Optimal Cluster Preserving Embedding of Nonmetric Proximity Data

Volker Roth, Julian Laub, Motoaki Kawanabe, and Joachim M. Buhmann, *Member*, *IEEE*

Abstract—For several major applications of data analysis, objects are often not represented as feature vectors in a vector space, but rather by a matrix gathering pairwise proximities. Such pairwise data often violates metricity and, therefore, cannot be naturally embedded in a vector space. Concerning the problem of unsupervised structure detection or *clustering*, in this paper, a new embedding method for pairwise data into Euclidean vector spaces is introduced. We show that all clustering methods, which are invariant under additive shifts of the pairwise proximities, can be reformulated as grouping problems in Euclidian spaces. The most prominent property of this *constant shift embedding* framework is the complete *preservation of the cluster structure* in the embedding space. Restating pairwise clustering problems in vector spaces has several important consequences, such as the statistical description of the clusters by way of *cluster prototypes*, the generic extension of the grouping procedure to a discriminative *prediction rule*, and the applicability of standard *preprocessing methods* like denoising or dimensionality reduction.

Index Terms—Clustering, pairwise proximity data, cost function, embedding, MDS.

1 Introduction

Modern data mining poses several major challenges to experimental scientists. Beside the inherent difficulty of interpretation and validation of, e.g., unsupervised methods, data arises in a variety of forms which require appropriate treatment. For several major applications, data is often not available as feature vectors in a vector space. For instance, genomics typically produce data represented as strings from some alphabet, psychology yields sets of similarity judgments, yet other fields like social sciences measure so-called preference data. The missing vector space representation precludes the use of well established clustering or classification techniques such as Principal Component Analysis [1] or Support Vector Machines [2].

Nonvectorial data sets as such are difficult to handle and, for data mining purposes, we need to relate them to some mathematical concept. A common approach is to replace the initial data by a collection of real numbers representing some "comparison" among the elements of the data set. This can be straightforward, as for similarity judgments, or highly nontrivial as for string data, where the similarity score may be derived by a complex alignment algorithm. This procedure yields a matrix gathering the pairwise relations between the original objects, which may be the starting point of intelligent data analysis, see, e.g., [3] for an example of such a procedure in the field of image retrieval. We like to stress here that such a matrix is by no means naturally related to the common viewpoint of objects being embedded in some "well-behaved" space with a vector space structure. In particular, for pairwise data, there is no well-established denoising method. In applications like string matching, however, noise reduction is an important issue. Many alignment algorithms produce noisy data, which, when fed to some clustering algorithm, typically yield poor results.

In this contribution, we therefore study properties of embedding strategies in the context of clustering. We will proceed as follows: We begin with a short overview of proximity-based data grouping and, then, we focus on reformulating such problems with vectorial data representations. For the class of pairwise clustering methods that are related to minimizing a shift-invariant cost function, our main contribution is a novel embedding strategy, which we call *constant shift embedding*. A surprising property of this embedding is the complete preservation of group structure. The original nonmetric pairwise clustering problem can be restated as a grouping problem over points in a vector space, yielding identical assignments of objects to clusters. Using the constant-shift embedding principle, we then demonstrate the equivalence between the *pairwise clustering* cost function and the classical k-means grouping criterion in the embedding space.

2 Proximity-Based Clustering

Unsupervised grouping or *clustering* aims at extracting hidden structure from data [4]. The term data refers to both a set of objects and a set of corresponding object representations resulting from some physical measurement process. Different types of object representations are possible, the two most common of which are *vectorial data* and *pairwise proximity data*. In the first case, a set of n objects is represented as n points in a d-dimensional vector space, whereas in the second case, we are given a $n \times n$ pairwise proximity matrix.

The problem of grouping vectorial data has been widely studied in the literature, and many clustering algorithms have been proposed [4], [5]. One of the most popular methods is k-means clustering. It derives a set of k prototype

- *V. Roth and J.M. Buhmann are with the Department of Computer Science III, University of Bonn, Roemerstr. 164, D-53117 Bonn, Germany. E-mail: {roth, jb}@cs.uni-bonn.de.*
- *J. Laub and M. Kawanabe are with Fraunhofer FIRST, Kekulestrasse 7, 12489 Berlin, Germany. E-mail: {julian.laub, nabe}@first.fhg.de.*

Manuscript received 20 Feb. 2003; revised 27 June 2003; accepted 3 July 2003. Recommended for acceptance by E. Hancock.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 118319.

vectors which quantize the data set with minimal quantization error.

Partitioning proximity data is considered a much harder problem since the inherent structure is hidden in n^2 pairwise relations. This datatype, however, is abundant in many applications such as molecular biology, psychology, linguistics, etc. In general, the proximities can violate the requirements of a distance measure, i.e., they may be nonsymmetric and negative and the triangle inequality does not necessarily hold. Thus, a loss-free embedding into a vector space is not possible, so that grouping problems of this kind cannot directly be transformed into vectorial problems by means of classical embedding strategies.

Presumingly, the most popular classical embedding method for nonmetric data is *Multidimensional Scaling* (MDS) (see, e.g., [6] for a recent overview), where one seeks a low-dimensional representation of data such that the distortion of the pairwise dissimilarities D_{ij} is minimal with respect to some cost function. One widely used cost function is the SSTRESS criterion, [7]:

$$J = \sum_{i,j=1}^{n} \omega_{ij} \left(d_{ij}^2 - D_{ij}^2\right)^2, \tag{1}$$

where $d_{ij} = \|x_i - x_j\|$ are the transformed distances in low-dimensional space, and ω_{ij} are weights. Typically, these weights read:

$$\omega_{ij} = \frac{1}{n(n-1)D_{ij}^2}, \quad \omega_{ij} = \frac{1}{\sum_{k,l} D_{kl}^2}, \text{ or } \omega_{ij} = \frac{1}{D_{ij} \sum_{k,l} D_{kl}}. \tag{2}$$

The choice in (2) relates to the minimization of relative, absolute, or intermediate error (see, e.g., [4]).

The problem with this MDS approach, however, is that clustering the embedded data-vectors, in general, yields partitionings *different* from those obtained by directly solving the pairwise problem. Even worse, by guaranteeing low (but nonzero) distortions of the proximities, it is still unclear how the object assignments are affected by the embedding.

Among several methods for clustering proximity-based data, in the following, we will focus on those techniques that explicitly minimize a certain cost function. This subset of clustering methods includes, e.g., graph-theoretic approaches like several variations of *Cut* criteria [8], and several methods derived from an axiomatization of pairwise cost functions in [9]. From a theoretical viewpoint, cost-based clustering methods are interesting insofar as many properties of the grouping solutions can be derived by analyzing invariance properties of the cost function.

Among the class of cost-based criteria, the main focus of this work concerns those cost functions which are invariant under constant additive shifts of the pairwise dissimilarities. For this subset of clustering criteria, we show that there always exists a set of vectorial data representations such that the grouping problem can be equivalently restated in terms of Euclidian distances between these vectors. A special cost function of this kind is the *pairwise clustering cost function*. It is of particular interest since it combines the properties of additivity, scale and shift invariance, and statistical robustness, see [9]. In [10], this grouping problem is stated as a combinatorial optimization problem which is optimized in a *deterministic annealing* framework after applying a mean-field approximation.

According to the Theorem 3, we can always find a vectorial data representation such that the optimal partitioning with respect to the pairwise cost function is *identical* to k-means partitioning in the embedding space. This property demonstrates that the embedding method is optimal with respect to to distortions of the *data partition*. This distortion preserving embedding has to be contrasted with alternative, in our view not consistent, approaches that are optimal with respect to some a priori chosen MDS distortion measure.

Formulating pairwise clustering as a k-means problem yields several advantages, both of theoretical and technical nature: 1) the availability of prototype vectors defines a generic rule for using the learned partitioning in a predictive sense, 2) we can apply standard noise- and dimensionality-reduction methods in order to separate the "signal" part of the data from underlying "noise," and 3) fast and efficient local search heuristics for optimizing the clustering cost functional often work much better in low-dimensional embedding spaces.

2.1 The Pairwise Clustering Cost Function

The modeling idea behind the Pairwise Clustering cost function is to minimize the sum of *pairwise* intracluster distances, emphasizing *compact* clusters. Optimizing a compactness criterion is certainly a very intuitive meta-principle for exploratory data analysis. It should be noticed, however, that other such metaprinciples have been proposed, such as *separation* measures, mixed *compactness/separation* measures, or *connectivity* measures. We will discuss the relation of Pairwise Clustering to some of these methods in Section 5.

