Simbed: Similarity-Based Embedding

John A. Lee[1,⋆] and Michel Verleysen[2]

[1] Imagerie Moléculaire et Radiothérapie Expérimentale,
Avenue Hippocrate, 54, B-1200 Bruxelles
[2] Machine Learning Group,
Place du Levant, 3, B-1348 Louvain-la-Neuve, Belgium
{john.lee,michel.verleysen}@uclouvain.be
http://www.ucl.ac.be/mlg

Abstract. Simbed, standing for similarity-based embedding, is a new method of embedding high-dimensional data. It relies on the preservation of pairwise similarities rather than distances. In this respect, Simbed can be related to other techniques such as stochastic neighbor embedding and its variants. A connection with curvilinear component analysis is also pointed out. Simbed differs from these methods by the way similarities are defined and compared in both the data and embedding spaces. In particular, similarities in Simbed can account for the phenomenon of norm concentration that occurs in high-dimensional spaces. This feature is shown to reinforce the advantage of Simbed over other embedding techniques in experiments with a face database.

Keywords: Nonlinear dimensionality reduction, similarity measure, manifold learning, stochastic gradient, multiscale optimization.

1 Introduction

Dimensionality reduction is the task of finding faithful, low-dimensional representations of high-dimensional data. Although the case of clustered data can be considered, it usually relies on the assumption that the data are sampled from a smooth manifold. For instance, if the underlying manifold is a linear subspace, then methods such as principal component analysis (PCA) [12] or classical metric multidimensional scaling [21] can be successfully applied. However, these techniques are not optimal if the manifold is heavily curved or folded [19]. This issue can be addressed by using methods of nonlinear dimensionality reduction [10] (NLDR) instead of a linear projection. The development of nonlinear variants of MDS lead in the early sixties to many techniques that are based on the principle of distance preservation. Nonmetric MDS [18,8] and Sammon's nonlinear mapping [15] (SNLM) are the best known methods in this family. The eighties and early nineties saw the advent of methods related to artificial neural networks and soft-computing. Auto-encoders with multilayer perceptrons [7] and

⋆ J.A.L. is a Postdoctoral Researcher with the Belgian National Fund for Scientific Research (FNRS).

C. Alippi et al. (Eds.): ICANN 2009, Part II, LNCS 5769, pp. 95–104, 2009.

Kohonen's self-organizing maps [6] (SOMs) are the most prominent examples in this trend. Since the late nineties and the seminal paper describing kernel PCA [17], many recent develoments in NLDR have targeted spectral embedding [16]. Isomap [19] and locally linear embedding [14] are probably the most representative methods in this branch. Spectral methods provide the guarantee of finding the global optimum of their cost function. In contrast, methods based on other optimization techniques generally do not offer this advantage. However, they usually compensate for this drawback by the capability of handling a broader range of cost functions. Successful nonspectral methods are for instance curvilinear component analysis [1,4] (CCA), stochastic neighbor embedding [5] (SNE), and its variant t-SNE [20].

This paper introduces Simbed, a new NLDR method that relies on similarity matching in order to embed data in a low-dimensional space. Simbed's most prominent feature is its principled way of computing pairwise similarities that accounts for the phenomenon of norm concentration [3]. Briefly put, this term refers to the fact that the norm of high-dimensional vectors tends to have a low variance/expectation ratio. Hence, Simbed owns a decisive advantage when it comes to real-life data that combine non-negligible noise with a high dimensionality. The paper also weaves connections with other methods that involve similarities, such as SNE and t-SNE. An unexpected relationship with CCA is also pointed out, which shows that CCA is closer to similarity matching than to distance preserving techniques such as SNLM.

The rest of this paper is organized as follows. Section 2 introduces notations for distances and gives a principled definition of pairwise similarities in high-dimensional spaces. Section 3 describes a cost function that assesses the similarity matching, along with an algorithm that optimizes it. Section 4 points out some connections with other techniques, such as CCA and t-SNE. Section 5 gathers some experimental results with both artificial and real data. Finally, Section 6 draws the conclusions.

2 Distances and Similarities

Let $\boldsymbol{\Xi} = [\boldsymbol{\xi}_i]_{1 \leq i \leq N}$ denote a data set of N vectors picked in an M dimensional space. The symbol δ_{ij} denotes the pairwise Euclidean distance $\|\boldsymbol{\xi}_i - \boldsymbol{\xi}_j\|_2$.

