The clustering model is given by:

$$e_i^{(l)} = w^{(l)^T} \varphi(x_i) + b_l, i = 1, \ldots, N_{\text{tr}} \quad (3)$$

where $\varphi : \mathbb{R}^N \to \mathbb{R}^{d_h}$ is the mapping to a high-dimensional feature space, b_l are bias terms, $l = 1, \ldots, k-1$. The projections $e_i^{(l)}$ represent the latent variables of a set of $k-1$ binary clustering indicators given by $\text{sign}(e_i^{(l)})$ which can be combined to form the final groups in an encoding/decoding scheme. The dual problem is:

$$D_\Omega^{-1} M_D \Omega \alpha^{(l)} = \lambda_l \alpha^{(l)} \quad (4)$$

where Ω is the kernel matrix with ij-th entry $\Omega_{ij} = K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$, D_Ω is the diagonal matrix with diagonal entries $d_i^\Omega = \sum_j \Omega_{ij}$, M_D is a centering matrix defined as $M_D = I_{N_{\text{tr}}} - (1/1_{N_{\text{tr}}}^T D_\Omega^{-1} 1_{N_{\text{tr}}})(1_{N_{tr}} 1_{N_{\text{tr}}}^T D_\Omega^{-1})$, the $\alpha^{(l)}$ are dual variables. The kernel function $K : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$ means the similarity function of the data (or the graph in case of community detection).

B. Encoding/decoding scheme

In the ideal case of k well separated clusters and properly chosen kernel parameters, the matrix $D^{-1} M_D \Omega$ has $k-1$ piecewise constant eigenvectors with eigenvalue 1 (see for example [11]). In the eigenvector space every cluster $\mathcal{A}_p$, $p = 1, \ldots, k$ is a point and is represented with a unique codeword $c_p \in \{-1, 1\}^{k-1}$. The codebook $\mathcal{CB} = \{c_p\}_{p=1}^k$ can then be obtained in the training process from the rows of the binarized projected variables matrix for training data $[\text{sign}(e^{(1)}), \ldots, \text{sign}(e^{(k)})]$. Since the first eigenvector $\alpha^{(1)}$ already provides a binary clustering $k-1$ score variables are sufficient to encode k clusters [5]. The decoding scheme consists of comparing the cluster indicators obtained in the validation/test stage with the codebook and selecting the nearest codeword in terms of Hamming distance (ECOC decoding procedure). In this way every point is assigned to only one cluster and thus soft boundaries are not allowed. The cluster indicators can be obtained by binarizing the score variables for out-of-sample points as follows:

$$\text{sign}(e_{\text{test}}^{(l)}) = \text{sign}(\Omega_{\text{test}} \alpha^{(l)} + b_l 1_{\text{Ntest}}) \quad (5)$$

with $l = 1, \ldots, k-1$. Ω_{test} is the $N_{\text{test}} \times N_{\text{tr}}$ kernel matrix evaluated using the test points with entries $\Omega_{\text{test,ri}} = K(x_r^{\text{test}}, x_i)$, $r = 1, \ldots, N_{\text{test}}$, $i = 1, \ldots, N_{\text{tr}}$.

C. BLF

The BLF criterion exploits the shape of the points in the projections space: it reaches its maximum value 1 when the clusters do not overlap, and in this ideal situation the clusters are represented as lines in this space. In particular the BLF is defined as follows [5]:

$$\text{BLF}(\mathcal{D}^V, k) = \eta \text{linefit}(\mathcal{D}^V, k) + (1-\eta)\text{balance}(\mathcal{D}^V, k) \quad (6)$$

where $\mathcal{D}^V$ represents the validation set and k as usual indicates the number of clusters. The linefit index equals 0 when the score variables are distributed spherically and equals 1 when the score variables are collinear (representing points in the same cluster). The balance index equals 1 when the clusters have the same number of points and tends to 0 otherwise. The parameter η controls the importance given to the linefit with respect to the balance index and takes values in the range $[0, 1]$. Then (6) can be used to select the number of clusters and the kernel tuning parameters by using a grid search approach. The main disadvantages of BLF are that it tends to reach always high values for $k = 2$, it is not clear how to select η and it can work poorly when the amount of overlap between clusters is not negligible. The first drawback is due to the fact that the linefit and balance for $k = 2$ are calculated using an ad-hoc dummy variable [5]. For what concerns η, we do not know beforehand if the clusters have or not the same size so we have to explore different possibilities. Finally, when the clusters overlap considerably, the line structure assumption for the points in the projection space does not hold any more and then the (6) criterion is not very suitable.