In order to formalize Pairwise Clustering, we define for each object a binary assignment variable that indicates its cluster membership. Let these variables be summarized in the $(n \times k)$ binary stochastic assignment matrix $M \in \{0,1\}^{n \times k} : \sum_{\nu=1}^{k} M_{i\nu} = 1$. Given a $(n \times n)$ dissimilarity matrix D, the Pairwise Clustering cost function reads:

$$H^{\text{pc}} = \frac{1}{2} \sum_{\nu=1}^{k} \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} M_{i\nu} M_{j\nu} D_{ij}}{\sum_{l=1}^{n} M_{l\nu}}. \tag{3}$$

The optimal assignments $\hat{M}$ are obtained by minimizing H^{pc}. The minimization itself is an $\mathcal{NP}$ hard problem [11], and some approximation heuristics have been proposed: In [10], a *mean field annealing* framework has been presented (see the discussion in Section 4 of this work for some comments and new results on annealing). In [9], it has been shown that the time-honored *Ward's method* can be viewed as a hierarchical approximation of H^{pc}.

2.2 A Special Case: k-Means Clustering

For the special case of squared Euclidean distances between vectors $\{x_i\}_{i=1}^{n}, \quad x_i \in \mathbb{R}^d$, it is well-known that H^{pc} is identical to the classical k-means cost function, see [4]. We now briefly review this relationship. The k-means cost function is defined as

$$H^{\mathrm{km}} = \sum_{\nu=1}^{k}\sum_{i=1}^{n} M_{i\nu}\|x_i - y_\nu\|^2. \tag{4}$$

It measures the sum of squared intracluster distances to the prototype vectors

$$y_\nu := \frac{\sum_{i=1}^{n} M_{i\nu}x_i}{n_\nu}, \tag{5}$$

where $n_\nu := \sum_{l=1}^{n} M_{l\nu}$ denotes the number of objects in cluster ν. H^{km} can be written in a pairwise fashion by exploiting a simple algebraic identity for squared Euclidian distances:

$$\begin{aligned}\|x_i - y_\nu\|^2 &= \frac{1}{n_\nu}\sum_{j=1}^{n} M_{j\nu}\|x_i - x_j\|^2 - \frac{1}{2n_\nu^2}\sum_{j=1}^{n}\sum_{l=1}^{n} \\ &\qquad M_{j\nu}M_{l\nu}\|x_j - x_l\|^2, \\ \sum_{i=1}^{n} M_{i\nu}\|x_i - y_\nu\|^2 &= \frac{1}{2n_\nu}\sum_{j=1}^{n}\sum_{l=1}^{n} M_{j\nu}M_{l\nu}\|x_j - x_l\|^2.\end{aligned} \tag{6}$$

Substituting the latter identity into (4), we obtain

$$H^{\mathrm{km}} = \frac{1}{2}\sum_{\nu=1}^{k}\frac{\sum_{i=1}^{n}\sum_{j=1}^{n} M_{i\nu}M_{j\nu}\|x_i - x_j\|^2}{\sum_{l=1}^{n} M_{l\nu}} = H^{\mathrm{pc}}. \tag{7}$$

From this viewpoint, k-means clustering can be interpreted as a method for minimizing the sum of squared *pairwise* intracluster distances $D_{ij} = \|x_i - x_j\|^2$. The reader should notice, however, that in the general case of arbitrary dissimilarities D_{ij}, a direct algebraic retransformation of H^{pc} into H^{km} is *not* possible. Despite this fact, we will show in the remainder of this paper that it is still possible to obtain the optimal assignment variables $\hat{M}$ with respect to $H^{\mathrm{pc}}(M)$ by minimizing a suitably transformed k-means problem. The key ingredient will be the *shift invariance property* of the Pairwise Clustering cost function described in the following section.

2.3 Invariances of Pairwise Clustering

The pairwise clustering cost function has two important invariance properties:

1. H^{pc} is invariant under symmetrizing transformations

$$\tilde{D}_{ij} = \frac{1}{2}(D_{ij} + D_{ji}) \quad \Rightarrow \quad \tilde{H} = H. \tag{8}$$

2. H^{pc} is invariant (up to a constant) under additive shifts of the *off-diagonal* elements of the dissimilarity matrix:

$$\begin{aligned}&\tilde{D}_{ij} = D_{ij} + d_0(1 - \delta_{ij}) \quad \Rightarrow \\ &\tilde{H} = H + (1/2)\cdot(n-k)d_0 = H + \mathrm{const}.\end{aligned} \tag{9}$$

Note that the optimal assignments of objects to clusters are not influenced by adding a constant to the cost function, i.e., $\hat{M}(\tilde{D}) = \hat{M}(D)$.

3 Constant Shift Embedding

In Section 2, we have introduced the cost function H^{pc} as a special instance of pairwise clustering problems. Due to the shift-invariance property (9), the partitioning of the data set (i.e., the assignments of a set of n objects to k clusters) is not affected by a constant additive shift on the off-diagonal elements of the pairwise dissimilarity matrix $D = (D_{ij}) \in \mathbb{R}^{n\times n}$. In the remainder of this paper, we will consider general dissimilarity matrices D, restricted only by the constraint that all self-dissimilarities are zero, i.e., that D has zero diagonal elements. We show that, by exploiting the above shift invariance, we can always embed such data into a Euclidean space without influencing the cluster structure. An off-diagonal shifted dissimilarity matrix reads

$$\tilde{D} = D + d_o\left(e_n e_n^\top - I_n\right), \tag{10}$$

where $e_n = (1, 1, \ldots 1)^\top$ is an n-vector of ones and I_n the $n \times n$ identity matrix. In other words, (10) describes a constant additive shift $\tilde{D}_{ij} = D_{ij} + d_o \quad \text{for all } i \neq j$.

Before developing the main theory, we have to introduce the notion of a *centralized matrix*. Let P be an $(n \times n)$ matrix and let $Q = I_n - \frac{1}{n}e_n e_n^\top$. Q is the projection matrix on the orthogonal complement of e_n. Define the *centralized* P by:

$$P^c = QPQ. \tag{11}$$

A centralized matrix has row and column-sum equal to zero, which can easily be seen by looking at the components of P^c

$$P^c_{ij} = P_{ij} - \frac{1}{n}\sum_{k=1}^{n} P_{ik} - \frac{1}{n}\sum_{k=1}^{n} P_{kj} + \frac{1}{n^2}\sum_{k,l=1}^{n} P_{kl}. \tag{12}$$

Let us now consider only *symmetric* dissimilarity matrices. Note that, for the clustering criterion H^{pc}, this requirement imposes no restrictions on the general applicability, since H^{pc} is invariant under symmetrizing transformations, see (8). Given such a symmetric and zero-diagonal matrix D, let us decompose it the following way by introducing a new matrix S:

$$D_{ij} = S_{ii} + S_{jj} - 2S_{ij}. \tag{13}$$

It is clear that this decomposition is not unique unless we specify the diagonal elements of S. Let $\mathbb{S}_D$ denote the equivalence class of all S yielding the same D. In particular, we note, by simple algebra, that for every matrix $S \in \mathbb{S}_D$, the centered version is contained in $\mathbb{S}_D$. Moreover, the following lemma states that for all members $S \in \mathbb{S}_D$, the centralized version S^c is identical and uniquely defined by the given matrix D.

Lemma 1. *For any symmetric and zero-diagonal matrix D, the following holds:*

$$S^c = -\frac{1}{2}D^c, \text{ with } D^c = QDQ.$$

Proof. Substituting (13) into the definition (12) of the centralized S^c yields

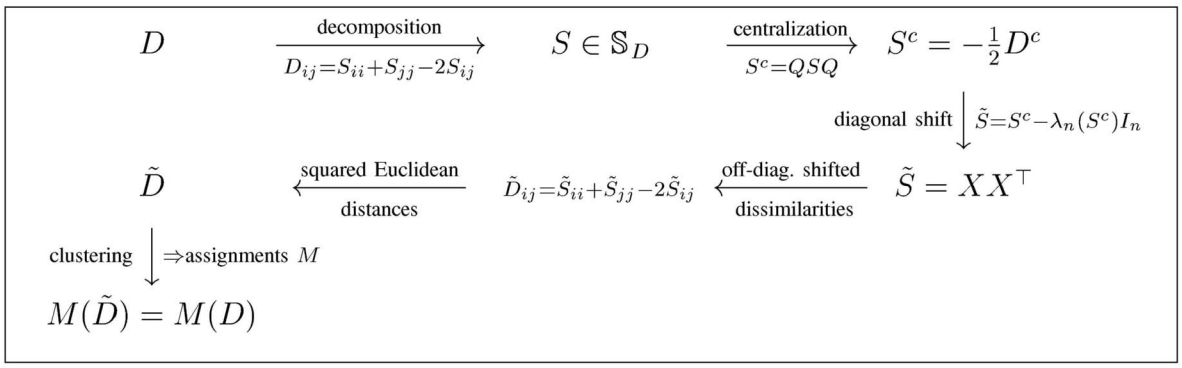

Fig. 1. Process flow of constant shift embedding (schematic).

$$S^c_{ij} = -\frac{1}{2}\Bigg[(D_{ij} - S_{ii} - S_{jj}) - \frac{1}{n}\sum_{k=1}^{n}(D_{ik} - S_{ii} - S_{kk}) - \frac{1}{n}\sum_{k=1}^{n}(D_{kj} - S_{kk} - S_{jj}) + \frac{1}{n^2}\sum_{k,l=1}^{n}(D_{kl} - S_{kk} - S_{ll})\Bigg]$$

$$= -\frac{1}{2}\left[D_{ij} - \frac{1}{n}\sum_{k=1}^{n} D_{ik} - \frac{1}{n}\sum_{k=1}^{n} D_{kj} + \frac{1}{n^2}\sum_{k,l=1}^{n} D_{kl}\right] = -\frac{1}{2}D^c_{ij}.$$

□

The matrix S^c is a particularly interesting member of $\mathbb{S}_D$, since the following theorem holds.

