A New Approach to Clustering Using Eigen Decomposition

Thomas A. Runkler and Florian Steinke

***Abstract*— We propose a novel approach to relational clustering: Given a matrix of pairwise similarity values between objects our algorithm computes a partition of the objects such that similar objects belong to the same cluster and dissimilar objects belong to different clusters. The proposed approach is based on the assumption that the given similarities are products of cluster membership variables. It is based on eigen vector decomposition and minimizes the squared error between the similarities and the products of membership vectors in an efficient, non–iterative way with guaranteed global optimality. In experiments with real world data we show superior performance to conventional iterative clustering approaches.**

I. Introduction

Clustering is the process of partitioning a set of $n \in \{1, 2, \ldots\}$ (abstract) objects $O = \{o_1, \ldots, o_n\}$ into $c \in \{1, 2, \ldots\}$ disjoint subsets (clusters) $C_1, \ldots, C_c$, $C_i \cap C_j = \emptyset$, $i \neq j$, $C_1 \cup C_2 \cup \ldots \cup C_c = O$, in such a way that objects in the same cluster are similar and/or objects in different clusters are dissimilar.

Similarities or dissimilarities between objects may be specified by a matrix of pairwise similarities

$$S = \begin{pmatrix} s_{11} & \cdots & s_{1n} \\ \vdots & \ddots & \vdots \\ s_{n1} & \cdots & s_{nn} \end{pmatrix} \tag{1}$$

$s_{ij} \geq 0$, $s_{ij} = s_{ji}$, $s_{ii} > s_{ij}$ $i, j = 1, \ldots, n$, or a matrix of pairwise dissimilarities

$$D = \begin{pmatrix} d_{11} & \cdots & d_{1n} \\ \vdots & \ddots & \vdots \\ d_{n1} & \cdots & d_{nn} \end{pmatrix} \tag{2}$$

$d_{ij} \geq 0$, $d_{ij} = d_{ji}$, $d_{ii} = 0$ $i, j = 1, \ldots, n$. Such similarity and dissimilarity matrices define relations on $O \times O$, hence clustering based on such relation matrices is called *relational clustering* [3]. Similarities or dissimilarities may be explicitly given, e.g. from human ratings in phychological experiments, or they may be derived from an object feature data set $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^p$ where each feature vector x_k, $k = 1, \ldots, n$, refers to the features associated with object o_k. Consider for example a production of wood fiber boards that are specified by their geometry (length, width) and their quality (strength, water absorption), leading to (normalized) four dimensional feature vectors [17]. The similarity of each pair of fiber boards o_i and o_j can then be computed from the associated feature vectors x_i and x_j using an appropriate similarity measure, for example the *cosine similarity*

$$s_{ij} = \frac{x_i x_j^T}{\sqrt{(x_i x_i^T)(x_j x_j^T)}} \tag{3}$$

and the dissimilarity can be computed with an appropriate dissimilarity measure, for example the *Euclidean norm*

$$d_{ij} = \sqrt{(x_i - x_j)(x_i - x_j)^T} \tag{4}$$

Clustering based on feature vectors is called *object feature data clustering*. Notice that any relational clustering method can be used for object feature data clustering if the given feature vectors are first used to compute a similariy or dissimilarity matrix.

In this paper we present a new approach to clustering based on the idea that any pair of objects (o_i, o_j) should belong to the same cluster in case of a high similarity s_{ij} and should belong to different clusters in case of a low similarity s_{ij}. This leads us to define a new functional for so–called *decomposite clustering*. Ding and He [5] have pointed out similarities between HCM and *principal component analysis (PCA)* [11]. Our approach is motivated by *multidimensional scaling (MDS)* [21], and yields a highly efficient one step algorithm to perform clustering using eigen decomposition.

This paper is structured as follows: In section II we give a brief overview of state of the art iterative clustering algorithms for relational and/or object feature data. In section III we define the objective function of decomposite clustering and outline a gradient descent approach to optimize it. In section IV we quickly review multidimensional scaling and its one step solution using eigen decomposition. In section V we show how eigen decomposition can be applied to decomposite clustering as well leading to a one step clustering algorithm. In section VI we present experiments with two real world data sets comparing decomposite clustering with a popular iterative clustering algorithms (NERFCM). In section VII we finally summarize our conclusions.