III. SKSC

The main idea behind soft kernel spectral clustering is to use KSC as an initialization step in order to find a first division of the data into clusters. Then this grouping is refined by re-calculating the prototypes in the score variable space, and the cluster assignments are performed by means of the cosine distance between each point and the prototypes. The final partitions are not only of better quality with respect to KSC results, but they give also better insight of the problem under study. The SKSC algorithm can be summarized as follows:

Algorithm SKSC Soft Kernel Spectral Clustering algorithm

Input: Training set $\mathcal{D} = \{x_i\}_{i=1}^N$ and test set $\mathcal{D}^{\text{test}} = \{x_m^{\text{test}}\}_{m=1}^{N_{\text{test}}}$, kernel function $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^+$ positive definite and localized ($K(x_i, x_j) \to 0$ if x_i and

x_j belong to different clusters), kernel parameters (if any), number of clusters k.

Output: Clusters $\{\mathcal{A}_1, \ldots, \mathcal{A}_p, \ldots, \mathcal{A}_k\}$, soft cluster memberships $cm^{(p)}, p = 1, \ldots, k$, cluster prototypes $\mathcal{SP} = \{s_p\}_{p=1}^{k}$, $s_p \in \mathbb{R}^{k-1}$.

1) Initialization by solving eq. (4).
2) Compute the new prototypes $s_1, \ldots, s_k$ (eq. (7)).
3) Calculate the test data projections $e_m^{(l)}$, $m = 1, \ldots, N_{\text{test}}$, $l = 1, \ldots, k-1$.
4) Find the cosine distance between each projection and all the prototypes (eq. (8))
5) $\forall m$, assign x_m^{test} to cluster A_p with membership $cm^{(p)}$ according to eq. (9).

A. Soft assignment

As already pointed out in Sec. II-C, in the projections/score variables space the points belonging to the same cluster appear aligned in the absence of overlap (see center of Fig. 1). In this ideal situation of clear and well distinct groupings, any soft assignment should reduce to a hard assignment, where every point must belong to one cluster with membership 1 and to the other clusters with membership 0. In fact, the membership reflects the certainty with which we can assign a point to a cluster and it can be thought as a kind of subjective probability. In order to cope with this situation, we can use the cosine distance from every point to the prototypes as the basis for the soft assignment. In this way, in the perfect above mentioned scenario, every point positioned along one line will be assigned to that cluster with membership or probability equal to 1, since the cosine distance from the corresponding prototype is 0, being the two vectors parallel (see right of Fig. 1). Given the projections for the training points $e_i^{(l)}, i = 1, \ldots, N_{tr}$ and the corresponding hard assignments q_i^p (found by applying the encoding scheme of KSC described in Sec. II-B), we can calculate for each cluster the new prototypes $s_1, \ldots, s_p, \ldots, s_k$, $s_p \in \mathbb{R}^{k-1}$ as:

$$s_p = \frac{1}{n_p} \sum_{i=1}^{n_p} \sum_{l=1}^{k-1} e_i^{(l)} \tag{7}$$

where n_p is the number of points assigned to cluster p during the initialization step by KSC. Then we can calculate the cosine distance between the $i-th$ point in the score variables space and a prototype s_p using the following formula:

$$d_{ip}^{\cos} = 1 - e_i^T s_p / (||e_i||_2 ||s_p||_2). \tag{8}$$

with $e_i = [e_i^{(1)}, \ldots, e_i^{(k-1)}]$. The membership of point i to cluster q can be expressed as:

$$cm_i^{(q)} = \frac{\prod_{j \neq q} d_{ij}^{\cos}}{\sum_{p=1}^{k} \prod_{j \neq p} d_{ij}^{\cos}} \tag{9}$$

with $\sum_{p=1}^{k} cm_i^{(p)} = 1$. As discussed in [13], this membership is given as a subjective probability and it indicates the strength of belief in the clustering assignment. Moreover, if on one hand the whole procedure described here resemble the classical spectral clustering algorithm [7], [8], on the other hand there are some clear differences:

- we do not apply k-means in the eigenspace (the $\alpha^{(l)}$ space) but we only calculate the centroids in the projections space (the $e^{(l)}$ space)
- we propose a soft assignment instead of a hard assignment
- we derive a new model selection technique
- we are able to perform out of sample extension on unseen data according to eq. (3).

