Neurocomputing 80 (2012) 19–30

Contents lists available at SciVerse ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Locally linear embedding based on correntropy measure for visualization and classification

Genaro Daza-Santacoloma [a,*], German Castellanos-Dominguez [b], Jose C. Principe [c]

[a] Universidad Antonio Nariño, Faculty of Electronic and Biomedical Engineering, Bogotá, Colombia
[b] Universidad Nacional de Colombia, Signal Processing and Recognition Group, Manizales, Colombia
[c] University of Florida, Computational NeuroEngineering Lab, Gainesville, FL, USA

ARTICLE INFO

Available online 6 November 2011

Keywords:
Nonlinear dimensionality reduction
Correntropy
Locally linear embedding
Class label information
Visualization
Classification

ABSTRACT

Linear dimensionality reduction (DR) is a widely used technique in pattern recognition to control the dimensionality of input data, but it does neither preserve discriminability nor is capable of discovering nonlinear degrees of freedom present in natural observations. More recently, nonlinear dimensionality reduction (NLDR) algorithms have been developed taking advantage of the fact that data may lie on an embedded nonlinear manifold within an high dimensional feature space. Nevertheless, if the input data is corrupted (noise and outliers), most of nonlinear techniques specially Locally Linear Embedding (LLE) do not produce suitable embedding results. The culprit is the Euclidean distance (cost function in LLE) that does not correctly represent the dissimilarity between objects, increasing the error because of corrupted observations. In this work, the Euclidean distance is replaced by the correntropy induced metric (CIM), which is particularly useful to handle outliers. Moreover, we also extend NLDR to handle manifold divided into separated groups or several manifolds at the same time by employing class label information (CLI), yielding a discriminative representation of data on low dimensional space. Correntropy LLE+CLI approach is tested for visualization and classification on noisy artificial and real-world data sets. The obtained results confirm the capabilities of the discussed approach reducing the negative effects of outliers and noise on the low dimensional space. Besides, it outperforms the other NLDR techniques, in terms of classification accuracy.

1. Motivation

The goal of the NonLinear Dimensionality Reduction (NLDR) is to represent in a low dimensional space high dimensional data. Techniques for NLDR usually assume that the data of interest lie on an embedded nonlinear manifold within a higher dimensional space. Therefore, reducing the data to their natural embedding space will reduce the dimensionality of models for data analysis. If the dimension of the manifold is low enough then objects can even be visualized (2D or 3D space), which is useful for interpreting and analyzing the underlying structure behind the measured features.

Conventional methods of dimensionality reduction for pattern classification are normally limited to the linear case [1,2], particularly, to PCA and MDS, which can not correctly discover underlying structures that lie on a nonlinear manifold. This limiting representation is overcome in [3], where the Locally Linear Embedding (LLE) technique is presented, which finds underlying data structures from non-linear manifolds.

Some other NLDR salient methods are the following: (a) *Kohonen maps* [4] implements a mapping from a higher dimensional input space to a lower dimensional map space, preserving approximately neighborhoods. The outputs are arranged according to some topology where the most common choice is a two-dimensional grid. Nonetheless, Kohonen maps also involve many free parameters, such as learning rates, initial conditions, convergence criteria, and architectural specifications [1]. (b) *Maximum Variance Unfolding* (*MVU*) [5] attempts to pull the inputs apart, maximizing the sum total of their pairwise distances without breaking (or stretching) rigid rods that connect nearest neighbors. MVU is computationally intensive due to the semidefinite program, it is very sensitive to outliers and noise not mentioning that there is not possible to map new samples after training. (c) *Isometric Feature Mapping* (*ISOMAP*) [6] is a nonlinear generalization of Multi-Dimensional Scaling (MDS) in which embeddings are optimized to preserve geodesic distances between pairs of data points, i.e. distances along the manifold from which the data is sampled. These distances are estimated by computing shortest paths through a graph associated with *k*-ary neighborhoods. ISOMAP needs considerable

* Corresponding author.
E-mail addresses: gdazas@unal.edu.co (G. Daza-Santacoloma), cgcastellanosd@unal.edu.co (G. Castellanos-Dominguez), principe@cnel.ufl.edu (J.C. Principe).

0925-2312/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2011.09.015

amount of points for a suitable estimation of the geodesic distance, but if the number of samples is large the computation of the classical MDS becomes intractable. (d) *Stochastic Neighbor Embedding* (*SNE*) [7] describes a probabilistic approach for placing high dimensional objects in a low dimensional space while the neighbor identities are preserved. A Gaussian is centered on each object and the densities under this Gaussian are used to define a probability distribution over all the potential neighbors of the object. The aim of the embedding is to approximate this distribution as well as possible. Although SNE constructs reasonably good visualizations, it is hampered by a cost function that is difficult to optimize.

Although LLE have shown to be an appropriate technique for NLDR, specially in visualization, it has some limitations when data jumps among different manifolds or when data is divided into separated groups [8], which are common cases in pattern recognition and classification. Besides, LLE does not consider class label information, which can be helpful for improving data representation making easier further analysis.

It is well known that LLE needs to ensure that chosen neighborhoods appropriately represent the manifold, which means that these neighborhoods actually must be well-sampled and lying in local linear patches. Nevertheless, the data collected from biological and industrial systems are usually corrupted by either artifacts or missing values. These observations produce low-density and unconnected neighborhoods, which distort the relations among neighbors and leads to unappropriated embeddings.

The suitable selection of the neighborhoods fundamentally depends on the distance metric employed inside the algorithm and its associated parameters. Particularly, the conventional LLE method uses the Euclidean distance as the criterion to determine neighborhoods. Since the Euclidean distance is very sensitive to artifacts and is unable to handle missing values, we seek a new metric to compare observations that would recover the relevant information, and at the same time would not be disturbed by artifacts or missing values.

Recent publications describing efforts to extract relevant information propose the use of informative cost functions as similarity measures for improving the performance of learning systems [9]. We are particularly interested in [10] where a new generalized correlation function (correntropy) is developed, and its probabilistic and geometric meaning is established [11]. The correntropy function can be understood as a similarity measure controlled by a kernel bandwidth directly related to the probability of how similar two random variables are in the bisector of the joint space. Moreover, correntropy induces a metric which is equivalent to the 2-norm if points are close, outside of this zone it behaves similarly to 1-norm, eventually, as two points are further apart,the metric saturates and becomes insensitive to distance approaching 0-norm (Fig. 1). Furthermore, the correntropy induced metric (CIM) has many interesting properties, establishing strong connections between information theoretic learning (ITL) [9], kernel methods [12], and robust statistics, particularly, robust M-estimation [13].

Therefore the artifacts (outliers) and missing values get much importance in the sample reconstruction when the Euclidian distance is employed. On the contrary, the Correntropy measure emphasizes the reconstruction of a sample with the closest neighbors, minimizing the influence of outliers.

Inspired by ITL, we are interested in exploiting the advantages of the CIM, that is robust to outliers and will improve neighborhood information. We propose a new technique for NLDR called Correntropy Locally Linear Embedding (Correntropy LLE) that improves the performance of LLE, when noisy data are employed.

Moreover, we extend LLE to deal with several manifolds (classes) employing class labels as extra information for guiding the embedding, effectively yielding a discriminative representation in the low dimensional space. The goal is to find a low dimensional output conformed by underlying variables from the high dimensional input data, which are closely related to the class label information.

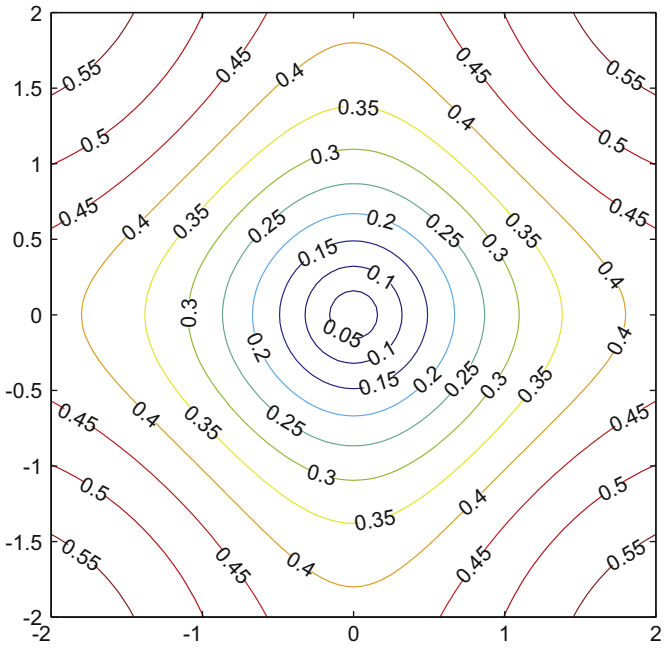


Fig. 1. Contours of the correntropy induced metric $(X;0)$ in 2-D sample space $(\sigma=1)$.

The rest of this paper is organized as follows. Section 2 briefly describes the LLE algorithm and its out-of-sample extension. Section 3 introduces the proposed Correntropy LLE approach. Next, Section 4 presents a supervised variant of the LLE-based methods to deal with multiple manifolds, which is called LLE with class label information. Section 5 shows the experimental framework used to evaluate the performance of the proposed methods. Finally, Sections 6 and 7 present a discussion of the obtained results and the conclusion of our work, respectively.

2. Locally linear embedding

Locally Linear Embedding (LLE) [3] is an unsupervised learning algorithm that attempts to compute a low dimensional embedding with the property that nearby points in the high dimensional space remain nearby and preserve the co-location with respect to one another in the low dimensional space. Let $\mathbf{X}$ be the $p\times n$ input data matrix, which contains the observation vectors $\mathbf{x}_i\in\mathbb{R}^p$, $i=1,\ldots,n$.

