Project acronym	SIMBAD
Project full title	Beyond Features: Similarity-Based Pattern Analysis and Recognition
Deliverable Responsible	Technische Universiteit Delft Mekelweg 4 Delft, 2628 CD Netherlands http://ict.ewi.tudelft.nl/
Project web site	http://simbad-fp7.eu
EC project officer	Teresa De Martino
Document title	Study on (non)geometricity
Deliverable	D 3.1
Document type	Report
Dissemination level	Public
Contractual date of delivery	M 12
Project reference number	213250
Status & version	Definitive version
Work package, deliverable responsible	WP 3,1, TUD
Author(s)	Robert P.W. Duin, Wan-Jui Lee, Marco Loog and Elżbieta Pękalska
Additional contributor(s)	-

This report is partially based on the below publications, published or submitted during the report period between April 2008 and April 2009. The report does not fully cover these publications and contains additional material as well.

Published in 2008

1. S.W. Kim and R.P.W. Duin, On Optimizing Dissimilarity-Based Classifier Using Multi-level Fusion Strategies (in Korean), *Journal of The Institute of Electronics Engineers of Korea*, Computer and Information (CI), vol. 45, no. 5, 2008, 15-24.

2. E. Pękalska and R.P.W. Duin, Beyond traditional kernels: classification in two dissimilarity-based representation spaces, *IEEE Transactions on Systems, Man Cybernetics*, vol. 38, no. 6, 2008, 729-744.

3. W.J. Lee and R.P.W. Duin, An Inexact Graph Comparison Approach in Joint Eigenspace, *Structural, Syntactic, and Statistical Pattern Recognition, Proc. SSSPR2008 (Orlando, Florida, USA, 4-6 Dec 2008)*, Lecture Notes in Computer Science, vol. 5342, Springer Verlag, Berlin, 2008, 35-44.

4. R.P.W. Duin, E. Pękalska, A. Harol, W.J. Lee, and H. Bunke, On Euclidean corrections for non-Euclidean dissimilarities, *Structural, Syntactic, and Statistical Pattern Recognition, Proc. SSSPR2008 (Orlando, Florida, USA, 4-6 Dec 2008)*, Lecture Notes in Computer Science, vol. 5342, Springer Verlag, Berlin, 2008, 551-561.

5. R.P.W. Duin and E. Pękalska, On refining dissimilarity matrices for an improved NN learning, *Proc. of the 19th Int. Conf. on Pattern Recognition (ICPR2008, Tampa, USA, Dec. 2008)*, IEEE Press, 2008.

6. B. Haasdonk and E. Pękalska, Indefinite Kernel Fisher Discriminant, *Proc. of the 19th Int. Conf. on Pattern Recognition (ICPR2008, Tampa, USA, Dec. 2008)*, IEEE Press, 2008.

Published or accepted in 2009

1. M. Orozco-Alzate, R.P.W. Duin, and C.G. Castellanos-Dominguez, A generalization of dissimilarity representations using feature lines and feature planes, *Pattern Recognition Letters*, vol. 30, no. 3, 2009, 242-254.

2. M. Bicego, E. Pękalska, D.M.J. Tax, and R.P.W. Duin, Component-based Discriminative Classification for Hidden Markov Models, *Pattern Recognition*, 2009.

3. W.J. Lee and R.P.W. Duin, A Labelled Graph Based Multiple Classifier System, *MCS 2009*, accepted.

Abstract

Two major steps can be distinguished in the construction of recognition systems for pattern classes of real world objects. These are *representation* and *generalization*. The step of generalization has been well studied for the case of vector representations in Euclidean spaces. However, in the pattern recognition practice non-Euclidean dissimilarity measures and/or indefinite kernels (or similarity measures) are frequently used for representation. They implicitly describe objects in non-Euclidean vector spaces (determined through embeddings) for which generalization is less well defined.

There are three ways to handle this problem: (1) suitable adaptation of the (dis)similarity measure, (2) transformation of the non-Euclidean space into a Euclidean space via a correction procedure, and (3) extension of the set of generalization procedures to non-Euclidean spaces.

Which solution is to be preferred may be related to the cause of the non-Euclidean relations between the objects in a particular problem. We will try to analyze them on the basis of examples from the real world as well as artificial ones. Non-Euclidean behavior can arise either by non-intrinsic or intrinsic causes. The first ones are the result of the lack of either computational or observational power. The second ones are the consequence of an essential non-Euclidean judgment of the object dissimilarities, often resulting from restricted, pairwise comparisons.

This report is concluded with a discussion on the possible identification of the cause of a non-Euclidean representation for the generalization step.

1 Introduction

Automatic systems for the recognition of objects like images, videos, time signals, spectra, etcetera, can be designed by learning from a set of object examples labeled with the desired pattern class. Two main steps can be distinguished in this procedure:

Representation: In this step the individual objects are characterized by a simple mathematical entity such as a vector, string of symbols or a graph. A condition for this representation is that objects can easily be related in order to facilitate the following step.

Generalization: The representations of the object examples should enable the mathematical construction of models for object classes or class discriminants such that a good class estimate can be found for the representation of new, unseen and, thereby, unlabeled objects.

The topic of generalization has been intensively studied within the research areas such as statistical learning theory [1] statistical pattern recognition [2, 3, 4, 5], artificial neural networks [6] and machine learning [7, 8]. The most popular representations are based on Euclidean vector spaces, next to strings and graphs. More recently it has also been studied how to use vector sets for representing single objects; see e.g. [9]. Representations like strings of symbols and attributed graphs are sometimes preferred over vectors as they model the objects more accurately and offer more possibilities to include domain expert knowledge [10].

Representations in Euclidean vector spaces are well suited for generalization. Many tools are available to build (learn) models and discriminants from sets of object examples (training sets) that may be used to classify new objects into the right class. Traditionally, the Euclidean vector space is defined by a set of features. These should ideally characterize the patterns well and also be relevant for class differences at the same time. Such features have to be defined by experts exploiting their knowledge of the application.

A drawback of the use of features is that different objects may have the same representation as they differ by properties that were not expressed in the chosen feature set. This results in class overlap: in some areas in the feature space objects of different classes are represented by the same feature vectors. Consequently, they cannot be distinguished, which leads to an intrinsic classification error, usually called the Bayes error.

An alternative to the feature representation is the dissimilarity representation based on direct pairwise object comparisons. If the entire objects are taken into account in the comparison, then only identical objects will have a dissimilarity zero (if the dissimilarity measure has the property of 'identity of indiscernibles'). For such a representation class overlap does not exist if the objects can be unambiguously labeled: there are no real world objects in the application that belong to more than one class.

Another advantage of the dissimilarity representation is that it uses the expert knowledge in a different way. Instead of features, a dissimilarity measures has to be supplied. Of course, when the features are available, a distance measure between feature vectors may be used as a dissimilarity measure. But instead, also other measures, comparing the entire objects may be considered and are even preferred. In some applications, e.g. shape recognition, good features are much more difficult to define than a dissimilarity measure. Even 'bad' dissimilarity measures may be used (at the cost of large training sets) as long as only identical objects have a zero dissimilarity.

Dissimilarities have been used in pattern recognition for a long time. The idea of 'template matching' is based on them: objects are given the same class label if their difference is sufficiently small [11]. This is identical to the nearest neighbor rule used in vector spaces [3]. Also many procedures for cluster analysis make use of dissimilarities instead of feature spaces [12]. To some extent, the concept of dissimilarities is analogous to the use of kernels (and the potential functions as studied in the sixties [13]). The main difference is that kernels were originally defined in vector spaces to preferably fulfill Mercer's conditions [14, 15]. Kernel values can be interpreted as inner products between feature vectors and are, as such, similarities. Because of their properties they are very well suited for finding non-linear classifiers in vector spaces using Support Vector Machines (SVMs) [7].

Inspired by the use of kernels in the machine learning area and the use of dissimilarities in pattern recognition, authors of this report started to experiment with building other classifiers than the ones based on template matching and the nearest neighbor rule for the dissimilarity representation [16, 17, 18, 19, 20], which they also discussed as generalized kernel approaches [21, 22]. Their target was to develop procedures for any type of dissimilarity matrix generated in pattern recognition applications. Many of the dissimilarity measures used in the pattern recognition practice appear to be indefinite: they cannot be understood as distances in a Euclidean vector space, they are sometimes even not metric and they do not satisfy the Mercer conditions.

The work on the general dissimilarity matrices touches the gradually raising interest of the machine learning community in indefinite kernels: [23, 24, 25, 26, 27]. There is however some doubt whether the non-Euclidean aspects of the relations between pairwise comparison of objects are informative [28, 29, 30].