Finally, it is worth to mention that a first attempt to provide a sort of probabilistic output in kernel spectral clustering was already done in [14]. However, in the cited work the authors did not explore deeply this possibility. Also they consider as prototypes the tips of the lines, but this is meaningful only when there is a clear line structure, i. e. if the clusters do not overlap too much. Moreover, the probabilistic output they propose depends on the euclidean distance (instead of cosine distance) between the prototypes and the points in the projections space. We think that this distance is more a measure of the size of the cluster than the certainity of the cluster assignment.

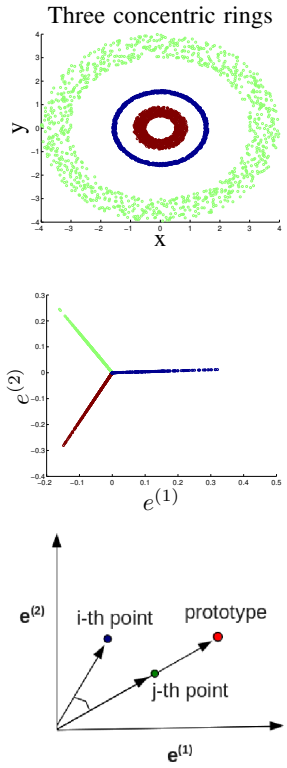


Fig. 1. **Clusters as lines**: example of how three distinct clusters (3 concentric rings on the top) appear in the projection space in the KSC formulation (center). On the bottom an illustration of the cosine distance in the score variable space is given: the $j-th$ point is parallel to the prototype so its membership is 1. This is not the case for the $i-th$ point.

B. AMS

From the soft assignment technique explained in the previous section a new model selection method can be derived. In fact, we can calculate a kind of mean membership per cluster indicating the average degree of belonging of the points to that cluster. If we do the same for every cluster and we take the mean, we have what we call the average membership strength (AMS) criterion:

$$\text{AMS} = \frac{1}{k}\sum_{p=1}^{k}\frac{1}{n_p}\sum_{i=1}^{n_p} cm_i^{(p)} \tag{10}$$

where we use the same notation introduced in Sec. III-A. Unlike BLF, AMS does not have any parameter to tune, it is not biased toward $k = 2$ and as we will show in the experimental section it can be used in an effective way also in case of overlapping clusters.

Algorithm MS AMS model selection algorithm for SKSC

Input: training and validation sets, kernel function $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^+$ positive definite and localized.
Output: selected number of clusters k, kernel parameters (if any)

1) Define a grid of values for the parameters to select
2) Train the related SKSC model using the training set
3) Compute the soft cluster assignments for the points of the validation set
4) For every partition calculate the related AMS score using eq. (10)
5) Choose the model with the highest score.

IV. Toy problems

Here we show the performance of KSC and SKSC on two synthetic experiments, regarding data clustering and community detection. We evaluate the cluster quality in terms of Adjusted Rand Index (ARI), Silhouette and Davies-Bouldin index (DBI) [15]. The former is an external evaluation criterion and is a measure of agreement between two partitions (ARI $= 0$ means complete disagreement and ARI $= 1$ means perfect agreement). In our case we use it to compare the partitions found by KSC and SKSC with the true division. Silouette and DBI criteria are internal criteria and measure how tightly grouped all the data in the clusters are. For the Silhouette the higher the better, for DBI the lower the better. Generally speaking, SKSC achieves better results than KSC. Moreover the AMS criterion appears to be more useful than BLF and Modularity for model selection, mostly when a big amount of overlap between clusters or communities is present.

