Laplacian Normalization for Deriving Thematic Fuzzy Clusters with an Additive Spectral Approach

Susana Nascimento[1], Rui Felizardo[1] and Boris Mirkin[2,3]
[1] Department of Computer Science and Centre for Artificial Intelligence (CENTRIA),
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa,
Caparica, Portugal
[2] Department of Computer Science, Birkbeck University of London,
London, UK
[3] School of Applied Mathematics and Informatics, Higher School of Economics,
Moscow, RF

November 24, 2012

Abstract

This paper presents a further investigation into computational properties of a novel fuzzy additive spectral clustering method, FADDIS, recently introduced by authors (Mirkin and Nascimento 2012). Specifically, we extend our analysis to "difficult" data structures from the recent literature and develop two synthetic data generators simulating affinity data of Gaussian clusters and genuine additive similarity data, with a controlled level of noise. The FADDIS is experimentally verified on these data in comparison with two state-of-the art fuzzy clustering methods. The claimed ability of FADDIS to help in determining the right number of clusters is experimentally tested and the role of the pseudo-inverse Laplacian data transformation in this is highlighted. A potentially useful extension of the method to biclustering is introduced.

1 Introduction

Unsupervised clustering provides exploratory techniques for finding hidden patterns in data. With the huge volumes of data generated from different systems nowadays, a crucial contribution to make a system intelligent is its ability to analyse the data for efficient decision-making based on new cluster discovery. Fuzzy clustering has been successfully applied in the construction of intelligent systems (Zimmermann, 2001; Drobics *et al.*, 2002; Casillas & Martínez López, 2010; Meyer & Zimmermann, 2011). Recently, relational data have become popular in several important application areas such as bioinformatics (Popescu *et al.*, 2006; Pal *et al.*, 2007; Xu *et al.*, 2008; Masullia & Mitra, 2009), recommender systems (Suryavanshi *et al.*, 2005; Abbassi & Mirrokni, 2009; Nanopoulos *et al.*, 2009), Web mining and text analysis (Krishnapuram *et al.*, 2001; Nasraoui *et al.*, 2002; Runkler & Bezdek, 2003; Castellano & Torsello, 2009).

Our motivation comes from our interest in knowledge analysis and engineering. Specifically, we use a hierarchical taxonomy, the core of an ontology of the domain, to map, generalize and interpret there thematic clusters derived from empirical data about the activities conducted on a organization under consideration. The prime objects here are activity topics of the taxonomy rather than the individual members or teams in the organization, and the information is organized as an index of similarity between the activity topics rather than the members. In such a setting, it seems rather natural to assume an additive action of the hidden research patterns as the underlying mechanism for the generation of the similarity index. This has led us to develop a relational fuzzy clustering method, the Fuzzy Additive Spectral Clustering (FADDIS), by combining a model-based approach of additive clustering and the spectral clustering approach (Mirkin & Nascimento, 2012).

In spite of the fact that many relational fuzzy clustering algorithms have been developed already (Roubens, 1978; Windham, 1985; Hathaway *et al.*, 1989; Hathaway & Bezdek, 1994; Bezdek *et al.*, 1999; Inoue & Urahama, 1999; Yang & Shih, 2001; Davé & Sen, 2002; Brouwer, 2009), they all involve manually specified parameters such as the number of clusters or threshold of similarity without providing any guidance for choosing them, which is a weakness to develop decision-support or expert systems. Indeed, the determination of the number of clusters in data is a fundamental problem in cluster analysis (see (Mirkin, 2011a) for a state of the art perspective). Concerning the FADDIS method, it does provide guidance for choosing the number of clusters according to its stop conditions. Moreover, it appears to be quite competitive in comparison to the state of the art relational fuzzy clustering algorithms.

The main goal of this paper is to experimentally compare the FADDIS algorithm with two state-of-the-art fuzzy clustering algorithms differently extending Fuzzy c-Means to relational data. One of these fuzzy clustering algorithms combines Fuzzy c-Means with a recently proposed fast-mapping technique proved superior to many other techniques, the Fast Map Fuzzy c-Means (FMFCM) (Brouwer, 2009), and the other is an extension of the c-means to dissimilarity data, the Non-Euclidean Relational Fuzzy c-Means (NERFCM) (Hathaway & Bezdek, 1994). Another subject concerns the study of FADDIS capability in deriving the number of clusters. We give special attention to the experimental analysis on the usage of the Laplacian data transformation, which is in the core of the spectral clustering approach (Shi & Malik, 2000; Ng *et al.*, 2002; Zelnik-Manor & Perona, 2004; Nadler *et al.*, 2006; von Luxburg, 2007; Huang *et al.*, 2009), to sharpen the cluster structure in the data, and its role on determining the number of clusters for FADDIS partitions. Also, we study FADDIS ability in recovering the cluster structure of genuine similarity data generated according to the FADDIS model. Finally, we present its extension to biclustering.

To be comprehensive in the experimentation, beyond considering a number of benchmark datasets from the literature, we developed two different cluster structure generators, each involving a controlled extent of noise. The first of them generates Gaussian entity-to-feature clusters with a different extent of intermix. The second produces genuine similarity data according to the additive fuzzy clustering model.

The rest of the paper is organized as follows. Section 2 describes the additive model and the FADDIS method. Section 3 describes the experiment and its results over entity-to-feature datasets. These data represent three different lines of research: those from real-world data repositories, artificial datasets with "difficult" geometric structures from the literature, and synthetic datasets representing Gaussian-generated clusters. Section 4 describes the experimental results over genuine similarity datasets. **Section 5 illustrates application of FADDIS for finding thematic clusters of research activities and representing them in a hierarchic taxonomy of the field. An extension of FADDIS to biclustering is proposed to highlight the relation between research teams and research topics.** Section 6 concludes the paper.

2 Additive Fuzzy Clustering Model and Spectral FADDIS Algorithm

The similarity, or relational, data is a matrix $W = (w_{ii'}), i, i' \in I$, of similarity indexes $w_{ii'}$, between objects i, i' from a set of objects I. Specifically, the elements of I can be leaves of a taxonomy tree such as a related hierarchical taxonomy such as Classification of Computer Subjects by ACM (ACM-CCS, 1998). Then individual projects or members of a research organization can be represented with fuzzy membership profiles over the subjects (leaves) of the taxonomy. Given a project/individual-to-subject profile matrix F, the similarity matrix can be defined as $W = F^T F$ so that $w_{ii'}$ is the inner product of subject columns i and i'. These subject-to-subject similarity values are assumed to be manifested expressions of some hidden patterns represented by fuzzy clusters.

To develop an additive model, we formalize a relational fuzzy cluster as represented by: (i) a membership vector $\mathbf{u} = (u_i)$, $i \in I$, such that $0 \leq u_i \leq 1$ for all $i \in I$, and (ii) an intensity $\mu > 0$ **that scales the membership values towards the impact of the cluster to the similarity values. This way, it is the product $\mu\mathbf{u}$ that expresses the hidden similarity pattern rather than its individual co-factors**. Given a value of the product μu_i, to separate μ and u_i, a conventional scheme applies: the scale of the membership vector $\mathbf{u}$ is constrained on a constant level by a condition such as $\sum_i u_i = 1$ or $\sum_i u_i^2 = 1$; then the remaining factor defines the value of μ. As will be seen from formula (5), the latter normalization suits our fuzzy clustering model well and thus is accepted further on. Also, to allow for a possible pre-processing transformation of the given similarity matrix W, we denote the matrix involved in the process of clustering as $A = (a_{ii'})$.

The additive fuzzy clustering model in (1) follows that of (Shepard & Arabie, 1979; Mirkin, 1987; Sato *et al.*, 1997) and involves K fuzzy clusters that reproduce the input similarities $a_{ii'}$ up to additive errors:

$$a_{ii'} = \sum_{k=1}^{K} \mu_k^2 u_{ki} u_{ki'} + e_{ii'}, \qquad (1)$$

where $\mathbf{u}_k = (u_{ki})$ is the membership vector of cluster k, μ_k its intensity $(k = 1, 2, ..., K)$, and $e_{ii'}$ is the residual similarity not explained by the model.

Each fuzzy cluster k is the fuzzy subset grouping the entities that share a common property. The item $\mu_k^2 u_{ki} u_{ki'}$ in (1) is the product of $\mu_k u_{ki}$ and $\mu_k u_{ki'}$ expressing k-th cluster's impact to similarity between i and i'. This value adds up to the others to form the similarity $a_{ii'}$ between entities i and i'. The value μ_k^2 summarizes the contribution of the intensity and will be referred to as the cluster's weight.

To fit the model in (1), the least-squares approach is applied, thus minimizing the sum of all $e_{ii'}^2$. Within that, the one-by-one principal component analysis strategy is attended for finding one cluster at a time by minimizing the corresponding one-cluster criterion

$$E = \sum_{i,i' \in I} (b_{ii'} - \xi u_i u_{i'})^2 \tag{2}$$

with respect to the unknown positive ξ weight and fuzzy membership vector $\mathbf{u} = (u_i)$, given similarity matrix $B = (b_{ii'})$.