Theorem 1. *D derives from a squared Euclidian distance, i.e., $D_{ij} = \|x_i - x_j\|^2$, if and only if S^c is positive semidefinite.*

Proof. [12] referring to [13]. □

For general dissimilarities, S^c will be indefinite. By shifting its diagonal elements, however, we can transform it into a positive semidefinite matrix: The following lemma states that, for any matrix A, a positive semidefinite matrix $\tilde{A}$ can be derived by subtracting the smallest eigenvalue from all of its diagonal elements.

Lemma 2. *Let $\tilde{A} = A - \lambda_n(A)I_n$, where $\lambda_n(\cdot)$ is the minimal eigenvalue of its argument. Then, $\tilde{A}$ is positive semidefinite.*

Proof. Due to the diagonal shift, the smallest eigenvalue becomes zero. □

We can now summarize the above results, cf. Fig. 1: Given a matrix D, there exists a unique matrix S^c by Lemma 1. If S^c is not positive semidefinite, Lemma 2 states that, by subtracting $\lambda_n(S^c)$ from its diagonal elements, we obtain a positive semidefinite $\tilde{S}$. Returning to (13) with our fixed matrix S^c, such a diagonal shift of S^c corresponds to an *off-diagonal* shift of the dissimilarities

$$\tilde{D}_{ij} = \tilde{S}_{ii} + \tilde{S}_{jj} - 2\tilde{S}_{ij} \;\Leftrightarrow\; \tilde{D} = D - 2\lambda_n(S^c)\big(e_n e_n^\top - I_n\big). \tag{15}$$

In other words, if we were given $\tilde{D}$ instead of our original D, then $\tilde{S}$ would be a positive semidefinite member of the equivalence class $\mathbb{S}_{\tilde{D}}$ of matrices fulfilling the decomposition $\tilde{D}_{ij} = \tilde{S}_{ii} + \tilde{S}_{jj} - 2\tilde{S}_{ij}$. Theorem 1 then tells us that this off-diagonally shifted matrix $\tilde{D}$ derives from a squared Euclidean distance. Since every positive semidefinite matrix is a dot product—(or *gram*)—matrix in some vector space, there exists a matrix X of vectors such that $\tilde{S} = XX^\top$. The matrix $\tilde{D}$ then contains squared Euclidean distances between these vectors. We can now insert $\tilde{D}$ into our clustering procedure (which is assumed shift-invariant), and we will obtain the same partition of the objects as if we had clustered the original matrix D. Contrary to directly using D, however, the matrix $\tilde{D}$ now contains squared Euclidean distances between a set of vectors $\{x_i\}_{i=1}^n$. So far, we have only shown the existence of these vectors. In Section 3.1, we will present how these vectors can be computed explicitly.

In principle, the above derivation holds true not only for the centralized matrix S^c, but for any member S of the of the equivalence class $\mathbb{S}_D$. Some of these members, however, will eventually have very large negative eigenvalues, which means that we would have to add a very large constant to all off-diagonal entries of D. For numerical reasons, we want to avoid these problems, which leads us to the question of the *minimal* necessary shift. The next theorem states that our above choice of using S^c is optimal in this sense:

Theorem 2 (Minimal shift) [6]. *$D_o = -2\lambda_n(S^c)$ is the minimal constant such that $\tilde{D} = D + D_o\big(e_n e_n^\top - I_n\big)$ derive from squared Euclidian distance.*

Proof. A proof is given in [6]. It also follows from Theorem 1 and Lemma 2. □

3.1 Reconstructing the Embedded Vectors

Given a general dissimilarity matrix D, in the last section, we have shown how to obtain a shifted matrix $\tilde{D}$ which derives from squared Euclidean distances between points $\{x_i\}_{i=1}^n$ in some vector space. This property of $\tilde{D}$ impies that the corresponding matrix $\tilde{S}^c$ is positive semidefinite and, thus, a dot product matrix $\tilde{S}^c = XX^\top$. According to Lemma 1, $\tilde{S}^c$ can be calculated as $\tilde{S}^c = -\frac{1}{2}\tilde{D}^c$. The following algorithm[1] describes how the vectors $\{x_i\}_{i=1}^n$ can be recovered by an eigenvalue decomposition of $\tilde{S}^c$.

1. Calculate the centralized dot product matrix $\tilde{S}^c = -\frac{1}{2}Q\tilde{D}Q$ from the matrix of squared Euclidean distances $\tilde{D}$.

1. This algorithm is also known as *kernel PCA*, see [14].

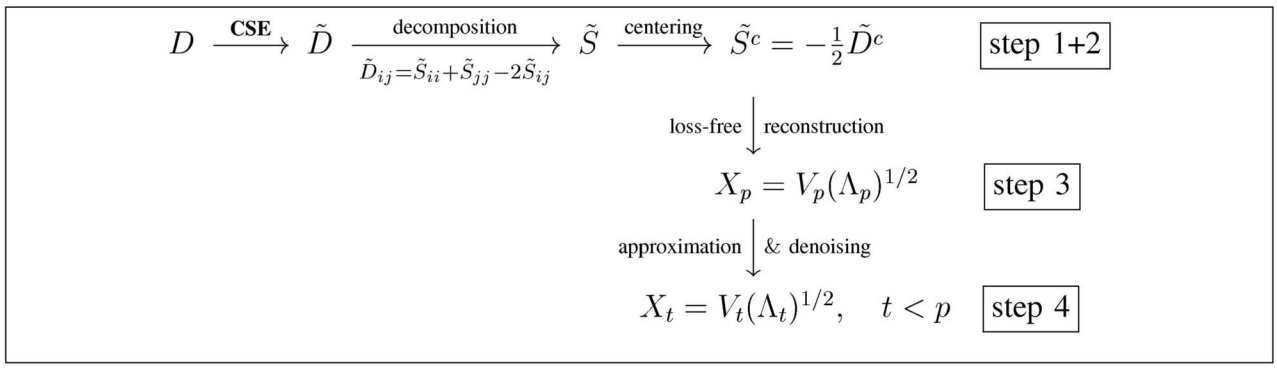

Fig. 2. Reconstructing the vector representation (schematic): constant shift embedding (CSE) & computing the centered dot product matrix $\tilde{S}^c$ (step 1 + 2). Loss-free reconstruction of the vectors x_i by projecting on the eigenvectors of $\tilde{S}^c$ (step 3). PCA-approximation & denoising (step 4, optional).

2. Express $\tilde{S}^c$ in its eigenbasis: $\tilde{S}^c = V\Lambda V^\top$, where $V = (v_1, \dots v_n)$ contains the eigenvectors v_i and $\Lambda = \text{diag}(\lambda_1, \dots \lambda_n)$ is a diagonal matrix of eigenvalues $\lambda_1 \geq \cdots \geq \lambda_p > \lambda_{p+1} = 0 = \cdots = \lambda_n$. Notice that, due to the centralization which introduces a linear dependency between all vectors, at least one eigenvalue equals zero, i.e., $p \leq n-1$.
3. Calculate the $n \times p$ map matrix

$$X_p = V_p(\Lambda_p)^{1/2}, \text{ with } V_p = (v_1, \dots v_p) \text{ and } \Lambda_p = \text{diag}(\lambda_1, \dots \lambda_p). \tag{16}$$

The rows of X_p contain the vectors $\{x_i\}_{i=1}^n$ in p dimensional space, whose mutual distances are given by $\tilde{D}$.