In this report preparations are discussed to study further the handling and possible informativeness of non-Euclidean dissimilarity matrices. From the observation that they arise often in the pattern recognition practice, it can be concluded that this is a significant issue. We will therefore discuss the various circumstances under which such dissimilarity matrices arise and will try to characterize them. Next, we will discuss three ways to approach this problem:

1. Avoiding the non-Euclidean dissimilarities by adapting the measure.
2. Correcting dissimilarity matrices such that they become Euclidean and by this traditional generalization procedures can be applied.
3. Leaving the data as they are and developing generalization procedures that can handle non-Euclidean dissimilarity data.

This will be illustrated by a series of examples based on artificially generated data sets as well as on real world problems. These are partially based on vector representations derived from dissimilarity matrices. There are two essentially different ways to construct vector spaces from dissimilarities. They are extensively described in the literature. For completeness they will be shortly summarized in the next section.

2 Vector spaces for the dissimilarity representation

The complete dissimilarity representation yields a square matrix with the dissimilarities between all pairs of objects. Traditionally, just the dissimilarities between the test objects and training objects are used. For every test object the nearest neighbors in the set of training objects are first found and used by the nearest neighbor classifier. This procedure does not use the relations between the training objects. The following two approaches construct a new vector space on the basis of the relations within the training set. The resulting vector space is used for training classifiers.

In the first approach, the dissimilarity matrix is considered as a set of row vectors, one for every object. They represent the objects in a vector space constructed by the dissimilarities to the other objects. Usually, this vector space is treated as a Euclidean space and equipped with the standard inner product definition.

In the second approach, an attempt is made to embed the dissimilarity matrix in a Euclidean vector space such that the distances between the objects in this space are equal to the given dissimilarities. This can only be realized error free, of course, if the original set of dissimilarities are Euclidean themselves. If this is not the case, either an approximate procedure has to be followed or the objects should be embedded into a non-Euclidean vector space. This is a space in which the standard inner product definition and the related distance measure are changed, resulting in indefinite kernels. It appears that an exact embedding is possible for every symmetric dissimilarity matrix with zeros on the diagonal. The resulting space is the so-called pseudo-Euclidean space.

These two approaches are more formally defined below, using an already published description [31].

2.1 Dissimilarity space

Let $\mathcal{X} = \{x_1, \ldots, x_n\}$ be a training set. Given a dissimilarity function and/or dissimilarity data, we define a data-dependent mapping $D(\cdot, R) : \mathcal{X} \to \mathbb{R}^k$ from $\mathcal{X}$ to the so-called *dissimilarity space* (DS) [16, 32, 21]. The k-element set R consists of objects that are representative for the problem. This set is called the representation or prototype set and it may be a subset of $\mathcal{X}$. In the dissimilarity space each dimension $D(\cdot, p_i)$ describes a dissimilarity to a prototype p_i from R. In this paper, we initially choose $R := \mathcal{X}$. As a result, every object is described by an n-dimensional dissimilarity vector $D(x, \mathcal{X}) = [d(x, x_1) \ \ldots \ d(x, x_n)]^T$. The resulting vector space is endowed with the traditional inner product and the Euclidean metric.

Any dissimilarity measure ρ can be defined in the DS. One of them is the Euclidean distance:

$$\rho_{DS}(x, y) = (\sum_{i=1}^{n} [d(x, x_i) - d(y, x_i)]^2)^{1/2} \tag{1}$$

This is the distance computed on vectors defined by original dissimilarities. For a set of dissimilarity measures ρ it holds asymptotically that the nearest neighbor objects are unchanged by ρ_{DS}. This is however not necessarily true for finite data sets. It will be shown later that this can be an advantage.

The approaches discussed in this report are originally intended for dissimilarities directly computed between objects and not resulting from feature representation. It is, however, still possible to study dissimilarity representations derived from features and yields sometimes interesting results [33]. In Fig. 1 an example is presented that compares an optimized radial basis SVM with a Fisher linear discriminant computed in the dissimilarity space derived from the Euclidean distances in a feature space. The example shows a large variability of the nearest neighbor distances. As the radial basis kernel

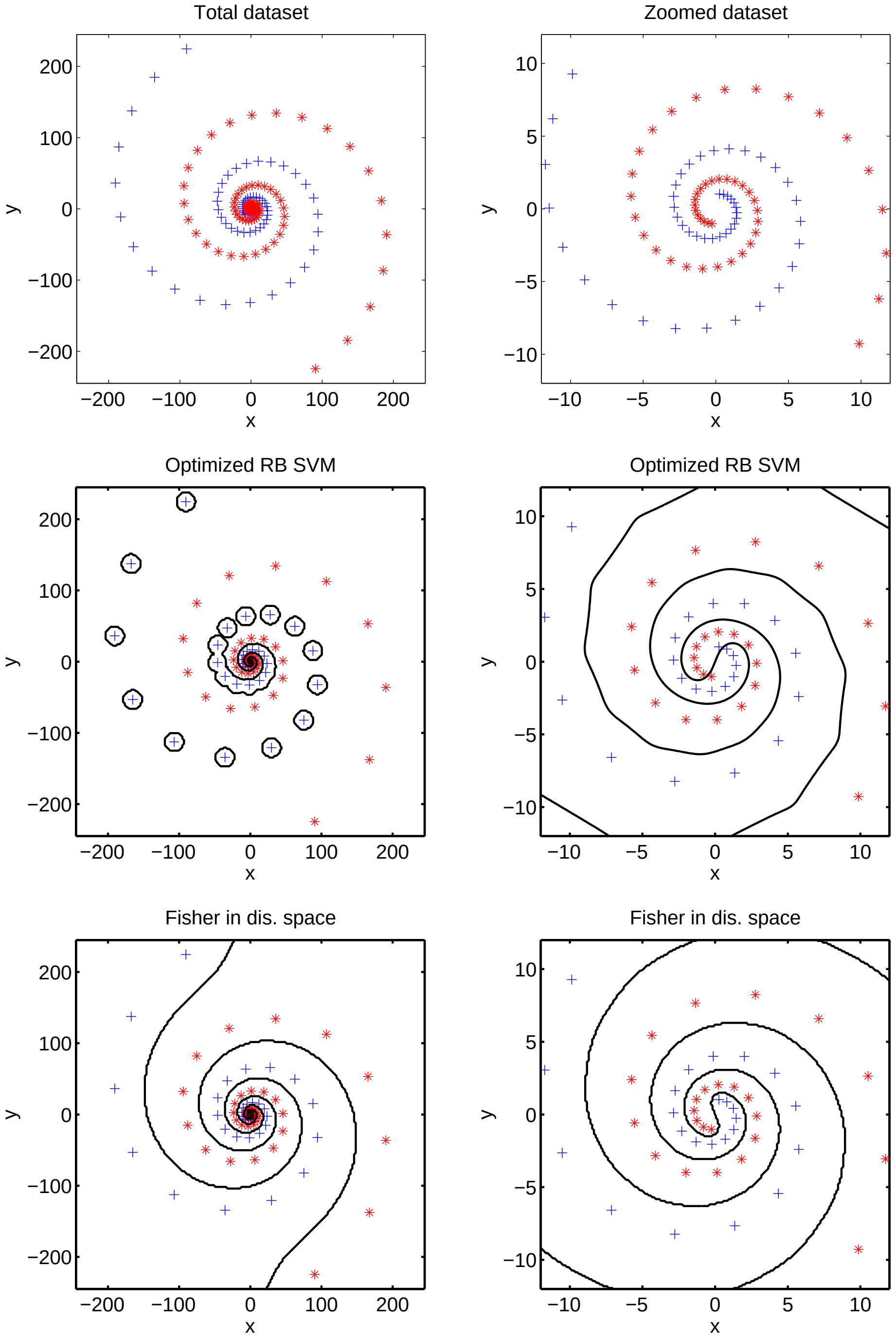

Figure 1: A spiral example with 100 objects per class. Left column shows the complete data sets, while the right column presents the zoom of the spiral center. 50 objects per class are used for training, systematically sampled. The middle row shows the training set and SVM with an optimized radial basis function; 17 out of 100 test objects are erroneously classified. The bottom row shows the Fisher Linear Discriminant (without regularization) computed in the dissimilarity space derived from the Euclidean distances. All test objects are correctly classified.

used by SVM is constant it cannot be optimal for all regions of the feature space. Fisher linear discriminant is computed in the dissimilarity space. Here the classes are linearly separable. Although the classifier is overtrained (the dissimilarity space is 100-dimensional and the training set has also 100 objects) it gives here perfect results. It should be realized that this example is specifically constructed to show the possibilities of the dissimilarity space.

2.2 Pseudo-Euclidean space

Before explaining the relation between pseudo-Euclidean spaces and dissimilarity representation, we start with definitions.