A. Three Gaussians

For this experiment we generate 1000 data points from a mixture of three 2-D Gaussians, with 500 samples drawn from each component of the mixture. We then consider different amounts of overlap between these clouds of points, respectively very small, small and large (see Fig. 2 from left to right). In Fig. 3 we show the model selection plots for these three cases. We can see that AMS, even in case of large overlap, is able to correctly identify the presence of the three distinct gaussian clouds. This is not the case for the BLF criterion. In tables I, II, III the clustering results of KSC and SKSC (when feeded with correct parameters) are evaluated. We can notice that SKSC outperforms KSC when the overlap between the clusters increases. Finally, in Fig. 4 the soft clustering results produced by SKSC for the large overlap case are depicted.

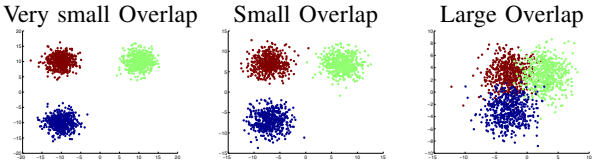


Fig. 2. **Three Gaussians dataset**.

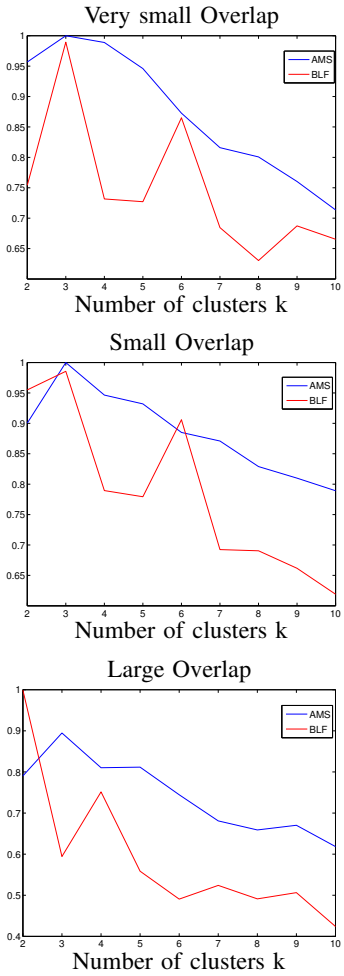


Fig. 3. **Model selection for the three Gaussians dataset**. The true number of clusters is $k = 3$.

	KSC	SKSC
Silhouette	0.96	0.96
DBI	0.24	0.24
ARI	1	1

TABLE I
THREE GAUSSIANS, VERY SMALL OVERLAP: CLUSTER QUALITY EVALUATION OF KSC AND SKSC. THE BEST PERFORMANCE ARE IN BOLD.

	KSC	SKSC
Silhouette	0.90	**0.91**
DBI	0.34	**0.33**
ARI	0.96	**0.99**

TABLE II
THREE GAUSSIANS, SMALL OVERLAP: CLUSTERING RESULTS OF KSC AND SKSC. THE BEST PERFORMANCE ARE IN BOLD.

	KSC	SKSC
Silhouette	0.59	**0.64**
DBI	0.79	**0.76**
ARI	0.65	**0.74**

TABLE III
THREE GAUSSIANS, LARGE OVERLAP: CLUSTER QUALITY MEASURES FOR KSC AND SKSC. THE BEST PERFORMANCE ARE IN BOLD.

B. Artificial network

This dataset consists of a synthetic network with 1 000 nodes and 4 communities with different degree of overlap ranging from small to large. The network has been generated by using the LFR benchmark [16], with mixing parameters equal to 0.05 (small overlap), 0.2 (medium overlap), 0.35 (large overlap). In Fig. 5 the related adjacency matrix is depicted. In Fig. 6 we compare the model selection criteria BLF, AMS and Modularity. BLF works only in case of small overlap, while Modularity and AMS criteria correctly identify 4 communities in the medium overlap case. Moreover, BLF and Modularity fail in case of large overlap, while AMS detects 3 communities, giving then an useful indication of the community structure to the user. Finally, in tables IV, V and VI the partitions found by SKSC and KSC are evaluated according to Modularity [17], Conductance [18], [19] and ARI. SKSC gives better results than KSC in terms of Modularity and ARI, while they perform the same according to Conductance.