II. Clustering Algorithms

An example for relational clustering is the *sequential agglomerative hierarchical nonoverlapping (SAHN)* clustering algorithm [19]. SAHN can be used for both similarity as well as dissimilarity matrices. The SAHN algorithm initially assigns each object to an individual cluster $C_i = \{o_i\}$, $i = 1, \ldots, n$, so initially we have $c = n$ clusters. In each step SAHN decreases the number of clusters by one by merging the two most similar (or least dissimilar) clusters to one new cluster. Similarity between two clusters C_i and C_j may be defined as the minimum, the maximum, or the average

Thomas A. Runkler and Florian Steinke are with Siemens Corporate Technology, 81730 Munich, Germany (email: thomas.runkler@siemens.com, phone: +49 89 636 40010).

similarity between any pair of objects from $C_i \times C_j$. Dissimilarity between clusters may be defined as the minimum, the maximum, or the average dissimilarity between object pairs. The corresponding algorithms are called *single linkage* (maximum similarity, minimum dissimilarity), *complete linkage* (minimum similarity, maximum dissimilarity), and *average linkage* (average similarity, average dissimilarity). SAHN terminates when the desired number of clusters is achieved or, latest, when all objects are merged to one single cluster $C_1 = O$.

An example for object feature data clustering is the *c–means* family, containing in particular *hard c–means (HCM)* [1], *fuzzy c–means (FCM)* [2], and *possibilistic c–means (PCM)* [10]. This family of clustering models uses a set of *prototypes* $V = \{v_1, \ldots, v_c\} \subset \mathbb{R}^p$ to represent the clusters, where v_i is the *center* of cluster C_i. Clustering is done by minimizing the dissimilarities between the cluster prototypes (centers) and the feature vectors of the objects belonging to the respective clusters, so the c–means clustering models are all based on dissimilarities not similarities. For example, FCM minimizes

$$J_m = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^m d(v_i, x_k)^2 \tag{5}$$

for the matrix U of memberships $u_{ik} \in [0,1]$ of object o_k in cluster C_i, with the normalization conditions

$$\sum_{k=1}^{n} u_{ik} = 1, \quad i = 1, \ldots, c \tag{6}$$

$d(v_i, x_k)$ denotes the dissimilarity between v_i and x_k, for example the Euclidean norm, and $m > 1$ is a suitable algorithmic parameter called *fuzziness*. The necessary conditions for extrema of J_m can be used for iterative minimization. If $d(\cdot,\cdot)$ is an inner product norm, then U and V are alternatingly updated by

$$u_{ik} = 1 \Bigg/ \sum_{j=1}^{c} \left(\frac{d(v_i, x_k)}{d(v_j, x_k)} \right)^{\frac{2}{m-1}} \tag{7}$$

$$v_i = \frac{\sum\limits_{k=1}^{n} u_{ik}^m x_k}{\sum\limits_{k=1}^{n} u_{ik}^m} \tag{8}$$

A relational extension of FCM can be obtained by inserting (8) into (5) [8]. This yields the *relational fuzzy c–means (RFCM)* objective function [3]

$$J_m = \sum_{i=1}^{c} \frac{\sum\limits_{j=1}^{n} \sum\limits_{k=1}^{n} u_{ij}^m u_{ik}^m d_{jk}^2}{\sum\limits_{j=1}^{n} u_{ij}^m} \tag{9}$$

which can be minimized by iteratively updating

$$u_{ik} = 1 \Bigg/ \sum_{j=1}^{n} \frac{\sum\limits_{s=1}^{n} \frac{u_{is}^m d_{sk}}{\sum\limits_{r=1}^{n} u_{ir}^m} - \sum\limits_{s=1}^{n} \sum\limits_{t=1}^{n} \frac{u_{is}^m u_{it}^m d_{st}}{2\left(\sum\limits_{r=1}^{n} u_{ir}^m\right)^2}}{\sum\limits_{s=1}^{n} \frac{u_{js}^m d_{sk}}{\sum\limits_{r=1}^{n} u_{jr}^m} - \sum\limits_{s=1}^{n} \sum\limits_{t=1}^{n} \frac{u_{js}^m u_{jt}^m d_{st}}{2\left(\sum\limits_{r=1}^{n} u_{jr}^m\right)^2}} \tag{10}$$

