Augmented Embedding of Dissimilarity Data into (Pseudo-)Euclidean Spaces

Artsiom Harol[1], Elżbieta Pękalska[2], Sergey Verzakov[1], and Robert P.W. Duin[1]

[1] Information and Communication Theory group
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, The Netherlands
{a.harol, e.pekalska, s.verzakov, r.p.w.duin}@ewi.tudelft.nl
[2] School of Computer Science
University of Manchester, United Kingdom
pekalska@cs.man.ac.uk

Abstract. Pairwise proximities describe the properties of objects in terms of their similarities. By using different distance-based functions one may encode different characteristics of a given problem. However, to use the framework of statistical pattern recognition some vector representation should be constructed. One of the simplest ways to do that is to define an isometric embedding to some vector space. In this work, we will focus on a linear embedding into a (pseudo-)Euclidean space.

This is usually well defined for training data. Some inadequacy, however, appears when projecting new or test objects due to the resulting projection errors. In this paper we propose an augmented embedding algorithm that enlarges the dimensionality of the space such that the resulting projection error vanishes. Our preliminary results show that it may lead to a better classification accuracy, especially for data with high intrinsic dimensionality.

1 Introduction

Pattern recognition relies on the description of regularities in observations of classes of objects. How this knowledge is extracted and represented is of importance for learning. Representations which are alternative to feature-based descriptions should be studied as they may capture different characteristics of a problem we want to analyze [1,4].

An example of such a representation is a proximity representation, where every object is described by some continuous nonnegative symmetric function of two variables. Learning from such representations relies on embedding of the proximity data into some vector space. It is usually desirable to find a mapping such that the initial topology is preserved as much as possible. The simplest way to do that is to construct an isometric mapping, which preserves all given distances.

However the broad range of proximity functions, satisfying only the conditions described above, may not allow one to construct an isometric embedding into

D.-Y. Yeung et al. (Eds.): SSPR&SPR 2006, LNCS 4109, pp. 613–621, 2006.

Euclidean space. In that case one needs to look for a more general space, with smaller number of restrictions. The solutions might be to design a mapping into pseudo-Euclidean space.

Embedding algorithms are usually defined on the basis of some representation objects, called prototypes. The projection accuracy for new data is proportional to the number of dominated intrinsic dimensions, described by them. If one has sufficient amount of prototypes, the projection error is of a little significance. But, if the cost to get more data for a space representation is very high, augmented embedding might be a good solution. It reconstructs given proximity information by means of one (Euclidean) or two (pseudo-Euclidean) extra dimensions. Nevertheless, it does not help much in cases when data has large intrinsic nonlinearities, since it is based on a global linear projection.

The paper is organized as follows. In section 2, a linear embedding of distance data into a pseudo-Euclidean space is presented. In section 3 augmented embedding for proximity data is presented. Data sets with experiments are described in Section 4. Conclusions are presented in Section 5.

2 Linear Embedding in (pseudo-) Euclidean Spaces

In this section we focus on linear isometric embedding of distance-based information into pseudo-Euclidean spaces. The results also hold for Euclidean cases, i.e. when the Gram operator derived from distances is positive definite, and coincide with the classical scaling [7,8]. The technique described in this chapter is standard and can be found in [1,4].

The formalism is as follows. Suppose we have a pair $(\mathbb{X}, d)$, where $\mathbb{X}$ is a finite set of n elements equipped with a pairwise continuous non-negative symmetric distance functions d_{ij}. These distance functions define a matrix $\mathbf{D}$ of size $n \times n$.

Having these properties of proximity functions, the whole finite representation $\mathbf{D}$ can be embedded into pseudo-Euclidean space.