The algorithm has three main steps. First, the k nearest neighbors per point as measured by Euclidean distance are found. Second, each point is represented as a weighted linear combination of its neighbors, minimizing

$$\varepsilon(\mathbf{W})=\sum_{i=1}^{n}\left\|\mathbf{x}_i-\sum_{j=1}^{n}w_{ij}\mathbf{x}_j\right\|^2, \tag{1}$$

subject to a sparseness constraint $w_{ij}=0$, if $\mathbf{x}_j$ is not k-neighbor of $\mathbf{x}_i$, and an invariance constraint $\sum_{j=1}^{n}w_{ij}=1$. In the third step, the low-dimensional embedding is calculated by minimizing

$$\Phi(\mathbf{Y})=\sum_{i=1}^{n}\left\|\mathbf{y}_i-\sum_{j=1}^{n}w_{ij}\mathbf{y}_j\right\|^2, \tag{2}$$

subject to $\sum_{i=1}^{n}\mathbf{y}_i=\mathbf{0}$, and $\sum_{i=1}^{n}\mathbf{y}_i\mathbf{y}_i^\top/n=\mathbf{I}$, where $\mathbf{y}_i\in\mathbb{R}^m$ are the low dimensional output vectors.

Let $\mathbf{M}=(\mathbf{I}_{n\times n}-\mathbf{W}^\top)(\mathbf{I}_{n\times n}-\mathbf{W})$, hence Eq. (2) can be rewritten as

$$\Phi(\mathbf{Y})=\mathrm{tr}(\mathbf{Y}^\top\mathbf{M}\mathbf{Y})\quad \text{s.t.}\quad \begin{cases}\mathbf{1}_{1\times n}\mathbf{Y}=\mathbf{0}_{1\times m},\\ \dfrac{1}{n}\mathbf{Y}^\top\mathbf{Y}=\mathbf{I}_{m\times m}.\end{cases} \tag{3}$$

It is possible to calculate $m+1$ eigenvectors of $\mathbf{M}$, which are associated with the ordered spectrum of eigenvalues. The first eigenvector is the unit vector which is discarded. The remaining m eigenvectors constitute the m embedding coordinates found by LLE.

2.1. Out-of-sample extension of LLE

In order to extend the results of LLE to new objects in the input space, an explicit mapping between the high and low dimensional spaces can be studied. That mapping must not need an expensive eigenvector calculation for each new query. In [1], the authors propose two approaches (parametric and non-parametric) for mapping new input samples to the low dimensional space. Particularly, in this work, we consider the non-parametric model. To compute the output $\mathbf{y}$ for a new input $\mathbf{x}$: (i) identify the k nearest neighbors of $\mathbf{x}$ among the training inputs; (ii) compute the linear weights w_j that best reconstruct x from its neighbors, subject to the sparseness and the invariance constraints (see Eq. (1)); (iii) output $\mathbf{y}=\sum_{j=1}^{n}w_j\mathbf{y}_j$, where the sum is over the outputs corresponding to the neighbors of $\mathbf{x}$ (previously computed).

3. Correntropy locally linear embedding

As mentioned above, the problem with traditional NLDR schemes like LLE is that they traditionally employ the Euclidean distance metric to determine neighbors within local patches on the manifold. While the Euclidean distance is appropriate for measuring the distance between uncorrupted samples, it is less appropriate for measuring the distance between noisy samples (or data with outliers). Non-Euclidean distance measures such as mutual information, entropy correlation coefficient, and the relative entropy have been shown to be more appropriate than L2 norm for measuring the similarity between corrupted data samples [14]. We propose to use a generalized correlation cost function called cross-correntropy [11] which contains higher-order moments of the probability density function (pdf), but it is much simpler to estimate directly from samples than conventional moment expansions.

Let $\mathbf{X}$ be the input data matrix of size $n\times L$, where the given object $\{x_i[l],\ l=1,\ldots,L\}$ is an univariate stochastic process of L samples, and $i=1,\ldots,n$ indexes the objects (input data). Similarly as in the LLE algorithm, observations can be approximated as combinations of their nearest neighbors and then be mapped to a lower dimensional space ($m,m\le L$) which preserves data local geometry.

At the beginning, it is necessary to find the k nearest neighbors of each observation, as measured by cross-correntropy $\mathcal{V}_\sigma$. As it was stated, cross-correntropy is a generalized similarity measure between two arbitrary scalar random variables X_i and X_j controlled by a kernel bandwidth [11], so that

$$\mathcal{V}_\sigma(X_i,X_j)=\mathbf{E}[\kappa_\sigma(X_i-X_j)], \tag{4}$$

where κ_σ denotes a kernel with an specific bandwidth σ which acts as a zoom lens, controlling the observation window in which similarity is assessed, providing an effective mechanism to avoid distortions in the estimation of the neighborhoods. In this work, σ is initialized by means of the Silverman's rule [15] and it is scaled by a suitable value in order to improve the low dimensional representation under visualization or classification criteria.

In practice, the joint pdf is unknown and only a finite number of samples $\{\mathbf{x}_i,\mathbf{x}_j\,|\,x_i[l],x_j[l]\}_{l=1}^{L}$ are available, leading to the sample estimator of correntropy

$$\hat{\mathcal{V}}_{n,\sigma}(\mathbf{x}_i,\mathbf{x}_j)=\frac{1}{L}\sum_{l=1}^{L}\kappa_\sigma(x_i[l]-x_j[l]). \tag{5}$$

Employing a Gaussian kernel, it is possible to rewrite (5) as

$$\hat{\mathcal{V}}_{n,\sigma}(\mathbf{x}_i,\mathbf{x}_j)=\frac{1}{L}\sum_{l=1}^{L}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(\frac{-(x_i[l]-x_j[l])^2}{2\sigma^2}\right), \tag{6}$$

where σ is the kernel bandwidth for a Gaussian kernel.

The Correntropy induces a metric in the sample space (CIM) as in Fig. 1, where the contours of constant distances to the center of the space are being plotted. As we can observe, the CIM is a continuous blend of L2 and L0, that is, the metric saturates when the distance from the center increases which emphasizes the reconstruction of a sample on the closest points, minimizing the influence of the outliers.

Once the neighborhoods are established, each object is represented as a weighted combination of its neighbors, in this case, we calculate weights $\mathbf{W}$ that maximize the cost function:

$$J(W)=\sum_{i=1}^{n}\frac{1}{L}\sum_{l=1}^{L}\kappa_\sigma\left(x_i[l]-\sum_{j=1}^{n}w_{ij}x_j[l]\right), \tag{7}$$

subject to a sparseness constraint $w_{ij}=0$ if $\mathbf{x}_j$ is not one of the k-nearest neighbors of $\mathbf{x}_i$, and an invariance constraint $\sum_{j=1}^{n}w_{ij}=1$. Now, considering a particular object $\{\mathbf{x}\,|\,x[l],l=1,\ldots,L\}$, and let $\mathbf{V}$ be a matrix of size $k\times L$, which contains the k-nearest neighbors of $\mathbf{x}$, where each row corresponds to a neighbor object. Using column vectors for writing $\mathbf{V}$:

$$\mathbf{V}=[\mathbf{v}[1]\ \mathbf{v}[2]\ \cdots\ \mathbf{v}[l]\ \cdots\ \mathbf{v}[L]], \tag{8}$$

where

$$\mathbf{v}[l]=[v_1[l]\ v_2[l]\ \cdots\ v_j[l]\ \cdots\ v_k[l]]^\top. \tag{9}$$

Let be $\mathbf{w}$ a column vector of size $k\times 1$, which contains the representation weights of the data $\mathbf{x}$ as a function of its k nearest neighbors $\mathbf{V}$, then we wish

$$\max_{\mathbf{w}}J(\mathbf{w})=\max_{\mathbf{w}}\left(\frac{1}{L}\sum_{l=1}^{L}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(\frac{-(x[l]-\mathbf{w}^\top\mathbf{v}[l])^2}{2\sigma^2}\right)\right)\quad \text{s.t. } \mathbf{1}^\top\mathbf{w}=1, \tag{10}$$

where $\mathbf{1}^\top=[1\ 1\ \cdots\ 1]$.

In order to optimize (10), it is possible to employ a Sequential Quadratic Optimization (SQO) strategy, in particular we use the Lagrange–Newton method [16]. In this case, to maximize $J(\mathbf{w})$ in (10) is equivalent to minimize $-J(\mathbf{w})$, then $f(\mathbf{w})=-J(\mathbf{w})$, that is

$$\min_{\mathbf{w}}f(\mathbf{w})=\min_{\mathbf{w}}\left(-\frac{1}{L}\sum_{l=1}^{L}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(\frac{-(x[l]-\mathbf{w}^\top\mathbf{v}[l])^2}{2\sigma^2}\right)\right)\quad \text{s.t. } \mathbf{1}^\top\mathbf{w}=1, \tag{11}$$

which could be solved via a series of approximations of the form:

$$f(\mathbf{w}+\boldsymbol{\delta})\simeq q(\boldsymbol{\delta})=\tfrac{1}{2}\boldsymbol{\delta}^\top\Omega\boldsymbol{\delta}+f'(\mathbf{w})^\top\boldsymbol{\delta}+f(\mathbf{w}), \tag{12}$$

subject to

$$c(w+\boldsymbol{\delta})\simeq r(\boldsymbol{\delta})=\mathbf{J}_{c1}(\mathbf{w})^\top\boldsymbol{\delta}+c_1(\mathbf{w}), \tag{13}$$

where $c_1(\mathbf{w})$ is the equality constraint in (11) and $\mathbf{J}_{c1}(\mathbf{w})$ is the Jacobian of the constraint $(\mathbf{J}_{c1})_j=\partial c_1/\partial w_j$.