	KSC	SKSC
Modularity	0.68	**0.69**
Conductance	0.0040	0.0040
ARI	0.98	**0.99**

TABLE IV
SYNTHETIC NETWORK, SMALL OVERLAP: CLUSTER QUALITY EVALUATION OF KSC AND SKSC. THE BEST PERFORMANCE ARE IN BOLD.

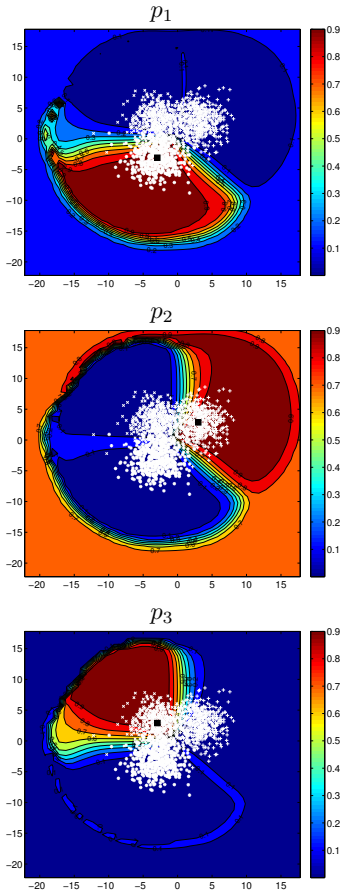


Fig. 4. **Three Gaussian dataset**: probability to belong to cluster 1 (p_1, top), to cluster 2 (p_2, center) and to cluster 3 (p_3, bottom) detected by SKSC for the large overlap case. We can notice that all the probabilities reach their maximum value around the centroids and then decrease in the overlapping region (see Sec. III-A for further details).

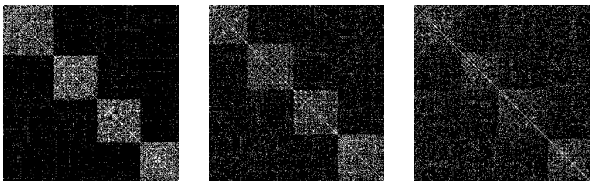


Fig. 5. **Artificial network**. Adjacency matrix of the synthetic network with small overlap (left), medium overlap (center) and large overlap (right).

	KSC	SKSC
Modularity	0.53	**0.54**
Conductance	0.0042	0.0042
ARI	0.96	**0.97**

TABLE V
SYNTHETIC NETWORK, MEDIUM OVERLAP: CLUSTERING RESULTS OF KSC AND SKSC. THE BEST PERFORMANCE ARE IN BOLD.

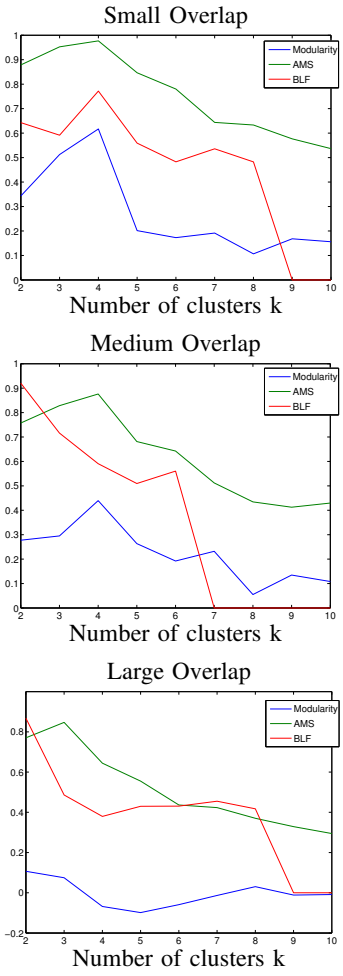


Fig. 6. **Model selection for the synthetic network**: BLF (red), Modularity (blu) and AMS (green) are compared. The true number of communities is $k = 4$.