By definition, a *pseudo-Euclidean* space $\mathbb{R}^{(p+q)}$ [4] of signature (p, q) is a pair $(V, \mathbf{\Phi})$, where V is a vector space under the field of real numbers of dimension $(p + q)$ and $\mathbf{\Phi}$ is a non-degenerate symmetric bilinear form, which represents the generalized inner product in V. Given an orthonormal (w.r.t $\mathbf{\Phi}$) basis $e = (e_1, e_2, \cdots, e_n)$, the generalized inner product between two vectors in $\mathbf{x}, \mathbf{y} \in V$ is expressed as

$$\langle \mathbf{x}, \mathbf{y} \rangle_{pq} = \sum_{i=1}^{p} x^{(i)} y^{(i)} - \sum_{j=p+1}^{p+q} x^{(j)} y^{(j)}. \tag{1}$$

Any *pseudo-Euclidean* space admits a decomposition into a direct orthogonal sum of two non-commensurate Euclidean subspaces of dimensions p and q respectively, i.e. $\mathbb{R}^{(p+q)} = \mathbb{R}^p \dot{+} \mathbb{R}^q$. The inner product is positive definite in $\mathbb{R}^p$ and negative definite in $\mathbb{R}^q$. The *pseudo-Euclidean* space corresponds to a Euclidean space in case of $q = 0$.

From the definition it is clear that the notion of inner product in *pseudo-Euclidean spaces* is relative, since its is not necessary positive definite and the *square-distance*, defined as $\|\mathbf{x}-\mathbf{y}\|^2 = \langle \mathbf{x}-\mathbf{y}, \mathbf{x}-\mathbf{y}\rangle = (\mathbf{x}-\mathbf{y})^T \mathcal{J}_{pq}(\mathbf{x}-\mathbf{y})$ can be negative. Here, $\mathcal{J}_{pq} = \begin{pmatrix} \mathbf{I}_{p\times p} & 0 \\ 0 & -\mathbf{I}_{q\times q} \end{pmatrix}$ is the canonical matrix of the symmetric bilinear form, corresponding to the orthogonal (w.r.t $\boldsymbol{\Phi}$) basis $e = (e_1, e_2, \cdots, e_n)$ of V and $\mathbf{I}$ represents an identity matrix.

Based on linear relations between square pseudo-Euclidean distances $\mathbf{D}^2 = (d_{ij}^2)$ and inner products in $\mathbb{R}^{(p+q)}$ space [4], one can write:

$$\mathbf{D}^2 = \mathrm{diag}(\mathbf{G})\mathbf{1}^T + \mathbf{1}\mathrm{diag}(\mathbf{G})^T - 2\mathbf{G}, \tag{2}$$

where $\mathbf{1}$ is a column vector of ones and $\mathbf{G}$ is a Gram operator, defined as:

$$\mathbf{G} = \mathbf{X}\mathcal{J}_{pq}\mathbf{X}^T. \tag{3}$$

Here, $\mathbf{X}$ is a matrix of object coordinates in that space.

Assuming that only distances between a set of objects are given, the sought coordinates can be determined based on the relations between distances and inner products, as presented above. Note that having found one set of coordinates, another one can be created by a rotation and(or) a translation.

The mapping is constructed such that the origin coincides with the mean of $\mathbf{X}$. It is done by using a centering matrix $\mathbf{J} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T$. So, $\mathbf{G} = -\frac{1}{2}\mathbf{J}\mathbf{D}^2\mathbf{J}$. The underlying configuration $\mathbf{X}$ can be found as an eigendecomposition:

$$\mathbf{G} = \mathbf{Q}|\Lambda|^{1/2} \begin{pmatrix} \mathcal{J}_{pq} & \\ & 0 \end{pmatrix} |\Lambda|^{1/2}\mathbf{Q}^T, \tag{4}$$

where Λ is a diagonal matrix of the first decreasing p positive and q negative eigenvalues ($k = p + q$), followed by zero(s). $\mathbf{Q}$ is a matrix of the corresponding eigenvectors. Consequently,

$$\mathbf{X} = \mathbf{Q}_k \Lambda_k^{\frac{1}{2}} \mathbf{P}^T, \quad k \leq n, \tag{5}$$

where only k eigenvectors are taken into account. Here, $\mathbf{P}$ is some matrix, which brings the unique solution by fixing the rotation and satisfying the constraint:

$$\mathbf{P}\mathcal{J}_{pq}\mathbf{P}^T = \mathcal{J}_{pq}. \tag{6}$$

$\mathbf{X}$ in a k-dimensional space is determined from the matrix $\mathbf{D}$. If $k \ll n$ then a smaller $\mathbf{D}$ could be used to determine this k-dimensional space. The projections of new objects, represented by the distances to objects from $\mathbb{X}$ can be done by linear operations.