A Pseudo-Euclidean Space (PES) $\mathcal{E} = \mathbb{R}^{(p,q)} = \mathbb{R}^p \oplus \mathbb{R}^q$ is a vector space with a non-degenerate indefinite inner product $\langle \cdot, \cdot \rangle_{\mathcal{E}}$ such that $\langle \cdot, \cdot \rangle_{\mathcal{E}}$ is positive definite on $\mathbb{R}^p$ and negative definite on $\mathbb{R}^q$ [34, 20]. The inner product in $\mathbb{R}^{(p,q)}$ is defined (wrt an orthonormal basis) as $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{E}} = \mathbf{x}^T \mathcal{J}_{pq} \mathbf{y}$, where $\mathcal{J}_{pq} = [I_{p\times p}\ 0;\, 0\ -I_{q\times q}]$ and I is the identity matrix. As a result, $d^2_{\mathcal{E}}(\mathbf{x}, \mathbf{y}) = (\mathbf{x}-\mathbf{y})^T \mathcal{J}_{pq} (\mathbf{x}-\mathbf{y})$. Obviously, a Euclidean space $\mathbb{R}^p$ is a special case of a pseudo-Euclidean space $\mathbb{R}^{(p,0)}$. An infinite-dimensional extension of a PES is a Kreĭn space. It is a vector space $\mathcal{K}$ equipped with an indefinite inner product $\langle \cdot, \cdot \rangle_{\mathcal{K}} \colon \mathcal{K} \times \mathcal{K} \to \mathbb{R}$ such that $\mathcal{K}$ admits an orthogonal decomposition as a direct sum, $\mathcal{K} = \mathcal{K}_+ \oplus \mathcal{K}_-$, where $(\mathcal{K}_+, \langle \cdot, \cdot \rangle_+)$ and $(\mathcal{K}_-, -\langle \cdot, \cdot \rangle_-)$ are separable Hilbert spaces with their corresponding positive and negative definite inner products.

A positive definite kernel function can be interpreted as a generalized inner product in some Hilbert space. This space becomes Euclidean when a kernel matrix is considered. In analogy, an arbitrary symmetric kernel matrix can be interpreted as a generalized inner product in a pseudo-Euclidean space. Such a PES is obviously data dependent and can be retrieved via an embedding procedure. Similarly, an arbitrary symmetric dissimilarity matrix with zero self-dissimilarities can be interpreted as a pseudo-Euclidean distance in a proper pseudo-Euclidean space. Since in practice we deal with finite data, dissimilarity matrices or kernel matrices can be seen as describing relations between vectors in the underlying pseudo-Euclidean spaces. These pseudo-Euclidean spaces can be either determined via an embedding procedure and directly used for generalization, or approached indirectly by the operations on the given indefinite kernel. The section below explains how to find the embedded PES.

2.2.1 Pseudo-Euclidean embedded space

A symmetric dissimilarity matrix $D := D(\mathcal{X}, \mathcal{X})$ can be embedded in a Pseudo-Euclidean Space (PES) $\mathcal{E}$ by an isometric mapping [34, 20]. The embedding relies on the indefinite Gram matrix G, derived as $G := -\frac{1}{2}HD^{\star 2}H$, where $D^{\star 2} = (d_{ij}^2)$ and $H = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is the centering matrix. H projects the data such that X has a zero mean vector. The eigendecomposition of G leads to $G = Q\Lambda Q^T = Q|\Lambda|^{\frac{1}{2}}[\mathcal{J}_{pq};\, 0]|\Lambda|^{\frac{1}{2}}Q^T$, where Λ is a diagonal matrix of eigenvalues, first decreasing p positive ones, then increasing q negative ones, followed by zeros. Q is the matrix of eigenvectors. Since $G = X\mathcal{J}_{pq}X^T$ by definition of a Gram matrix, $X \in \mathbb{R}^n$ is found as $X = Q_n\,|\Lambda_n|^{\frac{1}{2}}$, where Q_n consists of n eigenvectors ranked according to their eigenvalues Λ_n. Note that X has a zero mean and is uncorrelated, because the estimated pseudo-Euclidean covariance matrix $C = \frac{1}{n-1}X^TX\,\mathcal{J}_{pq} = \frac{1}{n-1}\Lambda_r$ is diagonal. The eigenvalues λ_i encode variances of the extracted features in $\mathbb{R}^{(p,q)}$.

Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. If this space is a PES $\mathbb{R}^{(p,q)}$, $p + q = n$, the pseudo-Euclidean distance is computed as:

$$\begin{aligned}\rho_{PES}(\mathbf{x}, \mathbf{y}) &= (\sum_{i=1}^{p}[x_i - y_i]^2 - \sum_{i=p+1}^{p+q}[x_i - y_i]^2)^{1/2}\\ &= (\sum_{i=1}^{n}\delta(i,p)[x_i - y_i]^2)^{1/2},\end{aligned}$$

where $\delta(i,p) = \text{sign}(p - i + 0.5)$. Since the complete pseudo-Euclidean embedding is perfect, $D(x,y) = \rho_{PES}(x,y)$ holds.

Other distance measures may also be defined between vectors in a PES, depending on how this space is interpreted. Two obvious choices are:

$$\rho_{PES+}(\mathbf{x}, \mathbf{y}) = (\sum_{i=1}^{p}[x_i - y_i]^2)^{1/2}, \tag{2}$$

which neglects the axes corresponding to the negative dimensions (derived from negative eigenvalues in the embedding), and

$$\rho_{AES}(\mathbf{x}, \mathbf{y}) = (\sum_{i=1}^{n}[x_i - y_i]^2)^{1/2}, \tag{3}$$

which treats the vector space $\mathbb{R}^n$ as Euclidean $\mathbb{R}^{p+q}$. This means that the negative subspace of PES is interpreted as a Euclidean subspace (i.e. the negative signs of eigenvalues are neglected in the embedding procedure).

To inspect the amount of non-Euclidean influence in the derived PES, we define a Non-Euclidean Coefficient (NEC) as:

$$NEC = \sum_{j=p+1}^{p+q} |\lambda_j| / \sum_{i=1}^{p+q} |\lambda_i| \in [0, 1] \tag{4}$$

Fig. 2 shows how NEC varies as a function of p of the Minkowski-p dissimilarity measure (k-dimensional spaces) for a two-dimensional example:

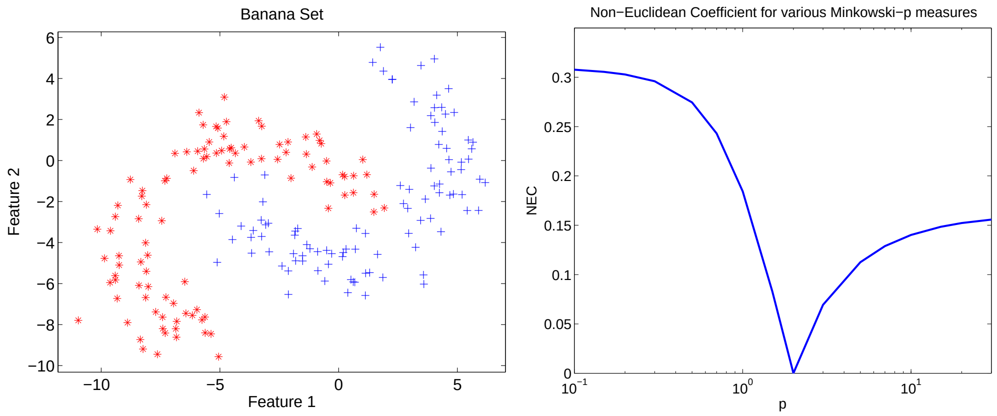

Figure 2: A two-dimensional data set (left) and the NEC as a function of p for various Minkowski-p dissimilarity measures.

$$\rho_{Min_p}(\mathbf{x}, \mathbf{y}) = (\sum_{i=1}^{k} [x_i - y_i]^p)^{1/p} \tag{5}$$

This dissimilarity measure is Euclidean for $p = 2$ and metric for $p > 1$. The measure is non-Euclidean for all $p \neq 2$. The value of NEC may vary considerably with a changing dimensionality. This phenomenon is illustrated in Fig. 3 for 100 points generated by a standard Gaussian distribution for various values of p. The one-dimensional dissimilarities obviously fit perfectly to a Euclidean space. For a vary high dimensionality, the sets of dissimilarities become again better embeddable in a Euclidean space.

2.3 Discussion on dissimilarity-based vector spaces

Now we want to make some remarks on the two procedures for deriving vector spaces from dissimilarity matrices, as discussed in previous subsection. On some aspects we will return at the end of this reports in relation to examples and experiments.