Initially matrix B is taken to be equal to matrix A. Each found cluster $(\mu,\ \mathbf{u})$ is subtracted from B, so that the residual similarity matrix applied for obtaining the next cluster is defined as $B - \mu^2 \mathbf{u}\mathbf{u}'$. In this way, A indeed is additively decomposed according to formula (1) and the number of clusters K can be determined in the process.

The optimal value of ξ at a given $\mathbf{u}$ is proven to be

$$\xi = \frac{\mathbf{u}' B \mathbf{u}}{(\mathbf{u}'\mathbf{u})^2}, \tag{3}$$

which is obviously non-negative if B is positive semidefinite.

By putting this ξ in equation (2), one arrives at $E = S(B) - \xi^2 \left(\mathbf{u}'\mathbf{u}\right)^2$, where $S(B) = \sum_{i,i' \in I} b_{ii'}^2$ is the similarity data scatter.

By denoting the last item as

$$G(\mathbf{u}) = \xi^2 \left(\mathbf{u}'\mathbf{u}\right)^2 = \left(\frac{\mathbf{u}' B \mathbf{u}}{\mathbf{u}'\mathbf{u}}\right)^2, \tag{4}$$

the similarity data scatter is decomposed as $S(B) = G(\mathbf{u}) + E$ where $G(\mathbf{u})$ is the part of the data scatter that is explained by cluster $(\mu, \mathbf{u})$, and E, the unexplained part. Therefore, an optimal cluster is to maximize the explained part $G(\mathbf{u})$ in (4) or its square root

$$g(\mathbf{u}) = \xi \mathbf{u}'\mathbf{u} = \frac{\mathbf{u}' B \mathbf{u}}{\mathbf{u}'\mathbf{u}}, \tag{5}$$

which is the Rayleigh quotient: its maximum value is the maximum eigenvalue of matrix B, which is reached at its corresponding eigenvector, in the unconstrained problem.

This shows that the spectral clustering approach can be applied to find a suboptimal maximizer of (5). According to this approach, one should find the maximum eigenvalue λ and corresponding normed eigenvector z for B, $[\lambda, z] = \Lambda(B)$, and take its projection to the set of admissible fuzzy membership vectors. **The normalization condition $\sum_i u_i^2 = 1$ leads to the spectral solution of criterion (5) and simplifies it.**

A number of criteria for halting the process of sequential extraction of fuzzy clusters follow from the above. The process stops if either of the conditions is true:

S1 The optimal value of ξ (3) for the spectral fuzzy cluster becomes negative (meaning that the residual similarity matrix becomes negative definite).

S2 The contribution of a single extracted cluster to the data scatter, $G(\mathbf{u})$, becomes less than a pre-specified $\tau > 0$ threshold.

S3 The residual data scatter becomes smaller than a pre-specified $\epsilon > 0$ proportion of the original similarity data scatter.

S4 A pre-specified number K_{max} of clusters is reached - in some real world problems that information can be set.

The described one-by-one Fuzzy ADDItive-Spectral cluster extraction method is referred to as FADDIS. It combines additive clustering (Shepard & Arabie, 1979; Mirkin, 1987; Sato *et al.*, 1997), spectral clustering (Shi & Malik, 2000; Ng *et al.*, 2002; von Luxburg, 2007; Zhang *et al.*, 2007), and relational fuzzy clustering (Hathaway *et al.*, 1989; Bezdek *et al.*, 1999; Davé & Sen, 2002; Brouwer, 2009). Since FADDIS extracts clusters one-by-one, in the order of their contribution to the data scatter, the algorithm is supposed to be oriented at cluster structures at which the clusters contribute differently: **the bigger the differences, the better. In fact, the higher the differences of the clusters contributions, the more contrastful are the fuzzy memberships of the entities, which manifests a more clear-cut cluster structure.** We refer to this supposed property of the data as the property of *'different contributions'*.

To make the cluster structure in the similarity matrix sharper, one may apply the spectral clustering approach to pre-process a raw similarity matrix W into A by using the so-called normalized Laplacian transformation which is related to the popular clustering criterion of normalized cut (Shi & Malik, 2000; von Luxburg, 2007). The normalized cut criterion can be expressed as the minimum non-zero eigenvalue of the Laplacian matrix. To change this to the criterion of maximum eigenvalue in (5), we further transform this matrix by using the LAplacian PseudoINverse transformation (Mirkin & Nascimento, 2012), Lapin for short, defined by :

$$L_n^+(W) = \tilde{Z}\tilde{\Lambda}^{-1}\tilde{Z}'$$

where $\tilde{\Lambda}$ and $\tilde{Z}$ are defined by the spectral decomposition $L_n = Z\Lambda Z'$ of the normalized Laplacian matrix $L_n = D^{-1/2}(D - W)D^{-1/2}$, **with D the diagonal degree matrix of the similarity adjacency matrix W.** To specify these matrices, first, set T' of indices of elements corresponding to non-zero elements of Λ is determined, after which the matrices are taken as $\tilde{\Lambda} = \Lambda(T', T')$ and $\tilde{Z} = Z(:, T')$. The choice of the Lapin transformation is explained by the fact that it leaves the eigenvectors of L_n unchanged while inverting the non-zero eigenvalues $\lambda \neq 0$ to those $1/\lambda$ of L_n^+. The maximum eigenvalue of L_n^+ is the inverse of the minimum non-zero eigenvalue λ_1 of L_n, corresponding to the same eigenvector.

The effect of the Lapin transformation is to increase the gaps between eigenvalues. Such an increase is experimentally verified further on in Section 3 as useful for deriving an appropriate stop-condition for the cluster-extracting process.

3 Testing FADDIS on Relational Data Derived from the Entity-to-Feature Data

3.1 General remarks

In this section, FADDIS is compared to two most effective methods for fuzzy clustering that are extensions of the popular c-means fuzzy clustering method to relational data:

NERFCM (Hathaway & Bezdek, 1994) and FMFCM (Brouwer, 2009). The NERFCM has been derived as an analogue to the classical c-means at the situation in which the Euclidean distance dissimilarity data is derived from the original entity-to-feature data. The FMFCM also starts from the distance data to produce a number of approximating features after which the fuzzy c-means itself applies to the extracted entity-to-feature data. FADDIS applies to the affinity similarity data derived from entity-to-feature data by using the Gaussian kernel defined as $w_{ij} = exp(-d^2(y_i, y_j)/2\sigma^2)$, where d is the Euclidean distance. **The scaling parameter σ controls the rapidity of decay of the similarity between data points and has to guarantee that the affinity similarity data is between $(0, 1]$. For the artificial and real-world data sets, we set $2\sigma^2$ as the local scaling approach proposed in (Zelnik-Manor & Perona, 2004): the product of the distances of points y_i, y_j to the corresponding K nearest neighbours (in the experiments we used $K = 7$). For the Gaussian cluster generated data we set $2\sigma^2 = 0.5$.** The diagonal elements of W are set to be equal to 0: $w_{ii'} = 0$ (Shi & Malik, 2000; Ng *et al.*, 2002). Then the Laplace Pseudo-Inverse applies to transform the affinity similarity matrix W into the matrix A to which FADDIS algorithm can be applied. The original entity-to-feature data is first normalized by shifting the origin of each feature to its mean and rescaling by the feature range.

In the following experiments the FADDIS algorithm (with or without Lapin) always stopped at condition S2. The threshold parameter τ was empirically fixed in the interval $[0.1, 0.3]$.

3.2 Several geometrically difficult datasets

Fig. 1 presents seven artificial bivariate datasets of "complex" cluster structure. The former three contain a number of rectangular-shaped objects whereas the next four contain a number of round-shaped objects. These types of data sets are considered in the literature as difficult data sets to find their cluster structures (see, for example, (Zelnik-Manor & Perona, 2004)). Each data set is sequentially labelled as D1/..., D2/..., etc, with the number after the slash representing the number of indicative clusters.

Fig. 2 and Fig. 3 present the results of application of FADDIS in two options, to the original affinity data (no-Lapin) and to the Lapin-transformed results, as well as the results for FMFCM and NERFCM algorithms. The bottom of each subfigure shows: the number of clusters found in the case of FADDIS versions, or setting as input in the case of FMFCM and NERFCM algorithms, and the Adjusted Rand Index (ARI) (Hubert & Arabie, 1985), to score the similarity between ground truth and computed partitions.

The number of clusters found by FADDIS is to be analysed based on the relative cluster contribution to the data scatter (stop condition $S2$).

