Front. Electr. Electron. Eng. China 2011, 6(1): 72–85
DOI 10.1007/s11460-011-0131-5

RESEARCH ARTICLE

Hujun YIN

Advances in adaptive nonlinear manifolds and dimensionality reduction

Abstract Recent decades have witnessed a much increased demand for advanced, effective and efficient methods and tools for analyzing, understanding and dealing with data of increasingly complex, high dimensionality and large volume. Whether it is in biology, neuroscience, modern medicine and social sciences or in engineering and computer vision, data are being sampled, collected and cumulated in an unprecedented speed. It is no longer a trivial task to analyze huge amounts of high dimensional data. A systematic, automated way of interpreting data and representing them has become a great challenge facing almost all fields and research in this emerging area has flourished. Several lines of research have embarked on this timely challenge and tremendous progresses and advances have been made recently. Traditional and linear methods are being extended or enhanced in order to meet the new challenges. This paper elaborates on these recent advances and discusses various state-of-the-art algorithms proposed from statistics, geometry and adaptive neural networks. The developments mainly follow three lines: multidimensional scaling, eigen-decomposition as well as principal manifolds. Neural approaches and adaptive or incremental methods are also reviewed. In the first line, traditional multidimensional scaling (MDS) has been extended not only to be more adaptive such as neural scale, curvilinear component analysis (CCA) and visualization induced self-organizing map (ViSOM) for online learning, but also to be more local scaling such as Isomap for enhanced flexibility for nonlinear data sets. The second line extends linear principal component analysis (PCA) and has attracted a huge amount of interest and enjoyed flourishing advances with methods like kernel PCA (KPCA), locally linear embedding (LLE) and Laplacian eigenmap. The advantage is obvious: a nonlinear problem is transformed into a linear one and a unique solution can then be sought. The third line starts with the nonlinear principal curve and surface and links up with adaptive neural network approaches such as self-organizing map (SOM) and ViSOM. Many of these frameworks have been further improved and enhanced for incremental learning and mapping function generalization. This paper discusses these recent advances and their connections. Their application issues and implementation matters will also be briefly enlightened and commented on.

Keywords dimensionality reduction, multidimensional scaling, nonlinear principal component analysis (PCA), principal manifold, neural networks, self-organizing maps (SOM), biologically inspired models, data projection, embedding and visualisation

1 Introduction

In an increasingly digitized and highly automated world, data collection has become a routine in many fields. Many modern scientific experiments and business operations generate huge amounts of data. With the data volume, complexity and dimensionality growing rapidly, analysis of data sets is becoming more and more cumbersome. The curse of dimensionality in particular has prompted a great deal of effort in finding effective ways of reducing data dimensionality and extracting data manifolds. Such a procedure has become an essential step of information processing in many fields. It has recently attracted a great deal of attention and a number of advanced techniques for extracting nonlinear manifolds and reducing data dimensions have been proposed from statistics, geometry and adaptive neural networks. A number of early reviews on some dimensionality reduction techniques have been conducted [1–4] and a recent review on adaptive nonlinear manifolds and their applications in face recognition is given in Ref. [5]. Al-

Received August 26, 2010; accepted December 16, 2010

Hujun YIN (✉)
The University of Manchester, Manchester, UK
E-mail: h.yin@manchester.ac.uk

though there are some applications in supervised or semi-supervised cases, fundamental dimensionality reduction and manifold learning are unsupervised for data embedding and data structure discovery. This paper provides a systematic review on various methods proposed and further comments on their recent advances and discusses the connections among them.

Traditional multidimensional scaling (MDS) is a popular methodology especially in social sciences for extracting data manifold and visualizing high dimensional data sets. It seeks a low (often two) dimensional representation of high dimensional data points that preserves as much as possible the inter-point distances or pair-wise dissimilarities [6]. Most (metric) MDS methods minimize a defined stress function in order to solve for the low dimensional coordinates of the data points. The mapping is generally nonlinear and can reveal the overall structure of the data on a manifold. In contrast to metric MDS, nonmetric MDS finds a monotonic relationship (instead of metric ones) between the dissimilarities of the data points in the data space and those of their corresponding coordinates in the projected space. Isomap [7] applies scaling on geodesic instead of Euclidean distances. MDS methods are generally point-to-point mappings and do not provide a generalizing mapping function or a generative manifold. MDS implementation and generalization has been undertaken before by using neural networks, for example, the feed-forward neural network based mapping [8] and radial-basis-function based MDS [9].

Principal component analysis (PCA) projects a data set onto its principal directions represented by the orthogonal eigenvectors of the covariance matrix of the data. It has long been used to reduce the number of variables and visualize data in scatter plots or linear subspaces. Singular value decomposition is adopted to perform the task due to various advantages such as direct operation on data matrix, stable results even when the data matrix is ill-conditioned, and decomposition at both feature and data levels. The linearity of PCA, however, limits its power for increasingly complex data sets, because it cannot capture nonlinear relationships defined by beyond second order statistics. Extension to nonlinear projection can tackle practical problems better; yet a unique, universal solution is still to be defined, if not impossible [10]. Various nonlinear methods along this line have been proposed, such as the auto-associative networks [11], generalized PCA [12], kernel PCA (KPCA) [13], local linear embedding (LLE) [14] and incremental LLE (e.g., Ref. [15]), as well as Laplacian eigenmap [16] and spectral clustering [17]. The eigen-decomposition based approach has gained a lot of interest and become the main advanced technique to embed nonlinear manifolds to data sets.

Principal curve/surface [18] is another framework of nonlinear PCA. It formally defines a principal manifold, though mostly in 1 or 2 dimensional space. Available algorithms for extracting such manifolds include the Hastie and Stuetzle (HS) algorithms [18], its extended version [19], principal oriented points [20], an incremental segment algorithm [21], and probabilistic principal surface [22].

Adaptive neural networks present alternative approaches to nonlinear data projection and dimension reduction. They can provide (implicit) generalizing mapping functions [23]. Although neural networks in general are nonparametric, "black-box" approaches, they have profound roots in emulating how living organisms perceive and process vast amounts of complex information. In neural terms, dimensionality reduction has been associated with retinotopic mapping for understanding cortical maps. Multisensory information is processed, fused and mapped onto an essentially 2-D cortex in an information preserving manner. Data processing and projection techniques inspired by this biologic mechanism are playing an increasingly important role in pattern recognition, computational intelligence, data mining, information retrieval and image recognition. Early examples include Refs. [8,9,11]. Self-organization is a fundamental pattern recognition process, in which intrinsic inter- and intra-pattern relationships within the sensory data are learnt. Kohonen's self-organizing map (SOM) [24,25] is a simplified, abstracted version of Willshaw and von der Malsburg's retinotopic mapping model [26]. Modeling and analyzing such mappings are important to understanding how the brain perceives, encodes, recognizes, and processes the patterns it receives and thus are beneficial to machine-based pattern recognition. SOM's topology-preserving property is utilized to extract and visualize relative mutual relationships among the data. Many variants and extensions have since been proposed, in particular the author's visualization induced SOM (ViSOM) [27], which preserves (local) distances on the map. SOM and ViSOM have been linked with principal curve and surface (e.g., Refs. [28,29]). It has also been widely observed that SOMs produce a similar effect to MDS. Further analysis has shown the exact connections [30].

The remaining of the paper is organized as follows. Section 2 describes MDS and related recent approaches in extracting nonlinear manifolds. Section 3 provides a review on the existing approaches to nonlinearizing PCA. Section 4 focuses on principal manifolds and Sect. 5 on neural based methods, in particular, SOM and ViSOM in learning nonlinear, generative manifolds of data sets, as well as their relationship with the previous approaches. Section 6 describes typical applications of these methods and further discusses various computational applications and potentials of these methods, followed by conclusions.

2 Multidimensional scaling based methods

MDS is a traditional subject related to dimension reduction and data visualization. MDS aims to project or embed high dimensional data points onto a low dimensional (often 2- or 3-D) plane by preserving as closely as possible inter-point metrics [6]. The projection, which is usually calculated via an optimization process of a stress function, is generally nonlinear and can reveal the overall structure of the data. Traditional MDS includes classical, metric and nonmetric MDS. Recent advances and extensions see the use of neighborhood to confine the stress locally and the use of eigen-decomposition method or adaptive learning algorithms instead of an optimization method for solving the objective functions.