The reformulation from FCM to RFCM implicitly assumes that the dissimilarity matrix D is computed from some feature data X using the Euclidean norm. If this is not the case, then RFCM might produce memberships $u_{ik} < 0$ or $u_{ik} > 1$. This problem can be avoided by transforming the dissimilarity matrix D into a Euclidean dissimilarity matrix D_β by applying a so–called β–spread transform [7]

$$D_\beta = D + \beta \cdot B \tag{11}$$

with a suitable $\beta \in \mathbb{R}^+$, where $B \in [0,1]^{n \times n}$ is the off–diagonal matrix with $b_{ij} = 1$ for all $i, j = 1, \ldots, n$, $i \neq j$, and $b_{ii} = 0$ for all $i = 1, \ldots, n$. The parameter β is sucessively increased, i.e. higher values of β are added to the off–diagonal elements of R, until the Euclidean case is achieved, i.e. all $u_{ik} \in [0,1]$. This yields *non–Euclidean relational fuzzy c–means* (NERFCM) [7]. In a similar way HCM can be advanced to RHCM and NERHCM, and PCM to RPCM and NERPCM [14].

III. Decomposite Clustering

In this paper we propose a new approach to relational clustering. The basic idea of this approach is that similar objects should have similar membership vectors $\mu_j = (u_{1j}, \ldots, u_{cj})$ and $\mu_k = (u_{1k}, \ldots, u_{ck})$, $j, k = 1, \ldots, n$. μ_j and μ_k specify fuzzy sets A_j and A_k whose similarity is defined as

$$w_{jk} = \frac{\|A_j \cap A_k\|}{\|A_j \cup A_k\|} = \frac{\sum\limits_{i=1}^{c} T(u_{ij}, u_{ik})}{\sum\limits_{i=1}^{c} C(u_{ij}, u_{ik})} \tag{12}$$

with a suitable t–norm T and t–conorm C. In this paper we choose the product t–norm and assume disjoint clusters of equal sizes, so we can set the denominators constant, which leads to the approximate (scaled) similarity

$$\tilde{w}_{jk} = \sum_{i=1}^{c} u_{ij} u_{ik} \tag{13}$$

So, the similarity of a pair of objects can be approximated by the scalar product of their membership vectors. Notice that scalar products of membership vectors (to the power of m) also occur in (9) and in *fuzzy nonlinear projection (FNP)* [13]. The scalar product of two membership vectors is high if the both objects have high memberships in the same cluster, and it is low if they have high memberships in different clusters, i.e. a pair of objects is considered similar if both objects belong to the same cluster, and dissimilar if they belong to different clusters, which matches the intuitive expectation. The matrix of the similarities $\tilde{w}_{jk}$ can be written as

$$\tilde{W} = U^T U \tag{14}$$

In relational clustering based on similarities, we have a given similarity matrix S and produce a partition matrix U. We want the matrix $\tilde{W}$ to represent the similarities, $\tilde{W} = S$, so for a given S we want to find a U that solves

$$S = \tilde{W} = U^T U \tag{15}$$

S is an $n \times n$ matrix. If we allow U to be an $n \times n$ matrix as well, and require that S is symmetric and positive definite, then we can find a lower triangular matrix U satisfying (15) by Cholesky decomposition which can be solved by Gaussian elimination in $O(\frac{1}{6}n^3)$ time [6]. In clustering, however, we are looking for a $c \times n$ not $n \times n$ partition matrix U, so Cholesky decomposition is not suitable here. Moreover, since U has $c \cdot n$ and S has n^2 free variables, we are in gereral not even able to satisfy but only approximate (15). Using a squared error criterion we therefore define the *decomposite clustering* functional

$$J = \|S - \tilde{W}\|^2 = \|S - U^T U\|^2 \tag{16}$$

A simple and straightforward approach to minimize the decomposite clustering functional (16) is gradient descent: U is randomly initialized and then iteratively updated by

$$U := U - \alpha \cdot U(S - U^T U) \tag{17}$$

with a suitable step size $\alpha > 0$. The authors have implemented and tested this approach but experienced that it required many iterations and very often got stuck in local extrema so that it had to be repeated many times with new initializations. Therefore, we pursue a different and much more efficient approach here that is motivated by *multidimensional scaling (MDS)* [21].