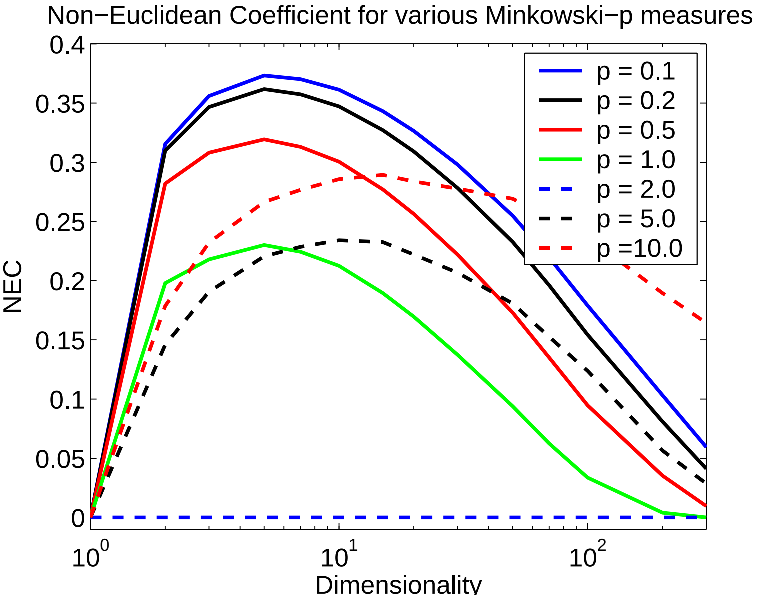

Figure 3: The Non-Euclidean Coefficient for various Minkowski-p dissimilarity measures as a function of the dimensionality of a set of 100 points generated by a standard Gaussian distribution.

The dissimilarity space in fact interprets the dissimilarities to particular prototypes (the representation set) as features. Their characteristics of dissimilarities is not used when a general classifier is applied. Special classifiers are needed to make use of that information. The good side of this 'disadvantage' is that the dissimilarity space can be used for any dissimilarity representation, including ones that are negative or asymmetric.

The embedding procedure is more restrictive. The dissimilarities are assumed to be symmetric and zero for the comparison with identical objects. Something like the pseudo-Euclidean embedding is needed in case of non-Euclidean data sets. The requirements of a proper metric or well-defined distances obeying the triangle inequality are not of use as they do not guarantee a Euclidean embedding. As we want to study more general data sets we use the name of dissimilarities instead of distances.

A severe drawback of both procedures is that they initially start with vector spaces that have as many objects as dimensions. Specific classifiers or dimension reduction procedures are thereby needed. For the dissimilarity representation this is somewhat more feasible than for the feature representation: features can be very different, some might be very good, others might be useless, or only useful in relation with particular other features. This is not true for dissimilarities. The initial representation is just based on objects. They have similar characteristics. It is not useful to use two objects that are much alike. Systematic, or even random procedures that reduce the initial representation set (in fact prototype selection) can be very effective [35] for this reason.

3 Non-Euclidean dissimilarity measures

The purpose of this study is to find good generalization procedures for dissimilarity data that arise in practical pattern recognition applications. In between is the step of representation. In the previous section two procedures for deriving vector spaces are presented. One is general, but neglects the dissimilarity characteristic of the data. The other is specific but suffers from the possible non-Euclidean relations that are present in the data. In order to analyze possible transformations of the derived vector spaces, especially of the pseudo-Euclidean space, we will first summarize and categorize the ways in which non-Euclidean dissimilarity data can arise.

Before becoming more specific, we like to emphasize how common non-Euclidean measures are. In [20] we already presented an extensive overview of such measures, but we encountered in many occasions that this fact is not sufficiently recognized.

Almost all probabilistic distance measures are non-Euclidean, including the Kolmogorov Variational Distance which is directly related to the classification error. This implies that when we want to build a classification system for a set of objects and each individual object is represented by a probability density function resulting from its invariants, the dissimilarity matrix resulting from the overlap between the object pdfs is non-Euclidean. Also the Mahalanobis class distance as well as the related Fisher criterion are non-Euclidean.

As a direct consequence of the above, many non-Euclidean distance measures are used in cluster analysis and in the analysis of spectra in chemometrics and hyperspectral image analysis. An energy spectrum can be considered as a pdf of energy contributions for different wavelengths. The popular absolute difference between two spectra is identical with the Minkowski-1 distance (related to the l_1-norm) between vector representations of the spectra.

In shape recognition, various dissimilarity measures are used based on the weighted edit distance as well as on variants of the Hausdorff distance. Usual parameters are optimized within an application w.r.t. the performance based on template matching and other nearest neighbor classifiers [36]. Most of them are still metric, some of them however are non-metric [37].

In the design and optimization of the dissimilarity measures it was in the past not an issue whether they were Euclidean. Just more recently, with the popularity of SVMs, it has became important to design kernels (similarity measures) which fulfill the Mercer conditions. This is equivalent to the possibility of Euclidean embedding. Next subsection discusses a number of reasons that give rise to violations of these conditions in applications, which lead to a set of non-Euclidean dissimilarities or indefinite kernels.

3.1 Non-intrinsic non-Euclidean dissimilarities

3.1.1 Numeric inaccuracies

A very simple reason why non-Euclidean dissimilarities arise is the numeric inaccuracies resulting from the use of computers with a finite word length. E.g., when we generate at random four points in an n-dimensional vector space and we follow the embedding procedure discussed in section 2.2 the projected vectors will fit in a 3-dimensional Euclidean space. In the procedure three eigenvalues larger than zero are expected to be found. In case $n = 2$ one of these eigenvalues will be zero. In a numeric procedure, however, there is a probability of almost 50% that the smallest eigenvalue has a very small negative value due to numeric inaccuracies (resulting from iterative procedures of determining the eigenvalues).

For this reason it is advisable to neglect all very small positive as well as negative eigenvalues. As a consequence, the dimensionality of the embedded space will be smaller than its maximum value of n-1.

3.1.2 Overestimation of large distances

When dissimilarities are not directly computed in a vector space but derived on raw data such as images or objects detected in images instead, more complicated measures may be used. They may still rely on the concept that the distance between two objects is the length or cost of the shortest path that has to be followed to transform one object into the other. Examples of such transformations are the weighted edit distance [38] and deformable templates [39]. In the optimization procedure that minimizes the length of the path, a minimization procedure may be used based on approximating the costs from above. As a consequence, too large distances are found.

The detection of too large distances is not easy, except when they are so large that the triangle inequality has been violated. In that case $d(A,C) > d(A,B) + d(B,C)$, indicating that a lower cost is possible in the transformation of A to C via a detour over B. This violates the result of the cost minimization. See [40] for an example. Such violations can easily be detected and corrected. The result is however just the replacement of a non-metric measure by a metric one. A possible non-Euclidean set of dissimilarities resulting from relations between more than three objects may still exist.

3.1.3 Underestimation of small distances

The underestimation of small distances has the same result as the above discussed overestimation of large distances. Similar correction procedures may be applied and again they only correct the metric property but not the Euclidean one.

There may be different causes of underestimated small distances. They may arise as the consequence of neglecting different particular object properties in different pairwise comparisons. For instance, in consumer preference data, the ranking of the most interesting books by every reader individually yields (dis)similarities based on different books by different pairwise comparisons of books or readers. Unread books by both readers in a comparison are thereby not taken into account, resulting in a too small estimate, especially for the small dissimilarities. E.g., it is possible to estimate a dissimilarity of zero if the ranking of the books read by both readers is identical, while it may be larger if additional books are taken into account.

Phrased in more abstract terms, the underestimation of small distances occurs when object pairs have to be compared from different points of view, or suffering from different partial (information) occlusions.

3.2 Intrinsic non-Euclidean dissimilarities

The causes discussed in the above may be judged as accidental. They result either from computational or observational problems. If better computers and observations were available, they would disappear. Now we will focuss on dissimilarity measures for which this will not happen. We will discuss three possibilities, without claiming completeness.

3.2.1 Non-Euclidean dissimilarities

As already indicated at the start of this section, there can be arguments from the application side to use another metric than the Euclidean one. An example is the Kolmogorov variational distance between pdfs as it is related to the classification error, or the l_1-distance between energy spectra as it is related to energy differences. Although the l_2-norm is very convenient

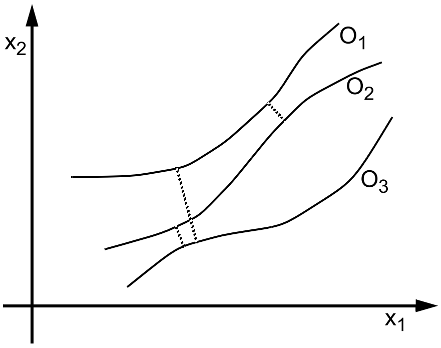

Figure 4: Vector space with the invariant trajectories for three objects O_1, O_2 and O_3. If the chosen dissimilarity measure is the minimal distance between these trajectories, triangle inequality can easily be violated, i.e. $d(O_1, O_2) + d(O_1, O_3) < d(O_1, O_3)$.

for computational reasons or because it is rotation invariant in a Euclidean space, the l_1-norm may naturally arise from the demands in applications.

3.2.2 Invariants

A very fundamental reason is related to the occurrence of invariants. Frequently, one is not interested in the dissimilarity between two objects A and B, but between two families of objects $A(\theta)$ and $B(\theta)$ in which θ controls an invariant, e.g. rotation in case of shape recognition. One may define the dissimilarity between two objects A and B as the minimum difference between the two sets defined by all their invariants.

$$d^*(A, B) = \min_{\theta_A} \min_{\theta_B} (d(A(\theta_A), B(\theta_B))) \tag{6}$$

In general, this measure is non-metric: the triangle inequality may be violated as for different pairs of objects different values of θ may be found that minimize (6). An example is given in figure 3.2.2, which is taken from [22].