So far, we have discussed an exact reconstruction of the structure preserving vectors in the embedding space. While this has both important theoretical and practical consequences (see Section 4), in many applications, we would like to insert some preprocessing step in our clustering procedure. A typical example of this kind would be the suppression of noise. When focusing on noise reduction, we are interested in some sort of approximative reconstructions of the exact vectors. The reader should notice that, given the vectorial representations $\{x_i\}_{i=1}^n$ in a Euclidean space, the issue of separating the "noisy" part of the data from the "signal" part can be handled within a well-defined framework. On the contrary, in the original pairwise setting without a common vector-space structure, to our knowledge, there exist no general purpose denoising methods. For instance, it is not clear how to define a global noise model that specifies the amount of noise by which each single object is corrupted. The semantics of a generative model which is responsible for the "signal" part is also unclear.

In Principal Component Analysis (PCA), one usually assumes that the directions corresponding to small eigenvalues contain the noise [15]. We can thus obtain a representation in a space of reduced dimension (with the well-defined error of PCA reconstruction) when choosing $t < p$ dimensions in step 3 of the above algorithm: $X_t = V_t(\Lambda_t)^{1/2}$, where V_t consists of the first t column vectors of V and Λ_t is the top $t \times t$ submatrix of Λ. The vectors in $\mathbb{R}^t$ then differ the least from the vectors in $\mathbb{R}^p$ in the sense of quadratic error. This means that the embedded vectors are the best least squares error approximation to the optimal vectors which preserve the group structure. The mathematical tractability of error constitutes the main difference to directly decomposing S^c (i.e., without shifting) and projecting onto a subset of eigenvectors with positive eigenvalue, a method which is usually called *classical scaling* or "*lossy*" *PCA*. In the latter case, there exist no optimal vectors (in the sense of structure preservation), since only the positive eigenvalues can be used for deriving a vector representation. For classical scaling, it is thus unclear what "objects" are approximated and with what error. The processing pipeline of both the loss-free vector reconstruction and the PCA approximation is depicted in Fig. 2.

It should be noticed, however, that given the exactly reconstructed vectors in $\mathbb{R}^p$, we can also apply any other standard method for dimensionality reduction or visualization, such as *projection pursuit* [16], *locally linear embedding* (LLE) [17], *Isomap* [18], or *Selforganizing maps* [19]. The latter methods can also be viewed as approximations of the optimal structure preserving vectors, employing, however, an approximation criterion different from the squared error as in the case of the above PCA framework.

3.2 Predicting the Cluster Membership of New Data

First, notice that, due to the eigenvalue equation $\tilde{S}^c V_p = V_p \Lambda_p$, we can rewrite (16) in the form:

$$X_p = \tilde{S}^c V_p (\Lambda_p)^{-1/2}. \tag{17}$$

Consider now the situation where we are given m new objects and the corresponding $m \times n$ matrix of pairwise dissimilarities D_{ij}^{new} between these new objects and all n original objects. In order to predict the cluster membership of the new objects, we first have to project them into the Euclidean space spanned by the eigenvectors V_p of the centered dot product matrix $\tilde{S}^c$. Then, we assign each new object to the cluster with the closest centroid. For the projection itself, two steps are required:

1. Compute the matrix S_{new} defined by

$$D_{ij}^{\text{new}} = S_{ii}^{\text{new}} + \tilde{S}_{jj}^c - 2S_{ij}^{\text{new}}. \tag{18}$$

Similar to the situation in (13), we still have the problem of ambiguities due to the freedom of

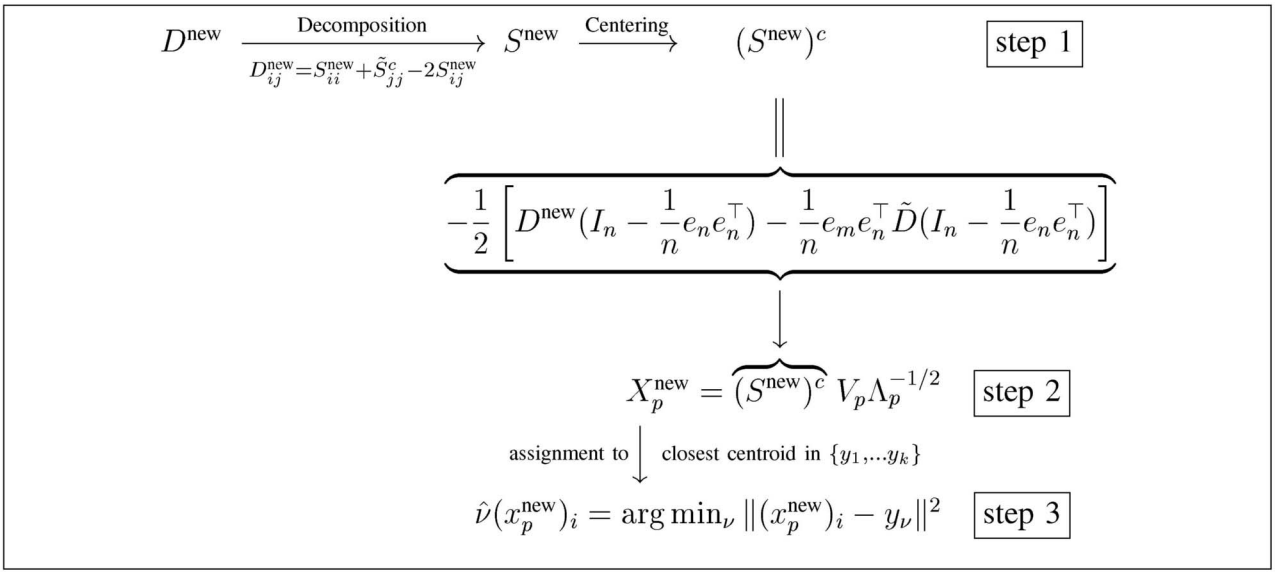

Fig. 3. Prediction (schematic): From the preceeding clustering step, we are given the squared Euclidean distances $\tilde{D}$, the centered dot-product matrix $\tilde{S}^c=-\frac{1}{2}\tilde{D}^c$, its eigenvectors & -values V_p, Λ_p, and the cluster centroids $\{y_\nu\}_{\nu=1}^k$. Prediction step 1: decomposing D^{new} and reexpressing the matrix S^{new} in the centered coordinate system of $\tilde{S}^c$. Step 2: projecting the new objects on the eigenvectors V_p of $\tilde{S}^c$. Step 3: assigning objects to the cluster with the closest centroid vector y_ν.

choosing S_{ii}^{new}. This problem, however, is automatically overcome by reexpressing the matrix S^{new} in the centered coordinate system:[2]

$$(S^{\text{new}})_{ij}^c = S_{ij}^{\text{new}} - \frac{1}{n}\sum_{k=1}^n S_{ik}^{\text{new}} - \frac{1}{n}\sum_{k=1}^n \tilde{S}_{kj}^c + \frac{1}{n^2}\sum_{k,l=1}^n \tilde{S}_{kl}^c. \tag{19}$$

Substituting (18) into the above equation and noticing that $\tilde{D}$ and $\tilde{S}^c$ are connected by $\tilde{D}_{ij}=\tilde{S}_{ii}^c+\tilde{S}_{jj}^c-2\tilde{S}_{ij}^c$, we can restate $(S^{\text{new}})^c$ solely in terms of D^{new} and $\tilde{D}$:

$$(S^{\text{new}})_{ij}^c = -\frac{1}{2}\left[D_{ij}^{\text{new}} - \frac{1}{n}\sum_{k=1}^n D_{ik}^{\text{new}} - \frac{1}{n}\sum_{k=1}^n \tilde{D}_{kj} + \frac{1}{n^2}\sum_{k,l=1}^n \tilde{D}_{kl}\right] \tag{20}$$

$$\Leftrightarrow \quad (S^{\text{new}})^c = -\frac{1}{2}\left[D^{\text{new}}(I_n-\frac{1}{n}e_ne_n^\top) - \frac{1}{n}e_me_n^\top \tilde{D}(I_n-\frac{1}{n}e_ne_n^\top)\right]. \tag{21}$$

2. Project the objects represented by $(S^{\text{new}})^c$ into the coordinate system spanned by the eigenvectors V_p of the matrix $\tilde{S}^c$:

$$X_p^{\text{new}} = (S^{\text{new}})^c V_p(\Lambda_p)^{-1/2}. \tag{22}$$

The whole process flow for predicting the cluster membership of new objects is summarized in Fig. 3.

2. Formally, this is the same centering mechanism for new objects as in the *kernel PCA* algorithm, see [14].

4 A k-Means Formulation for Pairwise Clustering

It is well-known that, for the special case of squared Euclidean distances, the Pairwise cost function and the k-means cost function can be transformed into each other by using a simple algebraic identity, cf. Section 2.2. With the results of the last section, we are now able to prove that a similar relationship between both cost functions holds in the general setting.