IV. Multidimensional Scaling

Multidimensional scaling (MDS) is a method to compute feature data $\tilde{X}$ that best approximate a given dissimilarity matrix R. Consider for example a set O of text documents with a given dissimilarity matrix R computed from word counts, i.e. the dissimilarity of a document pair is low when both documents share many common words with similar counts [9], [16]. A two dimensional feature data set $\tilde{X}$ with $\tilde{R} \approx R$ then represents the locations of each text document in O on a two–dimensional map so that similar documents are close and dissimilar documents are distant. Another application of MDS is nonlinear dimension reduction, i.e. (nonlinear) mapping of high–dimensional feature data X to low–dimensional feature data $\tilde{X}$ so that the associated dissimilarity matrices R and $\tilde{R}$ are as similar as possible.

For the case of Euclidean norms Young and Householder [21] present an approach to represent MDS as an eigen vector problem and solve it in a single step procedure. To do so, they show that the matrix $Y = XX^T$ can be written in double centered form as

$$Y = -\frac{1}{2}HDH, \quad H = I_{n\times n} - \frac{1}{n}J_{n\times n} \tag{18}$$

where $I_{n\times n}$ is the $n \times n$ identity matrix, and $J_{n\times n}$ is the $n \times n$ matrix of ones. Then, Y is diagonalized by eigen decomposition with the matrix $Q = (v_1, \ldots, v_n)$ of eigenvectors of Y and the diagonal matrix Λ whose diagonal elements are the corresponding eigenvalues of Y, $\Lambda_{ii} = \lambda_i$, $i = 1, \ldots, n$, so

$$Y = Q\Lambda Q^T = (Q\sqrt{\Lambda}^T) \cdot (\sqrt{\Lambda}Q^T) = XX^T \tag{19}$$

and $\tilde{X}$ can be finally extracted as

$$\tilde{X} = Q\sqrt{\Lambda}^T \tag{20}$$

This approach produces n dimensional feature vectors $\tilde{X} \subset \mathbb{R}^n$. For lower dimensional projections $\tilde{X} \subset \mathbb{R}^q$, $q < n$, only the first q dimensions are used and then scaled so that their squared norms are equal to the corresponding eigenvalues.

V. Clustering by Eigen Decomposition

To solve the decomposite clustering functional (16) for U we adopt the eigen decomposition approach in MDS and write (15) using the eigen decomposition of S with the matrix $Q = (v_1, \ldots, v_n)$ of eigenvectors of S and the diagonal matrix Λ whose diagonal elements are the corresponding eigenvalues of S, $\Lambda_{ii} = \lambda_i$, $i = 1, \ldots, n$.

$$S = Q\Lambda Q^T = (Q\sqrt{\Lambda}^T) \cdot (\sqrt{\Lambda}Q^T) = U^T U \tag{21}$$

which yields the membership values

$$U = \sqrt{\Lambda}Q^T \tag{22}$$

This approach produces square $n \times n$ matrices U. In clustering, we are looking for $c \times n$ matrices U, i.e. using the eigen decomposition from above we can only consider c eigenvectors and eigenvalues. In order to minimize the quadratic approximation error of the eigen decomposition we consider only the c largest eigenvalues and the corresponding eigenvectors. Thus, we still use (22) to compute U, but now with the $c \times n$ matrix

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 & \ldots & 0 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \lambda_c & 0 & \ldots & 0 \end{pmatrix} \tag{23}$$

with eigenvalues sorted in descending order. The decomposite clustering functional (16) is not only minimized by U (22) but also any rotation of U

$$U = RU, \quad RR^T = I \tag{24}$$

because

$$(RU)^T(RU) = U^T R^T RU = U^T(R^T R)U = U^T U \tag{25}$$

The solution U and its rotations are matrices in $\mathbb{R}^{c\times n}$ but in clustering we require partition matrices in $[0,1]^{c\times n}$. In particular, we want to avoid negative memberships u_{ik}. RU specifies n points in a c dimensional space. In the following we restrict to $c = 2$, so we have n points in the two–dimensional plane. Having non–negative memberships means that all points in RU should be in the first quadrant, so their angle

$$\varphi_k = \text{atan2}(u_{2k}, u_{1k}) \tag{26}$$

- input similarity matrix S
- compute eigenvectors Q and eigenvalues Λ of S
- truncate eigenvalue matrix Λ by (23)
- compute $U = \sqrt{\Lambda} Q^T$
- compute rotation matrix R by (27), (28), (26)
- output RU