	KSC	SKSC
Modularity	0.34	**0.36**
Conductance	0.0043	0.0043
ARI	0.65	**0.66**

TABLE VI
SYNTHETIC NETWORK, LARGE OVERLAP: CLUSTER QUALITY MEASURES FOR KSC AND SKSC. THE BEST PERFORMANCE ARE IN BOLD

V. IMAGE SEGMENTATION

For the image segmentation task, we utilized 10 color images from the Berkeley image dataset [20]. We use a subset of 1000 pixels for training and the whole image for testing, in order to obtain the final segmentation. After properly selecting the number of clusters and the kernel bandwidth for each image, we compared the segmentation obtained by KSC and SKSC by using three evaluation measures [21]:

- F-measure ($\frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$) with respect to human ground-truth boundaries
- Variation of information (VI): it measures the distance between two segmentations in terms of their average conditional entropy. Low values indicate good match between the segmentations.
- Probabilistic Rand Index (PRI): it operates by comparing the congruity of assignments between pairs of elements in the clusters found by an algorithm and multiple ground-truth segmentations.

The results are shown in Fig 7 (F-measure) and tables VII (VI) and VIII (PRI). In general, SKSC can improve the performance obtained by KSC. Finally, in Fig. 8 the segmentations accomplished by SKSC on three selected images are depicted for illustration purpose.

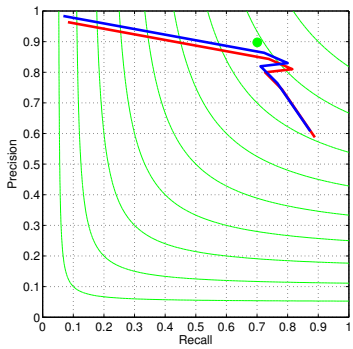


Fig. 7. **F-measure plot**: evaluation of KSC (red) and SKSC (blue) algorithms with respect to human ground-truth boundaries. Iso-F curves are shown in green. The green dot indicates human segmentation.

Image ID	KSC	SKSC
145086	3.113	**3.099**
42049	3.423	**3.410**
167062	1.847	**1.694**
147091	1.106	**1.105**
196073	0.184	**0.183**
62096	0.357	**0.313**
101085	2.772	**2.696**
69015	2.365	**2.361**
119082	0.903	**0.899**
3096	2.978	**2.974**

TABLE VII
VI: COMPARISON OF KSC AND SKSC SEGMENTATIONS IN TERMS OF VARIATION OF INFORMATION (THE LOWER THE BETTER). BEST PERFORMANCE IN BOLD.

VI. COMMUNITY DETECTION

In this last section we analyze the outcomes of SKSC and KSC in the community detection problem. We use three well-known real-world networks:

- **Karate**: the Zachary's karate club network [22] consists of 34 member nodes, and splits in two smaller clubs after a dispute arose during the course of Zachary's study between the administrator and the instructor.
- **Football**: this network [23] describes American college football games and is formed by 115 nodes (the teams)

Image ID	KSC	SKSC
145086	0.591	**0.596**
42049	0.705	**0.707**
167062	**0.836**	0.769
147091	0.838	**0.839**
196073	0.983	0.983
62096	**0.954**	0.938
101085	0.375	**0.381**
69015	0.578	**0.580**
119082	0.876	**0.877**
3096	0.198	**0.199**

TABLE VIII
PRI: COMPARISON OF KSC AND SKSC SEGMENTATIONS IN TERMS OF PROBABILISTIC RAND INDEX (THE HIGHER THE BETTER). BEST PERFORMANCE IN BOLD.

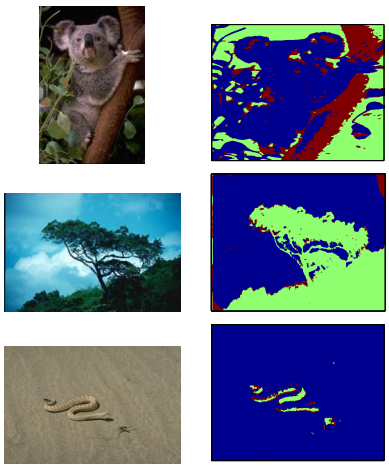


Fig. 8. **SKSC segmentations**: original images (left) and SKSC segmentations (right) for image IDs 69015, 147091, 196073.

and 616 edges (the games). It can be divided into 12 communities according to athletic conferences.