3.2.3 Sets of vectors

Finding relations between sets of vectors is an important issue in cluster analysis. Individual objects may be represented by single vectors, but in a hierarchical clustering procedure the (dis)similarities between already grouped vectors are used to establish a new cluster level. Dissimilarity measures as used in the complete linkage and single linkage procedures are very common. The second, which is defined as the distance between the two most neighboring points of the two clusters being compared, is non-metric. It even holds for this distance measure that if $d(A, B) = 0$, then it does not follow that $A \equiv B$, because different clusters may just be touching.

For the single linkage dissimilarity measure it can be understood why the dissimilarity space may be useful. Given a set of such dissimilarities between clouds of vectors, it can be concluded that two clouds are similar if the entire sets of dissimilarities with all other clouds are about equal. If just their mutual dissimilarity is (close to) zero, they may still be very different. Fig. 5 illustrates this point.

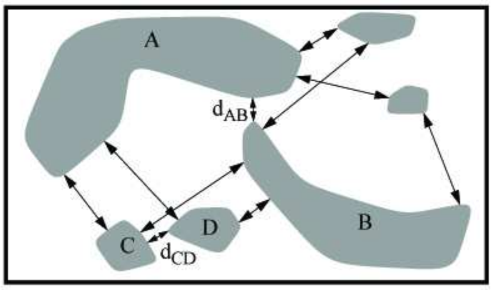

Figure 5: Single-linkage distance may be small for clusters which differ in position and shape.

The problem with the single linkage dissimilarity measure between two sets of vectors points to a more general problem in relating sets and even objects. In [9] an attempt has been made to define a proper Mercer kernel between two sets of vectors. Such sets are in this paper compared by the Hellinger distance derived from the Bhattacharyya's affinity between two pdfs $p_A(x)$ and $p_B(x)$ found for the two vector sets A and B:

$$d(A, B) = \left[\int (\sqrt{p_A(x)} - \sqrt{p_B(x)})^2 \right]^{1/2}. \tag{7}$$

The authors state that by expressing $p(x)$ in any orthogonal basis of functions, the resulting kernel K is automatically positive definite. This is correct, but it should be realized that it has to be the same basis for all vector sets $A, B, ...$ to which the kernel is applied. If in a pairwise comparison of sets different bases are derived, the kernel will become indefinite. This may happen if the numbers of vectors per set are smaller than the dimensionality of the vector space. It will happen most likely if this vector space is already a Hilbert space, e.g. when the vectors are already derived from a kernelization step.

This also makes it clear that indefinite relations may arise in any pairwise comparison of real world objects if they are first represented in some joint space for the two objects, followed by a dissimilarity measure. These joint spaces may be different for different pairs! Consequently, the total set of dissimilarities can be non-Euclidean, even if a single comparison is defined as Euclidean, as in (7).

3.3 Other non-Euclidean measures

There may be other factors leading to non-Euclidean dissimilarity measures. After further inspection, they may simplify to one or both of the above. We now mention two possibilities:

- Dis/similarity judgements by human experts. In some applications, e.g. psychometrical experiments, subjects are asked to judge the similarity between various sets of observations. It is not clear on which ground such judgements are made, as also in the consumer preference data.

- Weighted combinations of different dis/similarity measures that focus on different aspects of objects, e.g.

  $$d(x, y) = \sum_i \alpha_i d_i(x, y)$$

  where α_i is a constant and $d_i(x, y)$ is a dissimilarity w.r.t. particular i-th characteristics. An example is to derive the dissimilarity between images as a weighted average of dissimilarities computed w.r.t. texture, color and response to particular shape detectors.

4 Example classifiers in pseudo-Euclidean spaces

In our recent studies on analyzing dissimilarity data [20, 22, 41, 25, 29, 31], we have given many examples for classifiers that can be trained in indefinite (pseudo-Euclidean) spaces, e.g.

- The nearest mean rule as means and distances to points are well defined.

- The nearest neighbor rule for the same reason.

- The Parzen classifier, as it can be expressed in distances to points.

- The linear and quadratic classifiers based on class covariances. In Euclidean spaces they are related to normal distributions. In the pseudo-Euclidean spaces they can still be computed, but the relation with densities is unclear.

- A kernelized version of the Fisher discriminant for indefinite kernels.

Problematic classifiers are the ones based on general probability density estimates, as they are not (yet) properly defined for pseudo-Euclidean spaces and classifiers that rely on a distance to a linear or nonlinear classification boundary, such as SVM. The SVM classifier may still be computed but convergence and uniqueness are not guaranteed [23].

Below we present two artificial examples, taken from [25] in order to illustrate the work and performance of classifiers built in pseudo-Euclidean spaces. In this case, we do not explicitly determine the embedded PES, but consider classifiers that work on indefinite kernels instead. The considered classifiers are indefinite kernel Fisher discriminant (IKFD), indefinite SVM (ISVM) and indefinite kernel nearest mean classifier (IKNMC).

4.1 Checkboard example

The first example is an artificial 4×4 checkerboard data set based on a uniform distribution on $[-2, 2]^2 \subset \mathbb{R}^2$; see Fig. 6. A practical source of indefiniteness is incorporation of invariance into kernels. Here, it is done by combining different kernels into a new one. Let us denote $d(x, x') := \sum_{i=1,2} |(x)_i - (x')_i|^2$ and the kernel $k(x, x') := \exp(-d(x, x')^2/\sigma^2)$. As prior knowledge we observe that the problem is invariant w.r.t. the point reflection $\tau(x) := -x$ through the origin. We incorporate this by combining two kernels into a new one: $\bar{k}(x, x') := \max(k(x, x'), k(x, \tau(x')))$, which can alternatively be motivated by invariant distances. Application on a random training data set of $50 + 50$ samples yields an indefinite kernel matrix and corresponding data representations in a PES $\mathbb{R}^{(p,q)}$ for each σ. A 10-fold cross-validation was performed on the training set for each of the listed σ to determine the additional parameters of IKFD and ISVM. The chosen parameters and the corresponding test errors (on 500+500 samples) are reported in Table 1, as well as the signature (p, q) and NEC as an indefiniteness index. The resulting classifiers (with σ also being selected in cross-validation) are illustrated in Fig. 6. The parameters are: $\beta = 1, \sigma = 0.5$ for IKFD, $C = 1, \sigma = 0.1$ for ISVM, and $\sigma = 0.05$ for IKNMC. The test-errors equal $0.083, 0.121$ and 0.173, respectively. The perfect point symmetry of all classifiers in Fig. 6 occurs thanks to the invariant kernel. The table shows that IKNMC is here consistently worse than IKFD. ISVM performs as well or better than IKFD for marginal indefiniteness (here $\sigma \leq 0.1$). For predominantly indefinite data (here $\sigma > 0.1$), IKFD outperforms ISVM.

Table 1: Checkerboard example. Measures of indefiniteness and test errors.

σ	NEC	(p,q)	IKFD (β)	ISVM (C)	IKNMC
0.010	0.000	(98,2)	0.336 (10)	0.323 (10)	0.340
0.050	0.022	(82,18)	0.145 (10)	0.134 (10)	0.173
0.100	0.055	(66,34)	0.121 (10^{-1})	0.121 (1)	0.201
0.500	0.125	(51,49)	0.083 (1)	0.168 (1)	0.384
1.000	0.132	(52,48)	0.091 (10^{-3})	0.418 (1)	0.486
5.000	0.107	(50,50)	0.132 (10^{-2})	0.480 (1)	0.497
10.00	0.062	(51,49)	0.159 (10^{-3})	0.373 (10^2)	0.494

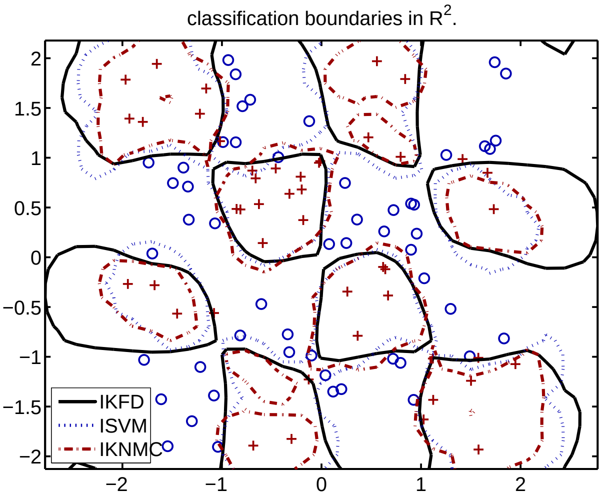

Figure 6: Indefinite invariant-kernel classifiers for the checkerboard data.