2.1 Classical, metric and nonmetric MDS

Let δ_{ij} denote the dissimilarity between data points, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. δ_{ij} is often calculated (but not necessarily) by the Euclidean distance of data vectors $\delta_{ij} = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$. Let $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ be the mapped points or coordinates of points i and j in the visual space, then the distance between the mapped points is $d_{ij} = \|\boldsymbol{y}_i - \boldsymbol{y}_j\|$.

Classical MDS seeks a configuration so that the distances between projected points match the original dissimilarities, i.e., $d_{ij} = \delta_{ij},\ \forall i, j$. *Metric* MDS ensures that dissimilarities are proportional to distances of the projected points, $d_{ij} = f(\delta_{ij}),\ \forall i, j$; where f is a continuous, monotonic function, or a metric transformation function, that transforms dissimilarities to distance metrics. In practice, exact dissimilarity-distance matches may not be possible due to data noise and imprecision, the equality is replaced by approximation, "$\approx$", meaning "as equal as possible" [31]. It is worth noting that PCA is a special case of classical MDS.

A typical MDS configuration is sought by minimizing the following raw *stress* function,

$$S = \sum\nolimits_{i,j} [f(\delta_{ij}) - d_{ij}]^2, \tag{1}$$

where f is a metric transformation function. In some cases, the above raw stress is normalized by $\sum_{i,j} d_{ij}^2$ or $\sum_{i,j} \delta_{ij}^2$ to give a relative reading of the overall stress. Other normalization schemes are also possible. For example, in Sammon mapping [32] an intermediate normalization (pair-wise distance of original space) was used to preserve good local distributions and at the same time maintain a global structure. The Newton optimization method was used to recursively solve for the optimal configuration. More general weighting schemes have been proposed recently, together with a partial embedding distance, and the resulting MDS is called generalized MDS [33], which is shown capable of deformable surface matching.

For general metric MDS, especially when original dissimilarities need to be transformed to a distance like form, f is a monotonic transformation function, e.g., a linear function. For classical MDS and many cases of metric MDS, f is simply the identify function, the stress becomes,

$$S = \sum_{i,j} (\delta_{ij} - d_{ij})^2. \tag{2}$$

That is, metric MDS configuration tries to preserve, as well as possible, the pair-wise distances of original data points on the projected space. When the distances are transformed to dot products, the embedding coordinates can be solved via an eigen decomposition of the transformed distance matrix [6].

Nonmetric MDS deals with rank order of the dissimilarities and seeks a configuration such that distances between pairs of mapped points match order-wise "as well as possible" the original dissimilarities. That is, a nonmetric MDS looks for a projection function that is monotonic and satisfies,

$$\text{if } \delta_{ij} \leqslant \delta_{kl},\ \text{then } d_{ij} \leqslant d_{kl},\ \forall i, j, k, l. \tag{3}$$

Nonmetric MDS produces an ordinal scaling rather than a metric one.

MDS usually relies on an optimization algorithm to search for a configuration that gives as low stress as possible. Inevitably, various computational problems such as local minima and divergence may occur to the process. The methods are often computationally intensive, especially with large data sets. The final solution depends on the starting configuration and parameters used in the algorithm. Converting to an eigen problem can greatly facilitate the solution solving [6].

Another major drawback of MDS is the lack of an explicit projection function, as MDS is usually a point-to-point mapping and cannot *naturally* accommodate new data points. Thus for any new input data, the mapping has to be recalculated based on all available data. Although some methods have been proposed to accommodate the new arrivals using triangulation [34,35], the methods are generally not adaptive.

2.2 Adaptive MDS

In Ref. [7], Isomap was proposed to use geodesic (curvature) distance for better scaling of nonlinear manifolds. Instead of the direct (global) Euclidean distance, the geodesic distance along the manifold is calculated (or cumulated) via neighborhood graphs or neighboring points. That is, δ_{ij} is computed by the shortest path on the manifold via neighborhood graphs. All such pairwise distances form a matrix $\boldsymbol{D}_G$. The scaling is thus re-

stricted along the manifold and this helps extract highly nonlinear manifolds. The cost function of Isomap is

$$E = \sum_{i,j} \left[\tau\left(\delta_{ij}\right) - \tau\left(d_{ij}\right)\right]^2, \tag{4}$$

where operator τ converts squared distance to centered dot-product. The minimization is achieved by setting the embedded coordinates to the m largest eigenvectors of the matrix $\tau\left(\boldsymbol{D}_G\right)$.

The method usually requires a large number of data points. Otherwise, the calculation of the geodesic distances can lead to large errors and the extracted manifold can become distorted. In addition, selecting a suitable neighborhood size can be a difficult task, which is highly dependent on the (unknown) data structure; and often requires a cross-validation procedure. Isomap has been reported to be unstable [36]. Two variants of Isomap have been proposed [37]: one is to further confine the geodesic distances to local by using a weighting scheme to make the large distance less important; the other is to use a subset of original data as landmark points to reduce the computational costs when the data set is large. Another similar method is the stochastic proximity embedding (SPE) [38], which restricts the scaling within a neighborhood and uses a stochastic learning algorithm to adapt the embedded coordinates.

Curvilinear component analysis (CCA) [39] is another method for nonlinear scaling. It detects the intrinsic geometric properties of data by preserving local distance relationships via minimizing an error function defined as

$$E = \frac{1}{2}\sum_i \sum_{j\neq i} \left(\delta_{ij} - d_{ij}\right)^2 \varphi\left(d_{ij}, \theta_y\right), \tag{5}$$

where $\varphi\left(d_{ij}, \theta_y\right)$ is a monotonically decreasing neighborhood function with regard to the distance in the projected space and is used for preserving local topology and maintaining shorter distances than longer ones.

These MDS extensions, though highly adaptive for nonlinear manifolds, do not generally accommodate new data points like traditional MDS. In Ref. [40], an incremental algorithm for Isomap was proposed and it efficiently updates the geodesic distances and re-estimates the eigenvectors using the previous computation results. The drawbacks of traditional MDS can also be naturally overcome by implementing or parameterizing MDS using neural networks. In Ref. [8], a feed-forward network was used to parameterize the Sammon mapping and an unsupervised training method has been derived to train the network. The derivation is similar to the back-propagation algorithm, by minimizing the Sammon stress instead of the total errors between desired and actual output. The network takes a pair of input points at each time in training. An evaluation has to be carried out, using all the data points, after a fixed number of iterations. In Ref. [9], the neural scale method uses a radial basis function (RBF) network to minimize the simple stress function, Eq. (2), to perform MDS. SOM and ViSOM can be regarded as MDS methods as to be explained in Sect. 5.

It is worth noting that while MDS generally minimizes the difference or squared difference between the dissimilarities in the original and mapped spaces, the c measure proposed as a unified objective for topographic mapping [41] maximizes the correlation between the two. One can argue that when the dissimilarities are normalized, the two are equivalent.

3 Eigen-decomposition based methods

3.1 PCA

PCA is conducted by solving an eigenvalue problem on, i.e., by eigen-decomposition of, the covariance matrix $\boldsymbol{C}$ of a data set, $\lambda\boldsymbol{U} = \boldsymbol{C}\boldsymbol{U}$, where columns of $\boldsymbol{U}$ are the eigenvectors and λ is a eigenvalue. By discarding the minor components, PCA can effectively reduce the number of variables and display the dominant ones in a linear, lower dimensional subspace. It is the optimal linear projection in the sense of the mean-square errors between original and projected points, i.e.,

$$\min \sum_{\boldsymbol{x}} \left[\boldsymbol{x} - \sum_{j=1}^{m} \left(\boldsymbol{u}_j^{\mathrm{T}}\boldsymbol{x}\right)\boldsymbol{u}_j\right]^2, \tag{6}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the n-dimensional data and $\{\boldsymbol{u}_j,\ j = 1, 2, \ldots, m,\ m \leqslant n\}$ are the orthogonal eigenvectors and represent principal directions. They are the first m principal eigenvectors of the covariance matrix. The second term in the bracket is the reconstruction or projection of $\boldsymbol{x}$ on these eigenvectors. The term $\boldsymbol{u}_j^{\mathrm{T}}\boldsymbol{x}$ represents the projection of $\boldsymbol{x}$ onto the jth principal dimension. Efficient and robust statistical methods such as singular value decomposition (SVD) exist for solving eigenvector problems, especially when data covariance matrix is ill-conditioned (e.g., when the dimensionality n is much greater than the sample size N). In this case, data matrix $\boldsymbol{X}$ of $n \times N$ is decomposed as $\boldsymbol{X} = \tilde{\boldsymbol{U}}\boldsymbol{\Gamma}\boldsymbol{V}^{\mathrm{T}}$, where $\tilde{\boldsymbol{U}}$ contains the first N eigenvectors of $\boldsymbol{C}$.