- **Dolphins**: the dataset, described in [24], depicts the associations between 62 dolphins with ties between dolphin pairs representing statistically significant frequent associations. As pointed out in [24], this network can be naturally split in two main groups (female-female and male-male associations), even if another cluster of mixed sex associations could also be considered.

The model selection plots for AMS, BLF and Modularity are shown in Fig. 9. In the Karate network case all methods detect the presence of the 2 communities. For what concerns the Football dataset BLF fails, while Modularity and AMS select 10 communities (AMS suggests also 3). Finally in the Dolphins network AMS suggests the possible presence of 3 communities, Modularity 6 and BLF 2. In tables IX, X and XI the partitions found by KSC and SKSC are evaluated according to Conductance, Modularity and ARI. The performance of the two algorithms is nearly the same, apart from the Dolphins network where KSC seems better than SKSC. This is probably due to the fact that these networks show a clear division into groups with few overlap between communities. Finally, if we analyze the memberships obtained by SKSC we can draw the following considerations:

- Karate network: node 31 has the most certain membership, being assigned with probability 1 to its community. This is reasonable since it is linked to the highest degree nodes present in the same module
- Football network: node 25 and 59 are assigned to every cluster with nearly the same probability (i.e. they are overlapping nodes)
- Dolphins network: the membership of node 39 is equally shared between the 2 communities of the network.

However, these insights need to be further studied in order to better interpret the results.

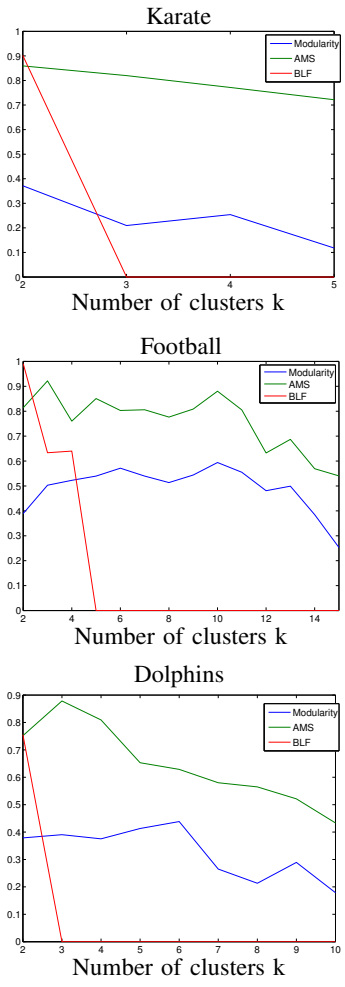


Fig. 9. **Model selection for the real-world networks**: BLF (red), Modularity (blu) and AMS (green) are compared. The true number of communities is $k = 2$ for Karate, $k = 12$ for Football and $k = 2$ for Dolphins.

VII. CONCLUSIONS

In this work we presented a soft clustering method called soft kernel spectral clustering (SKSC). It is based on kernel spectral clustering (KSC), but a fuzzy assignment rule is used.

	KSC	SKSC
Modularity	0.371	0.371
Conductance	0.083	0.083
ARI	1	1

TABLE IX
KARATE NETWORK: CLUSTER QUALITY MEASURES FOR KSC AND SKSC.

	KSC	SKSC
Modularity	0.594	**0.597**
Conductance	0.772	**0.766**
ARI	0.818	0.818

TABLE X
FOOTBALL NETWORK: CLUSTER QUALITY MEASURES FOR KSC AND SKSC.

The latter depends on the cosine distance from the cluster prototypes in the projections space. A new model selection technique derived from this soft assignment is also proposed. We call it average membership strength criterion (AMS). We show how AMS can solve the drawbacks of the previously proposed method called balanced linefit (BLF). Moreover, we illustrate on toy data and real problems (in particular community detection and image segmentation) that SKSC outperforms KSC, mainly in the most difficult tasks (that is when clusters overlap to a large extent).