3 Augmented Embedding

Suppose we have selected k prototype patterns $\mathbf{x}_i \in \mathbb{X}$. We may construct $(k-1)$ dimensional pseudo-Euclidean space based on them, where each object from this set of selected objects has coordinates $\mathbf{x}_i = (x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(p)}, x_i^{(p+1)}, \ldots,$

$x_i^{(p+q)})^T$, $p+q=(k-1)$ and $i=1,\ldots,k$. Let us now assume that our configuration lives in a (k+1)-dimensional space such that we add one dimension to represent the positive subspace p and one dimension to represent the negative subspace q. The configuration (5) stays the same, except that the coordinates for these two extra dimensions are zeros. Objects from some new set $\tilde{\mathbb{X}}$ may be projected on this space by given their distances to $\mathbf{x}_i$. For every $\mathbf{x}_s \in \tilde{\mathbb{X}}$ $(s=1,\ldots,m)$ it can be done as follows:

$$d_{si}^2 = \sum_{l=1}^{p}\left(x_s^{(l)} - x_i^{(l)}\right)^2 - \sum_{l=p+1}^{k-1}\left(x_s^{(l)} - x_i^{(l)}\right)^2 + \varepsilon^2, \tag{7}$$

where $\varepsilon^2 = \varepsilon_p^2 - \varepsilon_q^2$ stands for the projection error and might be negative. We also assume that the center of mass lies in the origin: $\sum_{i=1}^{k+1}\mathbf{x}_i = 0$, remembering that the last coordinates for each prototype in our space are $x_i^{(k)} = x_i^{(k+1)} = 0$. Summing up among all k prototypes we receive the following equation:

$$\sum_{i=1}^{k} d_{si}^2 = \sum_{i=1}^{k}\sum_{l=1}^{p}\left(x_s^{(l)} - x_i^{(l)}\right)^2 - \sum_{i=1}^{k}\sum_{l=p+1}^{k-1}\left(x_s^{(l)} - x_i^{(l)}\right)^2 + k\varepsilon^2 \tag{8}$$

Opening brackets and recalling that the norm of any vector $\mathbf{x}_s$ can be expressed as:

$$\|\mathbf{x}_s\|^2 = \sum_{l=1}^{p}\left(x_s^{(l)}\right)^2 - \sum_{l=p+1}^{k-1}\left(x_s^{(l)}\right)^2 + \varepsilon^2 \tag{9}$$

we receive:

$$\|\mathbf{x}_s\|^2 = \frac{1}{k}\sum_{i=1}^{k-1}\left(d_{si}^2 - \|\mathbf{x}_i\|^2\right) \tag{10}$$

Substituting this result into equations (7) and after some computations we receive the following solution for the projected object $\mathbf{x}_s$ into $(k-1)$-dimensional space as $\mathbf{x}_s{}'$:

$$\mathbf{x}_s^{'} = \frac{1}{2}|\Lambda|^{-1}\mathcal{J}_{pq}\mathbf{X}_i^{'T}\left(\mathrm{diag}(\mathbf{G}_i) - \mathbf{d}_s^2\right), \tag{11}$$

Here $\mathbf{X}_i^{'}$ is a matrix of prototype coordinates, $\mathbf{G}_i$ is a Gram matrix for objects from $\mathbf{X}_i^{'}$ and $\mathbf{d}_s^2$ is a vector of distances from an object $\mathbf{x}_s$ to all prototypes $\mathbf{x}_i$.

One should remember that the solution for $\mathbf{x}_s^{'}$ is unique within the fixed rotation: $\mathbf{x}_s^{'} = \mathbf{Q}|\Lambda|^{1/2}\mathbf{P}^T$, that satisfies the constraint (6). Finally, the sought vector of coordinates for the projected object can be derived as follows:

$$\|\mathbf{x}_s\|^2 = \|\mathbf{x}_s^{'}\|^2 + \varepsilon^2 \tag{12}$$

On the other hand, recalling (10) and rewriting it in the matrix form:

$$\|\mathbf{x}_s\|^2 = -\frac{\mathbf{1}^T}{k}(\mathrm{diag}(\mathbf{G}_i) - \mathbf{d}_s^2), \tag{13}$$

we can derive ε^2.