Theorem 3. *Given an arbitrary $(n\times n)$ dissimilarity matrix D with zero self-dissimilarities, there exists a transformed matrix $\tilde{D}$ such that*

1. *The matrix $\tilde{D}$ can be interpreted as a matrix of squared Euclidian distances between a set of vectors $\{x_i\}_{i=1}^n$ with dimensionality $\dim(x_i)\le n-1$.*
2. *The original pairwise clustering problem defined by the cost function $H^{\text{pc}}(D)$ is equivalent to the k-means problem with cost function H^{km} in this vector space, i.e., the optimal cluster assignment variables $\hat{M}_{i\nu}$ are identical in both problems: $\hat{M}^{\text{pc}}(D)=\hat{M}^{\text{km}}(\tilde{D})$.*

Proof.

1. Let $\tilde{D}$ be the symmetrized and off-diagonal shifted version of D:

$$D_{sym} := \frac{1}{2}(D+D^\top) \tag{23}$$

$$S^c := \ = -\frac{1}{2}QD_{sym}Q = -\frac{1}{2}D_{sym}^c\,, \text{ (cf. Lemma 1)} \tag{24}$$

$$\tilde{D} := D_{sym} - 2\lambda_n(S^c)\big(e_ne_n^\top - I_n\big)\,, \quad \text{(cf. (15))}. \tag{25}$$

According to Section 3 and the theorems therein, there exists a set of vectors $\{x_i\}_{i=1}^n$ with dimensionality $\dim(x_i)\le n-1$ such that $\tilde{D}$ contains squared Euclidean distances between these vectors.

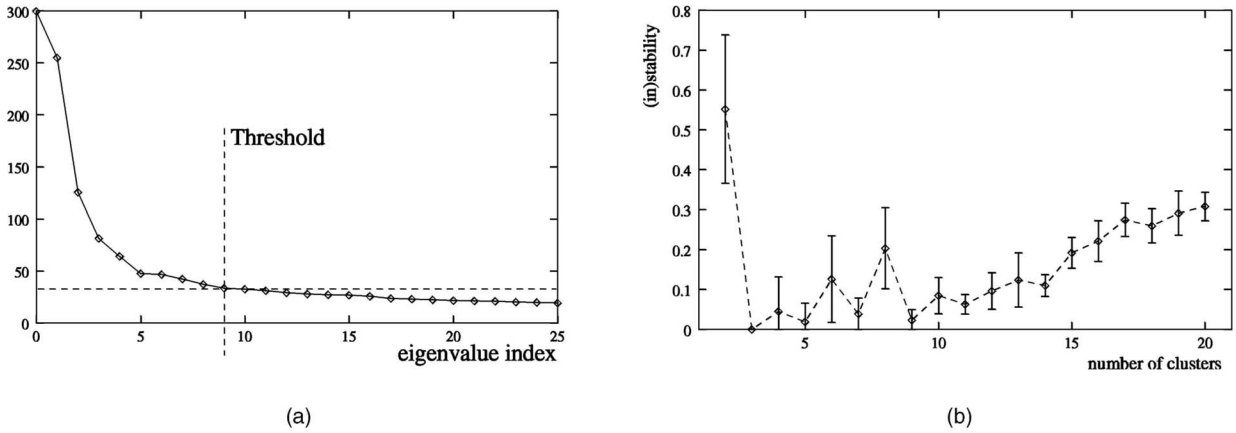

Fig. 4. Clustering of globin proteins. (a) Leading eigenvalues of the centered matrix S^c. (b) Instability of the partition versus the number of clusters k.

2. Since $\tilde{D}$ represents squared Euclidean distances, (7) implies that the Pairwise Clustering cost function is identical to the k-means function: $H^{\mathrm{pc}}(\tilde{D}) = H^{\mathrm{km}}(\tilde{D})$. According to the invariance properties (8) and (9), the optimal assignments $\{\hat{M}_{i\nu}\}$ of objects to clusters are not influenced by the transformations (23) and (25) of D into $\tilde{D}$, i.e., $\hat{M}(D) = \hat{M}(\tilde{D})$. □

The above theorem has several important consequences.

Interpretation and representation. Rewriting Pairwise Clustering as a k-means problem naturally introduces the notion of cluster centroids or cluster representants.

Prediction. The cluster prototypes define a generic prediction rule for new objects.

Data preprocessing and denoising. The vectorial representation of the objects allows us to apply standard preprocessing and denoising methods. Note that the usual semantics of "signal" and "noise" is closely related to some sort of generative model in a vector space.

Optimization. Minimizing the Pairwise Clustering cost function is an $\mathcal{NP}$-hard problem. The associated k-means problem with loss-free reconstructed vectors has the same complexity since the dimensionality of the vectors grows with n, see [20]. Thus, for handling real-word problems, in both cases, efficient approximation algorithms/schemes are necessary. In [10], it has been proposed to optimize H^{pc} by way of *deterministic annealing*. Since annealing methods are not in the main focus of this paper, we only mention that deterministic annealing is feasible only for *factorial* Gibbs distributions [9]. For $H^{\mathrm{pc}}(D)$, this constraint requires the use of a *mean-field approximation*. Applying Theorem 3, however, we are able to anneal the shifted k-means cost function $H^{\mathrm{km}}(\tilde{D})$, for which the mean-field approximation becomes *exact*. For details on annealing and mean-field approximations, the interested reader is referred to [10], [21].

If one decides to insert a denoising/dimensionality-reduction step into the clustering procedure, this will usually not only speed up the computations, but it will also "robustify" optimization heuristics for the k-means problem. For instance, applying PCA approximations according to Section 3.1, the energy landscape typically will be smoothed out, which makes local search heuristics (such as the classical iterative k-means algorithm) less sensitive to being trapped in local minima.

4.1 A Demo Application: Clustering of Protein Sequences

In this experiment with globin sequences, we present a worked-through example of combining constant-shift embedding, low-dimensional approximations, model selection, and clustering in the embedding space. From the SWISS-PROT and TrEMBL databases [22], we extracted all of the approximate 1,200 sequences annotated as "globins" or as "globin-like." The heuristic FASTA scoring method [23] was used for computing pairwise alignment scores which, in turn, were length-corrected (a Bayesian approach for correcting local alignments, following [24]) and normalized to the length of the alignment. From the pair-scores S_{ij}, we derived dissimilarities by setting $D_{ij} = S_{ii} + S_{jj} - 2S_{ij}$.[3] The eigenvalue spectrum of the centered matrix S^c shows some highly negative entries, indicating that the dissimilarities do not derive from squared Euclidean distances. By way of the constant shift embedding procedure, however, the sequences are represented as points in a vector space without distorting the grouping solution.

Given these vectors, we are left with two problems: 1) choosing an appropriate denoising mechanism and 2) minimizing the k-means cost function for different values of k and selecting the "optimal" number of clusters k. In the following, we present details for both the model selection procedure and the final clustering results.

Denoising. Fig. 4a shows the 25 leading eigenvalues of the centered matrix S^c. The eigenvalue curve suggests that there are only very few dominating directions in the embedding space. We thus decided to discard all but the first 10 leading eigenvectors. Since, in this control experiment, we have access to the ground-truth labels, we are able to test this hypothesis about "signal" and

3. Note that other transformations (e.g., of the form $D'_{ij} = \exp(-S_{ij})$) may be applied as well. Our experimental results, however, favor the above choice.

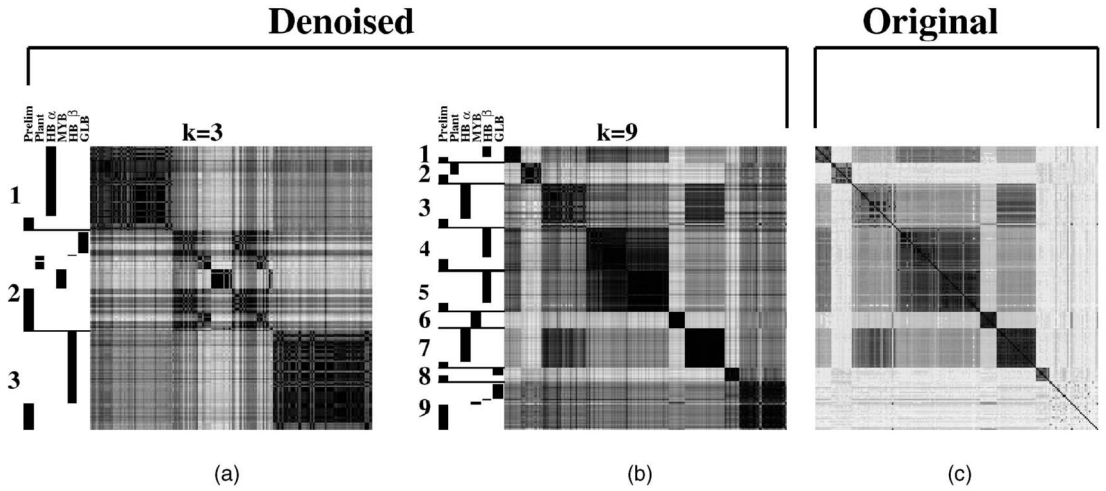

Fig. 5. Distance matrices for the embedded clustering problems, permuted with respect to cluster labels. (a) $k = 3$, (b) $k = 9$, and (c) original dissimilarities (without denoising, plotted in the permutation of the $k = 9$ solution).