Once $\mathbf{w}$ weights for all the input objects are calculated, a matrix $\mathbf{W}$ of size $n\times n$ is obtained and it is possible to find the data embedding to the low dimensional space. The low dimensional output $\mathbf{Y}$ can be

found by minimizing (2), which is done in the same way as in the third step of the LLE algorithm (Section 2).

In a more general framework, the CIM shows that it bears a close relationship with M-estimation [11]. M-estimation is a generalized maximum likelihood method to estimate parameters θ under the cost function $\min_\theta \sum_{i=1}^{N} \rho(e_i|\theta)$, where ρ is a differentiable function. This general estimation is also equivalent to a weighted least square problem as $\min_\theta \sum_{i=1}^{N} w(e_i)e_i^2$, where the weight function $w(e)$ is defined by $w(e)=\rho'(e)/e$ where ρ' is the derivative of ρ. Defining $\rho(e)=(1-\exp(-e^2/2\sigma^2))/(\sigma\sqrt{2\pi})$, it corresponds to the kernel of the error. This means that large errors get larger attenuation thus the estimation is resistant to outliers as in correntropy cost [17]. Notice that the weighting function is solely determined by the choice of ρ.

4. LLE with class label information (LLE+CLI)

In classification tasks, the interest is to analyze data which do not constitute necessarily just one manifold. In this sense, we have to fit account for several patterns, e.g. a biological signal which contains normal and pathological behaviors, or images of handwritten digits, which are depicting the natural numbers from 0 up to 9. In these cases, attempting to create a single manifold that represents the whole data set can be unfeasible.

Thus, we extend the LLE approach to deal with several manifolds, employing class labels as extra information to guide the procedure of dimensionality reduction. This approach allows for the construction of a nonlinear dimensionality reduction algorithm that preserves the local geometry of the data, and provides a discriminative strategy during the embedding procedure, improving the visualization and/or classification results in comparison to conventional LLE or other topologically constrained NLDR techniques.

Similar efforts on supervised locally linear embedding [2,18–20] have been already presented. These approaches drop the main objective of the LLE algorithm; they leave behind the local structure preservation, placing the objects of different classes very far away and losing the topology defined by high dimensional input data. The popular supervised LLE version presented in [18], that we named as δ-LLE, attempts to map separately the within-class structure from between-class structure. In order to conform the neighborhoods, δ-LLE computes the distance between objects as $\tilde{d}_{ij}=d_{ij}+\delta\max(d)\Delta_{ij}$, where d is the Euclidean distance, $\Delta_{ij}=1$ if the class label of $\mathbf{x}_i$ is the same of $\mathbf{x}_j$, else $\Delta_{ij}=0$, and $\delta\in[0,1]$ controls the amount to which class information should be incorporated.

Unlike the previously mentioned works, the supervised NLDR approach of [21] defines a dissimilarity function between points considering class label information, which preserves the local structure of the data. Nevertheless, it has two new free parameters that control the excessive increase of the dissimilarity measure and the overlap between classes, respectively. However, no well founded strategy for parameter selection exists.

Our approach looks for relevant underlying variables behind the observed high dimensional data which are related the class label information. These underlying variables span a low dimensional space, where the objects can be reconstructed as locally linear combinations of their neighbors, preserving the local geometry relations and the global structure of the manifold. Hence, we want to separate the classes as much as possible without distorting the local geometry relations.

Consider a data set with n samples and a particular low dimensional point $\mathbf{y}_i$, which belongs to the class A of n_A elements where $n_A<n$. Also, consider the class B conformed by the data set elements which do not belong to A, as can be seen in Fig. 2. Now

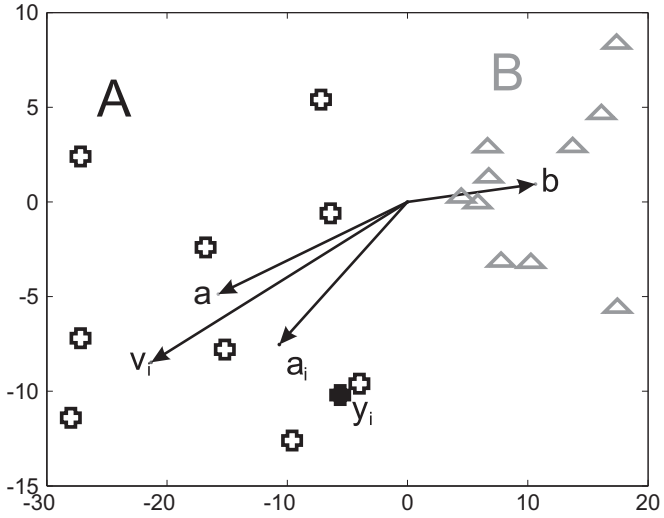


Fig. 2. Classes separation.

we set the vectors

$$\mathbf{a}=\frac{1}{n_A-1}\sum_{\substack{j=1\\ i\neq j}}^{n}\tau_A(j)\mathbf{y}_j \quad \text{and} \quad \mathbf{b}=\frac{1}{n-n_A}\sum_{j=1}^{n}\tau_R(j)\mathbf{y}_j, \tag{14}$$

$$\begin{aligned}\tau_A(j)&=\begin{cases}1, & \mathcal{P}(\mathbf{y}_i)=\mathcal{P}(\mathbf{y}_j),\\ 0 & \text{otherwise,}\end{cases}\\ \tau_B(j)&=\begin{cases}1, & \mathcal{P}(\mathbf{y}_i)\neq\mathcal{P}(\mathbf{y}_j),\\ 0 & \text{otherwise}\end{cases}\end{aligned} \tag{15}$$

where $\mathcal{P}(\cdot)$ is a function that determines the class label of the objects. The $\mathbf{a}$ vector corresponds to the mean of the class A computed without the point $\mathbf{y}_i$; A is also called the *inner-class*, because it is composed by all the elements that belong to the same class excluding $\mathbf{y}_i$. Moreover, $\mathbf{b}$ corresponds to the mean of the class B, which we call the *outer-class*.

In order to keep the elements of the inner-class close together and to reduce the effect of possible outliers, we compute the vector $\mathbf{a}_i=\frac{1}{2}(\mathbf{a}+\mathbf{y}_i)$, which helps to correct the position of $\mathbf{y}_i$ relative to the mean vector $\mathbf{a}$. Besides, to separate the classes, it is necessary to find the outputs $\mathbf{y}$ that maximizes the dissimilarity between the vectors mean vectors $\mathbf{a}_i$ and $\mathbf{b}$ as

$$\max_{\mathbf{Y}}\{\|\mathbf{a}_i-\mathbf{b}\|^2\}. \tag{16}$$

Eq. (16) means that we are pushing away each point $\mathbf{y}_i$ by maximizing the dissimilarity between classes. Additionally, note that the vector $\mathbf{v}_i=\mathbf{a}_i-\mathbf{b}$ points the direction of the force which must be applied to the point $\mathbf{y}_i$ in order to separate it from the outer-class, while it remains close to the other elements of the inner-class. In general, taking into account all the points of the data set, Eq. (16) can be rewritten as

$$\max_{\mathbf{Y}}\left\{\frac{1}{2}\sum_{i=1}^{n}\left\|\mathbf{y}_i-\left(\frac{2}{n-n_A}\sum_{j=1}^{n}\tau_B(j)\mathbf{y}_j-\frac{1}{n_A-1}\sum_{\substack{j=1\\ i\neq j}}^{n}\tau_A(j)\mathbf{y}_j\right)\right\|^2\right\}. \tag{17}$$

Since the main purpose is to pull apart objects from different classes while holding the neighborhood relationships of the data, we reformulate the conventional LLE algorithm by modifying the cost function (2), adding the term (17) that measures and

maximizes the separation between classes, that is

$$\min_{\mathbf{Y}} \Psi(\mathbf{Y},\beta) = \min_{\mathbf{Y}} \left\{ \sum_{i=1}^{n} \left\| \mathbf{y}_i - \sum_{j=1}^{n} w_{ij}\mathbf{y}_j \right\|^2 - \beta \sum_{i=1}^{n} \left\| \mathbf{y}_i - \sum_{j=1}^{n} \gamma_{ij}\mathbf{y}_j \right\|^2 \right\}, \tag{18}$$

subject to $\sum_{i=1}^{n} \mathbf{y}_i = \mathbf{0}$ and $\sum_{i=1}^{n} \mathbf{y}_i\mathbf{y}_i^\top/n = \mathbf{I}_{m\times m}$, where $\mathbf{Y}$ is the embedding data $n \times m$ matrix (being $m \le p$), and $\mathbf{y}_i \in \mathbb{R}^m$ is the output sample vector. Furthermore

$$\gamma_{ij} = \begin{cases} 0 & \text{if } i = j, \\ \dfrac{2}{n - n_{\mathcal{P}}(\mathbf{y}_i)} & \text{if } \mathcal{P}(\mathbf{y}_i) \neq \mathcal{P}(\mathbf{y}_j), \\ -\dfrac{1}{n_{\mathcal{P}}(\mathbf{y}_i) - 1} & \text{if } \mathcal{P}(\mathbf{y}_i) = \mathcal{P}(\mathbf{y}_j), \end{cases} \tag{19}$$

being $n_{\mathcal{P}(\mathbf{y}_i)}$ the number of elements of the inner-class.

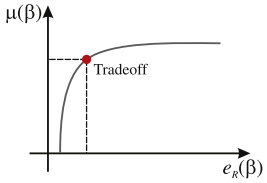


Fig. 3. Tradeoff between reconstruction error and margin.