Table 1 presents the differences of cluster contributions of the first seven FADDIS clusters for the original affinity data (on top) and the Lapin transformed similarities (on bottom). As one can see, for the Lapin-based contributions, the number of clusters (marked with (*)) is fixed according to the rule "*select the best number of clusters K^* for which at least the next two consecutive differences of cluster contributions are less than* 0.01". The correct number of clusters is found for data sets D3/4 to D7/2. Data sets D1/3 and D2/5 are the exceptions for which the number of found clusters is 2 and 4, respectively. These results are not that surprising since the background data points should not be counted as a cluster but instead as noise. Indeed, the Lapin based differences on cluster contributions supply a correct information on the number of clusters even if the clustering results on them are disastrous (as is the case of the result in Fig. 2-(j)– data D3/4).

In the case of no-Lapin FADDIS the differences of cluster contribution values provide no rule for stopping the computations, as is seen in Table 1. **This way, we had fine-tuned the threshold parameter τ individually in order to stop at condition S2 having the number of extracted clusters equal to the original ones**. In this case, the no-Lapin based FADDIS finds the correct cluster structure for all data sets except D4/3 and D6/3.

Overall, the clustering results presented in Fig. 2 and Fig. 3 show that at least one of the FADDIS options succeeds in finding the correct cluster structure for each data set. However, the structures populated with more than two rectangular objects on Fig. 1 are difficult for Lapin transformation: they are basically destroyed by it, as is seen from the second column of Fig. 2 in positions (b), (f), (j).

The FMFCM and NERFCM algorithms fail on finding the cluster structures for every data sets, given as input the correct number of clusters. This is not surprising since FCM-based algorithms experience difficulties when the clusters are not hyper-spherical.

3.3 Three datasets from UCI Machine Learning data repository

Fig. 4 presents two-dimensional projections to the plane of the first two singular vectors for three popular datasets from UCI Machine Learning data repository (Frank & Asuncion, 2010): Iris, Leukemia, and Wisconsin Breast Cancer. These datasets are pre-assigned with 3, 3, and 2 class labels, respectively. Yet the data structure suggests the numbers of clusters (rather than classes) in them are in fact, 2, 3, and 2, respectively.

The results of applying the FADDIS algorithm (no-Lapin/Lapin options) as well as the FMFCM and NERFCM are summarized in Table 2. The FADDIS differences of cluster contributions are also shown in Table 1. Applying the same rule as for the previous data collection, the Lapin-based FADDIS establishes the number of clusters for Iris ($D9/2$) as 2, 3 for Leukemia ($D10/3$), and 3 for Wisconsin Breast Cancer ($D11/2$).

For the Iris data set the two options of FADDIS algorithm find two clusters with a perfect partition (ARI=1), while FMFCM and NERFCM found worst partitions. In the case of Leukemia and Breast Cancer, FMFCM and NERFCM provide slightly better results than any of the FADDIS versions. However, the Lapin FADDIS has the advantage that it automatically determines the number of clusters, which is correct for Leukemia indeed. Despite the fact that Lapin FADDIS finds a three partition for Breast Cancer, if we force $K = 2$, it finds a similar partition as in the case of NERFCM and FMFCM.

3.4 Gaussian cluster generated datasets

This study has been conducted with generated data by extending the data generator used in (Brouwer, 2009). Specifically, 4 clusters of data points are generated from a bivariate spherical Gaussian distribution with standard deviation $\sigma = 950$. The centers of the clusters are defined as $c_1 = (1500, 1500)$, $c_2 = (-1500, 1500)$, $c_3 = (-1500, -1500)$, $c_4 = (1500, -1500)$, so that they are located on bisectors of the quadrants of the Cartesian plane at the same distance from the origin. The clusters have cardinalities of $50, 100, 200, 150$

data points, respectively, 500 entities altogether. In our extension, a scale parameter sn is introduced as a factor to the center of the cluster to be added to all data points, to model stretching the data points to or out of the origin. At $sn < 0$, the clusters stretch in to the origin, whereas they move out from the origin at $sn > 0$. Fig. 5 illustrates the type of generated data for different values of the scale parameter sn. A data set generated at $sn = 0$ on the left, and a stretched out dataset generated at $sn = 1$ on the right. The Lapin normalization does sharpen the cluster structure in this type of data, and so has been used in this experiment.

Ten data sets have been generated for each of the values of the scale parameter sn. The three algorithms have been run and the results have been evaluated according to the ARI index to score the similarity between generated and computed clusterings. Also, we tested the ability of Lapin-based FADDIS to recover the number of clusters. Since in FMFCM and NERFCM the number of clusters K must be prespecified, these algorithms have been run with $K = 3, 4, 5$, after which the results have been evaluated by the extended Xie-Beni validation index (Sledge *et al.*, 2010).

Table 3 shows the means and standard deviations of the ARI index for the 10 data sets generated at each level of the scale parameter. In each row the highest ARI value is marked in boldface and (*). For the FADDIS algorithm the mode of the number of clusters retrieved by the algorithm is also presented.

The results show that Lapin-based FADDIS algorithm always recovers the correct number of clusters with stop condition S2. Also, FADDIS finds the best ARI values for the data sets generated with the higher levels of cluster intermix ($sn \leq 0$). In these cases the NERFCM and FMFCM found their best partitions for a wrong number of clusters ($K = 3$). The NERFCM and FMFCM slightly outperform FADDIS for lower levels of cluster intermix ($sn > 0$), and show no differences for very clear-cut cluster structures (e.g. $sn \geq= 30$)[1]. Yet, one should notice that the number of clusters is an input to the former algorithms.

4 Testing FADDIS with Genuine Similarity Data

The main goal of this experiment is to study the ability of the FADDIS algorithm on recovering a cluster structure from genuine similarity data generated following the additive model (1) perturbed with different levels of Gaussian noise. In particular, we want to test the supposed property of FADDIS of 'different cluster contributions'. Therefore, we developed a similarity data generator following the additive model. We apply the same three algorithms to a pool of generated data.

4.1 The Fuzzy Core Cluster Data Generator

As usual in fuzzy clustering, we assume that each entity has one "core" cluster to which it belongs most. Therefore, the data generation process starts with the generation of the "core" clusters.

[1] The values of the extended Xie-Beni index are concordant with the ARI values for both NERFCM and FMFCM.

Given the size N of an entity set I, and the number of clusters K, the proposed Fuzzy Core Cluster Data Generator (FCC DG), generates an $N \times N$ similarity data matrix G according to the underlying (FADDIS) model $W = U\Lambda U^T$, as follows:

$$G = U\Lambda U^T + \alpha E, \tag{6}$$

where:

- $N \times K$ fuzzy membership matrix U is randomly generated using a fuzzy "core" clusters generating procedure.

- Positive real valued $K \times K$ diagonal weight matrix Λ with diagonal positive values λ_k of the cluster weights equal to $\lambda_k = \mu_k^2$ is defined according to model (1). Since the vectors $\mathbf{u}_k$ in (1) are assumed normed, the weights take in the norms of the generated vectors $\mathbf{u}_k$. To test the supposed property of different cluster contributions of the FADDIS, the weights are also made proportional to $(K-k+1)^\beta$, for $k = 1, 2, \ldots, K$, so that the greater the $\beta > 0$, the greater the difference. Therefore, the weights are defined by $\lambda_k = (K-k+1)^\beta * \|\mathbf{u}_k\|$.

- Elements of the $N \times N$ error matrix E are independently generated from a Gaussian distribution $N(0,1)$, and then symmetrized so that $e_{ii'} = (e_{ii'} + e_{i'i})/2$.

- The value $\alpha \in [0,1]$ is the parameter that controls the level of error introduced into the model $W = U\Lambda U^T$.

This generator builds a fuzzy cluster structure by conventionally relaxing a crisp partition. Given a crisp partition R of the entity set I with non-overlapping clusters R_k $(k = 1, \ldots, K)$, a fuzzy relaxation builds each k-th fuzzy cluster $\mathbf{u}_k$ having the corresponding crisp cluster R_k as its core in such a way that the maximum membership values u_{ik} will be at entities $i \in R_k (k = 1, \ldots, K)$ while the other components of $\mathbf{u}_k$ are close to 0. The cores are generated of different random sizes. The first core is taken to be $N/2$ or less, the size N_k of the next $K-2$ cores are at most half-size of the remaining part of the entity set, and the last core's size is taken to complement the cumulative core size to N.

Fixed the number K of core clusters covering the entire data set, I, the data generator builds each core cluster by filling in with fuzzy membership values, such that: (a) the membership values of k-th fuzzy cluster $\mathbf{u}_k$ are very high at k-th core (e.g. $u_{ik} > 2/3$ for $i \in R_k$); and (b) the fuzzy clusters form a fuzzy partition so that $\sum_k u_{ik} = 1$ at each entity $i \in I$. Each $N_k \times K$ core membership matrix U_k is defined independently of each other and then arranged into the final $N \times K$ membership matrix U, as follows: We start filling in the k-th column of U_k, as a N_k-dimensional vector $\mathbf{a} = (a_1, \cdots, a_{N_k})$ of uniformly random values a_i such that $a_i \in [2/3, 1]$. Then, to satisfy the probability constraint, we fill in each entity $i\,(i = 1, \cdots, N_k)$ in U_k with random numbers $u_{ik'} (k' \neq k)$, summing up to $1 - a_i$. For $K \geq 3$ we generate $K-2$ uniformly random values, each one less than $1 - a_i$. Then we sort them in the ascending order $r_1 < \cdots < r_{K-2}$ and set $p_1 = r_1$, $p_{K-1} = 1 - a_i - r_{K-2}$, and $p_{k'+1} = r_{k'+1} - r'_k$ $(2 \leq k' \leq K-3)$. For $K = 2$ the calculation is trivial.