However,

$$\varepsilon^2 = \varepsilon_p^2 - \varepsilon_q^2. \tag{14}$$

It means, that the all possible solutions for ε_p and ε_q, lie on a hyperbola (14) in the augmented subspace.

Our task is to optimize both positive and negative parts simultaneously to get a unique solution. It can be done in different ways. First, in a non-regularized version, one may just check the sign of ε^2 and depending on that assume the existence of only one ε_p or ε_q of the variable, calculating it as $sign(\varepsilon^2)\sqrt{|\varepsilon^2|}$. It means that the only one ε_p or ε_q encodes the projection error while the other is zero. As a result the objects will be projected directly on the axes of the augmented 2D subspace.

More advanced techniques, taking some assumptions about possible solutions, could also be constructed, assuming the simultaneous existence of both ε_p and ε_q variables. We will focus on looking for the so-called regularized normal solutions (solutions near the origin) that take the history into account, i.e. values close to the positive and negative class means, averaged among all axis in the space of dimension $(p+q)$. For this we will minimize the following functional:

$$F(\varepsilon_p, \varepsilon_q) = (\varepsilon_p - \hat{\mu_p})^2 + (\varepsilon_q - \hat{\mu_q})^2 \mapsto \min, \tag{15}$$

where

$$\begin{aligned} \hat{\mu_p} &= \frac{\mid \mu_p \mid}{p} \\ \hat{\mu_q} &= \frac{\mid \mu_q \mid}{q} \end{aligned} \tag{16}$$

expresses the averaged absolute values of positive and negative distribution means for each class in the $(k-1)$-dimensional space. This functional by the construction is convex and has a unique solution. It should be noted that the overall positive and negative means of the representation set is at the origin due to the centering procedure we have done. But, once the projection is made for the training set, the mean values μ_p and μ_q shift.

Moreover, in our regularization algorithm we choose to optimize the position of the test objects taking into account the class means from the training set. For each test object, the closest class mean is determined based on the pseudo-Euclidean distances. It means that $\hat{\mu}_p$ and $\hat{\mu}_q$ constitute now the mean vector of that class.

So, for every new object to project, the task (15) can be solved by the standard method of Lagrangian multipliers, taking into account the restriction (14).

$$L = F(\varepsilon_p, \varepsilon_q) + \lambda\left(\varepsilon^2 - \varepsilon_p^2 + \varepsilon_q^2\right), \tag{17}$$

where λ is some constant. Constructing Euler equations we receive:

$$\begin{aligned} \frac{\partial L}{\partial \varepsilon_p} &: \quad \varepsilon_p^2 = \frac{\hat{\mu_p}^2}{(1-\lambda)^2} \\ \frac{\partial L}{\partial \varepsilon_q} &: \quad \varepsilon_q^2 = \frac{\hat{\mu_q}^2}{(1+\lambda)^2} \end{aligned} \tag{18}$$

Substituting ε_p^2 and ε_q^2 we receive the following equations with respect to λ:

$$\varepsilon^2 = \frac{\hat{\mu_p}^2}{(1-\lambda)^2} - \frac{\hat{\mu_q}^2}{(1+\lambda)^2} \tag{19}$$

Solving this fourth-order equation, we get four solutions. Two of them we reject since they are imaginary. Among remaining two we select the one that brings the minimum to our functional (15).

4 Experimental Setup

Ionosphere data set. The data set describes radar returns from the ionosphere and is obtained from the UCI repository [5]. The targets are free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not; their signals pass through the ionosphere.

Received signals were processed using an autocorrelation function whose arguments are the time of a pulse and the pulse number. There were 17 pulse numbers for the used system. Instances in this database are described by 2 attributes per pulse number, corresponding to the complex values returned by the function resulting from the complex electromagnetic signal.