Several adaptive learning algorithms have also been proposed for performing PCA such as the subspace network [42] and the generalized Hebbian algorithm [43]. The limitation of linear PCA is obvious, as it cannot capture nonlinear relationships defined by higher than the second order statistics.

3.2 KPCA

Nonlinear extension of PCA can be intuitively approached using combined or piecewise local PCA models. That is, the entire input space is segmented, for example using a clustering algorithm, into non-overlapping regions and a local PCA can be performed in each region. However, the extension to nonlinear PCA is not unique due to the lack of a unified mathematical structure and an efficient and reliable algorithm, and in some cases due to excessive freedom in selection of representative basis functions [10,12]. Existing methods include the five-layer feedforward associative network (FFAN) [11] and KPCA [13]. The first three layers of the FFAN project the original data on to a curve or surface, providing an activation value for the bottleneck node. The last three layers define the curve and surface. The weights of the associative network are determined by minimizing the following objective function,

$$\min \sum_{\boldsymbol{x}} \left\| \boldsymbol{x} - \boldsymbol{f}\left\{ s_f(\boldsymbol{x}) \right\} \right\|^2, \tag{7}$$

where f: $\mathbb{R}^1 \to \mathbb{R}^n$ (or $\mathbb{R}^2 \to \mathbb{R}^n$), the function modeled by the last three layers, defines a curve (or a surface), $s_f : \mathbb{R}^n \to \mathbb{R}^1$ (or $\mathbb{R}^n \to \mathbb{R}^2$), the function modeled by the first three layers, defines the projection index.

KPCA uses nonlinear mapping and kernel functions to generalize PCA to nonlinear. The nonlinear function $\Phi(\boldsymbol{x})$ maps data onto high-dimensional feature space, where the standard linear PCA can be performed via kernel functions $k(\boldsymbol{x}, \boldsymbol{y}) = (\Phi(\boldsymbol{x}) \cdot \Phi(\boldsymbol{y}))$. The projected covariance matrix is then

$$\mathrm{Cov} = \frac{1}{N} \sum_{i=1}^{N} \Phi(\boldsymbol{x}_i) \Phi(\boldsymbol{x}_i)^{\mathrm{T}}. \tag{8}$$

The standard linear eigenvalue problem can now be written as $\lambda \boldsymbol{U} = \boldsymbol{K}\boldsymbol{U}$, where the columns of $\boldsymbol{U}$ are the eigenvectors and $\boldsymbol{K}$ is a $N \times N$ matrix with elements as kernels $K_{ij} := k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j))$.

Kernel techniques turn a nonlinear problem into a linear one and several other nonlinear dimensionality reduction methods can be regarded under the "kernel" framework [44]. Establishing such links may bring more insight to the method under study as many properties of the kernel methods can be applied to the method.

3.3 LLE

LLE [14] is another eigen-decomposition method that forms nonlinear principal subspace by piecewise linear approach. The local linearity is defined on a local neighborhood, via a ε ball or the κ nearest neighbors. Then the linear contributions or weightings, W_{ij}, of these neighboring points are calculated through minimizing the following cost function,

$$e(\boldsymbol{W}) = \sum_i \left\| \boldsymbol{x}_i - \sum_{j=1} W_{ij} \boldsymbol{x}_j \right\|^2. \tag{9}$$

This step seeks to reconstruct each data point $\boldsymbol{x}_i$ linearly from its κ nearest neighbors. The optimal weights, W_{ij}, satisfying $\sum_j W_{ij} = 1$, can be found by solving a constraint least-squares problem as shown in Refs. [45,46],

$$W_{ij} = \frac{\sum_{p=1}^{\kappa} R_{i,jp}}{\sum_{q=1}^{\kappa} \sum_{p=1}^{\kappa} R_{i,jp}},$$

where $\boldsymbol{R}_i = \boldsymbol{G}_i^{-1}$ and $\boldsymbol{G}_i$ is a $\kappa \times \kappa$ local Gram matrix with $G_{i,jk} = \langle (\boldsymbol{x}_i - \boldsymbol{x}_j), (\boldsymbol{x}_i - \boldsymbol{x}_k) \rangle$.

The embedding is computed via,

$$\min \sum_i \left\| Y_i - \sum_{j=1} W_{ij} Y_j \right\|^2, \tag{10}$$

where $\{Y\}$ are the embedding coordinates. With the optimal weightings solved, a matrix is formed as $\boldsymbol{M} = (\boldsymbol{I} - \boldsymbol{W})^{\mathrm{T}}(\boldsymbol{I} - \boldsymbol{W})$. Then the embedding on the m dimensional space is obtained from the $m+1$ smallest eigenvectors of $\boldsymbol{M}$, with the smallest discarded. That is, the remaining m eigenvectors define the embedding coordinates. Working with the smallest eigenvectors, however, means the result or projection can be sensitive to noise and outliers.

LLE is a batch operation and requires the entire data set of all available data points. Its inability to adapt to new data has attracted some criticism. Extending LLE to accommodate new data points on an already learned embedding has thus become a topic, esp. for online learning. In Ref. [45], an intuitive way of doing it is suggested. For a new data point, find its κ nearest neighbors and calculate its weightings to the neighbors as in the LLE. Then its embedding is simply the weighted average (by the calculated weightings) of its neighbors' corresponding embeddings. This method implies that the previous data set is sufficient or well-sampled from the manifold, and the new data points are well distributed on the manifold. In Ref. [15], an incremental LLE was proposed, which involves recalculation of matrix $\boldsymbol{M}$ when a new data point arrives, with certain simplification and assumptions. The method also requires that the manifold is already well-represented by the original data set. A similar method has also been proposed in Ref. [47].

3.4 Hessian LLE (HLLE)

Similar to LLE and Isomap, HLLE [48] aims to recover the embedding of a data set on a manifold that is locally isometric. It overcomes the convexity and global

isometry requirements of the Isomap and uses a similar procedure as to the LLE to establish the embedding coordinates. It also reduces the computational cost of Isomap from full matrix of graph distances to sparse (local) distance matrix. Its drawback is the requirement of estimation of second order derivatives, which can be noisy or difficult in high-dimensional data samples.

Similar to LLE, HLLE algorithm first identifies the nearest neighbors for each data point. Local tangent coordinates are calculated from these local neighborhood matrixes via SVD. Then the Hessian estimators are constructed by the least-squares estimation on the neighborhoods. A quadratic matrix, similar to $\boldsymbol{M}$, is built from the Hessian estimators and embedding coordinates obtained from the eigen analysis on the matrix.

3.5 Laplacian eigenmap and spectral clustering

Laplacian eigenmap [16] forms a local linear mapping by converting the problem to a generalized eigen problem and the solution becomes easily traceable. First, the weightings (of local neighboring points), also sometimes called heat kernels, are constructed,

$$W_{ij} = \exp\left(-\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{t}\right). \tag{11}$$

Then the embedding is computed via the generalized eigenvalue problem,

$$\boldsymbol{L}\boldsymbol{f} = \lambda \boldsymbol{D}\boldsymbol{f}, \tag{12}$$

where $\boldsymbol{D}$ is diagonal matrix $D_{ii} = \sum_j W_{ji}$ and $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$.

The data are then projected onto the subspace spanned by the principal eigenvectors $\{\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_m\}$, the $m+1$ smallest engenvectors with the smallest excluded. This approach is also related to the locality preserving projection (LPP) [49] and the spectral clustering [17].

These nonlinear PCA methods, which are all based on eigen-decomposition though with different approaches, have many common characters. There are many similar such methods proposed recently under different terms. They are all defined on a local neighborhood so transforming global structure to local linear structures. That is, the manifold is constructed on local (linear) graphs. It has been shown that these methods are closely related [44,48], or they can be described under the regularization theory or kernel methods [44,50], a unified embedding framework [47,51] or the local tangent space model [52]. The difference mainly lies in defining the local reconstruction or graph, either linearly as in LLE and HLLE or by local distance graph as in Laplacian eigenmap. Then the embedding is to project the data onto the local tangent subspaces, sought by either Hessian or Laplacian. The key advantage of eigen-decomposition based methods is that once a neighborhood or "heat kernel" is defined, the solution is unique. In addition, although the embedding is given on often a manifold of a pre-specified dimensionality, m, the solution given by the eigen-decomposition actually contains the results of all possible dimensionality, useful for validating or estimating the true or intrinsic dimensionality, if it is not known. Another interesting advance is to treat the manifold extraction as maximum variance unfolding [53]. Instead of minimizing the reconstruction errors, it maximizes the pairwise distances in the mapped space. The problem can be framed to a Semidefinite programming problem.