After all the membership vectors $\mathbf{u}_k$ are generated, the norms of $\mathbf{u}_k$'s are computed and assigned as factors in the clusters' weights, in order to "adjust" them to the additive fuzzy clustering model. Then the final membership matrix has its membership vectors $\mathbf{u}_k$ normalized.

The FCC generated data sets typically present clear-cut cluster structures.

4.2 Assessment of Cluster Structure Recovery

The FADDIS clustering recovery ability is evaluated according to the following parameters:

(i) Number of clusters retrieved by FADDIS and achieved stop condition;

(ii) Per generated cluster k and corresponding computed cluster $\widehat{k}$, measure:

(a) Recovery membership error (RME) of generated cluster k with membership vector $\mathbf{u}_k = [u_{ik}]$, and computed membership, $\widehat{\mathbf{u}}_k = [\widehat{u}_{ik}]$:

$$RME\left(\mathbf{u}_k\right) = \sum_{i=1}^{N} u_{ik}^2 \frac{|u_{ik} - \widehat{u}_{ik}|}{u_{ik}}$$

such that, $\sum_{i=1}^{N} u_{ik}^2 = 1$.

The RME error is an average relative difference weighted by normalization factor u_{ik}^2; its maximum value is one.

(b) Recovery intensity error (RIE) of generated and computed intensities, μ_k, and $\widehat{\mu}_k$,

$$RIE(\mu_k) = \frac{|\mu_k - \widehat{\mu}_k|}{\mu_k}.$$

(c) Percentage of the matching between the K generated R_k cores and the crisp cores retrieved from the computed partitions;

(iii) Similarity between generated and found partitions measured with ARI index.

The datasets have been generated in three groups for three different numbers of clusters: $K = 3, 4, 5$. The experiments were cross-combined according to the following settings: (i) Total number of entities of the data set $N = 50, 200, 400, 700$; (ii) α values of the standard deviation of noise, $\alpha = \{0, 0.05, 0.1, 0.15, 0.25, 0.5\}$. (iii) For each value of K, 10 distinct datasets had been generated for each tuple (N, α, β), resulting in a total of 720 datasets per K value, and so a total of 2160 datasets. In the case of NERFCM, the similarity data matrix G (6) is transformed into a dissimilarity matrix D, such that, $D = max(G) - G$.

The statistics are presented for $\alpha \in \{0, 0.05, 0.1\}$, since in preliminary experiments we observed that FADDIS clustering recovery significantly decreases for values of $\alpha > 0.1$. In Tables 4 and 5, the best value in each row is marked with (*). **We only present the no-Lapin FADDIS results since the Lapin FADDIS shown worse results.**

Table 4 shows the means/std and mode values of the recovered number of clusters by the FADDIS algorithm. For $K = 3, 4, 5$ the percentage of data sets for which the correct number of clusters is recovered increases with the increase of β values from $\beta = 0.0$ to $\beta = 1.0$. The only exception occurs for $K = 5, N = 200$, where the best values are achieved for $\beta = 0.5$. In all cases, the FADDIS algorithm stops at condition $S2$.

The analyses of the Recovery Membership Error (RME) and the Recovery Intensity Error (RIE) (Table 5), show that the minimum values are achieved for $\beta = 1.0$ for the collections of data sets with $K = 3$ and $K = 4$ clusters. For the data sets with $K = 5$, the minimum values are obtained for parameter $\beta = 0.5$. In any case, for $\beta = 1.0$ the RME and RIE errors are always inferior to 0.1 which is a very good value (the only exception is at $K = 3$ and $N = 700$). Also, the errors almost always decrease with the increase of β, which is in accordance with the expected property of different contributions of FADDIS.

On comparing FADDIS, FMFCM and NERFCM partitions, the highest ARI index values correspond to FADDIS partitions[2]. The best ARI values, with mean/std in the range $[0.8/0.2, 0.94/0.11]$, are achieved for data sets generated with $\beta = 1.0$ in the case of $K = 3$ and $K = 4$ clusters. For $K = 5$, the best values (ARI ≥ 0.93) are achieved at $\beta = 0.5$, in contrast to the expected property of 'different contributions'. The analysis of the best values for the percentages of the crisp core matching is concordant with the ARI ones.

Comparing between the FMFCM and NERFCM, in almost all the cases the NERFCM outperforms FMFCM for the data sets generated with $\beta = 0.0$, which is in contrast to the case of the entity-to-feature data at which FMFCM outperforms NERFCM (Brouwer, 2009). This is concordant to the idea that the NERFCM is a genuine relational clustering algorithm whereas the FMFCM is not.

5 Analysis of Activities and Biclustering

5.1 Deriving thematic clusters for the analysis of activities

The FADDIS method is part of a novel hybrid methodology called *cluster-lift* (Mirkin *et al.*, 2010) for the representation and interpretation of the activities conducted in a research organization, such as a University department, by mapping them to a related hierarchical taxonomy such as the Classification of Computer Subjects by ACM (ACM-CCS, 1998). This methodology combines: (i) the FADDIS method that allows to cluster research topics according to their thematic similarities without taking into account the topology of the taxonomy; (ii) a recursive method that lifts the clusters mapped onto the taxonomy to higher ranked nodes of the tree, leading to a parsimonious representation of the clusters in terms of topology derived concepts of "head subjects", "gaps" and "offshoots".

The cluster-lift method was applied on survey data about the research activities conducted in three University Computer Science units: one research centre and two university departments. The raw data have the format of a topic-by-respondent matrix with each entry being the proportion of a topic chosen by a respondent to express its contribution on the respondents' research activity. From this we constructed a topic-to-topic similarity matrix. A description of the raw data sets highlighting their cluster tendency, the found thematic clusters by FADDIS as well as their representation using the lifting method can be consulted in (Nascimento, 2011).

[2] The table with the ARI mean/std values is not shown due to lack of space.

Let us consider the survey data of a research centre[3] covering 46 topics of the 3rd layer of the ACM-CCS tree. The Lapin based FADDIS found two thematic clusters when applied to the topic-to-topic similarity matrix. To evaluate the consistency of those clusters with the topic-by-respondent raw matrix, we divided it into distinct submatrices . Each submatrix contains a subset of topics for which at least three respondents have the total of their research efforts covered by those topics. We obtained three submatrices. Comparing the FADDIS found thematic clusters with these submatrices we observed a rather good matching. Indeed, the two found clusters have all their topics coinciding with the topics covered by the two first submatrices of the original raw data. The remaining topics, covered by the third submatrix, are those not covered by FADDIS, thus being residual data.

The same kind of consistent results were obtained from the surveys conducted in two other University departments covering, respectively, 54 and 65 topics of the 3rd layer of ACM-CCS. For each department FADDIS found four thematic clusters characterizing their research activities.

This brings forth the idea of applying a biclustering method for automatically obtaining submatrices of high research efforts by simultaneously finding both thematic clusters and research teams engaged in them (for reviews of biclustering see (Prelić *et al.*, 2006; Yang *et al.*, 2007)).

5.2 Extending FADDIS to spectral fuzzy bi-clustering

Given a rectangular data matrix $F = (f_{iv})$ with a row-set I and column-set V, a bicluster is a pair (X, Y) where $X \subset I$ is a set of rows and $Y \subset V$ is a set of columns such that submatrix $F(X, Y)$ shows a remarkable relation between X and Y, as was stated in the pioneer paper by Hartigan (Hartigan, 1972). Currently this relation in most cases is assumed to be consistently higher values within the submatrix $F(X, Y)$ than in the rest of matrix F. In our case, set V corresponds to members of a research organization and set I to the research topics. Then f_{iv} would be the score given by member v to the topic i to express the proportion of their total research effort corresponding to topic i.