Fig. 1. The decomposite clustering algorithm, $c = 2$.

should be in $[0, \pi/2]$, $k = 1, \ldots, n$. From the infinite set of solutions RU we select the solution that maps the average of the minimum and maximum angles φ_k, $k = 1, \ldots, n$, to the diagonal in the first quadrant at the angle $\pi/4$, so

$$R = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \tag{27}$$

with the angle

$$\alpha = \frac{\pi}{4} - \frac{1}{2}\big(\max_{k=1,\ldots,n} \varphi_k + \min_{k=1,\ldots,n} \varphi_k\big) \tag{28}$$

The overall decomposite clustering algorithm is summarized in Fig. 1. In contrast to the gradient descent approach mentioned before this eigen decomposition approach is very fast and *always* finds the global optimum.

VI. Experiments

In the first set of experiments we consider a real world gene expression data set for cutaneous malignant melanoma, the most common type of human cancers [4]. The authors provide gene expression profiles for 31 melanoma and 7 control samples from a microarray containing probes for 8067 cDNAs. We ignore the control samples and only consider the 31×8067 data set X_1. The objects in the melanoma are sorted by class. The first 19 objects refer to one type of melanoma that we will for simplicity call class one, and the the other 12 objects refer to another type that we will call class two. As suggested in the original paper we compute the $(19 + 12) \times (19 + 12)$ matrix of Pearson correlation coefficients and use it as the dissimilarity matrix R_1. We then normalize the dissimilarities to the unit interval $[0, 1]$ and finally convert the dissimilarities to similarities by $s_{ij} = 1 - d_{ij}$. Fig. 2 shows a grey value visualization of the resulting data set S_1. Light boxes (as on the main diagonal) represent high similarities (close to 1), and dark boxes represent low similarities (close to 0).

The melanoma data set has more features than objects, $p > n$, which is often the case in bioinformatics data, especially in microarray data [12]. For $p > n$ it is usually quite easy to build a classifier with a very good classification rate on the training data set. However, because of overfitting the generalization ability of such a classifier is usually quite bad, so building good classifiers is very difficult for $p > n$. Also clustering is often difficult for $p > n$. We used the melanoma data set S_1 as described above and applied NERFCM and the decomposite clustering algorithm proposed in this paper, both for $c = 2$. For NERFCM, the

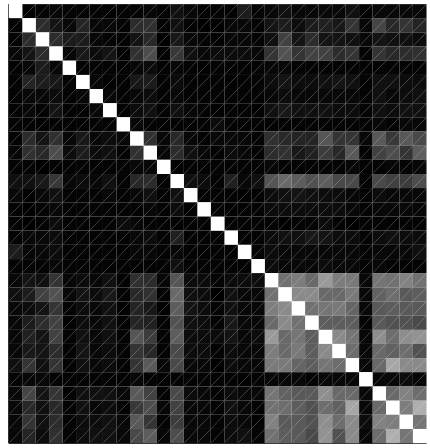
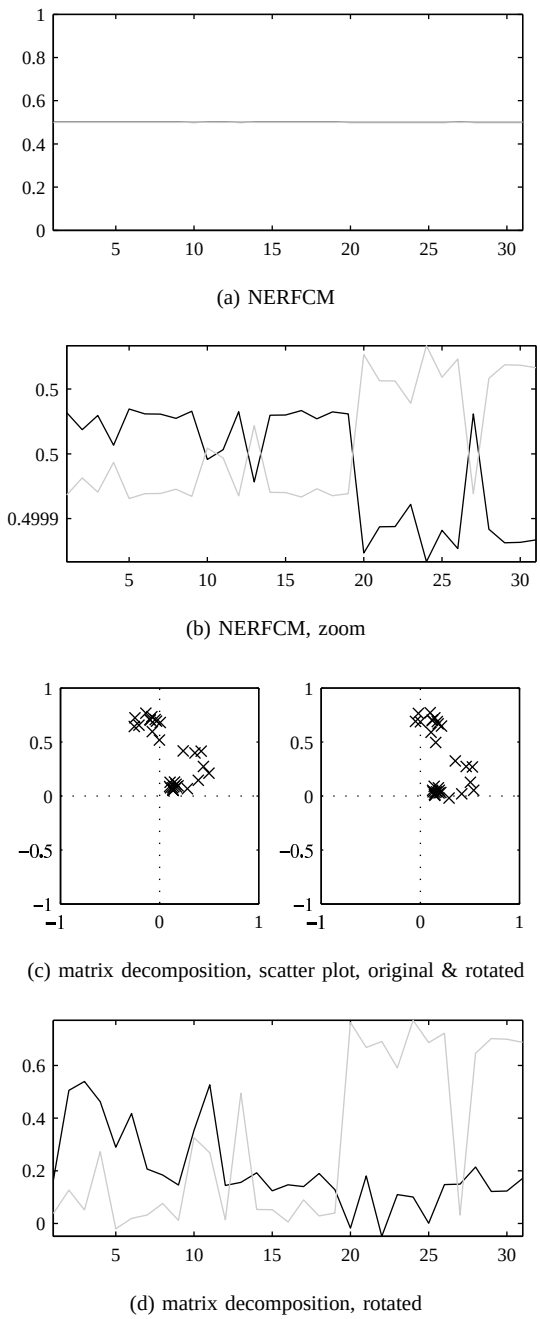