4 Principal curve/surface based methods

4.1 Principal curve/surface

The principal curve and surface [18,54,55] are the principled nonlinear extension of PCA. The principal curve is defined as a smooth and self-consistency curve, which does not intersect itself, passing through the middle of the data. Denote $\boldsymbol{x}$ as a random vector in $\mathbb{R}^n$ with density p and finite second moment. Let $f(\cdot)$ be a smooth unit-speed curve in $\mathbb{R}^n$, parameterized by the arc length ρ (from one end of the curve) over $\Lambda \in \mathbb{R}$, a closed interval.

For a data point $\boldsymbol{x}$, its projection index on f is defined as

$$\rho_f(\boldsymbol{x}) = \sup_{\rho\in\Lambda}\left\{\rho : \|\boldsymbol{x} - f(\rho)\| = \inf_{\vartheta} \|\boldsymbol{x} - f(\vartheta)\|\right\}. \tag{13}$$

The curve is called self-consistent principal curve of ρ if

$$f(\rho) = E\left[\boldsymbol{X}|\rho_f(\boldsymbol{X}) = \rho\right]. \tag{14}$$

The principal component is a special case of the principal curves if the distribution is ellipsoidal. Although principal curves have been mainly studied, extension to higher dimension, e.g., principal surfaces or manifolds is feasible in principle. However, in practice, a good implementation of principal curves/surfaces relies on an effective and efficient algorithm. The principal curves/surfaces are more of a concept and practical algorithms are needed for implementation. Hastie and Stuetzle (HS) algorithm is a nonparametric method [18] that directly iterates the two steps of the above definition. It is similar to the standard vector quantization (VQ) algorithm combined with some smoothing techniques when only a finite data set is available:

1) *Initialization*: Choose the first linear principal component as the initial curve, $f^{(0)}(\boldsymbol{x})$.

2) *Projection*: Project the data points onto the current curve and calculate the projections index, i.e., $\rho^{(t)}(\boldsymbol{x}) = \rho_{f(t)}(\boldsymbol{x})$.

3) *Expectation*: For each index, take the mean of data points projected onto it as the new curve point, i.e., $f^{(t+1)}(\rho) = E\left[\boldsymbol{X}\middle|\rho_{f(t)}(\boldsymbol{X}) = \rho\right]$.

The projection and expectation steps are repeated until a convergence criterion is met, for instance, when the change of the curve between iterations is below a threshold.

For a finite data set, the density is often unknown; the above expectation is replaced by a smoothing method such as the locally weighted running-line smoother or smoothing splines. For kernel regression, the smoother is,

$$f(\rho) = \frac{\sum_{i=1}^{N} \boldsymbol{x}_i \kappa(\rho, \rho_i)}{\sum_{i=1}^{N} \kappa(\rho, \rho_i)}. \tag{15}$$

The arc length is simply computed from the line segments. There are no proofs of convergence of the algorithm, but no convergence problems have been reported, though the algorithm is biased in some cases [18]. In Ref. [19] a modified HS algorithm was proposed by taking the expectation of the residual of the projections in order to reduce the bias. An incremental principal curve was proposed in Ref. [21]. It is an incremental, or segment by segment, and arc length constrained method for practical construction of principal curves.

In Ref. [55] a semi-parametric model for the principal curve was introduced. A mixture model was used to estimate the noise along the curve; and the expectation and maximization (EM) method was employed to estimate the parameters. Other adaptive learning approaches include the probabilistic principal surfaces (PPS) [22], which uses a manifold oriented covariance noise model based on the generative topographical mapping (GTM) [56]. PPS has been shown to significantly outperform GTM in terms of reconstruction errors on several benchmark data sets [22]. A recent work on local tangent space alignment model [52] interestingly links the principal manifold with eigen-decomposition based nonlinear PCA methods.

5 Self-organizing map based methods

SOM is an abstract, simplified mathematical model of the mapping between nerve sensory and cerebral cortex [24,25]. Modeling and analyzing such mappings are important to understanding how the brain perceives, encodes, recognizes, and processes the patterns it receives and thus are beneficial to machine-based pattern recognition. External stimuli are received by various sensory or receptive fields (e.g., visual-, auditory-, motor-, or somato-sensory), coded, combined and abstracted by the living neural networks, propagated through axons, and projected onto the cerebral cortex, often to distinct parts of cortex. Different areas of the cortex (cortical maps) respond to different sensory inputs, though many functions and actions require collective responses from various areas. Topographically ordered mappings are widely observed in the cortex. The main structures (primary sensory areas) of the cortical maps may be established genetically in a predetermined manner [57]. More detailed areas (associative areas) between the primary sensory areas, however, are developed through self-organization gradually during life and in a topographically meaningful fashion. Therefore, studying such topographic projections, which had been ignored during the early period of neural network research, is undoubtedly fundamental to understanding dimension-reduction mapping for effective representation of sensory information and feature extraction.

5.1 SOM as nonmetric scaling manifold

Von der Malsburg and Willshaw first developed in a mathematical form the self-organizing topographic mapping, from two-dimensional presynaptic sheets to two-dimensional postsynaptic sheets, based on retinatopic mapping: the ordered projection of visual retina to visual cortex [26]. Kohonen abstracted this self-organizing learning model and proposed a much simplified mechanism [24]. This simplified model can emulate the self-organization effect. Although the SOM algorithm was more or less proposed in a heuristic manner, it is an abstract and generalized model of the self-organization learning process.

In the SOM, a set of neurons, often arranged in a 2-D rectangular or hexagonal grid or map, is used to form a discrete, topological mapping of an input space, $\boldsymbol{X} \in \mathbb{R}^n$. At the start of the learning, all the weights $\{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_M\}$ are initialized to either small random numbers or some specific values (e.g., principal subspace), where $\boldsymbol{w}_i$ is the weight associated to neuron i and is a vector of the same dimension, n, of the input, and M is the total number of neurons. Denote $\boldsymbol{r}_i$ the discrete vector defining the position (coordinates) of neuron i on the map grid. Then the algorithm iterates the following steps:

1) At each time t, present an input, $\boldsymbol{x}(t)$, select the winner,

$$v(t) = \arg\min_{k \in \Omega} \|\boldsymbol{x}(t) - \boldsymbol{w}_k(t)\|. \tag{16}$$

2) Update the weights of the winner and its neighbors,

$$\Delta \boldsymbol{w}_k(t) = \alpha(t)\eta(v, k, t)\left[\boldsymbol{x}(t) - \boldsymbol{w}_k(t)\right], \tag{17}$$

where $\alpha(t)$ is the learning rate and $\eta(v, k, t)$ is the neighborhood function and Ω is the set of neuron indexes. Although one can use a top-hat type of neighborhood

function, a Gaussian, $\eta(v,k,t) = \mathrm{e}^{-\frac{d_{vk}^2}{2\sigma(t)^2}}$, is often used in practice with $\eta(v,k,t)$ representing the effective range of the neighborhood and $d_{vk} = \|\boldsymbol{r}_v - \boldsymbol{r}_k\|$, the distance between neurons v and k on the map grid.

SOM was proposed to model the sensory-to-cortex mapping or an associative memory mechanism. Such a model is also related to VQ in coding terms. The SOM has been shown to be an asymptotically optimal VQ [58]. More importantly, with the neighborhood learning, the SOM is an error tolerant VQ and Bayesian VQ [59,60]. SOM has been linked with minimal wiring of cortex-like maps [61,62].

SOM has been primarily used for data visualization. However, the inter-neuron distances, when referred to the data space, have to be crudely or qualitatively marked by colors or gray levels on the trained map. The coordinates of the neurons (the resulting of scaling) are fixed on a lower dimensional (often 2-D) grid and do not resemble the distances (dissimilarities) in the data space. The similarities between SOMs and MDS in terms of topographic mapping, mostly the qualitative likeness of the mapping results, have been reported before [63]. However, limitations of using the SOM for MDS have also been noted [64] – the main one being that SOM does not preserve distance. Many applications combine the SOM and MDS for improved visualization of the SOM projection results. In fact, it is argued that the SOM is closer to MDS than to principal manifold [63]. In Ref. [30], SOM has been shown to be a nonmetric MDS.