The penalty term β is a tradeoff between the preservation of the local geometry of the high dimensional data and the representation induced by the class labels.

In particular, the first term in (18) computes the reconstruction error of the objects in the low dimensional space. The second term maximizes the dissimilarity between the inner-class and the outer-class. To solve the minimization problem, it is possible to rewrite (18) as

$$\min_{\mathbf{Y}} \Psi(\mathbf{Y},\beta) = \min_{\mathbf{Y}} \{\mathrm{tr}(\mathbf{Y}^\top(\mathbf{M} - \beta\tilde{\mathbf{M}})\mathbf{Y})\}$$
$$\text{s.t.} \begin{cases} \mathbf{1}_{1\times n}\mathbf{Y} = \mathbf{0}_{1\times n}, \\ \dfrac{1}{n}\mathbf{Y}^\top\mathbf{Y} = \mathbf{I}_{m\times m}, \end{cases} \tag{20}$$

where $\mathbf{M}$ is defined as in Section 2 and $\tilde{\mathbf{M}} = (\mathbf{I}_{n\times n} - \Gamma^\top)(\mathbf{I}_{n\times n} - \Gamma)$, while Γ is the matrix of size $n \times n$ with elements γ_{ij} according to (19).

It is possible to calculate the m eigenvectors of $\mathbf{M} - \beta\tilde{\mathbf{M}}$, which are associated to the m smallest eigenvalues after discarding the eigenvector related to some eigenvalue equal or close to zero. These eigenvectors constitute the m embedding coordinates required to the low dimensional mapping.

4.1. Automatic tradeoff choice

The β parameter in (18) is a tradeoff between the reconstruction error and the margin between objects belonging to different classes. If $\beta = 0$, we have the original mapping of LLE, and as β

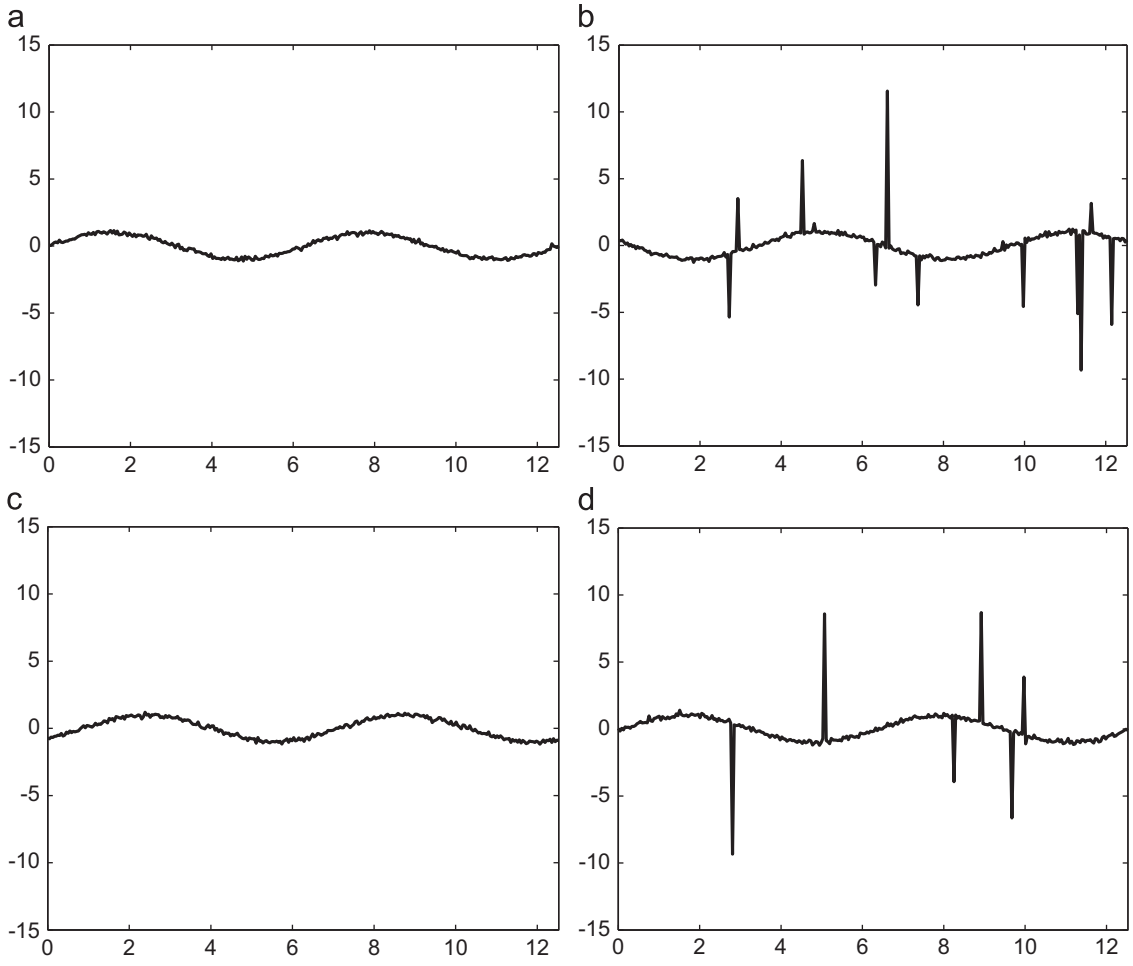


Fig. 4. Examples on the artificial data set: (a) $\phi = 0^\circ$, (b) $\phi = 165^\circ$, (c) $\phi = 310^\circ$, (d) $\phi = 355^\circ$.

increases the separation between classes is larger. Nonetheless, the size of the low dimensional space is bounded by the constraint $\sum_{i=1}^{n} \mathbf{y}_i \mathbf{y}_i^\top / n = \mathbf{I}_{m\times m}$, for this reason, when the value of β is large the low dimensional representation of the manifolds (classes) is deformed, because the points are pushed against the limits of the space.

For a given β, it is possible to find the output $\mathbf{Y}_\beta$ that minimizes the cost function (18). Next, the reconstruction error e_R and the margin μ can be also computed as a function of $\mathbf{Y}_\beta$, for each value of β, as

$$e_R(\beta) = \mathrm{tr}(\mathbf{Y}_\beta^\top \mathbf{M} \mathbf{Y}_\beta),$$

$$\mu(\beta) = \mathrm{tr}(\mathbf{Y}_\beta^\top \tilde{\mathbf{M}} \mathbf{Y}_\beta). \tag{21}$$

This procedure requires the simultaneous minimization of the reconstruction error $e_R(\beta)$ and the maximization of the margin $\mu(\beta)$. For such a purpose, we consider the parametric plot $e_R(\beta)$ versus $\mu(\beta)$ (Fig. 3) as a tool to study the behavior of these quantities [22]. Similar to the L-curve criteria for Tikhonov regularization, the point with maximum curvature is a good choice for the parameter β because it balances the two errors being plotted [23].

5. Experimental methodology

In order to verify the proposed improvements on the nonlinear dimensionality reduction technique, we set two kind of experiments.

5.1. Testing the correntropy LLE algorithm

First, we test the embedding procedure based on the Correntropy LLE algorithm discussed in Section 3, on visualization tasks, particularly, for non-Gaussian noisy data sets. We employ one artificial and one real-world data bases to visually determine whether the embedding is correctly calculated and to confirm the capabilities of the proposed approach.

The artificial data set consists of $n=72$ different objects $f[l]_i = \sin(lT+\phi_i)+Z_1$, where $Z_1 \sim \mathcal{N}(\mu=0,\sigma=0.1)$, $\phi_i = 0°,5°,\ldots,355°$, $l=1,2,\ldots,300$, and $T=4\pi/300$. Additionally, half of the objects have been corrupted with artifacts, such that these objects are defined by $f[l]_i = \sin(lT+\phi_i)+Z_1+Z_2\cdot B$, where $Z_2 \sim \mathcal{N}(\mu=0, \sigma=6)$ and $B \sim$ Bernoulli$(p=0.03)$. Some examples of these objects are presented in Fig. 4.

The real-world data set depicting 72 pictures of the Pitcher, belongs to COIL-100 [24]. Pictures are taken while the object is rotated 360° in intervals of 5°. Fig. 5 shows some examples. We transform original color images to gray scale and we crop them to 64×64, so that $L=4096$. Moreover, all images were corrupted with salt and pepper noise, affecting 5% of their pixels.

For visualization, we compare Correntropy LLE, against LLE, PCA, and Maximum Variance Unfolding (MVU) [5]. The dimension of the output space is set to $m=2$ for the artificial data set and $m=3$ for the real-world data set. Moreover, the number of nearest neighbors is chosen using the approach in [25], which computes a specific number of neighbors for each input object.

In Fig. 6(a–d) the embedding results are presented for the artificial data set using PCA, MVU, LLE with Euclidean distance, and Correntropy LLE, respectively. Moreover, the comparisons of the embedding results for Pitcher database are shown in Fig. 7. Color bar of Figs. 6(e) and 7(h) helps to understand the rotation of the objects in the low dimensional space.

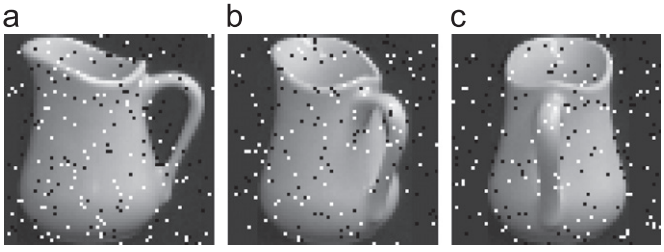


Fig. 5. Pitcher data set: (a) Pitcher 0°, (b) Pitcher 45°, (c) Pitcher 90°.