Fig. 2. The melanoma data set.

termination threshold for successive partitions is $\varepsilon = 10^{-4}$. Because of its iterative nature, NERFCM always requires significantly more time than decomposite clustering. We do not use SAHN for comparison here because it produces hard (non–fuzzy) clusters.

Fig. 3 shows the results of the experiments with the melonoma data set. The three graphs (a), (b), and (d) show the membership functions of the two clusters (black and grey) for the $19 + 12 = 31$ objects on the horizontal axis. Graph (c) shows the scatter plots of the memberships in the first cluster on the horizontal and the second cluster on the vertical axis. Fig. 3a shows the memberships obtained by NERFCM. All memberships are very close to $1/c = 0.5$, so the two membership functions can not be distinguished but appear as two overlapping horizontal lines at $u = 0.5$. This is a well known phenomenon that occurs with NERFCM when applied to very high–dimensional data $p > n$ [15]. Fig. 3b shows a zoom of these memberships revealing that one cluster (black) mainly captures class one (first 19 objects), and the other cluster (grey) class two (last 12 objects). If used as a classifier, this would imply two misclassifications in class one (objects 10 and 13), and one misclassification in class two (object 27). The left view of Fig. 3c shows the original matrix U obtained by decomposite clustering in a scatter plot, i.e. the points (u_{1k}, u_{2k}), $k = 1, \ldots, n$. As mentioned in the previous section, U is an optimal solution of (16) but so is any rotation RU. In order to obtain a partition matrix, we require positive memberships, so we rotate U (left) so that all the points RU move to the first quadrant (right). The corresponing membership function is shown in Fig. 3d. Apparently, the memberships of each object do not sum up to one, so the normalization condition of the FCM family (6) does not hold. The black cluster represents class one with only one misclassification (object 13), and the grey cluster class two with the same misclassification as NERFCM (object 27). So, besides being faster and always finding the global optimum, decomposite clustering outperforms NERFCM on the melanoma data set.



(a) NERFCM

(b) NERFCM, zoom

(c) matrix decomposition, scatter plot, original & rotated

(d) matrix decomposition, rotated

Fig. 3. Memberships for the lung cancer data set.

In the second set of experiments we consider the congressional voting records data set `house-votes-84` available from the UCI machine learning repository, originally from the Congressional Quarterly Almanac, 98th Congress, 2nd session 1984, Volume XL: Congressional Quarterly Inc. Washington, D.C., 1985, provided by J. C. Schlimmer [18]. This data set includes votes (yea or nay) for each of the U.S. House of Representatives Congressmen, 168 democrats and 267 republicans, on the 16 key votes (handicapped infants, water project cost sharing, etc.), summarized in a binary $(168 + 267) \times 16$ feature data set X_2. From this data set