In order to define a similarity measure, let us consider vector $\boldsymbol{\xi}_i$ and an isotropic k-dimensional normal distribution centered on it, with $k \leq M$. We define the similarity of $\boldsymbol{\xi}_j$ with respect to $\boldsymbol{\xi}_i$ to be the probability of the event $\|\boldsymbol{\xi} - \boldsymbol{\xi}_i\|_2 \geq \delta_{ij}$ where $\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{\xi}_i, \lambda \mathbf{I})$. In other words, the similarity is the probability of observing a larger distance than the measured value and thus varies between 0 and 1. The normality assumption allows us to write $\|\boldsymbol{\xi} - \boldsymbol{\xi}_i\|_2 / \lambda \sim \chi_k$, where χ_k denotes a chi distribution with k degrees of freedom [2]. The probability density function of $c \sim \chi_k$ is given by

$$p(c,k) = \frac{\sqrt{2}}{\Gamma(k/2)} \left(\frac{c}{2^{1/2}} \right)^{k-1} \exp(-c^2/2) \ , \tag{1}$$

where Γ is the Gamma function. Therefore, the similarity between $\boldsymbol{\xi}_i$ and $\boldsymbol{\xi}_j$ can be defined by

$$\sigma_{ij}(\lambda, k) \doteq \text{Prob}\left[\delta_{ij} \geq \lambda c\right] = \int_{\delta_{ij}}^{\infty} \frac{p(z/\lambda, k)}{\lambda}\, dz = Q\left(\frac{\delta_{ij}^2}{2\lambda^2}, \frac{k}{2}\right) \quad , \tag{2}$$

where $c \sim \chi_k$ and Q is the regularized upper incomplete gamma function. This definition contains two free parameters, namely λ and k.

Standard deviation λ reflects the fact that the proposed similarity measure is a scale-dependent concept. Hence, this parameter basically sets up the threshold between the 'local' neighborhood of some vector and the rest of the space. Its value can then be arbitrarily fixed by the user. If we consider that $\boldsymbol{\Xi}$ contains noisy vectors sampled from some manifold, λ should not go below the (local) noise standard deviation. Similarly, larger values than $\max_{1\leq i,j\leq N} \delta_{ij}$ make little sense. As will be shown later on, a multiscale or multiresolution approach that explores several values of λ can be useful.

As to k, which specifies the number of degrees of freedom, its optimal value depends on λ. If λ is close to the standard deviation of the noise, k should be equal to M, since noise indifferently spans all dimensions of space. For slightly larger values of λ, noise becomes negligible and the manifold can be locally approximated by a linear subspace with as many degrees of freedom as the manifold intrinsic dimensionality. For larger values of λ, k should be chosen according to the global shape of the manifold, which is difficult to investigate.

It is noteworthy that for appropriate values of k and λ the proposed similarity can account for the phenomenon of norm concentration [3]. Defining the similarity with a Gaussian kernel (this turns out to be the case $k = 2$) does not offer this possibility. The model behind the proposed similarity can be refined, for instance by using anisotropic normal distributions, but this introduces additional parameters.

3 Matching the Pairwise Similarities

NLDR aims at finding a low-dimensional representation of data set $\boldsymbol{\Xi}$. Let $\mathbf{X} = [\mathbf{x}_i]_{1\leq i\leq N}$ denote this representation and let P be its dimensionality. Symbol d_{ij} refers to the pairwise Euclidean distance $\|\mathbf{x}_i - \mathbf{x}_j\|_2$. As in the previous section, we can define the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$ in the same way, i.e.

$$s_{ij}(\lambda, l) \doteq \text{Prob}\left[d_{ij} \geq \lambda c\right] = Q\left(\frac{d_{ij}^2}{2\lambda^2}, \frac{l}{2}\right) \quad , \tag{3}$$

where $c \sim \chi_l$. Scale parameter λ can be reused from the previous definition, since we look at the same scale in both high- and low-dimensional spaces. The key difference lies in l, whose value is simply equal to embedding dimensionality P in this case.