5.2. Testing correntropy LLE with class label information

These experiments test high dimensional input data containing several classes. The main idea is to validate visualization and classification results by taking into account class labels, adding information to the mapping.

Three real-world databases are employed for the second kind of experiments. The two first databases are subsets of the full MNIST database, which consists of 28×28 pixels pictures, we employ three classes (handwritten digits 3, 5, and 8) and 10 classes, distributed as shown in Table 1. Each picture was also corrupted with salt and pepper noise, affecting 8% of its pixels. Some examples of these pictures are shown in Fig. 8. The MNIST data set is tested for visualization and classification purposes.

Our approach Correntropy LLE+CLI is compared against: PCA, MVU, ISOMAP [26], unsupervised LLE, Correntropy LLE, and LLE added with class label information (LLE+CLI). These results are shown in Figs. 9 and 10. The dimension of the output space is set

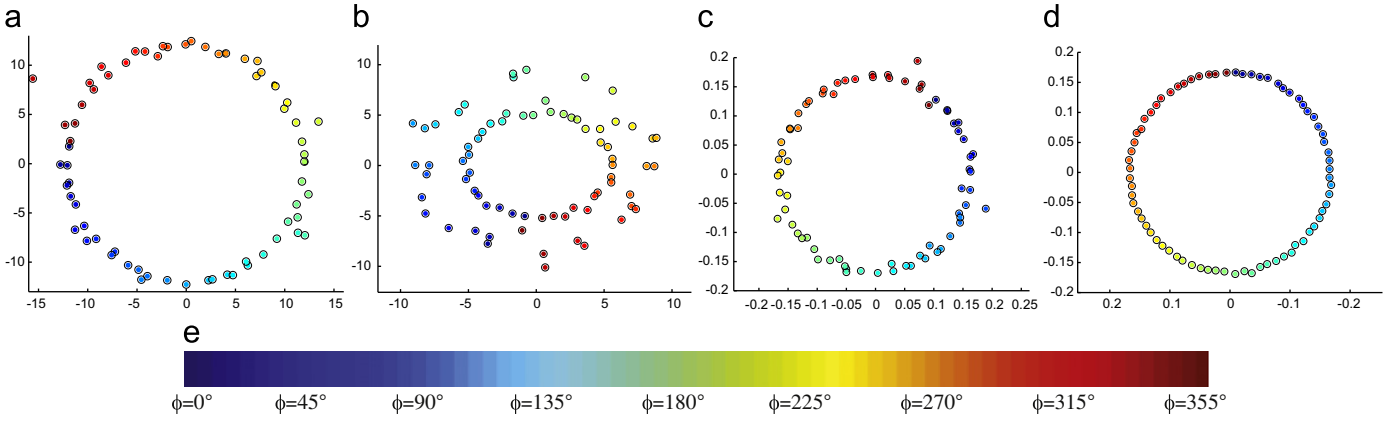


Fig. 6. Embedding results on artificial data set: (a) PCA, (b) MVU, (c) LLE, (d) Correntropy LLE $\sigma=0.365$, (e) color legend. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

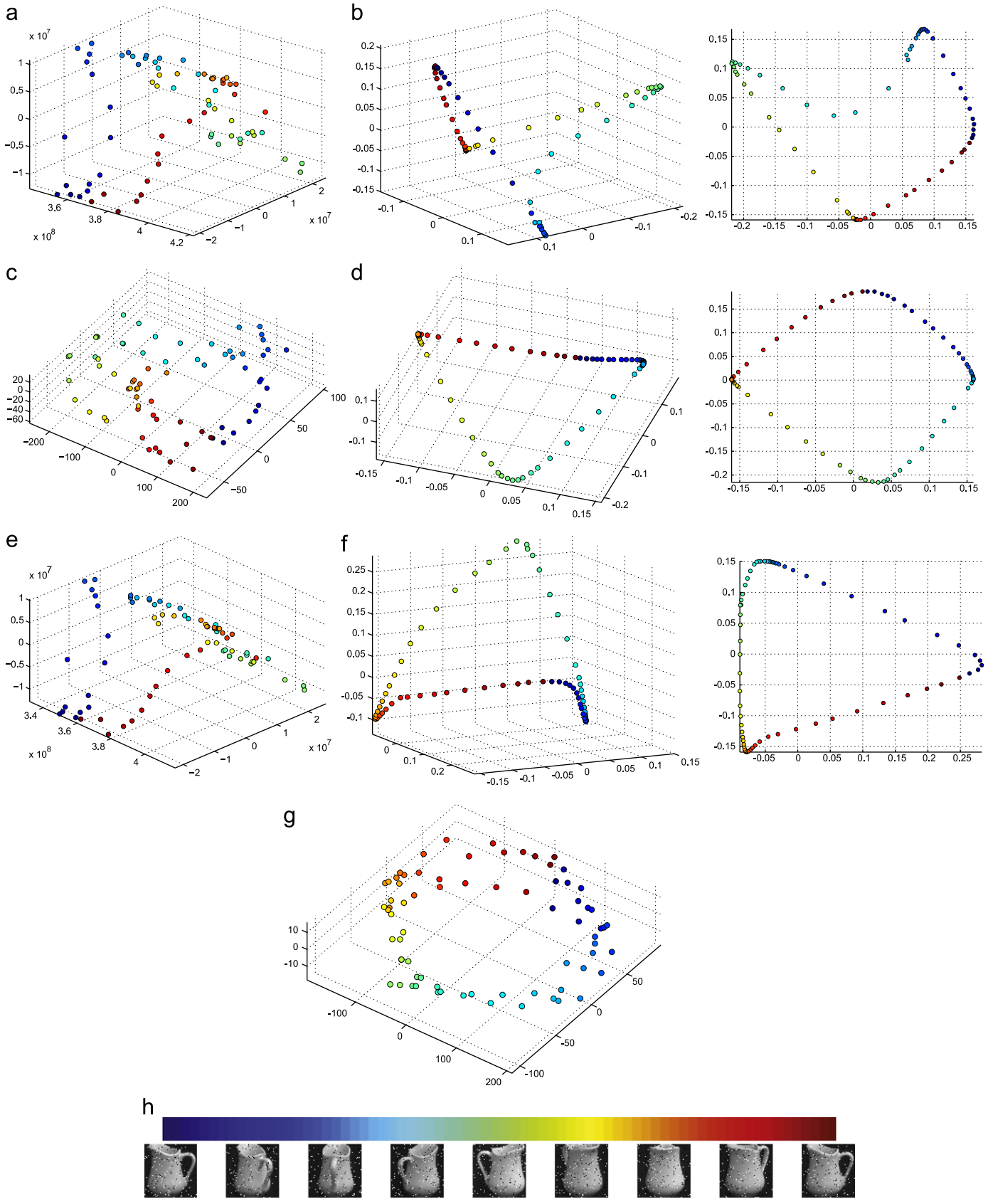


Fig. 7. Embedding results on Pitcher data set: (a) PCA, (b) LLE, *left:*3D embedding, *right:* upper view, (c) MVU, (d) Correntropy LLE, $\sigma = 25.732$, *left:*3D embedding, *right:* upper view, (e) Median filter and PCA, (f) Median filter and LLE, *left:*3D embedding, *right:* upper view, (g) Median filter and MVU, (h) color legend. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

to $m=3$. Moreover, the number of nearest neighbors is chosen by means of the method presented in [25].

The other real-world data set contains phonocardiographic (PCG) signals, which is composed of 548 (274 normals and 274 with murmur) beat sequences of 45 volunteers. Every recording lasted 12 s approximately, and was obtained from the patient standing in dorsal decubitus position. The beats are characterized by means of the spectrogram obtained by short-time Fourier transform (38 frequencies and 480 instants of time). The dimension of the input space is $p=18\,240$. The PCG database is employed for classification.

Table 1
Number of objects in the MNIST data set.

Digit	0	1	2	3	5	5	6	7	8	9
Objects	97	116	99	93	105	92	94	117	87	100

Fig. 8. Examples of MNIST data set.

The goal for classification is to map the data into a feature space in which the members from different classes are clearly separated. The generalization abilities of the classifier have to be tested following a cross-validation scheme with different sets for training and validation (ν-*folds*) [27]. These sets are chosen randomly from the full data set. In this work, 10-fold cross validation has been used, splitting the 90% of the objects for training, and the remaining 10% for testing. The training objects are used to calculate the low dimensional embedding, then, an out-of-sample extension (Section 2.1) is used to map and classify the testing objects. We compute the classification accuracy on the testing set, as the ratio of number the samples well classified to the whole number of samples. Finally, we calculate the expected accuracy rate (Acc) as the average of the 10 accuracy results computed. Moreover, the Confidence Interval (CI) is calculated on the basis of a normal distribution for the mean with unknown variance, and using a t-student distribution with nine degrees of freedom with a confidence level $\alpha=0.05$ [28]. We use the k-nearest neighbor classifier (KNNC), where k is optimized with respect to the leave-one-out error.

In this case, we compare the Correntropy LLE+CLI against PCA, ISOMAP, LLE, δ-LLE [18], Correntropy LLE, and LLE+CLI. The parameter $\delta=1$ for the δ-LLE algorithm. The intrinsic dimension of the MNIST three classes, MNIST 10 classes, and PCG databases are $m=4$, $m=6$, and $m=15$, respectively. The size of the low dimensional space is selected by examining the eigenvalue spectra of local covariance matrices, retaining 95% of the variance of the data [18]. The classification accuracy and the confidence interval are presented in Table 2.