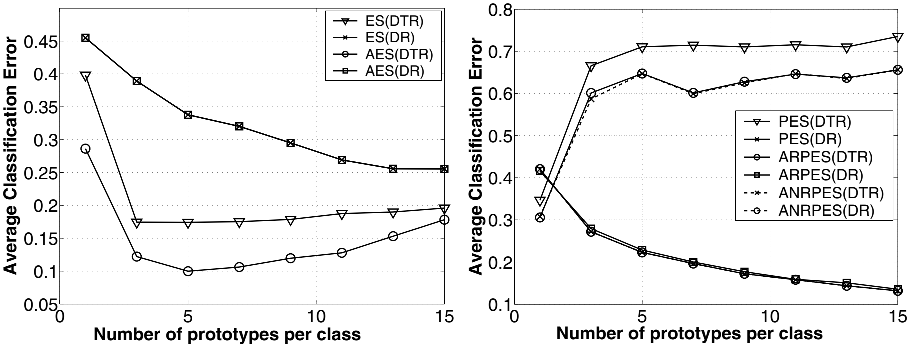
The number of instances is 351, the number of attributes is 34 plus the class attribute. All 34 predictor attributes are continuous; the 35th attribute is either "good" or "bad". This is a binary classification task with no missing values.

The dissimilarity matrix computed on the Ionosphere data set and used in our experiments is Euclidean. Moreover, distances in the matrix are scaled to be in $[0, 1]$.

Chicken pieces data set. This data set consists of 446 images of chicken pieces [2]. Each piece belongs to one of five categories, which represents specific parts of the chicken: wing (117 samples), back (76), drumstick (96), thigh and back (61), and breast (96). Each image is in binary format containing the silhouette of a particular piece. Pieces were placed in a natural way without considering orientation.

To extract string representations, some preprocessing had been done and provided to us by the group of prof. Bunke [6]. First, edge detection was performed. Secondly, the edges were approximated by straight line segments of fixed length. The sequence of angles between the segments were chosen as the string representation. Such string representations are then compared by edited distances. The cost of substitution is the absolute difference between the angles, while the costs of insertion and deletions are fixed. In our experiments we have used the segments of length 25 and the insertion and deletion costs 60. Final dissimilarity matrix computed on a data set appears to be non-Euclidean. Again, distances are rescaled to be in $[0, 1]$.

In all our experiments we use the classification error of the 1-NN classifier averaged over 20 repetitions as a performance criterion for our embedding techniques. For both data sets we set uniform prior probabilities for each of the



(a) Ionosphere data set

(b) Chicken pieces data set

Fig. 1. Averaged classification error (over 20 repetitions) for 1-NN classifier. Euclidean distance matrix is computed on Ionosphere data set, while pseudo-Euclidean distances are computed for Chicken pieces data set.

classes. Training data is divided into two parts. The first part is used for the representation, randomly chosen from the training set, and defining the pseudo-Euclidean embedding described in section 2. The remaining part is projected in this space. In such an augmented space the complete training data is used for the performance evaluation of the 1-NN rule. The test data are also projected to this augmented space and the distances to the training objects are recomputed according to the pseudo-Euclidean distance of that space. The obtained results are averaged out.

The choice of the 1-NN is justified since all high level classifiers require the construction of probabilistic models in pseudo-Euclidean spaces, which are not defined yet in pattern recognition literature, while the 1-NN rule operates directly with distances obtained via an embedding algorithm. However, the whole idea described in this paper should be seen as a first step towards the construction of advanced classification methods which are left for our future research.

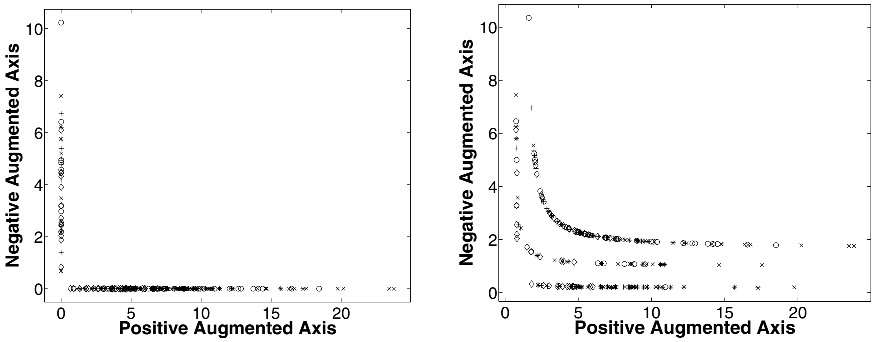
In figure 1 we use the following notation. “ES” and “PES” denote the usage of Euclidean or pseudo-Euclidean spaces. The entire training data is denoted as “DTR”, while for the selected representation set as “DR”. The regularized or not regularized versions of the augmented embedding are denoted either as “AR” or “ANR”.