5.2 ViSOM as metric scaling manifold

For metric scaling and data visualization, an isometric display of data structure and distribution is highly desirable. ViSOM extends the SOM for distance preservation on the map [27]. In order for the map to capture the data manifold structure directly, (local) distance quantities must be preserved on the map, along with the topology. The map can be seen as a smooth and graded mesh embedded into the data space, onto which the data points are mapped and the inter-point distances are approximately preserved.

To achieve that, the updating force, $[\boldsymbol{x}(t) - \boldsymbol{w}_k(t)]$, of the SOM algorithm is decomposed into two elements $[\boldsymbol{x}(t) - \boldsymbol{w}_v(t)] + [\boldsymbol{x}_v(t) - \boldsymbol{w}_k(t)]$. The first term, $[\boldsymbol{x}(t) - \boldsymbol{w}_v(t)]$, represents the updating force from the winner v to the input $\boldsymbol{x}(t)$, and is the same to the updating force used by the winner v. The second term, $[\boldsymbol{x}_v(t) - \boldsymbol{w}_k(t)]$, is a lateral contraction force bringing neighboring neuron k to the winner. In the ViSOM, this lateral contraction force is regulated in order to help maintain uniform inter-neuron distances locally on the map. The update rule is

$$\Delta \boldsymbol{w}_k(t) = \alpha(t)\eta(v,k,t)\Big([\boldsymbol{x}(t) - \boldsymbol{w}_v(t)] + \beta\,[\boldsymbol{x}_v(t) - \boldsymbol{w}_k(t)]\Big), \tag{18}$$

where β is a constraint coefficient–the simplest form being $\beta = \delta_{vk}/(d_{vk}\lambda) - 1$, δ_{vk} is the distance of neuron weights in the input space, d_{vk} is the distance of neuron indexes on the map, and λ is a resolution constant. A further refresh step (using neurons weights as the input) is added to SOM algorithm to ensure smooth expansion of the map in areas where the data are sparse or empty [29].

The ViSOM regularizes the inter-neuron contraction so that local distances between the nodes on the map are analogous to the distances of their weights in the data space. In addition to SOM's objective to minimize the quantization error, the aim is also to maintain constant inter-neuron distances locally. When the data points are eventually projected onto the trained map, the distance between data points i and j on the map is proportional to the distance of these two points in the data space, at least locally, subject to the quantization error (the distance between a data point and its neural representative). That is, $d_{ij} \propto \delta_{ij}$ or $\lambda d_{ij} \approx \delta_{ij}$. This makes data visualization more direct and quantitatively measurable. The resolution of the map can be enhanced by interpolating a trained (small) map or by incorporating the local linear projection (LLP) method [65]. Instead of projecting onto the winning node v (or $\boldsymbol{w}_v$), the data point $\boldsymbol{x}$ is projected to the sub plane spanned by two closest edges. Projected point is therefore,

$$\boldsymbol{x}' = \boldsymbol{w}_v + \max_{v'=v\pm1}\left\{ \frac{(\boldsymbol{x} - \boldsymbol{w}_v)\cdot(\boldsymbol{w}_v - \boldsymbol{w}_{v'})}{\|\boldsymbol{w}_v - \boldsymbol{w}_{v'}\|^2}, 0 \right\}, \tag{19}$$

where '$\cdot$' denotes dot-product.

The size or covering range of the neighborhood function decreases from an initially large value to a final small one. The final neighborhood, however, should not shrink to contain only the winner, but can be made adaptive. The rigidity or curvature of the map is controlled by the size of the neighborhood. The larger this size the flatter the final map is in the data space. The computational costs of the ViSOM are higher than the SOM, due to the regularization and that often a larger map is used for fine resolutions.

Several improvements have since been made to the ViSOM for improved stability, flexibility and scalability [30,66,67]. In Ref. [29], it is shown that the distance preserving ViSOM approximates a discrete principal manifold. One advantage of the ViSOM is that its neighborhood is usually made adaptive according to data characteristics in various regions, unlike that in other nonlinear PCA or manifold methods, a preset neighborhood size has to be set empirically. ViSOM has also been shown

to produce a similar mapping result as to metric MDS. Such a connection has been proven recently [30]. ViSOM is a metric MDS, while order-preserving SOM is a kind of nonmetric MDS.

It has been noted that SOMs of pre-fixed size are difficult to converge to highly nonlinear manifolds. To improve the local distance-preserving capability of ViSOM, an incremental or growing ViSOM (gViSOM) has been proposed [30] for embedding and metric-scaling nonlinear manifolds. Such an incremental algorithm can reduce the computational costs of the ViSOM. Details of the gViSOM algorithm are as follows,

Step 1 Start with a small initial map (e.g., 5×5) in either rectangular or hexagonal shape. Place the initial map onto a linear subspace of either the entire or a local region of the data space. Set the desired resolution and the neighborhood size (i.e., locality).

Step 2 Randomly draw a data sample from the data space and find the winning neuron with the shortest Euclidean distance.

Step 3 If the sample falls within the neighborhood, update the weights of all the neurons within the neighborhood using the ViSOM algorithm; otherwise go back to Step 2.

Step 4 At regular iteration intervals (e.g., 1000 iterations), if the growing condition is met (that is, the data are underrepresented by the existing map), grow the map by adding a column or row to the side with the highest activities (measured by the winning frequencies or quantization errors). The added column or row is a linear extrapolation of the existing map. Other growing structures can be used, such as incrementing polygons instead of entire column or row for a free structure of the map and efficient use of neurons.

Step 5 As in the ViSOM, at regular intervals (every certain number of iterations), refresh the map (neurons) probabilistically.

Step 6 Check if the map has converged. If not go back to Step 2; if so go to the next step.

Step 7 Project the data samples onto the map, either to the neurons or by the LLP resolution enhancement.

Other SOM-based or motivated approaches for finding the nonlinear manifold include adaptive subspace SOM (ASSOM) [68], GTM [56], self-organizing mixture network (SOMN) [69] and Topological product of experts [70]. These methods model the data by means of a latent space. They belong to the semi-parameterized mixture model, although types and orientations of the local distributions vary from method to method. These approaches also resemble or can produce nonlinear PCA. SOM is used to quantize or segment the input space into local regions (Voronoi tessellation) and local probabilistic models formed in these regions can interpret PCA locally. Other similar neural approaches also include the early elastic net [61] and elastic maps [71].

6 Applications of dimensionality reduction and nonlinear manifolds

Dimension reduction and manifolds have found numerous applications in the literature and practice in various fields, such as biology, neuroscience, computer vision, knowledge management, finance, social networking and astronomy. In many pattern recognition tasks, manifold methods are used for dimensionality reduction and feature extraction. Along with clustering algorithms, dimension reduction can (optimally) reduce the quantity of the data and thus greatly facilitate the analysis and interpretation. Herein, two representative examples are given. Further discussions on applications can be found in Ref. [5]. Various new developments such as in multiple manifolds and temporal manifolds have not been covered here.

6.1 Data visualization and manifold extraction

The primary purpose of dimensionality reduction and multidimensional scaling is to visualize high dimensional data on a 2-D or 3-D plot to discover patterns and distributions of the data. The success of a proposed method is often tested on some benchmark data sets such as the S-curve/surface, Swissroll and the UCI data repository.

Here, we provide a set of typical experimental results on the Swissroll nonlinear manifold data set. Three basis methods are included: Isomap, LLE and ViSOM, as they represent the essence of the various methods described in Sect. 2–5. In total 2000 3-D data points, shown in Fig. 1 as colored crosses, were generated according to [14]. Typical results of Isomap and LLE, obtained using the code provided by their authors, are shown in Figs. 2(a) and (b), respectively. The neighborhood size was set to 12 in LLE. Other values do not seem to change the result much. Large neighborhoods incur higher computational costs. The incremental version of ViSOM, gViSOM [30], is used. It started with a 5×5 grid and finally settled to 18×70. The resolution was set to 1.5. ViSOM result is shown in Fig. 2 (bottom) and its embedding in the data space with its final flattened manifold revealed in Fig. 1 as the blue grid.