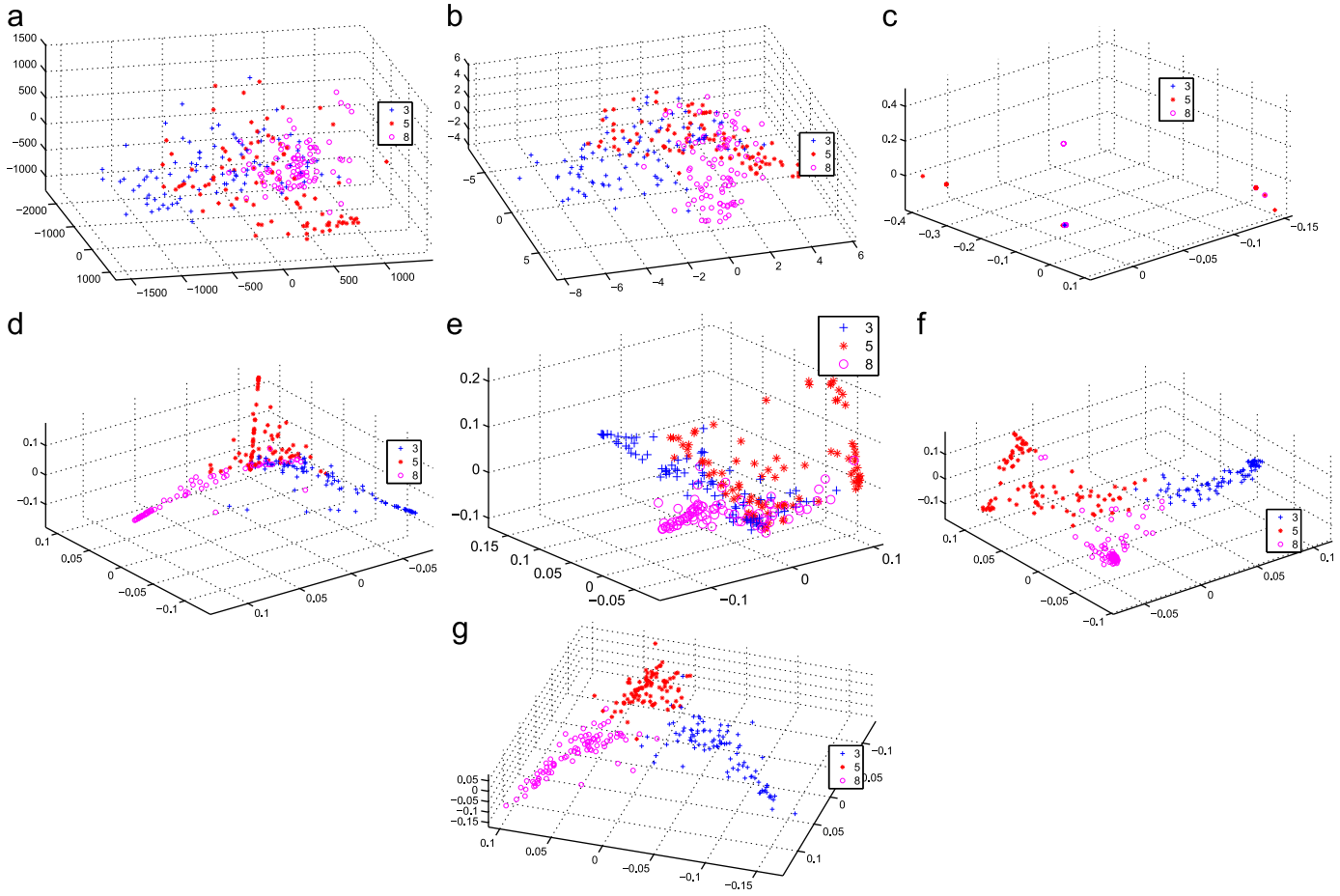


Fig. 9. Embedding results on the MNIST (three classes) database: (a) PCA, (b) MVU, (c) ISOMAP, (d) LLE, (e) Correntropy LLE, $\sigma=0.835$, (f) LLE+CLI, (g) Correntropy LLE+CLI, $\sigma=0.682$.

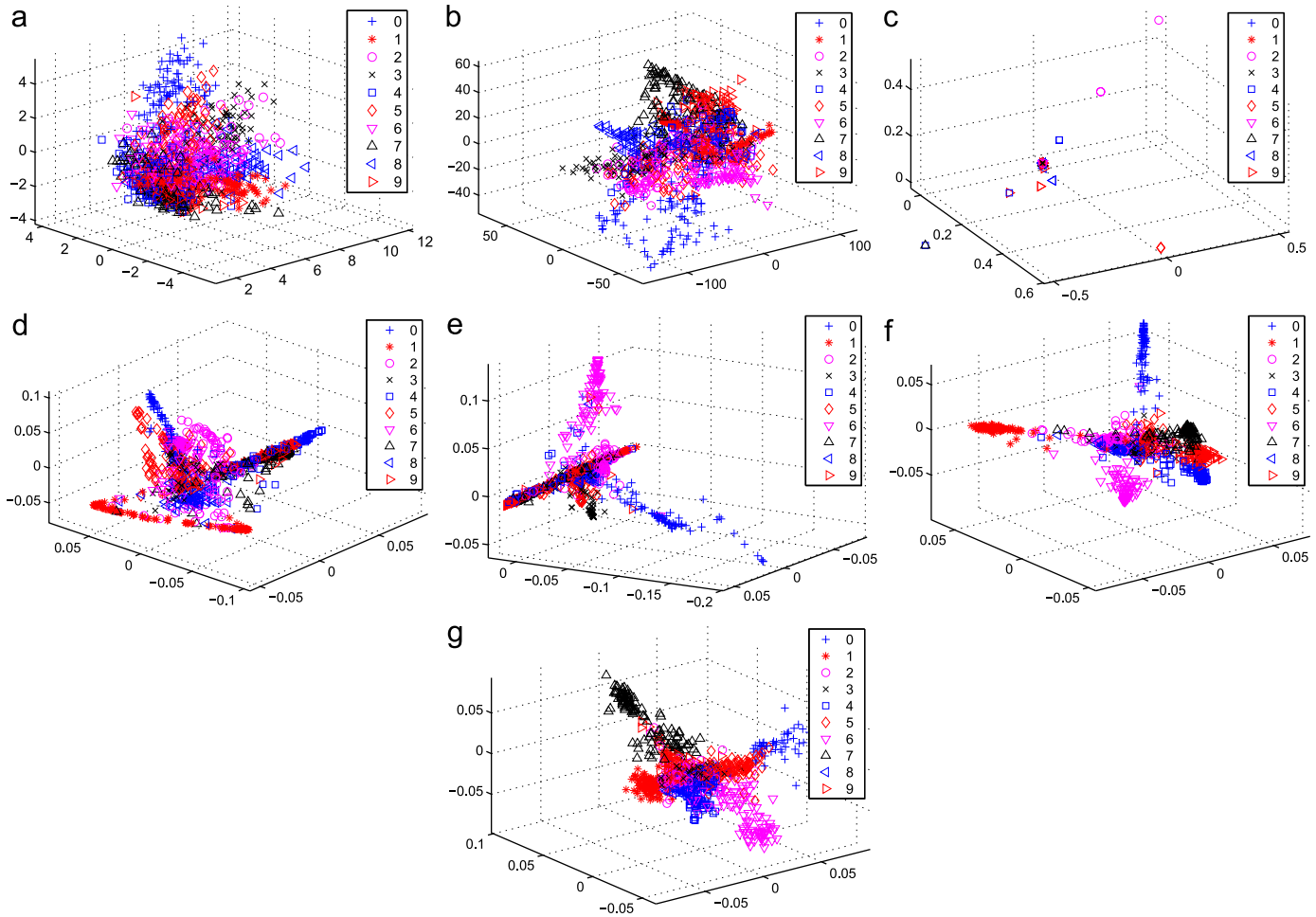


Fig. 10. Embedding results on the MNIST (10 classes) database: (a) PCA, (b) MVU, (c) ISOMAP, (d) LLE, (e) Correntropy LLE, $\sigma = 0.709$, (f) LLE+CLI, (g) correntropy LLE+CLI, $\sigma = 0.184$.

Table 2
Classification accuracy and confidence interval on MNIST and PCG databases.

	MNIST 3 classes $m=4$		**MNIST 10 classes** $m=6$		**PCG** $m=15$	
Method	**Acc**	**CI**	**Acc**	**CI**	**Acc**	**CI**
PCA	0.758	[0.706, 0.809]	0.750	[0.723, 0.778]	0.891	[0.841, 0.941]
ISOMAP	0.327	[0.285, 0.370]	0.108	[0.091, 0.125]	0.690	[0.629, 0.751]
LLE	0.765	[0.686, 0.843]	0.729	[0.699, 0.759]	0.930	[0.863, 0.998]
δ-LLE	**0.802**	[0.756, 0.848]	0.788	[0.757, 0.818]	0.945	[0.917, 0.973]
Correntropy LLE	0.743	[0.694, 0.792]	0.763	[0.739, 0.787]	0.879	[0.844, 0.914]
LLE+CLI	0.764	[0.552, 0.976]	**0.825**	[0.787, 0.871]	0.939	[0.929, 0.949]
Correntropy LLE+CLI	**0.801**	[0.720, 0.882]	**0.826**	[0.789, 0.864]	**0.960**	[0.921, 0.999]
	$\sigma = 1.346$		$\sigma = 7.238$		$\sigma = 3.89$	

In all the cases where the Correntropy measure is employed, the kernel bandwidth (σ) is initially computed by means of the Silverman's rule [15], and after that, σ is scaled by a value in the grid [0.001, 0.01, 0.1, 1, 10, 100, 1000]. There are two criteria employed for choosing the appropriate σ. In case of visualization tasks, we tune σ in the grid to maximize the number of common neighbors between the high-dimensional space local patches and the low-dimensional space local patches. For classification, the appropriate σ must maximize the accuracy rate (Acc), the behavior of the accuracy as a function of the σ parameter for the MNIST 3 classes and PCG databases are shown in Fig. 11.