In figure 1 for both different data sets the idea of augmentation helps, especially when one wants to operate with sufficiently small-dimensional spaces. However, in pseudo-Euclidean spaces the projection of the training set does not lead to better classification accuracy, like traditionally in Euclidean spaces. Moreover, it decreases drastically. Our opinion is that the data is linearly projected on a very nonlinear space, possibly equipped with curvature and torsion. In cases the representation set is small to describe all nonlinearities present in data, the classification possibilities are weak.

Standard deviations for the Ionosphere data set are less than 0.0251, while for the Chicken data set are less 0.0182.

Figures 2(a) and 2(b) illustrate the regularized vs. non-regularized versions of the augmented embedding in pseudo-Euclidean spaces, and bring an intuition behind them for future high-level pseudo-Euclidean classifiers, despite the fact that for the 1-NN rule the difference is of little significance. Here, the axes represent two augmented dimensions, the positive and the negative ones. These plots visualize how objects from the chicken pieces data are projected into this 2D augmented subspace in both cases: on the axes themselves (non-regularized version) or when their positions are optimized (regularized version).

Of course, other regularization of the augmented embedding may be constructed within this framework. For example, the position on the augmented subspace may be found in such a way, that some trained distortion function for projected training objects is minimal and applied to test data. However, this is only feasible when one has small number of prototypes (to make use of the whole idea of augmentation) but sufficiently large number of projected training objects (to train distortion parameters).



(a) No regularization

(b) With regularization

Fig. 2. Chicken pieces data set. Projection of test objects in a space, spanned by 10 prototypes. Two pictures represent augmented subspaces, constructed either with and without regularization. The use of regularization helps to prevent object overlap.

5 Conclusion

In this paper we have presented an idea of an augmented embedding which can be seen as a first step towards statistical learning in pseudo-Euclidean spaces. The method helps to reconstruct projection errors made by existing linear embedding algorithms. It may bring higher level of topology preservation than the standard methods, especially in cases of small amount of prototypes to construct a proper space. We have showed that by adding one (in a Euclidean case) or two (in a pseudo-Euclidean case) extra dimensions it becomes possible to retrieve projection errors back made by existing linear embedding methods, leading to better classification. Our experiments support this statement. However, we should accept that the projection distortion may take high values, especially in spaces with large initial non-linearities between objects.

Acknowledgments

This work is supported by the Dutch Organization for Scientific Research (NWO). The authors would like to thank Prof. Dr. Horst Bunke for providing the *Chicken pieces data set.*

References

1. Pekalska E and Duin RPW. The Dissimilarity Representation for Pattern Recognition. Foundations and Applications. World Scientific, Singapore, 2005.
2. Andreu G, Crespo A, and Valiente JM. Selecting the toroidal self-organizing feature maps (TSOFM) best organized to object recognition. Proceedings of ICNN'97, 2:1341-1346, June 1997. Houston, Texas (USA). IEEE.
3. Pekalska E, Paclik P, and Duin, RPW. A generalized kernel approach to dissimilarity based classification. Journal of Machine Learning Research, 2:175-211, 2002.
4. Goldfarb L. A New Approach to Pattern Recognition. Progress in Pattern Recognition, Elsevier Science Publishers BV, 2:241-402, 1985.
5. Newman DJ, Hettich S, Blake CL and Merz CJ. UCI Repository of machine learning databases. University of California, Irvine, Dept. of Information and Computer Sciences, 1998, http://www.ics.uci.edu/~mlearn/MLRepository.html
6. Spillmann B. Description of the Distance Matrices, University of Bern, Institute of Computer Science and Applied Mathematics, Computer Vision and Artificial Intelligence (FKI), 2004.
7. Borg I. and Groenen P. Modern Multidimensional Scaling, Springer-Verlag, 1997.
8. Cox T.F. and Cox M.A.A. Multidimensional Scaling, Chapman & Hall, 1994.