The advantages of the gViSOM are evident as it produces much more faithful metric scaling and is able to extract highly nonlinear manifold function [30], while Isomap has some distortions on the projected space and LLE result appears sensitive as commented before. SOM-based methods have better abilities handling noise and outliers. In addition, ViSOM (not SOM however) can cope well with discontinuities of the manifold (for ex-

ample, holes in the manifold or disjoint manifolds), and can adapt to the dynamics (slow changes) of the manifold, for which both Isomap and LLE (and other scaling methods) would have to re-capture the entire data set once any part or whole is updated. Although the gViSOM is much more computationally efficient than the ViSOM, it is still more demanding than Isomap and LLE, as the latter are framed to an eigen problem and solved by efficient algorithms.

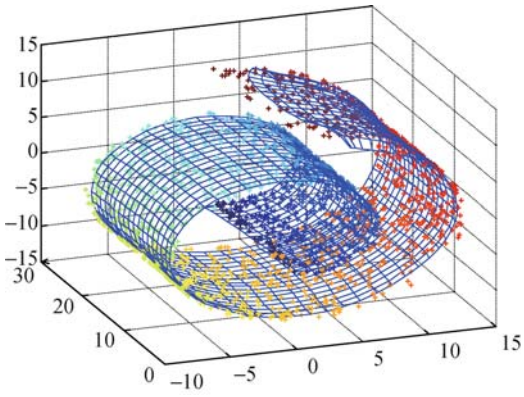

Fig. 1 Swissroll data of 2000 points (colored crosses) and embedded ViSOM (blue grid)

6.2 Dimension reduction for pattern recognition

High dimensionality of data poses many difficulties to pattern recognition and classification. Dimensionality reduction is often applied to the data in order to reduce the number of features or to make the classification problem well-defined. Here a face recognition example is used to illustrate the usefulness of the dimensionality reduction and the advantages of nonlinear approaches.

In the experiment, various manifold methods were used for dimension reduction in the preprocessing of raw face images, and then one of the classifiers, the nearest neighbor (NN), soft k-NN [72], linear discriminant analysis (LDA) [73] and support vector machine (SVM), were used for classification. The performances of the dimension reduction methods were evaluated and compared based on the same classifier. The experiment was conducted on a publicly available database, the Olivetti Research Laboratory (ORL) database, which consists of 40 subjects with 10 different face images for each subject. All images in the database were taken against a dark homogeneous background with an up-right, frontal position and have the same size of 92×112. Face images vary in terms of lighting conditions, facial expressions or facial details.

In PCA, nonlinear PCA and MDS-based methods, where the images are treated as vectors, the number of dimensions ($92 \times 112 = 10304$) of the face image was reduced to 60. Two types of kernel-PCA, *polynomial* (KPCA1) and *Gaussian radial basis* (KPCA2), were

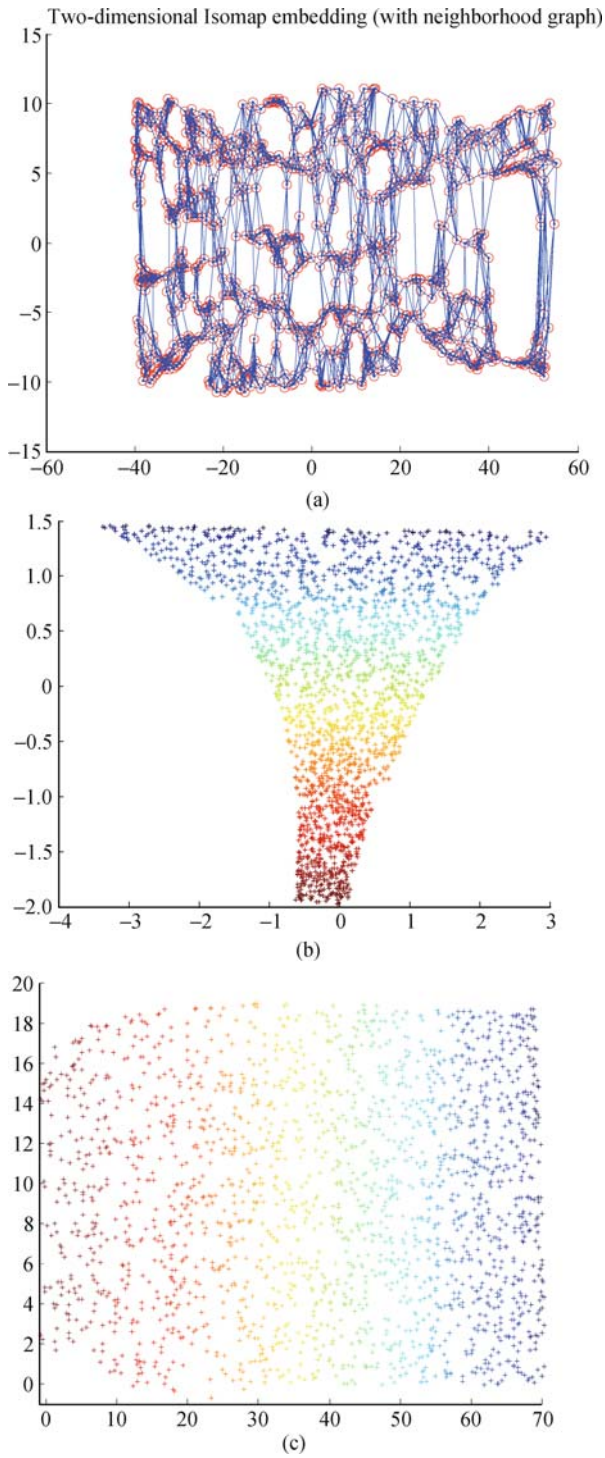

Fig. 2 Embedding of typical methods. (a) Isomap; (b) LLE; (c) gViSOM

used with degree of 2 and radius of 30, respectively. The sizes of the neighborhood used by LLE and Isomap were varied from 6 to 30 and 12 to120, respectively, with the best performances selected. Then the classification was conducted by using the NN, soft k-NN, LDA and SVM classifiers.

In block PCA [74], 2DPCA [75] and SOM-based approaches for dimension reduction, each face image was

first locally sampled by moving a window of size 5×5 over the entire image by four pixels each time. That is, each sampled face image contains $23 \times 28 = 644$ 25-dimensional subsamples. These 25-dimensional samples were used as the inputs for training SOM-based manifolds. For each method, its size and parameters had been optimized to its best performance. For example, the sizes of SOM, ViSOM and gViSOM varied from 5×5 to 30×30, and the chosen sizes represent the cases with the best performances. Then all 25-dimensional samples in each face image are passed through the trained SOM, ViSOM and gViSOM maps, and represented by the 2-D index values of the corresponding winners on the maps. Thereby, on the trained map, each face image has a corresponding 2-D face projection [5], which is used for further classification. Each dimension of the face projection can be reconstructed as a feature face, which resembles features of the original face images. It has been shown that ViSOM methods provide better feature faces due to the metric preserving manifold property.

The classification performances of these methods under various splits of training and test samples were conducted. The results reported are the average results of ten independent implementations with different randomly chosen training and test images. Meanwhile, the same choices of training and test images were used by all the methods to ensure an unbiased comparison. The results of PCA-based methods and MDS-based methods followed by an NN, soft k-NN, LDA and SVM classifier are shown in Table 1. The results of block PCA and SOM-based methods are listed in Table 2. ViSOM and gViSOM in the tables further use the LLP resolution enhancement. Similar results have been produced on the Yale data set [76].

The tables show that with more training samples, the classification rates increase in all methods as expected. The SOM has the similar performances to PCA-based methods with the NN classifier; ViSOM and gViSOM yield markedly improved performances over the SOM and other nonlinear PCA methods by about 2% in rate in each implementation. With soft k-NN classifier, LLE has slightly lower error rates than other PCA-based methods, while SOM-based methods have about 2-3% improvements over the LLE, and gViSOM with LLP has even better performances than the SOM with more than 1% further improvement.