An extension of the spectral clustering approach to biclustering has been proposed by (Dhillon, 2001), due to reformatting the rectangular data matrix F into a format of a square symmetric matrix. Specifically, the $(|I| + |V|) \times (|I| + |V|)$ square matrix B is defined by the $|I| \times |V|$ matrix F as

$$B = \begin{bmatrix} 0 & F \\ F' & 0 \end{bmatrix}$$

where F' is the transpose of F. Consider a singular triplet (μ, c, z) of F so that $Fc = \mu z$ and $F'z = \mu c$. It is quite easy to see then that μ is an eigenvalue of B corresponding to a $(|I|+|V|)$-dimensional eigenvector y consisting of c and z, $y = (c, z)$ in such a way that norms of z and c are equal to each other (Mirkin, 2011b). That means that the first eigenvector of B corresponds to the first singular vectors of F, that form the best fitting solution to the

[3] Survey conducted in Centre for Artificial Intelligence (CENTRIA) of Universidade Nova de Lisboa in 2009. There was used the Electronic Survey tool of Scientific Activities (ESSA) available at `https://copsro.di.fct.unl.pt/`.

model $f_{iv} = \mu c_i z_v + e_{iv}$ at which f_{iv} is observed and μ, c_i, z_v are sought values according to the principle of least-squares (see, for example, (Mirkin, 2011b)). Accordingly, the corresponding projection of the pair $y = (c, z)$ to the set of fuzzy membership values would be the fuzzy bicluster found according to the same model.

That means that FADDIS can be applied as is to our raw data matrix F reformatted into B rather then the topic-to-topic similarity matrix discussed previously. Of course, B then should be Lapin-transformed into a matrix A.

When applied to the research center raw data, the FADDIS bi-clustering found fuzzy biclusters almost coincide with the FADDIS thematic clusters. Additionally, they provide the information of member teams behind the thematic clusters. Each of the biclusters is tight in the sense that almost all of the individuals taken have 100% of their contribution efforts covered by the topics of corresponding bicluster. Specifically, in the case of cluster 1 four of five individuals have 100% of their contributions on this cluster whereas one individual has 40% of its contribution in the cluster, and the remaining 60% not covered by any cluster. For bicluster 2, seven individuals have 100% of their contributions covered by the topics of this bicluster, while one individual has 10% covered by the topics of bicluster 1 and the remaining 40% is covered by no cluster.

However, the biclustering results on the other research organizations do not have such a high matching score with the FADDIS ones.

6 Conclusion

The paper presents and experimentally studies an unconventional model of fuzzy clustering in which the observed similarity between entities is approximated by the weighted product of their fuzzy membership values that contribute towards the similarity. This is motivated by the idea that the similarity between research topics is obtained by adding up the working of different groups on them so that the clusters according to this model can be considered thematic clusters indeed. The spectral fuzzy clustering method FADDIS is accompanied with a set of model-based cluster extracting stop-conditions.

The presented work demonstrates that FADDIS is competitive on both the conventional data formats, two types of generated cluster structures, and in real world data. In most of our experiments the Lapin based FADDIS has been able to determine the correct number of clusters in data indeed. **The Lapin works quite well on affinity data of Gaussian clusters with different levels of intermix, on various difficult geometric data, as well as on genuine similarity data having a no clear-cut cluster structure, as is the case of the research activities data. Oppositely, for clusterdness data the Lapin transformation tends to destroy that structure (a situation also reported in (von Luxburg *et al.*, 2010)), while the no-Lapin FADDIS gets better results.**

Yet, there are some irregularities in FADDIS working that deserve to be investigated further. One of the irregularities is the experimentally observed deviations from the property of different cluster contributions. According to the definition of FADDIS, the more different the cluster weights in the data, that is, the greater the β at the genuine similarity data generator, the better should be the correspondence between the generated clusters and those FADDIS computed. This is true in most cases, but sometimes it is not. We are going to address this in our future work. **Another direction for future work is further exploring the Lapin transformation and its role in the recovery of a correct number of clusters. One more line of the future research is in the analysis of competitiveness of FADDIS biclustering and its adequacy to the tasks of the analysis of research activities.**

FADDIS is potentially useful in applications where there exists interest to model pairwise similarities by additive properties. This is the case of finding thematic clusters of the activities conducted in an organization or, for example, in collaborative filtering by clustering of items based on user preferences. FADDIS sequential extraction of clusters maximizing their contribution to the similarity data scatter allows to obtain the most K relevant thematic clusters towards their strength to similarity, typically having contrastful memberships indeed. These are good characteristics for decision-making processes.

Acknowledgments

This work has been supported by project grant PTDC/EIA/69988/2006 from the Portuguese Foundation for Science & Technology. The partial support of the Decision Analysis and Choice Laboratory and the Programme of Fundamental Studies at the National Research University Higher School of Economics, Moscow RF, to BM is acknowledged.

The authors are indebted to the anonymous Reviewers and the Editors for their comments that allowed to improve the paper.

References

ACM Computing Classification System (1998) `http://www.acm.org/about/class/1998` (Cited 9 Sep 2008).

ABBASSI, Z. and V. MIRROKNI (2009) A recommender system based on local random walks and spectral methods, *Advances in Web Mining and Web Usage Analysis*, LNCS, Volume **5439/2009**, 139-153.

BEZDEK, J., J. KELLER, R. KRISHNAPURAM and T. PAL (1999) Fuzzy Models and Algorithms for Pattern Recognition and Image Processing, Kluwer Academic Publishers.

BROUWER, R. (2009) A method of relational fuzzy clustering based on producing feature vectors using FastMap, *Information Sciences*, **179**, 3561-3582.

CASILLAS, J. and F. J. MARTÍNEZ LÓPEZ (2010) Marketing Intelligent Systems Using Soft Computing, *Studies in Fuzziness and Soft Computing*, **258**, Springer, 476p.

CASTELLANO, G. and M.A. TORSELLO (2009) How to Derive Fuzzy User Categories for Web Personalization. *Web Personalization in Intelligent Environments, Studies in Computational Intelligence*, **229/2009**, Springer, 65-79.

DAVÉ, R. and S. SEN (2002) Robust fuzzy clustering of relational data, *IEEE Transactions on Fuzzy Systems*, **10**, 713-727.

DHILLON, I. S. (2001) Co-clustering Documents and Words Using Bipartite Spectral Graph Partitioning, *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'01)*, ACM, NY, 269-274.

DROBICS, M., U. BODENHOFER and W. WINIWARTER (2002) Mining Clusters and Corresponding Interpretable Descriptions - A Three-Stage Approach, *Expert Systems*, **19**(4), 224-234.

FRANK, A. and A. ASUNCION (2010) UCI Machine Learning Repository [`http://archive.ics.uci.edu/ml`], Irvine, CA: University of California, School of Information and Computer Science.

HARTIGAN, J.A. (1972) Direct clustering of a data matrix, *Journal of American Statistical Association*, **67**, 123-129.

HATHAWAY, R., J. DAVENPORT and J. BEZDEK (1989) Relational duals of the c-means algorithms, *Pattern Recognition*, **22**, 205-212.

HATHAWAY, R.J. and J. BEZDEK (1994) NERF c-means: Non-Euclidean relational fuzzy clustering. *Pattern Recognition*, **27**, 429-437.

HUANG, L., D. YAN, M.I. JORDAN and N. TAFT (2009) Spectral clustering with perturbed data. In: Koller, D., Schuurmans, D., Bengio, Y., Bottou, L. (Eds.): Advances in Neural Information Processing Systems 21, *Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems (Vancouver)*, MIT Press, 705-712.

HUBERT, L.J. and P. ARABIE (1985) Comparing partitions, *Journal of Classification*, **2**, 193-218.

INOUE, K. and K. URAHAMA (1999) Sequential fuzzy cluster extraction by a graph spectral method, *Pattern Recognition Letters*, **20**, 699-705.

KRISHNAPURAM, R., A. JOSHI, O. NASRAOUI and L. YI (2001) Low-complexity fuzzy relational clustering algorithms for Web mining, *IEEE Transactions on Fuzzy Systems*, **9**(4), 595-607.

MASULLIA, F. and S. MITRA (2009) Natural computing methods in bioinformatics: A survey, *Information Fusion*, **10**(3), 211-216.

MEYER, A. and H.-J. ZIMMERMANN (2011) Applications of Fuzzy Technology in Business Intelligence, *International Journal of Computers, Communications & Control*, **VI**(3), 428-441.

MIRKIN, B. (1987) Additive clustering and qualitative factor analysis methods for similarity matrices, *Journal of Classification*, **4**(1), 7-31.

MIRKIN, B. (2011a) Choosing the number of clusters, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, **1**(3), 252-260.

MIRKIN, B. (2011b) Core Concepts in Data Analysis: Summarization, Correlation, Visualization, Springer, London.

MIRKIN, B. and S. NASCIMENTO (2012) Additive Spectral Method for Fuzzy Cluster Analysis of Similarity Data Including Community Structure and Affinity Matrices, *Information Sciences*, **183**(1), 16-34.

MIRKIN, B., S. NASCIMENTO, T. FENNER and L.M. PEREIRA (2010) Constructing and Mapping Fuzzy Thematic Clusters to Higher Ranks in a Taxonomy. In: Bi, Y., Williams, M.A. (Eds.), *4th Intl. Conf. on Knowledge Science, Engineering & Management (KSEM 2010)*, Springer LNAI **6291**, 329-340.