"noise." The plotted denoised and original distance matrices in Fig. 5 indicate that the space spanned by the first ten eigenvectors in deed accentuates the main structure of the protein (sub)families.

Optimization and model selection. For minimizing the k-means functional in the embedding space, a deterministic annealing method was applied. Concerning the selection of the "correct" number of clusters, we used the concept of *cluster stability* which has been introduced in [25] and refined in [26]. The main idea is to draw resamples from the data set and then to compare the inferred data-partitions across these resamples. The variations of the partitions are transformed into an instability index, which is normalized such that a *random* procedure yields instability 1, and a perfect correspondence between solutions yields instability 0. Fig. 4b depicts the estimated instability for different numbers of clusters. The bars show the standard deviations estimated in the resampling procedure. The most stable solution partitions the data into three clusters, and two another distinct local minima occur for $k = 5$ and $k = 9$.

Clustering results. For the solutions with $k = 3$ and $k = 9$, we have plotted the corresponding distance matrices in Fig. 5. In Figs. 5a and 5b we have also depicted the "true" group membership of the proteins, as annotated in the SWISS-PROT database. The groups are: *Plant* (plant globins), HB-α (hemoglobin-α), *MYG* (myoglobin), HB-β (hemoglobin-β), and *GLB* (other globins, e.g., globin I-IV or insect globins). The column marked *Prelim* indicates "preliminary" sequences from the TrEMBL database with missing or uncertain annotations. It is obvious that the automatically found solutions divide the sequences into biologically meaningful groups: the 3-cluster solution separates both hemoglobin-α and hemoglobin-β from the rest. The 9-cluster solution defines a refinement of these groups, in the following sense: the α-hemoglobins are split into two subgroups (cluster no. 3 and no. 7), the hemoglobin-β cluster is split into three clusters, both the myoglobins and the plant globins are now contained in individual clusters, and the other globins are also separated into two subclusters (the first of which now mainly contains insect globins). It is interesting to notice that successively increasing the number of clusters automatically leads to a natural hierarchical representation of the group structure, which has *not* been introduced by the algorithm as a modeling bias.

Comparison with MDS. From a theoretical viewpoint, the constant shift embedding principle has one major advantage over classical MDS embedding: For shift-invariant clustering cost functions, CSE yields cluster preserving embeddings in $n - 1$ dimensional vector spaces while, to our knowledge, for MDS, no such guarantees are available. Taking a practical perspective, however, one might be interested in differences between CSE and MDS in *low-dimensional* embedding spaces. Designing experiments which allows "fair" comparisons of this kind, however, is difficult, since both the CSE method (different reduction methods like PCA, LLE, etc.) and MDS (different cost functions, choice of weights, etc.) can be varied in several ways. Nevertheless, we conclude this section with a comparison of k-means clustering results in two dimensions, once directly embedded using MDS (SSTRESS cost function, relative weights), and the second time embedded with CSE and PCA. In the upper left panel of Fig. 6 the two-dimensional MDS embedding of the above data set is depicted. The different point symbols refer to the SWISS-PROT labels. Given these two-dimensional data set, we then minimized the k-means clustering cost function with $k = 3$, leading to the labels shown in the lower left panel. It is interesting to note that the typical "ring artifacts" of MDS embedding produce elongated structures which cannot be recovered by the compactness-based k-means clustering criterion. In the case of CSE with succeeding PCA embedding, the situation looks very different: The embedded data clearly show three relatively compact groups (upper right panel): one corresponds to hemoglobin-α

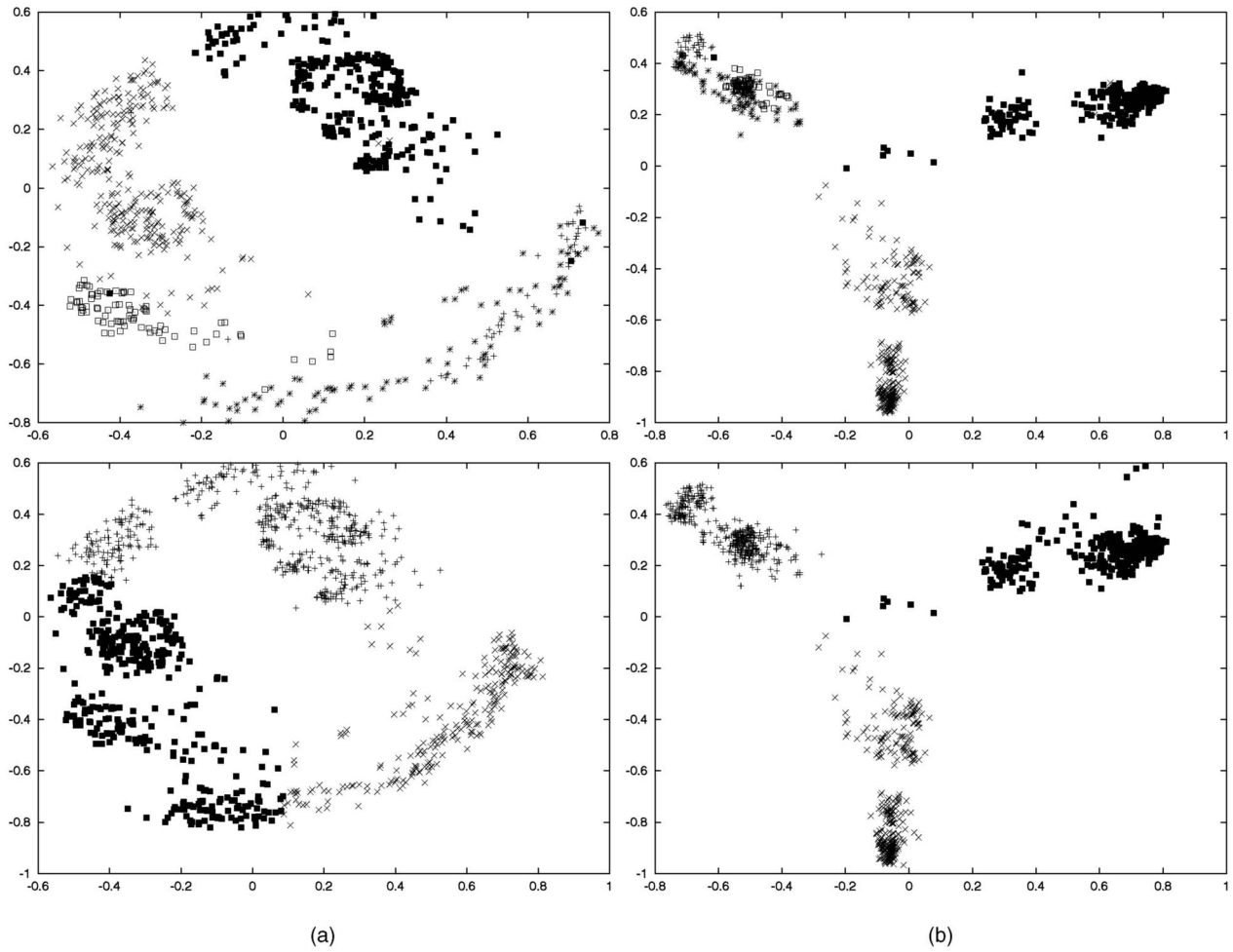

Fig. 6. Comparison of clustering results in two dimensions. Upper row: embedded proteins with original SWISS-PROT labels, lower row: data with inferred 3-means labels. (a) MDS (SSTRESS, local weights), (b) CSE with PCA embedding.

proteins, another to hemoglobin-β proteins; the third one is a mixture of the other protein families. These three compact groups are perfectly recovered in the 3-means solution (lower right panel).

5 Relations to Graph-Theoretic Clustering Methods

In this section, we discuss the relations between graph-theoretic grouping principles and the constant shift embedding method for pairwise clustering. As the main result, we show that both the *Averaged Association* and the *Averaged Cut* cost function are shift-invariant. With this invariance property, the *Averaged Association* principle turns out to be equivalent to the k-means clustering algorithm in the embedding space. Using the same strategy, we show that *Averaged Cut* is equivalent to the *pairwise separation* cost function. The latter can also be stated in terms of Euclidian distances between embedded vectors. For the *Normalized Cut* method; on the other hand, the constant-shift embedding method is not applicable. In the case of balanced partitions with similar structure among all clusters, however, the differences between *Averaged Association*, *Averaged Cut*, and *Normalized Cut* become vanishingly small. In such situations, all three methods can be reasonably well approximated by k-means.