Three findings can be drawn from the results: 1) little difference between vector- and matrix-based implementation in linear methods (PCA, 2DPCA and BPCA), 2) slight but insignificant improvements by nonlinear PCA methods and SOM, and 3) marked improvements by ViSOM and gViSOM. The classification rates of gViSOM followed by a soft k-NN classifier are significantly higher than all other methods in the same training/test scheme

Table 1 Classification rates of vector-based dimension reductions (ORL database)

No. of training faces	classification rates/%				
	PCA	KPCA	LLE	Isomap	CCA
	NN classifier				
3	86.75	87.75	**88.78**	85.79	**87.75**
4	91.92	**92.83**	92.51	91.46	91.83
5	94.35	94.20	**94.60**	93.15	94.45
6	**96.44**	95.94	96.07	95.87	96.04
	Soft k-NN classifier				
3	86.75	88.25	**88.71**	85.86	87.32
4	91.31	90.92	**92.75**	91.21	91.54
5	93.85	93.00	**94.50**	93.10	94.15
6	95.74	94.13	96.06	95.25	**96.19**
	LDA classifier				
3	**90.64**	89.93	90.29	86.54	87.85
4	**94.92**	94.04	93.25	91.63	92.79
5	96.20	95.05	**96.25**	93.10	94.60
6	96.88	96.81	**97.31**	95.69	95.96
	SVM				
3	90.21	86.75	**91.07**	86.61	89.71
4	**94.79**	91.42	94.46	91.08	94.46
5	**96.70**	94.20	95.65	93.30	96.25
6	**97.75**	96.26	97.13	96.19	97.06

Table 2 Classification rates of matrix-based dimension reductions (ORL database)

No. of training faces	classification rates/%				
	2DPCA	BPCA	SOM	ViSOM	gViSOM
	NN classifier				
3	88.86	87.14	88.43	**89.50**	89.21
4	93.33	91.63	92.50	**93.63**	93.46
5	95.60	93.65	94.15	**95.65**	95.50
6	97.13	95.44	96.19	**97.37**	97.12
	Soft k-NN classifier				
3	89.96	89.54	91.96	92.68	**93.29**
4	94.04	93.33	95.54	96.21	**96.33**
5	96.10	94.80	96.80	97.60	**97.90**
6	97.56	96.63	98.12	98.81	**99.25**

with ORL data. In training five and six, these rates reach 97.9% and 99.25%, respectively. This is largely due the better (metric-based) feature faces extracted [5]. Although the matrix-based and ViSOM methods seem to capture better face features by using local sampling and yield higher classification rates, but require a larger number of retained dimensions, thus increasing the computational and storage demand. In addition, the larger number of retained dimensions of matrix-based methods may cause a singularity problem when applying LDA on these data [77], as the number of reduced dimensions is still greater than the number of face images in both face databases.

7 Conclusions

The paper provides an overview on nonlinear principal manifolds and dimensionality reduction and a review on their recent advances. The existing methods have been categorized into four main groups: MDS-based, eigen-decomposition based, principal curve based, and SOM-based. Eigen-decomposition approaches are particularly advantageous, especially when they are framed under the kernel method or spectral analysis so that the problem is converted to a linear eigen problem with a unique solution expected. There are a number of geometrical or kernel parameters to be set empirically and the mapping result can be sensitive to these parameters. They usually are not adaptive learning methods and some recent work exists on extending them to incremental learning. Adaptive learning approaches such as SOM-based can gradually extract low dimensional manifolds. SOM-based methods have been shown as either nonmetric or metric MDS. The flexibility and diversity of these adaptive learning approaches make them widely applicable and integrateable with other computational paradigms. Their application for dimensionality reduction has been discussed. Experimental results show that the adaptive manifolds based on ViSOM or gViSOM can outperform marginally but consistently linear PCA and various other nonlinear methods, though at increased computational costs. It is largely due to the fact that ViSOM is an adaptive and metric preserving manifold. Various other applications of dimensionality reduction and manifolds have also been reviewed, together with some challenges ahead. The research and application in nonlinear manifolds are set to flourish in many disciplines in this data-rich era. Further advances will be gained in dealing with temporal data, heterogeneous data and data of multiple structures, where single, convex manifolds can rarely fit well in practice.

Acknowledgements The author would like to thank the reviewers and editors for their valuable comments and suggestions.

References

1. Fodor I K. A survey of dimension reduction techniques. Technical Report UCRL-ID-148494. San Francisco: Lawrence Livermore National Laboratory, 2002
2. Gorban A N, Kégl B, Wunsch D C, Zinovyev A. Principal Manifolds for Data Visualization and Dimension Reduction. Berlin: Springer, 2008
3. Van der Maaten L J P, Postma E O, van den Herik H J. Dimensionality reduction: a comparative review. *Online Preprint* (2008)
4. Yin H. Nonlinear dimensionality reduction and data visualization: A review. International Journal on Automation and Computing, 2007, 3(3): 294–303
5. Yin H, Huang W. Adaptive nonlinear manifolds and their applications to pattern recognition. Information Science, 2010, 180(14): 2649–2662
6. Cox T F, Cox M A A. Multidimensional Scaling. London: Chapman & Hall, 1994
7. Tenenbaum J B, de Silva V, Langford J C. A global geometric framework for nonlinear dimensionality reduction. Science, 2000, 290(5500): 2319–2323
8. Mao J, Jain A K. Artificial neural networks for feature extraction and multivariate data projection. IEEE Transactions on Neural Networks, 1995, 6(2): 296–317
9. Lowe D, Tipping M E. Feed-forward neural networks and topographic mappings for exploratory data analysis. Neural Computing & Applications, 1996, 4(2): 83–95
10. Malthouse E C. Limitations of nonlinear PCA as performed with generic neural networks. IEEE Transactions on Neural Networks, 1998, 9(1): 165–173
11. Kramer M A. Nonlinear principal component analysis using autoassociative neural networks. American Institute of Chemical Engineers Journal, 1991, 37(1): 233–243
12. Karhunen J, Joutsensalo J. Generalization of principal component analysis, optimization problems, and neural networks. Neural Networks, 1995, 8(4): 549–562
13. Schölkopf B, Smola A, Müller K R. Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation, 1998, 10(5): 1299–1319
14. Roweis S T, Saul L K. Nonlinear dimensionality reduction by locally linear embedding. Science, 2000, 290(5500): 2323–2326
15. Kouropteva O, Okun O, Pietikäinen M. Incremental locally linear embedding. Pattern Recognition, 2005, 38(10): 1764–1767
16. Belkin M, Niyogi P. Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation, 2003, 15(6): 1373–1396
17. Weiss Y. Segmentation using eigenvectors: a unified view. In: Proceedings of IEEE International Conference on Computer Vision. 1999, 975–982
18. Hastie T, Stuetzle W. Principal curves. Journal of the American Statistical Association, 1989, 84(406): 502–516
19. Banfield J D, Raftery A E. Ice floe identification in satellite images using mathematical morphology and clustering about principal curves. Journal of the American Statistical Association, 1992, 87(417): 7–16
20. Delicado P. Another look at principal curves and surfaces. Journal of Multivariate Analysis, 2001, 77(1): 84–116
21. Kégl B, Krzyzak A, Linder T, Zeger K. A polygonal line algorithm for constructing principal curves. Advances in Neural Information Processing Systems, 1998, 11: 501–507
22. Chang K Y, Ghosh J. A unified model for probabilistic principal surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2001, 23(1): 22–41
23. Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks. Science, 2006, 313(5786): 504–507
24. Kohonen T. Self-organized formation of topologically cor-