4.2 Polygon example

The other example is based on non-Euclidean dissimilarities, a common cause of indefiniteness. We consider the Polydist_m57 data set as described in Section 5. It consists of 2000+2000 polygons corresponding to two classes of polygons with five and seven vertices, respectively. The modified Hausdorff-distance is applied for computing the pairwise distances. We convert this dissimilarity d into similarity by considering the kernel $k(x,x') := -d(x,x')^\gamma$ for $\gamma > 0$. The experiment setting is as before, taking $50+50$ samples for a 10-fold cross-validated parameter selection and training, and then testing on the remaining 3900 examples. In order to address the statistical significance, we repeat this 10 times. The resulting mean and standard deviations of the test-errors, as well as of the signatures are reported in Table 2 Here, we see that IKFD and ISVM clearly outperform IKNMC. In the dominant positive definite case, $\gamma \leq 0.7$, there is no significant difference between the

Table 2: Polygon example. Average signatures and test errors.

γ	avr. (p,q)	IKFD	ISVM	IKNMC
0.2	(99.0,1.0)	0.021±0.006	0.021±0.006	0.089±0.027
0.5	(99.0,1.0)	0.019±0.006	0.018±0.004	0.110±0.034
0.7	(98.9,1.1)	0.020±0.004	0.018±0.004	0.118±0.037
1.0	(85.9,14.1)	0.019±0.009	0.029±0.007	0.129±0.041
2.0	(48.9,51.1)	0.017±0.008	0.094±0.057	0.152±0.051
5.0	(44.8,55.2)	0.102±0.021	0.131±0.030	0.218±0.081
7.0	(47.4,52.6)	0.111±0.027	0.237±0.058	0.253±0.093

performance of IKFD and ISVM, while in the remaining indefinite cases IKFD is obviously beneficial, with the overall best result for $\gamma = 2$.

5 Examples of Euclidean corrections

We used classifiers mentioned in the previous section to analyze various transformations from the pseudo-Euclidean space to the Euclidean space via Euclidean corrections [29]. We found many examples were such corrections are counterproductive, suggesting that indefinite spaces can be informative. More subtle corrections have to be investigated further.

The above mentioned transformations are topology preserving. This does not hold for the construction of the dissimilarity space out of a dissimilarity representation. In this case, a new Euclidean space is postulated based on the relations of objects with all other objects. This may remove or diminish noise, or defects arisen in the construction of the original dissimilarities. Possible information of original indefinite relations will thereby only be maintained if it can be expressed in the totality of the relation of objects to all other objects in a Euclidean way.

In the remainder of this section we will report a few experiments not published before that we use in our investigation of Euclidean corrections and the properties of the dissimilarity space. They are partially based on public domain data sets, and partially based on data generated for this purpose. Two of them are the results of the embedding of objects based on graph representations. In the next subsection this procedure is explained.

5.1 Graph matching

This paragraph summarizes a procedure developed by the authors, presented in [42] and used in [43].

A graph is a set of nodes connected by edges in its most general form. Consider an undirected graph $G = (V, E, W)$ with the node set $V = \{v_1, v_2, \ldots, v_n\}$, the edge set $E = \{e_1, e_2, \ldots, e_m\} \subset V \times V$, and the weight function $W : E \to (0, 1]$. If the graph edges are weighted, the adjacency matrix A for the graph G is the $n \times n$ matrix with elements

$$A_{ij} = \begin{cases} W(v_i, v_j), & \text{if } (v_i, v_j) \in E; \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

Clearly if the graph is undirected, the matrix A is symmetric. The Laplacian of the graph is defined by $L = D - A$, where D is the diagonal node degree matrix whose elements $D_{ii} = \sum_{k=1}^{n} A_{ik}$. The Laplacian matrix of G is positive semidefinite and singular, and it is more often adopted for spectral analysis than the adjacency matrix because of its properties.

Our approach for computing graph dissimilarity is based on spectral graph theory that is concerned with characterizing the structural properties of graphs using the eigenvectors of the adjacency matrix or the closely related Laplacian matrix (the degree matrix minus the adjacency matrix). To compute the graph dissimilarity, we first project each pair of two graphs into their joint eigenspace. This joint eigenspace (JoEig) is expanded by both sets of eigenvectors derived from the Laplacian matrices of graphs. Then the Frobenius norm of the difference between these two projected graphs is taken as their distance.

Let G and H be weighted undirected graphs and L_G and L_H be their Laplacian matrices, respectively. The eigendecomposition of L_G and L_H are performed as

$$\begin{aligned} L_G &= V_G D_G V_G^T, \\ L_H &= V_H D_H V_H^T, \end{aligned} \tag{9}$$

where V_G and V_H are orthonormal matrices and D_G and D_H are diagonal matrices of the eigenvalues (in ascending order) of G and H, respectively. With the joint projection vector $V_G V_H^T$, both graphs G and H will be projected to their joint eigenspace as $L_G V_G V_H^T$ and $V_G V_H^T L_H$. The difference between two graphs using JoEig is defined as

$$||V_G D_G V_H^T - V_G D_H V_H^T||^2. \tag{10}$$

The JoEig approach approximates a graph by relocating its eigenvalues in the joint eigenspace constructed by the eigenvectors of both graphs.

However, the sizes of graphs might be different, and therefore it might not be possible to make a matrix product between V_G and D_H or D_G and

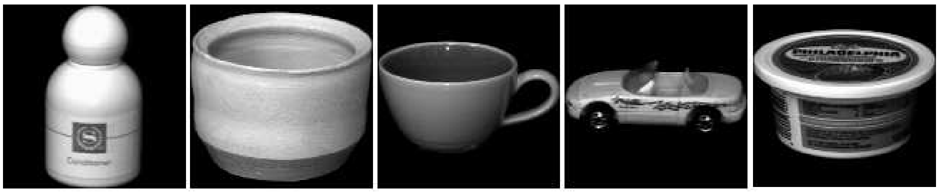

Figure 7: Example pictures from five different classes.

V_H. A feasible solution is to fix the number of eigenvectors for both graphs. One possible choice is to make full use of the eigenvectors from the smaller graph and keep the same number of eigenvectors and eigenvalues in the larger graph as in the smaller graph by removing less important eigenvalues and eigenvectors from the larger graph. Less important eigenvectors are those with smaller eigenvalues. The other possibility is ignoring the size of graphs and just pick a reasonable fix small number of eigenvectors and eigenvalues for all graphs.

The Coil-20 data set contains multiple views of the same object in different poses with respect to the camera. There are originally 20 objects (classes) in the data, but we only use five objects and 360 images in total (72 views per object) to form the data. Example pictures of these five classes are shown in Fig. 7.

For each picture, we extract the feature points using the scale-invariant feature transform (SIFT) method and then compute the Voronoi tessellations of the feature points to construct the region adjacency graph, i.e., the Delaunay triangulation, of the Voronoi regions. As a result, each picture is represented as a graph with the adjacency matrix. The adjacency matrices are further transformed into Laplacian matrices. The average number of nodes of these 360 graphs is 35.9 with the standard deviation of 21.4. The smallest graph has only five nodes.

Here, we consider two different settings for computing the distances between graphs. First, the dimensionalities of the joint eigenspaces from different pairs of graphs are different. This is done by making full use of the eigenvectors from the smaller graph and keep the same number of eigenvectors and eigenvalues in the larger graph as in the smaller graph by removing less important eigenvalues and eigenvectors from the larger graph. For different pair of graphs, the sizes of smaller graphs are very likely also different, and therefore different pairs of graphs are most probably compared in the eigenspaces with different dimensionalities. The other setting is to make sure all the eigenspaces where graphs are compared are with the same dimensionality. The dimensionality of these spaces is set to 5, which is the size of the smallest graph. Each graph keeps only 5 sets of eigenvectors and eigenvalues by removing less important eigenvalues and eigenvectors. Therefore, all pairs of graphs are compared in the eigenspaces with the same dimensionality.

5.2 Data sets

In the next section we will report some simple experiments based on the following data sets. A number of them are also used and discussed in [20] and [29].

Chicken data: dissimilarities are based on the weighted edit distances between 446 shapes representing five classes of chicken pieces. They depend on two parameters [44]. We used Chicken_30_45. It is a five-class set with 446 objects in total. Dissimilarities are computed by using a weighted edit-distance measure [38]. Formally, the measure is metric, but due to approximative optimizations as discussed in Section 3.1 many of the larger dissimilarities cause non-metric behavior. We did not correct for that. Earlier it was found [40] that after such a correction not only the data set yields about the same Non-Euclidean Coefficient, but also classification performances hardly change.

Zongker data: dissimilarities between 2000 handwritten digits in 10 classes based on deformable template matching [39]. The dissimilarity measure is the result of an iterative optimization of the non-linear deformation of the grid.

Polydist_m57 data: modified Hausdorff distances [37] between two classes, pentagons and heptagons, of artificially generated polygons. Each class consists of 2000 examples. Although polygons with five vertices are a subset of polygons with seven vertices, we put the restriction that all polygons of the later class have seven vertices, thereby avoiding ambiguity. The modified Hausdorff distance is non-metric but it may yield significantly better classification results than the original metric Hausdorff distance. Our implementation is rotation invariant as it finds the minimum distance over all possible rotations. Centers of gravity are aligned.