A graph $G = (V, E)$ can be partitioned into disjoint sets $A^\nu, \quad \nu = 1, \ldots, k$ by removing edges: $\bigcup_{\nu=1}^{k} A^\nu = V, \quad A^\nu \cap A^\mu = \emptyset$ for $\nu \neq \mu$. Following [8], we define the dissimilarity between the sets A^ν and $V - A^\nu$ by the total weight of the edges that have been removed

$$\mathrm{cut}(A^\nu, V - A^\nu) = \sum_{u \in A^\nu, v \in (V - A^\nu)} w(u, v), \tag{26}$$

where the weight on each edge, $w(u, v)$, is a function of the similarity between nodes u and v. We further introduce a measure of association between two sets, $\mathrm{assoc}(A, B)$, as the total connection from nodes in set A to the nodes in set B. It follows immediately that both measures are connected by the formula

$$\mathrm{cut}(A^\nu, V - A^\nu) = \mathrm{assoc}(A^\nu, V) - \mathrm{assoc}(A^\nu, A^\nu). \tag{27}$$

We further denote by W the similarity (weight) matrix with unit self-similarities: $W_{ii} = 1, \forall i = 1, \ldots, n$. Based on this similarity matrix, we define a dissimilarity matrix by

$D := e_n e_n^\top - W$, with $e_n = (1, \dots 1)^\top$ as before. ($e_n e_n^\top$ is the $(n \times n)$ matrix of ones.) Together with the notation of the binary assignment variables $M_{i\nu}$ and the definition $n_\nu := |A^\nu|$, we can write the association measure in the form

$$\begin{aligned}\mathrm{assoc}(A^\nu, A^\nu) &= \sum_{i=1}^{n} M_{i\nu} \sum_{j=1}^{n} M_{j\nu} W_{ij} = \sum_{i=1}^{n} M_{i\nu} \sum_{j=1}^{n} M_{j\nu}(1 - D_{ij}) \\ &= n_\nu^2 - \sum_{i=1}^{n} M_{i\nu} \sum_{j=1}^{n} M_{j\nu} D_{ij}.\end{aligned} \tag{28}$$

For two sets, $A \cup B = V$, $A \cap B = \emptyset$, in [8], the *Averaged Association* cost function has been defined as

$$\mathrm{AvAssoc} = \frac{\mathrm{assoc}(A, A)}{|A|} + \frac{\mathrm{assoc}(B, B)}{|B|}. \tag{29}$$

It can be easily extended for a k-partitioning problem:

$$\mathrm{AvAssoc}_k = \sum_{\nu=1}^{k} \frac{\mathrm{assoc}(A^\nu, A^\nu)}{n_\nu}. \tag{30}$$

Inserting $(D := e_n e_n^\top - W)$ and (28), we see that maximizing the averaged association is equivalent to minimizing the *Pairwise Clustering* cost function H^{pc}:

$$\mathrm{AvAssoc}_k(W) = \sum_{\nu=1}^{k} \frac{\mathrm{assoc}(A^\nu, A^\nu)}{n_\nu} = n - 2H^{\mathrm{pc}}(e_n e_n^\top - W). \tag{31}$$

According to Theorem 3, it is always guaranteed that the (possibly shifted) matrix $S^c := -\frac{1}{2} D^c$ is a positive semidefinite dot-product matrix which can be used to embed the data into a Euclidian space. In this space, the problem of minimizing the pairwise clustering function reduces to a standard k-means problem.

The *Averaged Cut* cost function, cf. [8], is defined as

$$\begin{aligned}\mathrm{AvCut}_k &= \sum_{\nu=1}^{k} \frac{\mathrm{cut}}{(A^\nu, V - A^\nu)} n_\nu \\ &= \sum_{\nu=1}^{k} \frac{\mathrm{assoc}(A^\nu, V) - \mathrm{assoc}(A^\nu, A^\nu)}{n_\nu}.\end{aligned} \tag{32}$$

In the following, we will show that AvCut is equivalent to the *Pairwise Separation* cost function H^{ps} (in [9], this cost function is denoted by H^{ps1a}):

$$\begin{aligned}H^{\mathrm{ps}} &= -\sum_{\nu=1}^{k} \sum_{i=1}^{n} M_{i\nu} \frac{1}{k-1} \sum_{\mu \neq \nu} \frac{\sum_{j=1}^{n} M_{j\mu} D_{ij}}{\sum_{j=1}^{n} M_{j\mu}} \\ &= -\frac{1}{k-1} \left[\sum_{\nu=1}^{k} \frac{1}{n_\nu} \sum_{i=1}^{n} M_{i\nu} \sum_{j=1}^{n} D_{ij} - 2H^{\mathrm{pc}} \right].\end{aligned} \tag{33}$$

With (31) and the identity

$$\mathrm{assoc}(A^\nu, V) = \sum_{i=1}^{n} M_{i\nu} \sum_{j=1}^{n} W_{ij} = n n_\nu - \sum_{i=1}^{n} M_{i\nu} \sum_{j=1}^{n} D_{ij}, \tag{34}$$

AvCut can be reformulated in terms of H^{ps}:

$$\mathrm{AvCut}_k = \sum_{\nu=1}^{k} \frac{\mathrm{assoc}(A^\nu, V)}{n_\nu} - n + 2H^{\mathrm{pc}} \tag{35}$$

$$= kn - \sum_{\nu=1}^{k} \frac{1}{n_\nu} \sum_{i=1}^{n} M_{i\nu} \sum_{j=1}^{n} D_{ij} - n + 2H^{\mathrm{pc}} \tag{36}$$

$$= (k-1)n + (k-1)H^{\mathrm{ps}}. \tag{37}$$

Minimizing the averaged cut cost function based on the affinity matrix W is thus equivalent to minimizing H^{ps} with distances $D := e_n e_n^\top - W$. Note that the separation measure H^{ps} has the same shift-invariance property as its compactness counterpart H^{pc}:

$$H^{\mathrm{ps}}\big(D + d_0(1 - \delta_{ij})\big) = H^{\mathrm{ps}} + Const. \tag{38}$$

We can thus directly apply the constant-shift embedding framework of Section 3.

The *Normalized Cut* cost function, cf. [8], is an intermediate grouping criterion that combines both the compactness and separation principle. The k-cluster version is defined as

$$\mathrm{Ncut}_k = \sum_{\nu=1}^{k} \frac{\mathrm{cut}(A^\nu, V - A^\nu)}{\mathrm{assoc}(A^\nu, V)} = k - \sum_{\nu=1}^{k} \frac{\mathrm{assoc}(A^\nu, A^\nu)}{\mathrm{assoc}(A^\nu, V)}. \tag{39}$$

Rewriting this in terms of distances $D = e_n e_n^\top - W$, we arrive at

$$\mathrm{Ncut}_k = k - \sum_{\nu=1}^{k} \left[\frac{n_\nu - (1/n_\nu) \sum_{i=1}^{n} M_{i\nu} \sum_{j=1}^{n} M_{j\nu} D_{ij}}{n - (1/n_\nu) \sum_{i=1}^{n} M_{i\nu} \sum_{j=1}^{n} D_{ij}} \right]. \tag{40}$$

Contrary to AvAssoc and AvCut, the Ncut cost function is not shift invariant. For nonmetric (dis)similarities, it is thus not possible to apply the constant-shift embedding trick to obtain a grouping problem in a vector space. However, for the special case of balanced partitionings, $n_\nu = n/k \; \forall \nu$, and similar distribution of intracluster distances among all groups, all the row-sums of the distance matrix tend to be similar. Assuming $\sum_{j=1}^{n} D_{ij} = const$ and substituting this into (36) or (40), respectively, we see that in this special case, both the AvCut_k and the Ncut_k criteria become equivalent to the $\mathrm{AvAssoc}_k$ criterion and, hence, equivalent to the H^{pc} cost function. This equivalence means that, for clustering problems with similar group structure and balanced partitions, large differences between the models will become vanishingly small. The somewhat surprising results of a large-scale comparison study of graph partitioning algorithms for image segmentation tasks in [27] are in our view explained by this analysis.

6 Conclusion

We have introduced an optimal embedding procedure for pairwise clustering by means of constant shift embedding (CSE). For the class of shift-invariant clustering methods, it optimizes a fundamentally different criterion compared to classical embedding approaches based on MDS. The most prominent property of CSE is the complete preservation of the group structure in the embedding space. For MDS methods, on the other hand, such a preservation can only be

guaranteed in the special (and rather uninteresting) case of zero distortions ("stress") of the pairwise dissimilarities. For nonzero distortions, to our knowledge, no bounds on *structural* distortions are known.