rect feature map. Biological Cybernetics, 1982, 43(1): 56–69

25. Kohonen T. Self-Organizing Maps. 2nd ed. Berlin: Springer, 1997
26. Willshaw D J, von der Malsburg C. How patterned neural connections can be set up by self-organization. Proceedings of the Royal Society of London. Series B. Biological Sciences, 1976, 194(1117): 431–445
27. Yin H. ViSOM-A novel method for multivariate data projection and structure visualization. IEEE Transactions on Neural Networks, 2002, 13(1): 237–243
28. Ritter H, Martinetz T, Schulten K. Neural Computation and Self-Organizing Maps: An Introduction. Menlo Park: Addison-Wesley Publishing Company, 1992
29. Yin H. Data visualization and manifold mapping using the ViSOM. Neural Networks, 2002, 15(8-9): 1005–1016
30. Yin H. On multidimensional scaling and the embedding of self-organizing maps. Neural Networks, 2008, 21(2-3): 160–169
31. Borg I, Groenen P J F. Modern Multidimensional Scaling: Theory and Applications, 2nd ed. New York: Springer, 2005
32. Sammon J W. A nonlinear mapping for data structure analysis. IEEE Transactions on Computers, 1969, 18(5): 401–409
33. Bronstein A M, Bronstein M M, Kimmel R. Generalized multidimensional scaling: A framework for isometry-invariant partial surface matching. Proceedings of the National Academy of Sciences of the United States of America, 2006, 103(5): 1168–1172
34. Lee R C T, Slagle J R, Blum H. A triangulation method for the sequential mapping of points from n-space to two-space. IEEE Transactions on Computers, 1977, 27(3): 288–292
35. De Ridder D, Duin R P W. Sammon mapping using neural networks: a comparison. Pattern Recognition Letters, 1997, 18(11-13): 1307–1316
36. Balasubramanian M, Schwartz E L. The Isomap algorithm and topological stability. Science, 2002, 295(5552): 7
37. De Silva V, Tenenbaum J B. Global versus local methods in nonlinear dimensionality reduction. Advances in Neural Information Processing Systems, 2003, 15: 705-712
38. Agrafiotis D K. Stochastic proximity embedding. Journal of Computational Chemistry, 2003, 24(10): 1215–1221
39. Demartines P, Hérault J. Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets. IEEE Transactions on Neural Networks, 1997, 8(1): 148–154
40. Law M H, Jain A K. Incremental nonlinear dimensionality reduction by manifold learning. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(3): 377–391
41. Goodhill G J, Sejnowski T. A unifying objective function for topographic mappings. Neural Computation, 1997, 9(6): 1291–1303
42. Oja E. Neural networks, principal components, and subspaces. International Journal of Neural Systems, 1989, 1(1): 61–68
43. Sanger T D. Optimal unsupervised learning in a single-layer linear feedforward network. Neural Networks, 1991, 2(6): 459–473
44. Ham J, Lee D D, Mika S, Schölkopf B. A kernel view of the dimensionality reduction of manifolds. In: Proceedings of the 21st International Conference on Machine Learning. 2004, 369–376
45. Saul L K, Roweis S T, Singer Y. Think globally, fit locally: unsupervised learning of nonlinear manifolds. Journal of Machine Learning Research, 2003, 4(2): 119–155
46. De Ridder D, Duin R P W. Locally linear embedding for classification. Technical Report PH-2002–01. Netherlands: Delft University of Technology, 2002
47. Bengio Y, Paiement J F, Vincent P, Delalleau O, Le Roux N, Ouimet M. Out-of-sample extensions for LLE, Isomap, MDS, eigenmaps, and spectral clustering. Advances in Neural Information Processing Systems, 2004, 16: 177–184
48. Donoho D L, Grimes C E. Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. Proceedings of the National Academy of Sciences of the United States of America, 2003, 100(10): 5591–5596
49. He X, Niyogi P. Locality preserving projections. Advances in Neural Information Processing Systems, 2003, 16: 152–160
50. Smola A J, Mika S, Schölkopf B, Williamson R C. Regularized principal manifolds. In: Proceedings of the 4th European Conference on Computational Learning Theory. 1999, 214–229
51. Yan S C, Xu D, Zhang B, Zhang H, Yang Q, Lin S. Graph embedding and extension: a general framework for dimensionality reduction. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(1): 40–51
52. Zhang Z, Zha H. Principal manifolds and nonlinear dimension reduction via local tangent space alignment. SIAM Journal on Scientific Computing, 2005, 26(1): 313–338
53. Weinberger K Q, Saul L K. Unsupervised learning of image manifolds by semidefinite programming. International Journal of Computer Vision, 2006, 70(1): 77–90
54. LeBlanc M, Tibshirani R J. Adaptive principal surfaces. Journal of the American Statistical Association, 1994, 89(425): 53–64
55. Tibshirani R. Principal curves revisited. Statistics and Computing, 1992, 2(4): 183–190
56. Bishop C M, Svensén M, Williams C K I. GTM: The generative topographic mapping. Neural Computation, 1998, 10(1): 215–235
57. Kohonen T. Self-organization and associative memory. Berlin: Springer-Verlag, 1984
58. Yin H, Allinson N M. On the distribution and convergence of the feature space in self-organizing maps. Neural Computation, 1995, 7(6): 1178–1187
59. Luttrell S P. Derivation of a class of training algorithms. IEEE Transactions on Neural Networks, 1990, 1(2): 229–232
60. Luttrell S P. A Bayesian analysis of self-organizing maps. Neural Computation, 1994, 6(5): 767–794
61. Durbin R, Mitchison G. A dimension reduction framework for understanding cortical maps. Nature, 1990, 343(6259):

644–647

62. Mitchison G. A type of duality between self-organizing maps and minimal wiring. Neural Computation, 1995, 7(1): 25–35
63. Ripley B D. Pattern Recognition and Neural Networks. Cambridge: Cambridge University Press, 1996
64. Flexer A. Limitations of self-organizing maps for vector quantization and multidimensional scaling. Advances in Neural Information Processing Systems, 1997, 10: 445–451
65. Yin H. Resolution enhancement for the ViSOM. In: Proceedings of Workshop on Self-Organizing Maps. 2003, 208–212
66. Wu S, Chow T W S. PRSOM: A new visualization method by hybridizing multidimensional scaling and self-organizing map. IEEE Transactions on Neural Networks, 2005, 16(6): 1362–1380
67. Estévez P A, Figueroa C J. Online data visualization using the neural gas network. Neural Networks, 2006, 19(6-7): 923–934
68. Kohonen T. The adaptive-subspace SOM (ASSOM) and its use for the implementation of invariant feature detection. In: Proceedings of International Conference on Artificial Neural Networks. 1995, 3–10
69. Yin H, Allinson N M. Self-organizing mixture networks for probability density estimation. IEEE Transactions on Neural Networks, 2001, 12(2): 405–411
70. Fyfe C. Two topographic maps for data visualization. Data Mining and Knowledge Discovery, 2007, 14(2): 207–224
71. Gorban A, Zinovyev A. Method of elastic maps and its applications in data visualization and data modeling. International Journal of Computing Anticipatory Systems, 2001, 12: 353–369
72. Tan X, Chen S, Zhou Z, Zhang F. Recognizing partially occluded, expression variant faces from single training image per person with SOM and soft k-NN ensemble. IEEE Transactions on Neural Networks, 2005, 16(4): 875–886
73. Fisher R A. The use of multiple measures in taxonomic problems. Annals of Eugenics, 1936, 7(2): 179–188
74. Kim M, Kim D, Lee S. Face recognition using the embedded HMM with second-order block specific observations. Pattern Recognition, 2003, 36(11): 2723–2735
75. Yang J, Zhang D, Frangi A F, Yang J. Two-dimensional PCA: A new approach to appearance-based face representation and recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(1): 131–137
76. Huang W, Yin H. Nonlinear dimensionality reduction for face recognition. In: Proceedings of the 10th International Conference on Intelligent Data Engineering and Automated Learning. 2009, 424–432
77. Ji S, Ye J. Generalized linear discriminant analysis: a unified framework and efficient model selection. IEEE Transactions on Neural Networks, 2008, 19(10): 1768–1782

Dr. Hujun YIN has been with The University of Manchester School of Electrical and Electronic Engineering since 1996. He received BEng and MSc degrees from Southeast University and Ph.D degree from University of York in 1983, 1986 and 1996, respectively. His main research interests include neural networks, self-organising systems in particular, image processing, pattern recognition, and bio-neuro-informatics. He has studied and extended the self-organising map (SOM) and related topics (principal manifolds and data visualisation) extensively and proposed a number of extensions including Bayesian SOM and ViSOM, a principled nonlinear manifold and data visualisation method. He has published over 100 peer-reviewed articles in a range of topics from density modelling, image processing, face recognition, text mining and knowledge management, gene expression analysis and peptide sequencing, novelty detection, to financial time series modelling, and recently decoding neuronal responses (spike trains and local field potentials). He is a senior member of the IEEE and a member of the UK EPSRC College. He has been an Associate Editor of the IEEE Transactions on Neural Networks (2006–2009) and a member of the Editorial Board of the International Journal of Neural Systems (since 2005), among other editorial duties. He has been the Organising Chair, Programme Committee Chair, and General Chair for a number of conferences, such as 2001 International Workshop on Self-Organising Maps (WSOM'01), International Conference on Intelligent Data Engineering and Automated Learning (IDEAL) (2002-2008), 2006 International Symposium on Neural Networks (ISNN'06). He has received research funding from the UK EPSRC, BBSRC and DTI. He is a regular assessor for the EPSRC, BBSRC, Royal Society, Hong Kong Research Grant Council, and Netherlands Organisation for Scientific Research. URL: http://personalpages.manchester.ac.uk/staff/hujun.yin/