NADLER, B., S. LAFON, R. R. COIFMAN and I.G. KEVREKIDIS (2006) Diffusion Maps, Spectral Clustering and Reaction Coordinates of Dynamical Systems, *Applied and Computational Harmonic Analysis*, **21**, 113-127.

NANOPOULOS, A., H-H. GABRIEL and M. SPILIOPOULOU (2009) Spectral Clustering in Social-Tagging Systems. In Vossen, G., Long, D., Yu, J.X. (eds.), *Procs. of the 10th International Conference on Web Information Systems Engineering (WISE 2009)*, Springer-Verlag LNCS **5802**, 87-100.

NASCIMENTO, S. (2011) Analysis of the Research Activities using the ACM-CCS Taxonomy, Case Studies of CS Departments/Research-Centres in Portugal and UK, COPSRO Project (grant PTDC/EIA/269988/006), *Technical Report*, 45p., `http://centria.di.fct.unl.pt/~snt/papers/ESSA2009_FinalReport.pdf`.

NASRAOUI O., R. KRISHNAPURAM, A. JOSHI and T. KAMDAR (2002) Automatic Web User Profiling and Personalization using Robust Fuzzy Relational Clustering. In: J. Segovia, J., Szczepaniak, P., and Niedzwiedzinski, M. (Eds.), *E-Commerce and Intelligent Methods, Series Studies in Fuzziness and Soft Computing*, Springer-Verlag.

NASRAOUI, O. and H. FRIGUI (2000) Extracting Web User Profiles Using Relational Competitive Fuzzy Clustering, *International Journal on Artificial Intelligence Tools (IJAIT)*, **9**(4), 509-526.

NG, A., M. JORDAN and Y. WEISS (2002) On spectral clustering: analysis and an algorithm. In: Ditterich, T.G., Becker, S., Ghahramani, Z. (Eds.), *Advances in Neural Information Processing Systems*, **14**, MIT Press, Cambridge Ma., 849-856.

PAL, N.R., K. AGUAN, A. SHARMA and S. AMARI (2007) Discovering biomarkers from gene expression data for predicting cancer subgroups using neural networks and relational fuzzy clustering, *BMC Bioinformatics*, **8**, 5.

POPESCU, M., J. KELLER and J. MITCHELL (2006) Fuzzy Measures on the Gene Ontology for Gene Product Similarity, *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, **3**(3), 263-274.

PRELIĆ, A., S. BLEULER1, P. ZIMMERMANN, A. WILLE, P. BHLMANN, W. GRUISSEM, L. HENNIG, L. THIELE and E. ZITZLER (2006) A systematic comparison and evaluation of biclustering methods for gene expression data, *Bioinformatics*, **22**(9), 1122-1129.

ROUBENS, M. (1978) Pattern classification problems and fuzzy sets, *Fuzzy Sets and Systems*, **1**, 239-253.

RUNKLER, T. A. and J. BEZDEK (2003) Web mining with relational clustering. International Journal of Approximate Reasoning, *Elsevier Science*, **32**(2-3), 217-236.

SATO, M., Y. SATO and L.C. JAIN (1997) Fuzzy Clustering Models and Applications, Physica-Verlag, Heidelberg.

SHEPARD, R.N. and P. Arabie (1979) Additive clustering: representation of similarities as combinations of overlapping properties, *Psychological Review*, **86**, 87-123.

SHI, J. and J. MALIK (2000) Normalized cuts and image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(8), 888-905.

SLEDGE, I.J., J. BEZDEK, T.C. HAVENS and J. KELLER (2010) Relational Generalizations of Cluster Validity Indices, *IEEE Transactions on Fuzzy Systems*, **18**(4), 771-786.

SURYAVANSHI, B.S., N. SHIRI and S.P. MUDUR (2005) An Efficient Technique for Mining Usage Profiles Using Relational Fuzzy Subtractive Clustering, *Procs. of the International Workshop on Challenges in Web Information Retrieval and Integration (WIRI'05)*, 23-29.

TREERATTANAPITAK, K. and C. JARUSKULCHAI (2012) Exponential Fuzzy *C*-Means for Collaborative Filtering, *Journal of Computer Science and Technology*, **27**(3), 567-576.

VON LUXBURG, U. (2007) A tutorial on spectral clustering. Statistics and Computing, **17**, 395-416.

VON LUXBURG, U., A. RADL and M. HEIN (2010), Getting lost in space: large sample analysis of the commute distance, *Proceedings of the 23th neural information processing systems conference (NIPS'10)*, 2622-2630.

WINDHAM, M.P. (1985) Numerical classification of proximity data with assignment measures, *Journal of Classification*, **2**, 157-172.

XU, D., J. M. KELLER, M. POPESCU and R. BONDUGULA (2008) Applications of Fuzzy Logic in Bioinformatics, Imperial College Press London, UK.

YANG, E., P. T. FOTEINOU, K. R. KING, M.L. YARMUSH and I.P. ANDROULAKIS (2007) A novel non-overlapping bi-clustering algorithm for network generation using living cell array data, *Bioinformatics*, **23**(17), 2306-2313.

YANG, M. and H. SHIH (2001) Cluster analysis based on fuzzy relations, *Fuzzy Sets and Systems*, **120**, 197-212.

ZIMMERMANN, H.-J. (2001) Fuzzy Set Theory and its Applications, fourth edition, Kluwer Academic Publishers.

ZELNIK-MANOR, L. and P. PERONA (2004) Self-tuning spectral clustering, *Advances in Neural Information Processing Systems (NIPS'04)*, **17**, 1601-1608.

ZHANG, S., R.-S. WANG and X.-S. ZHANG (2007) Identification of overlapping community structure in complex networks using fuzzy c-means clustering, *Physica A*, **374**, 483-490.

Vitae

Susana Nascimento's bio: Susana Nascimento received a Ph.D. degree in Computer Science from Universidade Nova de Lisboa in 2002. She is an Assistant Professor in the Department of Computer Science, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Lisboa, Portugal. Her main research interests include fuzzy cluster analysis and classification with applications to machine learning and data mining problems in general and remote sensing imagery.

Rui Felizardo's bio: Rui Felizardo received a M.Sc. degree in Computer Science from Universidade Nova de Lisboa in 2011, and he worked in the COPSRO project (2009-2011). His main research interests include data mining, clustering, operational research, and decision support models.

Boris Mirkin's bio: Boris Mirkin is Emeritus Professor of Computer Science in the Department of Computer Science, Birkbeck University of London, UK, and Professor in the Division of Applied Mathematics and Informatics, National Research University Higher School of Economics, Moscow, RF. He is interested in mathematical models, computational algorithms and programs for clustering and interpreting data in such applications as genomics, sociology, and text analysis. He published a hundred of refereed papers and several books on these; the latest text is "Core concepts in data analysis: Summarization, Correlation, Visualization" (Springer, 2011).

Tables

Table 1: FADDIS differences of clusters' contributions to the data scatter for the seven first clusters in two options of the algorithm: no-Lapin and Lapin.

Diff. Contribus		1-2	2-3	3-4	4-5	5-6	6-7
no Lapin	*D*1/3	0.0008	0.0062	0.0135	0.0013	0.0020	0.0019
	*D*2/5	0.0002	0.0001	0.0003	0.0039	0.0028	0.0016
	*D*3/4	0.0008	0.0034	0.0001	0.0044	0.0044	0.0021
	*D*4/3	0.0045	0.0043	0.0001	0.0100	0.0008	0.0020
	*D*5/3	0.0045	0.0093	0.0055	0.0000	0.0073	0.0026
	*D*6/3	0.0057	0.0023	0.0027	0.0037	0.0010	0.0013
	*D*7/2	0.0022	0.0011	0.0021	0.0005	0.0029	0.0038
	*D*9/2	0.0337	0.0565	0.0051	0.0253	0.0044	0.0033
	*D*10/3	0.3458	0.1146	0.0571	0.0096	0.0008	0.0003
	*D*11/2	0.7778	0.0122	0.0274	0.0005	0.0007	0.0005
Lapin	*D*1/3	0.2975	**0.1234***	0	0	0	0
	*D*2/5	0.2575	0.0665	0.0901	**0.0114***	0.0003	0.0003
	*D*3/4	0.0043	0.0807	0.0076	**0.0349***	0.0044	0.0046
	*D*4/3	0.3005	0.1013	**0.0109***	0.0005	0.0004	0.0002
	*D*5/3	0.0967	0.20836	**0.0334***	0.0002	0	0.0001
	*D*6/3	0.0172	0.2288	**0.0132***	0.0001	0.0001	0
	*D*7/2	0.0134	**0.2432***	0	0	0	0
	*D*9/2	0.3782	**0.0966***	0	0	0	0
	*D*10/3	0.0354	0.0104	**0.039***	0.0009	0.0008	0.0002
	*D*11/2	0.5373	0.0606	**0.0269***	0.0039	0.0018	0.0014

Table 2: The four algorithms clustering results for artificial datasets D9/2 - D11/2.