6. Discussion

In the experiments where the databases are composed by one only class, the Correntropy LLE algorithm, could control the distortion caused by artifacts on the low dimensional output. Although the linear transformation (PCA) found out the low-dimensional manifold for the artificial data set (Fig. 6(a)), results were strongly affected by the type of noise (e.g. Gaussian or non-Gaussian). Additionally, PCA could not capture the underlying structure for the Pitcher data set, neither avoided distortions produced by the artifacts: its embedding results exhibit overlapped trajectories and

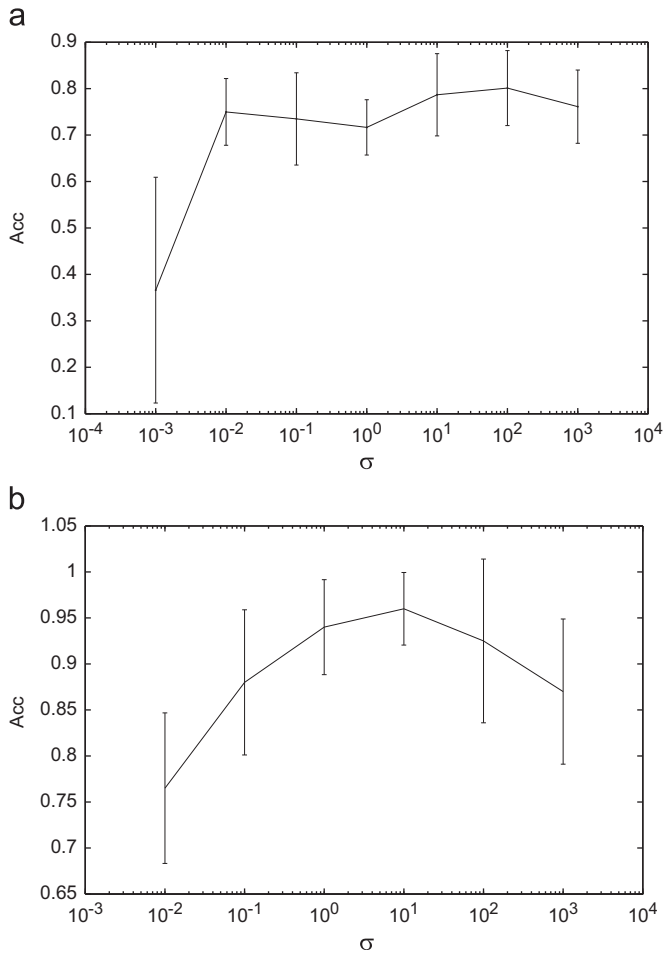


Fig. 11. Classification accuracy vs. σ for correntropy LLE+CLI: (a) MNIST 3 classes, (b) PCG.

neighborhoods conformed by dissimilar views (Fig. 7(a)). The results obtained using LLE are as discouraging as PCA, at least for the artificial data set (Fig. 6(c)). For the Pitcher, the manifold was not suitably unfolded. LLE could avoid overlapped trajectories but it could not represent the soft rotation seen in the images, see Fig. 7(b). Indeed, Euclidean distance could not capture the appropriated relationships among pixels, and we conclude that it is not suitable for measuring distances between this data set. For the artificial data set, MVU shows a double low-dimensional manifold, one for the corrupted signal and other for the signals without noise, in Fig. 6(b) a double ring is presented. Similarly, the embedding results on the real-world data set, using MVU, show distorted manifolds (Fig. 7(c)), since the points are scattered in the low dimensional space and they are not following a clear trajectory.

In contrast, results achieved with Correntropy LLE are consistent in both (artificial and real) cases. Low dimensional structures were found and the effect of the non-Gaussian noise was removed. Very small oscillations (Fig. 6(d)) are due to Gaussian noise added to signals. Underlying structures for the Pitcher data set (Fig. 7(d)) are smooth and accurately represent the hidden process behind the pictures.

We also test the Pitcher data set after a median filtering, which should be the straightforward solution for avoiding the outliers in this kind of images. However, Fig. 7(e) and (g) shows that PCA and MVU cannot discover a suitable embedding for the analyzed manifold, the low dimensional representations do not describe the motion of the object. On the other hand, when the conventional LLE algorithm is tested after the median filtering (Fig. 7(f)), the embedding results exhibit a softer rotation than when the filter is not applied (Fig. 7(b)), but the low dimensional representation is not as smooth as the achieved result by means of Correntropy LLE (Fig. 7(d)).

We choose three dimensions for represent the real-world data set because original pictures have angle changes (rotation), which need two dimensions, and besides they have scale changes, requiring at least one more dimension. The three dimensional representations in Fig. 7(b)-left, (d)-left, and (f)-left could not be conclusive in order to show what is the best embedding. Nevertheless, when we analyze Fig. 7(b)-right, (d)-right, and (f)-right, which should display the rotation of the object (it must be a circumference), clearly it is possible to see how the conventional LLE can not describes such behavior. The median filtering helps to improve the LLE representation, but it is not enough in order to appropriately illustrate the rotation. Unlike these results, the Correntropy LLE representation (Fig. 7(d)-right) seems to be similar to a circumference, which is stretched at 90° and 270° (according to the color reference). The elongation can be attributed to the fact that the space occupied by the pitcher at 90° and 270° in the pictures is more narrow than at other angles. Considering the above, the Correntropy LLE embedding is better than the low-dimensional representations achieve by LLE and prefiltered LLE.

Often the achieved results of the Correntropy LLE are better than the low dimensional representations reached by means of the other presented methods. Nonetheless, in some cases the optimization stage in Correntropy LLE (11) may stagnate in a local minima because of the non-convex nature of the optimization. To overcome local minima, a very small variation in the bandwidth parameter is needed to obtain a better performance of the low dimensional representation, but this procedure is computational demanding, therefore it is still required to derive a robust method to compute the bandwidth of the kernel avoiding as much as possible the local minima in the optimization process.

For the visualizations experiments on the MNIST databases, the 3-dimensional representation obtained by means of PCA shows that all the classes are overlapped (Figs. 9(a) and 10(a)), which can be easily explained because PCA does not consider class labels. The embeddings computed using MVU (Figs. 9(b) and 10(b)) shows a similar behavior to PCA embedding, because MVU is also unsupervised. ISOMAP represents most of the point cloud from a class on the same point in low dimensions (Figs. 9(c) and 10(c)). In these cases, the local geometry of the high dimensional data is not preserved. The conventional LLE algorithm tries to preserve the local structure of the data, but because of this technique is not supervised, then the class overlapping can not be avoided (Figs. 9(d) and 10(d)). The Correntropy LLE method improves the conventional LLE representation but Correntropy LLE can not separates the classes because it neither takes into account the label information (Figs. 9(e) and 10(e)). Although, the LLE+CLI technique captures the structure of the data and separates the most of the classes, few points remain mixed, and others are placed far away from the class centroid (Figs. 9(f) and 10(f)), because the noisy pictures can be consider as outliers. Finally, Correntropy LLE+CLI clearly separates the objects while maintaining the structure of the high dimensional data even when the database contains 10 classes (Figs. 9(g) and 10(g)).

In regard to classification, Correntropy LLE+CLI presents the best classification accuracy for the analyzed PCG database (Table 2). On the contrary, ISOMAP shows the poorest performance. PCA and Correntropy LLE reach low accuracy rates. The major issue with PCA and Correntropy LLE is the lack of class label information. The methods: LLE, δ-LLE, and LLE+CLI, show high scores. Nevertheless, they can not obtain maximum performance, because, unlike the Correntropy LLE+CLI they are not robust against outliers. Regarding the classification performance for the

MNIST 3 classes data set, the proposed method achieves a high accuracy while the δ-LLE technique ensures an appropriate separability. The ISOMAP performance is the worst. PCA, LLE, Correntropy LLE, and LLE+CLI show poor classification accuracy. Note, however that, when the number of classes are increased, testing the MNIST 10 classes database, the results achieved by δ-LLE got worsen. But the Correntropy LLE+CLI remains reaching a high accuracy rate. The LLE+CLI technique exhibits a similar performance than Correntropy LLE+CLI. The other unsupervised methods (PCA, LLE, δ-LLE, and Correntropy LLE) shows low scores. ISOMAP also reports in this case the worst performance.

For the MNIST 3 classes database, the classification accuracy is high for a wide range of σ values (Fig. 11(a)), because the kernel bandwidth, which removes the impulsive noise from the images accomplish its mission for a large range of σ values. Nevertheless, if σ is too small the similarity measure is very strict, and then, the density of the neighborhoods is low, which produces misleading low dimensional representations. The same behavior is observed for the MNIST 10 classes database. On the other hand, Fig. 11(b) shows a narrower range for the σ parameter to obtain a suitable classification accuracy, because if the kernel window is too large most observations looks similar while if the kernel window is too short the neighborhoods will be disconnected deforming the embedded representation. In particular, it is easier to remove the impulsive noise from the MNIST database than the noise from the PCG signals, because the latter is scattered along a wider frequency range.