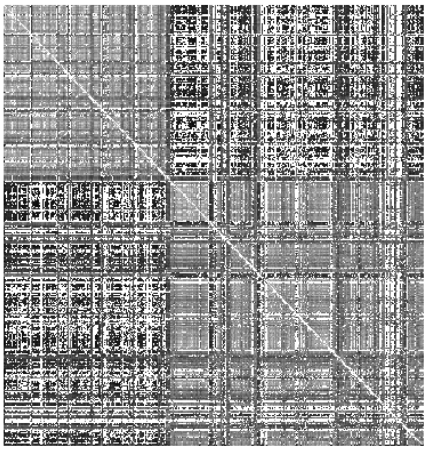


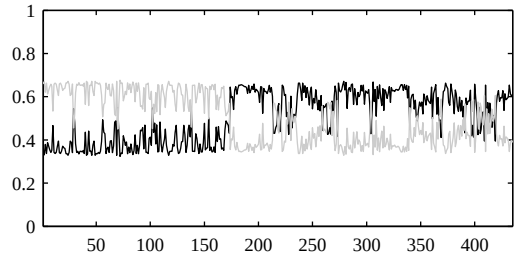
Fig. 4. The house votes data set.

we compute a $(168 + 267) \times (168 + 267)$ dissimilarity data set R_2 using the Euclidean norm. As with the melanoma data set, we normalize the dissimilarities to $[0, 1]$ and compute the similarities as $s_{ij} = 1 - d_{ij}$. Fig. 4 shows a visualization of the resulting data set S_2.

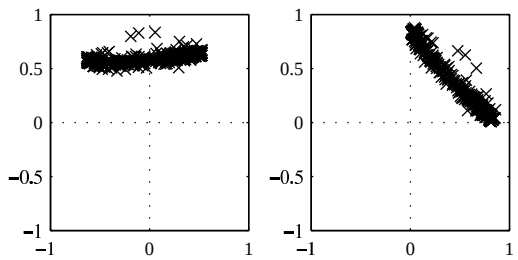
Fig. 5 shows the results of the experiments with the house votes data set. Fig. 5a shows the memberships obtained by NERFCM. Again, the memberships are close to $1/c = 0.5$ but this time both membership functions can be visually distinguished. The grey cluster represents the democrats class (first 168 objects) with 5 misclassifications (objects 29, 70, 102, 138, 154), and the black cluster represents the republicans class (last 267 objects) with 42 misclassifications (objects 1, 2, 3, 4, 5, 46, 47, 48, 49, 52, 53, 60, 61, 63, 66, 91, 96, 97, 98, 99, 101, 105, 136, 172, 179, 193, 199, 200, 201, 223, 229, 230, 231, 235, 237, 238, 241, 243, 245, 248, 250, 251). The left view of Fig. 5b shows the scatter plots of the original and rotated matrices U and RU obtained by decomposite clustering. Again, R is chosen so that all points are in the first quadrant. The corresponing membership function is shown in Fig. 5c. The grey cluster represents the democrats class also with 5 misclassifications (for the same objects as NERFCM), and the black cluster represents the republicans class with only 40 misclassifications (the same objects as NERFCM except 53 and 193). Again, decomposite clustering produces better clustering results than NERFCM, with lower computational effort and the guarantee to always find the global optimum.

VII. Conclusions

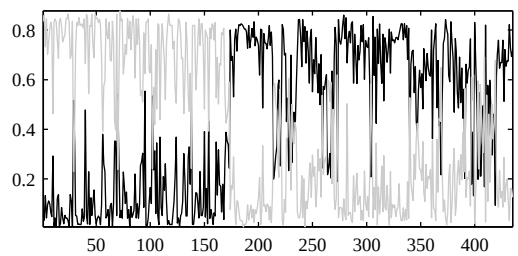
We have introduced a new approach to clustering based on the idea that similar objects should belong to the same cluster, and dissimilar objects should belong to different clusters. We have shown that the similarity of the memberships of a pair of points can be (at least approximately) computed using scalar products. We have developed the decomposite clustering functional by requiring that the similarities computed from memberships should be approximately equal



(a) NERFCM



(b) matrix decomposition, scatter plot, original & rotated



(c) matrix decomposition, rotated

Fig. 5. Memberships for the house votes data set.

to the given similarity matrix. Besides a simple gradient descent approach we have shown how the decomposite clustering functional can be optimized using eigen vector decomposition. This approach is computationally efficient and is guaranteed to find the global optimum, in contrast to conventional iterative clustering algorithms. For $c = 2$ we have shown how the resulting partition can be rotated so that all the memberships are positive. In experiments with two real world data sets (melanoma data set and house votes data set) we compared decomposite clustering with non–Euclidean relational fuzzy c–means (NERFCM) and found that decomposite clustering is not only faster and always finds the global optimum, but even finds clusters that better match the given class distributions.