	FADDIS- no Lapin		FADDIS- Lapin		FastMap FCM		NERFCM	
D	Output K	ARI	Output K	ARI	Input K	ARI	Input K	ARI
*D*9/2	2	1	2	1	2	0.69	2	0.920
*D*10/3	3	0.880	3	0.867	3	0.922	3	0.922
*D*11/2	2	0.227	3	0.812	2	0.841	2	0.830

Table 3: Bivariate Normal DG with different scale values of cluster intermix – Adjusted Rand Index (ARI) avg/std for FADDIS, NERFCM and FMFCM

	FADDIS		NERFCM			FastMap FCM		
sn	Lapin	K	K = 3	K = 4	K = 5	K = 3	K = 4	K = 5
-5	**0.47/0.048***	4	0.47/0.05	0.44/0.05	0.37/0.035	0.47/0.05	0.44/0.045	0.37/0.03
0	**0.68/0.029***	4	0.66/0.034	0.64/0.058	0.53/0.032	0.66/0.035	0.61/0.096	0.54/0.013
5	0.83/0.022	4	0.76/0.018	**0.84/0.016***	0.67/0.036	0.76/0.018	**0.84/0.016***	0.67/0.031
10	0.91/0.029	4	0.82/0.015	**0.93/0.021***	0.74/0.025	0.82/0.015	**0.93/0.021***	0.75/0.029
20	0.98/0.022	4	0.86/0.008	**0.99/0.009***	0.85/0.07	0.86/0.008	**0.99/0.009***	0.82/0.067
50	**1/0***	4	0.87/0.007	**1/0***	0.87/0.075	0.87/0.007	**1/0***	0.87/0.07

Table 4: FCC DG - Summary data of the percentage avg/std of correct extracted clusters and mode of the number of extracted clusters for std of added Gaussian noise=[0, 0.1] for FADDIS in best conditions for $K = \{3, 4, 5\}$

		FADDIS					
		$\beta = 0.0$		$\beta = 0.5$		$\beta = 1.0$	
	N	(%)	Mode	(%)	Mode	(%)	Mode
$K = 3$	50	50.0/0.0	3	62.5/9.6	3	85.0/5.8*	3
	200	60.0/0.0*	3	32.5/20.6	2	60.0/0.0*	3
	400	30.0/21.6	3	62.5/15.0	3	80.0/0.0*	3
	700	17.5/17.1	2	40.0/35.6	2	65.0/19.1*	3
$K = 4$	50	47.5/9.6	4	60.0/8.2	4	70.0/18.3*	4
	200	50.0/35.6	4	50.0/0.0	4	65.0/5.8*	4
	400	27.5/18.9	5	55.0/10.0	4	72.5/5.0*	4
	700	17.5/20.6	1	67.5/5.0	4	77.5/5.0*	4
$K = 5$	50	40.0/21.6	5	60.0/8.2	5	67.5/5.0*	5
	200	37.5/26.3	5	52.5/5.0*	5	40.0/8.2	5
	400	45.0/46.5	5	50.0/0.0	5	65.0/10.0*	5
	700	25.0/23.8	1	35.0/5.8	6	42.5/5.0*	5

Table 5: Summary Table of the RME and RIE errors' avg/std for std of added Gaussian noise=[0, 0.1] for FADDIS in best conditions for $K = \{3, 4, 5\}$

		RME				RIE		
	N	$\beta = 0.0$	$\beta = 0.5$	$\beta = 1.0$		$\beta = 0.0$	$\beta = 0.5$	$\beta = 1.0$
$K = 3$	50	0.25/0.08	0.24/0.02	0.14/0.02*		0.14/0.03	0.15/0.01	0.08/0.01*
	200	0.28/0.08	0.54/0.12	0.15/0.01*		0.13/0.02	0.29/0.08	0.07/0.00*
	400	0.45/0.34	0.18/0.11	0.14/0.01*		0.30/0.27	0.09/0.05	0.09/0.00*
	700	0.56/0.33	0.39/0.18	0.21/0.05*		0.35/0.24	0.25/0.14	0.13/0.05*
$K = 4$	50	0.22/0.07	0.13/0.05	0.13/0.02*		0.11/0.01	0.07/0.01*	0.08/0.04
	200	0.44/0.35	0.12/0.01	0.12/0.01*		0.29/0.30	0.06/0.00	0.06/0.00*
	400	0.41/0.33	0.20/0.01	0.10/0.03*		0.28/0.29	0.10/0.00	0.05/0.01*
	700	0.59/0.36	0.13/0.05*	0.17/0.01		0.43/0.30	0.07/0.02	0.07/0.00*
$K = 5$	50	0.28/0.12	0.14/0.02*	0.17/0.01		0.15/0.04	0.07/0.01	0.07/0.00*
	200	0.36/0.26	0.14/0.04	0.13/0.02*		0.21/0.18	0.07/0.01	0.06/0.01*
	400	0.40/0.34	0.07/0.01*	0.12/0.01		0.28/0.29	0.04/0.00*	0.05/0.01
	700	0.49/0.37	0.17/0.03*	0.18/0.01		0.35/0.33	0.10/0.01	0.06/0.00*

Figures

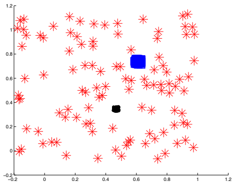
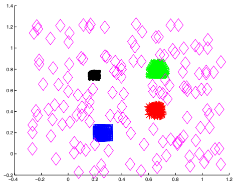
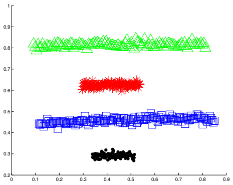
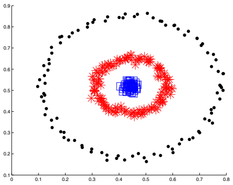
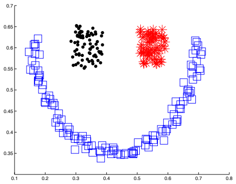
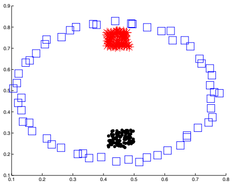
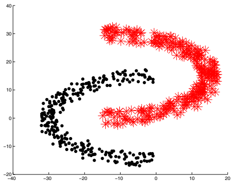
 D1/3	 D2/5	 D3/4	 D4/3
Number of entities: 622 Number of Classes: 5 Number of Attributes: 2	Number of entities: 622 Number of Classes: 5 Number of Attributes: 2	Number of entities: 512 Number of Classes: 4 Number of Attributes: 2	Number of entities: 299 Number of Classes: 3 Number of Attributes: 2
 D5/3	 D6/3	 D7/2	
Number of entities: 266 Number of Classes: 3 Number of Attributes: 2	Number of entities: 238 Number of Classes: 3 Number of Attributes: 2	Number of entities: 600 Number of Classes: 2 Number of Attributes: 2	

Figure 1: Seven artificial bivariate datasets.