7. Conclusion

In this paper, a new technique for NLDR called Correntropy LLE was proposed. This new approach is designed to work with noisy high dimensional data unlike most approaches for analyzing data on nonlinear manifolds. When the Correntropy similarity measure is employed instead of Euclidean distance in the LLE algorithm, the unfolded structure describes more clearly the underlying process behind the input data. Besides, the low dimensional representations are smooth, without distortions, and generally, the neighborhoods are conformed by very similar objects. The algorithm takes advantage of the metric induced by the Correntropy similarity measure, which reduces the effects of artifacts, distortions and non-Gaussian noise present in the objects.

Another contribution of this work is an extension of the Correntropy LLE algorithm for dealing with databases of multiclass data. When the classes are known, this approach preserves the local geometry of the high dimensional data and provides a discriminative strategy during the embedding procedure. Specifically, the algorithm searches relevant underlying variables behind the observed high dimensional data, which must be related to another reference signal, that is, the Class Label Information (CLI). The underlying variables span a low dimensional space, where the objects can be reconstructed as combinations of its neighbors. The low dimensional data preserves the local geometry relations, the global structure of the manifolds, and it describes the behavior defined by the class label information. In this sense, the proposed Correntropy LLE+CLI outperforms other NLDR methods, in classification and visualization tasks, because it is robust against outliers, distortions, and non-Gaussian noise. Moreover, the knowledge about the class is useful to avoid overlapping and over-fitting or excessive clustering. Our approach exhibits high classification accuracy compared to LLE+CLI and δ-LLE.

In summary, this family of algorithms based on Correntropy shows the ability of suitable representing several manifolds at the same time in a low dimensional space, taking into account the similarity among objects and the class labels. As a future work, we plan to discuss and analyze alternative metrics in order to find the optimal reconstruction weights, which allow a better visualization/classification performance. Equally, find the links between our proposed method and other non-spectral NLDR methods such as SNE or t-SNE is in our interest.

Acknowledgments

This research is carried out under grant "*Centro de Investigación e Innovación de Excelencia - ARTICA*", funded by COLCIENCIAS. The author JCP was partially supported by NSF ECCS 0856441.

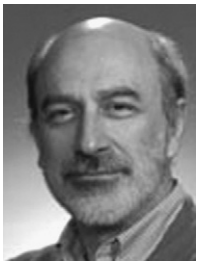
References

[1] L.K. Saul, S.T. Roweis, Think globally, fit locally: unsupervised learning of low dimensional manifolds, Mach. Learn. Res. 4 (2003) 119–155.
[2] O. Kouropteva, O. Okun, M. Pietikäinen, Supervised locally linear embedding algorithm for pattern recognition, in: F. Perales, A. Campilho, N. de la Blanca, A. Sandfeliu (Eds.), Pattern Recognition and Image Analysis, Lecture Notes in Computer Science, vol. 2652, Springer, 2003, pp. 386–394.
[3] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323–2326.
[4] D. Merkl, Text classification with self-organizing maps: some lessons learned, Neurocomputing 21 (1-3) (1998) 157–163.
[5] K.Q. Weinberger, L.K. Saul, An introduction to nonlinear dimensionality reduction by maximum variance unfolding, in: AAAI'06: Proceedings of the 21st National Conference on Artificial Intelligence, AAAI Press, 2006, pp. 1683–1686.
[6] J. Tenenbaum, Mapping a manifold of perceptual observations, in: NIPS '97: Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems, vol. 10, MIT Press, 1998, pp. 682–688.
[7] G. Hinton, S. Roweis, Stochastic neighbor embedding, in: NIPS '02: Proceedings of the 2002 Conference on Advances in Neural Information Processing Systems, vol. 15, MIT Press, 2003, pp. 833–840.
[8] M. Polito, P. Perona, Grouping and dimensionality reduction by locally linear embedding, in: NIPS '01: Proceedings of the 2001 Conference on Advances in Neural Information Processing Systems, vol. 14, MIT Press, 2002.
[9] J.C. Principe, D. Xu, J.W.F. III, Information theoretic learning, in: S. Haykin (Ed.), Unsupervised Adaptive Filtering, John Wiley & Sons, New York, 2000 (Chapter 7).
[10] I. Santamaría, P.P. Pokharel, J.C. Principe, Generalized correlation function: definition, properties, and applications to blind equalization, IEEE Trans. Signal Process. 54 (6) (2006) 2187–2197.
[11] W. Liu, P.P. Pokharel, J.C. Principe, Correntropy: properties and applications in non-gaussian signal processing, IEEE Trans. Signal Process. 55 (11) (2007) 5286–5298.
[12] V.N. Vapnik, The Nature of Statistical Learning Theory, Statistics for Engineering and Information Science, 2nd ed., Springer, New York, NY, 2000.
[13] P.J. Huber, Robust Statistics, Wiley Series in Probability and Statistics, John Wiley & Sons, Hoboken, NJ, 1981.
[14] A. Jog, A. Joshi, S. Chandran, A. Madabhushi, Classifying ayurvedic pulse signals via consensus locally linear embedding, in: International Conference on Bio-inspired Systems and Signal Processing (Biosignals), Springer Verlag, 2009, pp. 388–395.
[15] G. Daza-Santacoloma, G. Castellanos-Domínguez, J. Principe, Functional data representation using correntropy locally linear embedding, in: IEEE International Workshop on Machine Learning for Signal Processing, Kittilä, Finland, 2010, pp. 7–12.
[16] K. Madsen, H.B. Nielsen, O. Tingleff, Optimization with Constraints, 2nd ed., Informatics and Mathematical Modelling – Technical University of Denmark, 2004.
[17] W. Liu, P.P. Pokharel, J. Principe, Error entropy, correntropy and m-estimation, in: Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing, 2006, IEEE, 2006, pp. 179–184.
[18] D. de Ridder, R.P.W. Duin, Locally linear embedding for classification, Technical Report, Pattern Recognition Group, Delft University of Technology, Delft, The Netherlands, 2002.
[19] M. Pillati, C. Viroli, Supervised locally linear embedding for classification: an application to gene expression data analysis, in: S. Zani, A. Cerioli (Eds.), Book of Short Papers, CLADAG 2005, Parma, Italy, 2005, pp. 147–150.
[20] M. Loog, D. de Ridder, Local discriminant analysis. in: The 18th International Conference on Pattern Recognition (ICPR), vol. 3, IEEE Computer Society, 2006, pp. 328–331.
[21] X. Geng, D.-C. Zhan, Z.-H. Zhou, Supervised nonlinear dimensionality reduction for visualization and classification, IEEE Trans. Syst. Man Cybern. Part B Cybern. 35 (2005) 1098–1107.
[22] P.C. Hansen, Analysis of discrete ill-posed problems by means of the l-curve, SIAM Rev. 34 (4) (1992) 561–580.
[23] P.C. Hansen, Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion, Monographs on Mathematical Modeling and Computation, SIAM, 2000.

[24] S.A. Nene, S.K. Nayar, H. Murase, Columbia object image library: Coil-100, Technical Report, Columbia University, New York, 1996.
[25] A. Álvarez-Meza, J. Valencia-Aguirre, G. Daza-Santacoloma, G. Castellanos-Domínguez, Global and local choice of the number of nearest neighbors in locally linear embedding, Pattern Recognition Lett. 32 (2011) 2171–2177.
[26] J. Tenenbaum, V. de Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000) 2319–2323.
[27] A.R. Webb, Statistical Pattern Recognition, 2nd ed., John Wiley & Sons, Ltd, Indianapolis, IN, USA, 2002.
[28] D.C. Montgomery, G.C. Runger, Applied Statistics and Probability for Engineers, 3rd ed., John Wiley & Sons, 2003.

Genaro Daza-Santacoloma received the B.S. degree in electronic engineering (2005), the M.Sc. degree in engineering-industrial automation with honors (2007), and the Ph.D. degree in engineering-automatics with honors (2010), from the Universidad Nacional de Colombia. Currently, he is Assistant Professor at Universidad Antonio Nariño, Bogotá where he is researching about human motion and subspace learning in collaboration with Signal Processing and Recognition Group from Universidad Nacional de Colombia. His research interests are feature extraction/selection for training pattern recognition systems, artificial vision, computer animation, and machine learning.

German Castellanos-Dominguez received his undergraduate degree in radiotechnical systems and his Ph.D. in processing devices and systems from the Moscow Technical University of Communications and Informatics, in 1985 and 1990, respectively. Currently, he is a professor in the Department of Electrical, Electronic and Computer Engineering at the Universidad Nacional de Colombia at Manizales. In addition, he is Chairman of the GCPDS at the same university. His teaching and research interests include information and signal theory, digital signal processing and bioengineering.



Jose C. Principe received the B.S. degree from the University of Porto, Portugal, in 1972 and the M.Sc. and Ph.D. degrees from the University of Florida in 1974 and 1979, respectively. He is a Distinguished Professor of Electrical and Biomedical Engineering with the University of Florida, Gainesville, where he teaches advanced signal processing and artificial neural networks (ANNs) modeling. He is a BellSouth Professor and Founder and Director of the University of Florida Computational NeuroEngineering Laboratory (CNEL). He is involved in biomedical signal processing, in particular, the electroencephalogram (EEG) and the modeling and applications of adaptive systems. He has more than 129 publications in refereed journals, 15 book chapters, and over 300 conference papers. He has directed more than 50 Ph.D. dissertations and 61 master's degree theses. Dr. Principe is Editor-in-Chief of the IEEE transactions on biomedical engineering, President of the International Neural Network Society, and formal Secretary of the Technical Committee on Neural Networks of the IEEE Signal Processing Society. He is an AIMBE Fellow and a recipient of the IEEE Engineering in Medicine and Biology Society Career Service Award. He is also a member of the Scientific Board of the Food and Drug Administration, and a member of the Advisory Board of the McKnight Brain Institute at the University of Florida.