ACKNOWLEDGEMENTS

Research supported by Research Council KUL: GOA/10/09 MaNet, PFV/10/002 (OPTEC), several PhD/postdoc and fellow grants; Flemish Government IOF: IOF/KP/SCORES4CHEM, FWO: PhD/postdoc grants, projects: G.0588.09 (Brain-machine), G.0377.09 (Mechatronics MPC), G.0377.12 (Structured systems), IWT: PhD Grants, projects: SBO LeCoPro, SBO Climaqs, SBO POM, EUROSTARS SMART iMinds 2013, Belgian Federal Science Policy Office: IUAP P7/19 (DYSCO, Dynamical systems, control and optimization, 2012-2017), EU: FP7-EMBOCON (ICT-248940), FP7-SADCO (MC ITN-264735), ERC ST HIGHWIND (259 166), ERC AdG A-DATADRIVE-B, COST: Action ICO806: IntelliCIS. Johan Suykens is a professor at the Katholieke Universiteit Leuven, Belgium.

REFERENCES

[1] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern Recogn. Lett.*, vol. 31, no. 8, Jun. 2010.

[2] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

[3] H. Hsin-Chien, C. Yung-Yu, and C. Chu-Song, "Multiple kernel fuzzy clustering," *Fuzzy Systems, IEEE Transactions on*, vol. 20, no. 1, pp. 120–134, Feb. 2012.

[4] K. Yu, S. Yu, and V. Tresp, "Soft clustering on graphs," in *in Advances in Neural Information Processing Systems (NIPS)*, 2005, p. 05.

[5] C. Alzate and J. A. K. Suykens, "Multiway spectral clustering with out-of-sample extensions through weighted kernel PCA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 335–347, February 2010.

	KSC	SKSC
Modularity	**0.378**	0.370
Conductance	**0.146**	0.148
ARI	**0.935**	0.872

TABLE XI
DOLPHINS NETWORK: CLUSTER QUALITY MEASURES FOR KSC AND SKSC.

[6] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.

[7] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA: MIT Press, 2002, pp. 849–856.

[8] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, no. 8, pp. 888–905, 2000.

[9] R. Langone, C. Alzate, and J. A. K. Suykens, "Modularity-based model selection for kernel spectral clustering," in *Proc. of the International Joint Conference on Neural Networks (IJCNN 2011)*, 2011, pp. 1849–1856.

[10] C. Alzate and J. A. K. Suykens, "Out-of-sample eigenvectors in kernel spectral clustering," in *Proc. of the International Joint Conference on Neural Networks (IJCNN 2011)*, 2011, pp. 2349–2356.

[11] F. R. K. Chung, *Spectral Graph Theory*. American Mathematical Society, 1997.

[12] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[13] B.-I. Adi and I. Cem, "Probabilistic d-clustering," *J. Classification*, vol. 25, no. 1, pp. 5–26, 2008.

[14] C. Alzate and J. A. K. Suykens, "Highly sparse kernel spectral clustering with predictive out-of-sample extensions," in *Proc. of the 18th European Symposium on Artificial Neural Networks (ESANN 2010)*, 2010, pp. 235–240.

[15] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *Journal of Intelligent Information Systems*, vol. 17, pp. 107–145, 2001.

[16] A. Lancichinetti and S. Fortunato, "Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities," *Phys. Rev. E*, vol. 80, no. 1, p. 016118, Jul 2009.

[17] M. E. J. Newman, "Modularity and community structure in networks," *Proc. Natl. Acad. Sci. USA*, vol. 103, no. 23, pp. 8577–8582, 2006.

[18] R. Kannan, S. Vempala, and A. Vetta, "On clusterings: Good, bad and spectral," *J. ACM*, vol. 51, no. 3, pp. 497–515, May 2004.

[19] J. Leskovec, K. J. Lang, and M. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proceedings of the 19th international conference on World wide web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 631–640.

[20] D. R. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," EECS Department, University of California, Berkeley, Tech. Rep. UCB/CSD-01-1133, Jan 2001.

[21] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.

[22] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. 33, pp. 452–473, 1977.

[23] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, no. 12, pp. 7821–7826, 2002.

[24] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, "The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations," *Behavioral Ecology and Sociobiology*, vol. 54, no. 4, pp. 396–405, 2003.