NIST_m38 data: The modified Hausdorff distances between the contours of two classes ('3' and '8') of the NIST character database. The classes consist of 2000 objects each.

Cat-cortex data: 65 objects in four classes represented by ordinal dissimilarity values [45]. This is a very small data set. We have chosen it as it is an example of the use of ordinal data.

Newsgroups data: 600 messages in four newsgroups related by a non-metric correlation measure [20].

Gauss data This is based on a 2-class dataset in 20 dimensions. The two classes have identical spherical distributions. We generated 2000 objects per class and computed Euclidean distances.

Gauss_noise data In this example the dissimilarities of the above example are heavily disturbed by a multiplicative random factor with mean one and standard deviation 0.3.

Gauss_m1 data Instead of Euclidean distances the Minkowski-1 distances (sum of absolute differences) between the objects of the Gauss examples are computed. This dataset is thereby is metric but non-Euclidean.

Gauss_m02 data Instead of Euclidean distances the Minkowski-0.2 distances between the objects of the Gauss examples are computed. This dataset is thereby is non-metric and thereby also non-Euclidean.

Coil_diff data: This is the set of graph distances as explained in the previous section. Graphs are compared in the eigenspace with a dimensionality determined by the smallest graph in every pairwise comparison. There are five classes, of 72 objects each.

Coil_same data: In this set of graph distances all graphs are compared in a pairwise fashion in a 5D space of eigenvectors derived from the two graphs.

5.3 Experiments

In order to simplify the computational procedures and to make results for various data easier comparable all experiments have been performed on randomly chosen subsets of 50 objects per class. Only for the Cat-cortex data we had to take just 10 objects per class due to the small size.

Several vector spaces are considered for every data set chosen in this way. For each of these spaces two classifiers are investigated, the 1-Nearest Neighbor (NN) rule and the linear SVM. We always derive the kernel matrix K for SVM in the same way. Based on the given (or computed) dissimilarities D, we find K as:

$$K = -\frac{1}{2} H D^{\star 2} H, \text{ where}$$
$$H = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T.$$

One of the considered vector spaces is non-Euclidean, which gives rise to an indefinite kernel. Consequently, SVM may be suboptimal. The following four spaces, and related dissimilarities and kernels are considered:

PES: Original Data. This is the original set of dissimilarities. It can be fully embedded in a Pseudo-Euclidean Space (PES). The kernel matrix K as described by 5.3 is the linear kernel for this space obeying its special inner product definition [20].

DS: Dissimilarity Space. This is the Euclidean space postulated by using the dissimilarities to all training objects as feature dimensions. Distances in this space follow from $d_DS(x,y) = ||d(x,\cdot) - d(y,$ $centerdot)||$.

AES: Associated Euclidean Space. Instead of using the pseudo-Euclidean metric, the Euclidean metric is used in the pseudo-Euclidean space. This is similar to the standard procedure used in classical scaling.

PES+: Positive part of the pseudo-Euclidean space. In this case all directions in the pseudo-Euclidean space are neglected that correspond to the negative eigenvalues. This is similar to the standard procedure used for eigenspaces: all small eigenvalues are neglected, in this case including the negative ones.

The following procedure was followed in the experiments:

1. The original data sets are ten times randomly sampled with the class sizes mentioned above (usually 50 objects per class).

2. The dissimilarity matrices D for the three derived spaces are computed.

3. The kernel matrix K is computed by 5.3 for each of the four vector spaces.

4. For the sampled original space the Non-Euclidean Coefficient NEC is derived by 5.3.

5. The dissimilarity matrices D and the kernel matrices K are ten times used to generate training and test sets of the same size (usually 25 objects per class). Note that in all dissimilarity matrices, but D_org, and all kernel matrices the representations for the test objects implicitly depend on the training set (not on their labels!).

6. The 1-nearest neighbor classifier is applied to the matrices D and SVM to the matrices K.
7. The error rates for all ten classifiers are estimated from the test sets.
8. The errors obtained ten times (from random sampling) are averaged and the standard deviations are computed.

The results are summarized in the tables 3, for the 1-Nearest Neigbor rule (NN) and 4. for the linear Support Vector Machine (SVM). The tables list the errors $*$ 1000 for various problems for the original data, which is equivalent to a full pseudo-Euclidean embedding (PES), the dissimilarity space (DS), the associated Euclidean space of the PES (AES) and the positive part of the pseudo-Euclidean space (PES+). Between brackets are the standard deviations of the means. Underlined results show significant improvements over the results in the PES. The original data sets are sampled with n objects per class. c is the number of classes.

Table 3: Experiment results for the NN classifier.

Data	c	n	NEC	PES	DS	AES	PES+
Chicken	5	50	0.313	98(3)	$\underline{78}$(2)	365(4)	196(3)
Zongker	10	50	0.353	144(2)	$\underline{81}$(2)	310(4)	$\underline{128}$(2)
Polydist_m57	2	50	0.205	146(5)	142(5)	165(5)	154(5)
NIST_m38	2	50	0.178	116(3)	146(4)	140(4)	126(4)
Cat-cortex	4	10	0.160	152(6)	$\underline{106}$(5)	$\underline{96}$(6)	$\underline{104}$(6)
Newsgroup	4	50	0.136	360(3)	365(3)	400(4)	373(4)
Gauss	2	50	0.000	296(7)	$\underline{258}$(5)	296(7)	296(7)
Gauss_noise	2	50	0.374	444(6)	$\underline{380}$(7)	477(6)	470(7)
Gauss_m1	2	50	0.172	316(7)	$\underline{287}$(5)	326(7)	319(7)
Gauss_m02	2	50	0.309	369(5)	371(6)	390(6)	377(6)
CoilDiff	5	50	0.354	443(3)	$\underline{380}$(4)	$\underline{423}$(4)	$\underline{415}$(4)
CoilSame	5	50	0.437	552(4)	603(4)	634(4)	633(4)

The results are shown in the tables 3 and 4. The following observations can be made.

- All datasets are non-Euclidean, except Gauss of course. In some cases the non-Euclideaness is rather strong.
- As could be expected, Gauss_m02, which is non-metric, is more non-Euclidean than Gauss_m1, which is metric.

Table 4: Experiment results for the SVM.

Data	c	n	NEC	PES	DIS	AES	PES+
Chicken	5	50	0.313	131(2)	82(2)	99(2)	107(2)
Zongker	10	50	0.353	81(2)	71(2)	82(2)	78(1)
Polydist_m57	2	50	0.205	103(5)	50(4)	94(4)	96(4)
NIST_m38	2	50	0.178	222(6)	263(6)	231(7)	225(6)
Cat-cortex	4	10	0.160	134(7)	95(5)	106(6)	116(6)
Newsgroup	4	50	0.136	302(3)	320(4)	299(4)	298(3)
Gauss	2	50	0.000	243(6)	232(6)	243(6)	243(6)
Gauss_noise	2	50	0.374	393(7)	312(5)	405(5)	371(6)
Gauss_m1	2	50	0.172	247(6)	240(6)	222(5)	238(6)
Gauss_m02	2	50	0.309	439(9)	310(7)	284(6)	443(9)
CoilDiff	5	50	0.354	389(3)	413(3)	385(3)	404(3)
CoilSame	5	50	0.437	690(6)	665(7)	685(6)	625(4)

- The noise of the Gauss_noise dataset is rather severe, still the CoilSame graph example yields a larger value of NEC, suggesting that the graph matching procedure operating in eigenspaces of the same dimension is far from Euclidean.

- For the CoilDiff dataset eigenspaces of different dimensionality where used for the pairwise comparisons, matching the complexity of the simpler graph. This resulted in a less non-Euclidean behavior and in better classification results (not necessarily related).

- The three Euclidean spaces studied here show many improvements over the original representation. There is just one example in which the original, non-Euclidean space produced a clear better result (the SVM for CoilSame). The statement however that non-Euclidean space may be informative thereby still holds.

- In general the dissimilarity space DS yields good performances. This holds for the pure noise example, Gauss_noise, as well as for the pure non-Euclidean measures used in Gauss_m1 and Gauss_m02.

- The corrections made for the pseudo-Euclidean space, the associated space (AES) as well as the positive space (PES+) yield sometimes good performances and occasionally very bad, e.g. the nearest neighbor results for the Chicken and the CoilSame example. item The traditional way of handling non-Euclidean dissimilarities and indefinite kernels, neglecting all small and negative eigenvalues is equivalent to the AES.

Just occasionally this yields a significant best result: Cat-cortex for the NN rule, and Gauss_m1 and Gauss_m02 for the SVM.