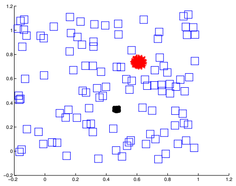
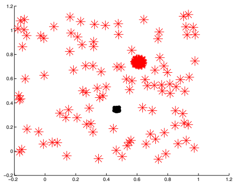
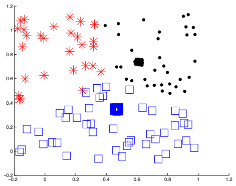
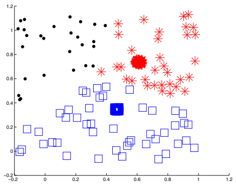
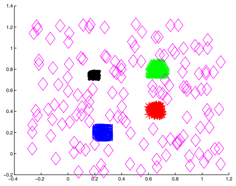
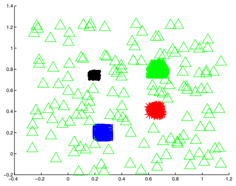
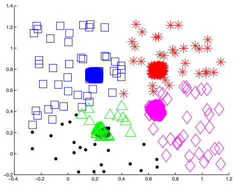
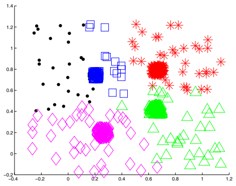
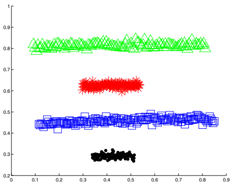
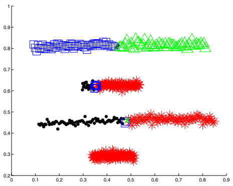
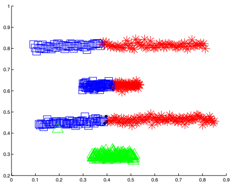
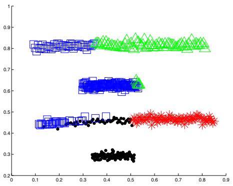
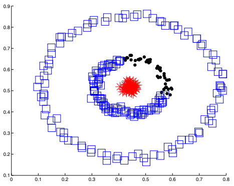
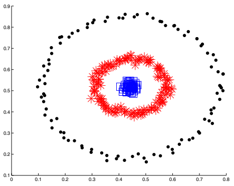
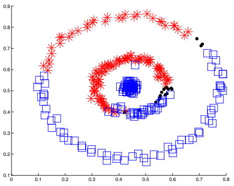
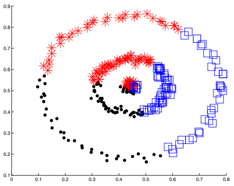
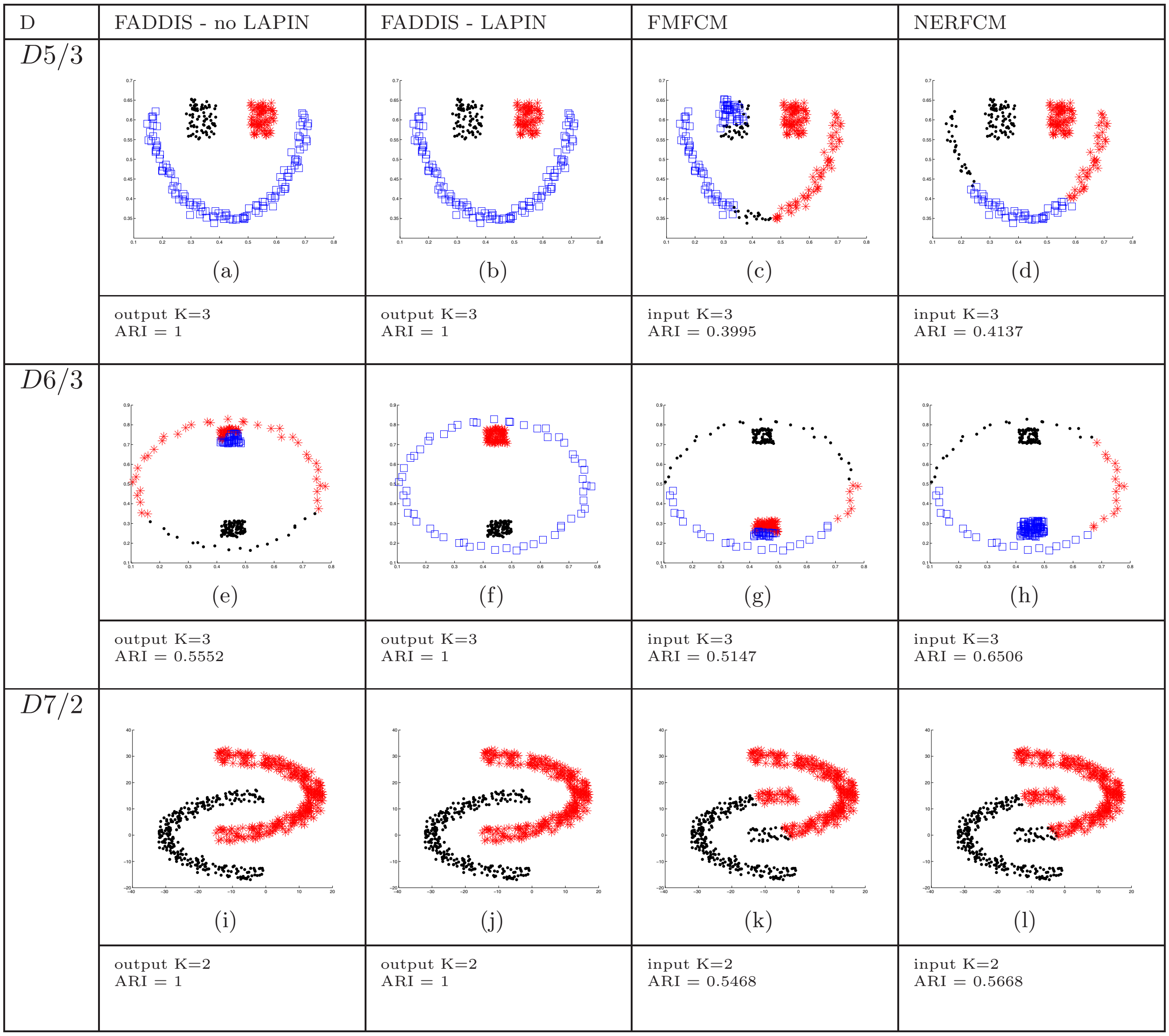
D	FADDIS - no LAPIN	FADDIS - LAPIN	FMFCM	NERFCM
$D1/3$	 (a)	 (b)	 (c)	 (d)
	output K=3 ARI = 1	output K=2 ARI = 0.5747	input K=3 ARI = 0.4734	input K=3 ARI = 0.4709
$D2/5$	 (e)	 (f)	 (g)	 (h)
	output K=5 ARI = 1	output K=4 ARI = 0.7259	input K=5 ARI = 0.6225	input K=5 ARI = 0.6857
$D3/4$	 (i)	 (j)	 (k)	 (l)
	output K=4 ARI = 1	output K=4 ARI = 0.2756 Error(%) = 0.4902	input K=4 ARI = 0.3184	input K=4 ARI = 0.4665
$D4/3$	 (m)	 (n)	 (o)	 (p)
	output K=3 ARI = 0.3894	output K=3 ARI = 1 Error(%) = 0	input K=3 ARI = 0.3507	input K=3 ARI = 0.0005

Figure 2: The four algorithms clustering results for the artificial datasets D1/3 - D4/3.



D	FADDIS - no LAPIN	FADDIS - LAPIN	FMFCM	NERFCM
$D5/3$	(a)	(b)	(c)	(d)
	output K=3 ARI = 1	output K=3 ARI = 1	input K=3 ARI = 0.3995	input K=3 ARI = 0.4137
$D6/3$	(e)	(f)	(g)	(h)
	output K=3 ARI = 0.5552	output K=3 ARI = 1	input K=3 ARI = 0.5147	input K=3 ARI = 0.6506
$D7/2$	(i)	(j)	(k)	(l)
	output K=2 ARI = 1	output K=2 ARI = 1	input K=2 ARI = 0.5468	input K=2 ARI = 0.5668

Figure 3: The four algorithms clustering results for artificial datasets D5/3 - D7/2.

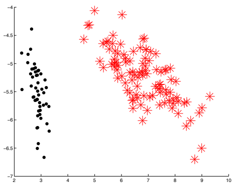
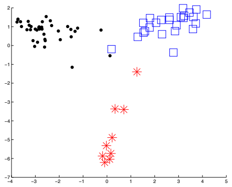
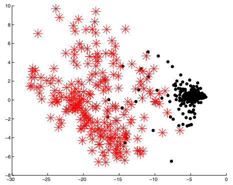
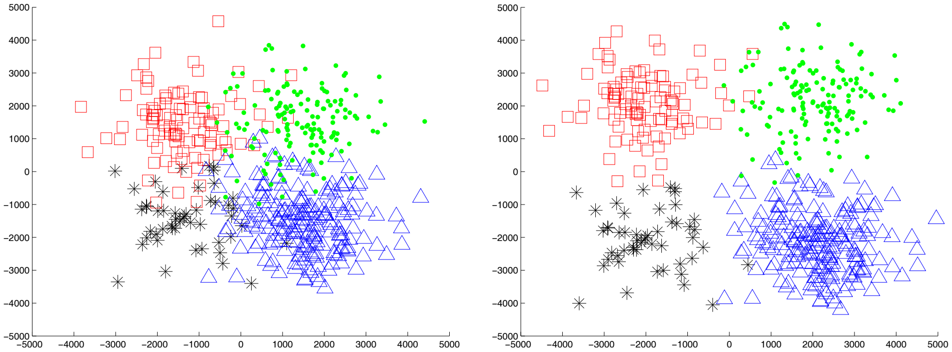
 D9/2	 D10/3	 D11/2
Number of entities: 150 Number of Clusters: 2 Number of Attributes: 4	Number of entities: 72 Number of Clusters: 3 Number of Attributes: 70	Number of entities: 683 Number of Clusters: 2 Number of Attributes: 9

Figure 4: Three datasets from UCI Machine Learning Repository: Iris($D9/2$), Leukemia($D10/3$), and Wisconsin Breast Cancer($D11/2$).



(a) Dataset generated at $sn = 0$

(b) Dataset generated at $sn = 1$

Figure 5: Dataset with two different levels of intermix.

Figure Captions

- Figure 1: Seven artificial bivariate datasets.
- Figure 2: The four algorithms clustering results for the artificial datasets D1/3 - D4/3.
- Figure 3: The four algorithms clustering results for artificial datasets D5/3 - D7/2.
- Figure 4: Three datasets from UCI Machine Learning Repository: Iris(D9/2), Leukemia(D10/3), and Wisconsin Breast Cancer(D11/2).
- Figure 5: Dataset with two different levels of intermix.