The possibility of restating a pairwise grouping problem in a vector-space has important theoretical consequences. For instance, we are able to statistically describe the clusters by defining cluster prototypes in the embedding space and by measuring the variance in each of the clusters. These prototypes, in turn, define a generic rule for extending the grouping solution to a predictive discrimination rule for estimating the cluster membership of new objects. Concerning the problem of finding efficient optimization algorithms for minimizing clustering cost functions, the shown equivalence of Pairwise Clustering and k-means shed light on the probabilistic structure of the solution space: The problem of minimizing H^{pc} belongs to the class of combinatorial optimization problems for which the classical *mean-field approximation* becomes *exact*. There are also a couple of practical consequences of CSE: A common vector-space representation renders the data accessible to standard dimensionality and noise-reduction methods which lack a clear meaning for pairwise data. Such preprocessing methods, however, have to be chosen carefully, depending on the requirements and/or the prior knowledge available for each special application. For the task of clustering the globin proteins, it turned out that a classical PCA denoising worked surprisingly well. A comparison with the known family structure of these proteins revealed that the low-dimensional PCA embedding space accentuated the relevant structure while suppressing the alignment noise. It should be noticed, however, that in general unsupervised situations, such high-level domain knowledge may be hardly available. In these situations, one should rely on general statistical descriptors such as the form of the eigenvalue spectrum of the covariance matrix.

Despite the fact that "wrong" preprocessing methods clearly have the potential to distort the cluster structure (which we naturally want to preserve), the CSE framework at least tells us that these distortions are not caused by the general restrictions of a vector space. We know that there always exists a Euclidean space which contains the optimal structure preserving vectors, which means that there might be hope to find more suitable low-dimensional approximations.

ACKNOWLEDGMENTS

Special thanks go to Klaus Robert Müller and Mikio Braun for fruitful discussion and thorough reading of the manuscript. Julian Laub is supported by DFG Grant MU 987/1-1 and Volker Roth by DFG Grant BU 914/4-1.

REFERENCES

[1] I.T. Jolliffe, *Principal Component Analysis.* New York: Springer-Verlag, 1986.
[2] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An Introduction to Kernel-Based Learning Algorithms," *IEEE Trans. Neural Networks,* vol. 12, no. 2, pp. 181-201, 2001.
[3] D.W. Jacobs, D. Weinshall, and Y. Gdalyahu, "Classification with Nonmetric Distances: Image Retrieval and Class Representation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 6, pp. 583-600, June 2000.
[4] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification.* John Wiley & Sons, second ed., 2001.
[5] A.K. Jain, M.N. Murty, and P.J. Flynn, "Data Clustering: A Review," *ACM Computing Surveys,* vol. 31, no. 3, pp. 264-323, 1999.
[6] T.F. Cox and M.A.A. Cox, *Multidimensional Scaling.* London: Chapman & Hall, 2001.
[7] Y. Takane, F.W. Young, and J. de Leeuw, "Nonmetric Individual Differences Multidimensional Scaling: An Alternating Least Squares Method with Optimal Scaling Features," *Psychometrica,* vol. 42, pp. 7-67, 1977.
[8] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 8, pp. 888-905, Aug. 2000.
[9] J. Puzicha, T. Hofmann, and J. Buhmann, "A Theory of Proximity Based Clustering: Structure Detection by Optimization," *Pattern Recognition,* vol. 33, no. 4, pp. 617-634, 1999.
[10] T. Hofmann and J. Buhmann, "Pairwise Data Clustering by Deterministic Annealing," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 1, pp. 1-14, Jan. 1997.
[11] P. Brucker, "On the Complexity of Clustering Problems," *Optimization and Operations Research: Lecture Notes in Economics and Math. Systems,* M. Beckman and H.P. Kunzi, eds. pp. 45-54, Springer, 1978.
[12] W.S. Torgerson, *Theory and Methods of Scaling.* New York: John Wiley and Sons, 1958.
[13] G. Young and A.S. Householder, "Discussion of a Set of Points in Terms of Their Mutual Distances," *Psychometrika,* vol. 3, pp. 19-22, 1938.
[14] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," *Neural Computation,* vol. 10, pp. 1299-1319, 1998.
[15] S. Mika, B. Schölkopf, A. Smola, K.-R. Müller, S. Mika, M. Scholz, and G. Rätsch, "Kernel PCA and De-Noising in Feature Spaces," *Advances in Neural Information Processing Systems,* M.S. Kearns, S.A. Solla, and D.A. Cohn, eds., vol. 11, pp. 536-542, MIT Press, 1999.
[16] P.J. Huber, "Projection Pursuit," *The Annals of Statistics,* vol. 13, no. 2, pp. 435-475, 1985.
[17] S. Roweis and L. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science,* vol. 290, pp. 2323-2326, 2000.
[18] J.B. Tenenbaum, V. Silva, and J.C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," *Science,* vol. 290, pp. 2319-2323, 2000.
[19] T. Kohonen, *Self-Organizing Maps.* Berlin: Springer-Verlag, 1995.
[20] P. Drineas, A. Frieze, R. Kannan, S. Vempala, and V. Vinay, "Clustering in Large Graphs and Matrices," *Proc. Symp. Discrete Algorithms,* 1999.
[21] K. Rose, E. Gurewitz, and G.C. Fox, "A Deterministic Annealing Approach to Clustering," *Pattern Recognition Letters,* vol. 11, no. 9, pp. 589-594, 1990.
[22] B. Boeckmann, A. Bairoch, R. Apweiler, M.-C. Blatter, A. Estreicher, E. Gasteiger, M.J. Martin, K. Michoud, C. O'Donovan, I. Phan, S. Pilbout, and M. Schneider, "The SWISS-PROT Protein Knowledgebase and Its Supplement TrEMBL," *Nucleic Acids Research,* vol. 31, pp. 365-370, 2003.
[23] W.R. Pearson and D.J. Lipman, "Improved Tools for Biological Sequence Analysis," *Proc. Nat'l Academy of Sciences,* vol. 85, pp. 2444-2448, 1988.
[24] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological Sequence Analysis.* Cambridge Univ. Press, 1998.
[25] S. Dudoit and J. Fridlyand, "A Prediction-Based Resampling Method for Estimating the Number of Clusters in a Data Set," *Genome Biology,* vol. 3, no. 7, 2002.
[26] T. Lange, M. Braun, V. Roth, and J.M. Buhmann, "Stability-Based Model Selection," *Proc. Conf. Neural Information Processing Systems,* to appear, 2003.
[27] P. Soundararajan and S. Sarkar, "Investigation of Measures for Grouping by Graph Partitioning," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 239-246, 2001.

Volker Roth received the diploma degree in physics in 1997, and the PhD degree in computer science in 2001, both from the University of Bonn, Germany. Currently, he is a postdoctorate researcher in the Computer Vision and Pattern Recognition Group headed by Professor J.M. Buhmann. His research interests include support vector machines and kernel-based learning algorithms, unsupervised learning and clustering, bioinformatics, and computational biology.

Julian Laub received the Diplom in physics from the Swiss Federal Institute of Technology Lausanne in 2000. He is a doctoral student in the Intelligent Data Analysis group of the Fraunhofer Gesellschaft. His scientific interests are in the fields of unsupervised learning, embedding, and visualization of pairwise data.

Motoaki Kawanabe received both the masters and PhD degrees in mathematical engineering from the University of Tokyo in Professor Amari's Lab. Since 2000, he has joined the IDA group at Fraunhofer FIRST. His research interests are focused on statistics, information theory, information geometry and, recently, also on blind source separation and independent component analysis.

Joachim M. Buhmann received the PhD degree in theoretical physics from the Technical University of Munich, Germany, in 1988. He has held postdoctoral and research faculty positions at the University of Southern California, Los Angeles, and the Lawrence Livermore National Laboratory, Livermore, California between 1988 and 1992. Until October 2003, he has headed the Research Group on Pattern Recognition, Computer Vision, and Bioinformatics in the Computer Science Department, Rheinische Friedrich-Wilhelms-Universität Bonn, Germany. In October 2003, he will join the Computer Science Department of the Swiss Federal Institute of Technology (ETH) in Zurich. His current research interests cover statistical learning theory and its applications to image understanding and signal processing. Special research topics include exploratory data analysis and data mining in bioinformatics, stochastic optimization, graphical models, and computer vision. Dr. Buhmann has been a member of the Technical Committee of the German Pattern Recognition Society since 1995. He is a member of the IEEE and the IEEE Computer Society.