6 Discussion

In this report two main causes of non-Euclidean behavior have been identified: non-intrinsic and intrinsic ones. The former are related to computational and computational problems. In case there are no other effects Euclidean representations can be expected asymptotically for increasing computational and observational resources. The latter, the intrinsic causes will remain to yield non-Euclidean dissimilarity matrices.

The question now raises whether the correction and classification procedures should be different for these two cases. It may be argued that if it is to be expected that for some circumstances an Euclidean space is appropriate, that then an approximation of this space by some correction of the originally non-Euclidean dataset may approximate the desired representation well. In case of intrinsicly non-Euclidean problems approximative Euclidean spaces might be less effective.

Table 3, in which the NN classifier has been used, shows a few datasets for which the original space (PES) hardly could be improved: the two modified Hausdorff problems, Polydist_m57 and NIST_m38, Newsgroup, Gauss_m02 based on an extreme Minkovski-p measure and CoilSame, the worst of the two graph matching procedures. For the NN rule corrections of the original dissimilarities are not needed or not helpful. This holds less clearly also for the SVM results. Although the performances of the SVM are often better than those of the NN classifier, we should take into account that its results in the PES are based on an indefinite kernel, thereby not optimal and consequently not comparable to the other, Euclidean spaces.

The SVM relates all objects to each other by its use of a kernel. It is thereby a global procedure. The NN rule operates very locally and may thereby be more suitable to study the effect of small changes in the topology than the more powerful SVM. At the end we are, of course, mainly interested in classification performances.

The experiments in this report are in addition to the ones we presented in [29] and in [31]. In both papers we studied more subtle correction procedures in which we interpolated between the PES and several Euclidean spaces. Some of these interpolations change the dissimilarities in a monotonous way, by which the NN classification results don't change and thereby also don't improve. Such transformations are nevertheless important they show that for every classifier in the PES, so on the original representation, there exist an equivalent classifier in an Euclidean space. Nevertheless, from all our experiments, presented here and elsewhere it can be concluded that for many case the pseudo-Euclidean space can be transformed in a non-topology-preserving way into an Euclidean space in which better classifiers can be computed.

In case there exist an Euclidean space in which several classifiers obtain there best results, we may conclude that the corresponding problem is not intrinsic non-Euclidean. If this space has been found by a correction or transformation of a pseudo-Euclidean space this just suggests that sufficient knowledge lacks to construct such a representation directly from an appropriate set of features or (dis)similarity measure.

References

[1] Vapnik, V.: Statistical Learning Theory. John Wiley & Sons, Inc. (1998)

[2] Fukunaga, K.: Introduction to Statistical Pattern Recognition. 2^{nd} edn. Academic Press, London (1990)

[3] Duda, R., Hart, P., Stork, D.: Pattern Classification. John Wiley and Sons (2001) 0-471-05669-3.

[4] Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: A review. IEEE Trans. Pattern Anal. Mach. Intell **22** (2000) 4–37

[5] Duin, R., Tax, D.: Statistical pattern recognition. In Chen, C.H., Wang, P.S.P., eds.: Handbook of Pattern Recognition and Computer Vision, Third Edition. World Scientific (2005) 3–24

[6] Schalkoff, R.J.: Artificial Neural Networks. McGraw-Hill Higher Education (1997)

[7] Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines. Cambridge University Press, UK (2000)

[8] Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2006)

[9] Kondor, R.I., Jebara, T.: A kernel between sets of vectors. In: ICML. (2003) 361–368

[10] Bunke, H., Sanfeliu, A., eds.: Syntactic and Structural Pattern Recognition Theory and Applications. World Scientific (1990)

[11] Duda, R.O., Hart, P.E.: Pattern Classification and Scene Analysis. Wiley, New York (1972)

[12] Theodoridis, S., Koutroumbas, K.: Pattern Recognition, 4th Edition. Academic Press (2008)

[13] Aizerman, M.A., Braverman, E.M., Rozonoér, L.I.: Theoretical foundations of the potential function method in pattern recognition learning. Automation and Remote Control **25** (1964) 821–837

[14] Schölkopf, B., Smola, A.: Learning with Kernels. MIT Press, Cambridge (2002)

[15] Shawe-Taylor, J., Cristianini, N.: Kernel methods for pattern analysis. Cambridge University Press, UK (2004)

[16] Duin, R., de Ridder, D., Tax, D.: Experiments with object based discriminant functions; a featureless approach to pattern recognition. Pattern Recognition Letters **18** (1997) 1159–1166

[17] Duin, R., Pękalska, E., de Ridder, D.: Relational discriminant analysis. Pattern Recognition Letters **20** (1999) 1175–1181

[18] Duin, R.: Relational discriminant analysis and its large sample size problem. In: ICPR. (1998) Vol I: 445–449

[19] Pękalska, E., Duin, R.P.W.: Dissimilarity representations allow for building good classifiers. Pattern Recognition Letters **23** (2002) 943–956

[20] Pękalska, E., Duin, R.: The Dissimilarity Representation for Pattern Recognition. Foundations and Applications. World Scientific, Singapore (2005)

[21] Pękalska, E., Paclík, P., Duin, R.: A Generalized Kernel Approach to Dissimilarity Based Classification. J. of Machine Learning Research **2** (2002) 175–211

[22] Pękalska, E., Duin, R.: Beyond traditional kernels: Classification in two dissimilarity-based representation spaces. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on **38** (2008) 729–744

[23] Haasdonk, B.: Feature space interpretation of SVMs with indefinite kernels. IEEE TPAMI **25** (2005) 482–492

[24] Haasdonk, B., Burkhardt, H.: Invariant kernel functions for pattern analysis and machine learning. Machine Learning **68** (2007) 35–61

[25] Haasdonk, B., Pękalska, E.: Indefinite kernel fisher discriminant. In: ICPR. (2008) 1–4

[26] Ong, C., Mary, X.and Canu, S., A.J., S.: Learning with non-positive kernels. In: Int. Conference on Machine Learning, Brisbane, Australia (2004) 639–646

[27] Ong, C.S.: Kernels: Regularization and optimization (2005)

[28] Pękalska, E., Harol, A., Duin, R., Spillmann, B., Bunke, H.: Non-euclidean or non-metric measures can be informative. In: SSPR/SPR. (2006) 871–880

[29] Duin, R., Pękalska, E., Harol, A., Lee, W.J., Bunke, H.: On euclidean corrections for non-euclidean dissimilarities. In: SSPR/SPR. (2008) 551–561

[30] Laub, J., Roth, V., Buhmann, J.M., Müller, K.R.: On the information and representation of non-euclidean pairwise data. Pattern Recognition **39** (2006) 1815–1826

[31] Duin, R., Pękalska, E.: On refining dissimilarity matrices for an improved nn learning. In: ICPR. (2008) 1–4

[32] Graepel, T., Herbrich, R., Bollmann-Sdorra, P., Obermayer, K.: Classification on pairwise proximity data. In: Advances in Neural Information System Processing 11. (1999) 438–444

[33] Pękalska, E., Duin, R.: Dissimilarity-based classification for vectorial representations. In: ICPR (3). (2006) 137–140

[34] Goldfarb, L.: A new approach to pattern recognition. In Kanal, L., Rosenfeld, A., eds.: Progress in Pattern Recognition. Volume 2. Elsevier (1985) 241–402

[35] Pękalska, E., Duin, R.P.W., Paclik, P.: Prototype selection for dissimilarity-based classifiers. Pattern Recognition **39** (2006) 189–208

[36] Bunke, H., Shearer, K.: A graph distance metric based on the maximal common subgraph. Pattern Recognition Letters **19** (1998) 255–259

[37] Dubuisson, M., Jain, A.: Modified Hausdorff distance for object matching. In: Int. Conference on Pattern Recognition. Volume 1. (1994) 566–568

[38] Bunke, H., Bühler, U.: Applications of approximate string matching to 2D shape recognition. Pattern recognition **26** (1993) 1797–1812

[39] Jain, A.K., Zongker, D.E.: Representation and recognition of handwritten digits using deformable templates. IEEE Trans. Pattern Anal. Mach. Intell **19** (1997)

[40] Duin, R., Pękalska, E.: Structural inference of sensor-based measurements. In: Structural, Syntactic, and Statistical Pattern Recognition. (2006) 41–55

[41] Pękalska, E., Haasdonk, B.: Kernel discriminant analysis with positive definite and indefinite kernels. IEEE Transactions on Pattern Analysis and Machine Intelligence (2009, accepted)

[42] Lee, W., Duin, R.: An inexact graph comparison approach in joint eigenspace. In: SSPR/SPR. (2008) 35–44

[43] Lee, W., Duin, R.: A labelled graph based multiple classifier system. (In: MCS 2009, accepted)

[44] Pękalska, E., Harol, A., Duin, R., Spillmann, B., Bunke, H.: Non-Euclidean or non-metric measures can be informative. In: S+SSPR. (2006) 871–880

[45] Scannell, J., Blakemore, C., Young, M.: Analysis of connectivity in the cat cerebral cortex. Journal of Neuroscience **15** (1995) 1463–1483