Although decomposite clustering yields highly impressive results, we have to leave some aspects open that still require further studies:

- Is it possible that the matrix U does not fit into a single quadrant? How should such a matrix be treated?
- How can the rotation of U be performed for $c > 2$?
- How can the positive membership constraint be integrated into the clustering functional? This would avoid the post–rotation of the result. How can the resulting functional be solved by eigen decomposition?
- How can decomposite clustering be explicitly extended to dissimilarity matrices?
- How can prototypes be included in decomposite clustering?
- How does this novel approach relate to spectral clustering [20]?

References

[1] G. B. Ball and D. J. Hall. Isodata, an iterative method of multivariate analysis and pattern classification. In *IFIPS Congress*, 1965.

[2] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York, 1981.

[3] J. C. Bezdek and R. J. Hathaway. Clustering with relational c–means partitions from pairwise distance data. *International Journal of Mathematical Modelling*, 8:435–439, 1987.

[4] M. Bittner, P. Meltzer, Y. Chen, Y. Jiang, E. Seftor, M. Hendrix, M. Radmacher, R. Simon, Z. Yakhini, A. Ben-Dor, N. Sampas, E. Dougherty, E. Wang, F. Marincola, C. Gooden, J. Lueders, A. Glatfelter, P. Pollock, J. Carpten, E. Gillanders, D. Leja, K. Dietrich, C. Beaudry, M. Berens, D. Alberts, V. Sondak, N. Hayward, and J. Trent. Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature*, 406:536–538, August 2000.

[5] C. Ding and X. He. K–means clustering via principal component analysis. In *International Conference on Machine Learning*, pages 225–232, 2004.

[6] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, 3rd edition, 1996.

[7] R. J. Hathaway and J. C. Bezdek. NERF c–means: Non–Euclidean relational fuzzy clustering. *Pattern Recognition*, 27:429–437, 1994.

[8] R. J. Hathaway and J. C. Bezdek. Optimization of clustering criteria by reformulation. *IEEE Transactions on Fuzzy Systems*, 3(2):241–245, May 1995.

[9] K. Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.

[10] R. Krishnapuram and J. M. Keller. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1(2):98–110, May 1993.

[11] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.

[12] M. Popescu, J. M. Keller, and J. Mitchell. Fuzzy measures on the gene ontology for gene product similarity. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 3(3):263–274, July 2006.

[13] T. A. Runkler. Fuzzy nonlinear projection. In *IEEE International Conference on Fuzzy Systems*, St. Louis, May 2003.

[14] T. A. Runkler. Kernelized non–euclidean relational possibilistic c–means clustering. In *IEEE Three Rivers Workshop on Soft Computing in Industrial Applications*, Passau, August 2007.

[15] T. A. Runkler. Pareto optimality of cluster objective and validity functions. In *IEEE International Conference on Fuzzy Systems*, pages 79–84, London, July 2007.

[16] T. A. Runkler and J. C. Bezdek. Automatic keyword extraction with relational clustering and Levenshtein distances. In *IEEE International Conference on Fuzzy Systems*, pages 636–640, San Antonio, May 2000.

[17] T. A. Runkler, E. Gerstorfer, M. Schlang, E. Jünnemann, and J. Hollatz. Modeling and optimization of a refining process for fiber board production. *IFAC Control Engineering Practice*, 11(11):1229–1241, 2003.

[18] J. C. Schlimmer. *Concept acquisition through representational adjustment*. PhD thesis, Department of Information and Computer Science, University of California, Irvine, CA, 1987.

[19] P. Sneath and R. Sokal. *Numerical Taxonomy*. Freeman, San Francisco, 1973.

[20] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 4(17):395–416, 2007.

[21] G. Young and A. S. Householder. Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3:19–22, 1938.