Having defined the similarity measures in both spaces, one can try to match them by minimizing the mean square error

$$E(\mathbf{X}, \Xi, \lambda, k, l) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\sigma_{ij}(\lambda, k) - s_{ij}(\lambda, l)\right)^2 \tag{4}$$

with respect to $\mathbf{X}$. The intuition behind this cost function is similar to that behind distance preservation: building a low-dimensional representation that preserves pairwise distances or similarities hopefully keeps neighboring data items close to each other, while maintaining the gap between dissimilar ones. Simbed —which stands for similarity-based embedding— minimizes (4) by performing a stochastic gradient descent such as in [1]. For this purpose, we separately consider all terms of the outer sum in the cost function. They can be written as

$$E_i(\mathbf{X}, \Xi, \lambda, k, l) = \frac{1}{N^2} \sum_{j=1}^{N} \left(\sigma_{ij}(\lambda, k) - s_{ij}(\lambda, l)\right)^2 \quad . \tag{5}$$

The partial derivative of $E_i(\mathbf{X}, \Xi, \lambda, k, l)$ with respect to $\mathbf{x}_j$ is

$$\frac{\partial E_i(\mathbf{X}, \Xi, \lambda, k, l)}{\partial \mathbf{x}_j} = \frac{2}{N^2} \left(\sigma_{ij}(\lambda, k) - s_{ij}(\lambda, l)\right) \frac{p(d_{ij}/\lambda, l)}{\lambda} \frac{\mathbf{x}_j - \mathbf{x}_i}{d_{ij}} \quad . \tag{6}$$

This leads to the iterative update $\mathbf{x}_j := \mathbf{x}_j + \Delta\mathbf{x}_j^{(t,i)}$ for $1 \leq j \leq N$, where

$$\Delta\mathbf{x}_j^{(t,i)} = -\alpha^{(t)} \frac{\partial E_i(\mathbf{X}, \Xi, \lambda^{(t)}, k^{(t)}, l)}{\partial \mathbf{x}_j} \quad . \tag{7}$$

Index t denotes the current iteration (or 'epoch') and $\alpha^{(t)}$ is the step size (or learning rate). At each iteration, index i runs over the whole data set. Each iteration thus performs N^2 updates for a time complexity of $\mathcal{O}(N^2)$.

In the spirit of a pseudo-Newton optimization scheme, an appropriate value of $\alpha^{(t)}$ should be related to the second-order derivative. A single value of $\alpha^{(t)}$ that fits for all i and j should satisfy the second inequality in

$$\left\| \frac{\partial^2 E_i(\mathbf{X}, \Xi, \lambda^{(t)}, k^{(t)}, l)}{\partial \mathbf{x}_j^2} \right\|_2 \leq \frac{2}{(\lambda^{(t)} N)^2} \leq \frac{1}{\alpha^{(t)}} \quad . \tag{8}$$

The first inequality gives an upper bound of the second partial derivative. Therefore, $\alpha^{(t)}$ should be lower than $(N\lambda^{(t)})^2/2$. In a classical stochastic gradient descent, $\alpha^{(t)}$ should slowly decay [13]. This can be indirectly achieved by progressively reducing $\lambda^{(t)}$ in a multiscale approach. The principle of a multiscale optimization is rather simple and consists in 'blurring' the cost function in the early part of the process. Doing so smooths out narrow pits and peaks, thus putting the emphasis on the widest and deepest basins. Next, the amount of blur is slowly reduced as the optimization progresses. Hence, a multiscale strategy increases the probability of finding the global optimum. As a matter of fact,

Simbed's cost function can be optimized with a multiscale scheme because it can be written as a continuous function of smooth and rapidly decaying kernels. The parameters to be optimized, namely the coordinates in $\mathbf{X}$, exclusively appear in the arguments of these similarity kernels.

The efficiency of a multiscale optimization increases if scale parameter λ slowly decays. In the case of a stochastic gradient descent, we can combine the decay of λ with that of the step size. Simbed relies on the schedules given by $\lambda^{(t)} = 4 \max_{1 \leq i,j \leq N} \delta_{ij}/t$ and $\alpha^{(t)} = 0.1(N\lambda^{(t)})^2$. The latter fulfills the inequality mentioned in the previous section and ensures that $\|\Delta \mathbf{x}_j^{(t,i)}\|_2$ slowly vanishes as t grows to infinity. In practice, Simbed runs for at most T iterations but an early stop is possible when the criterion

$$\sum_{i=1}^{N} \sum_{j=1}^{N} \left| \left(d_{ij}^{(t)}\right)^2 - \left(d_{ij}^{(t-1)}\right)^2 \right| \leq \epsilon \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\left(d_{ij}^{(t)}\right)^2 + \left(d_{ij}^{(t-1)}\right)^2 \right) \quad , \tag{9}$$

is met, in which $d_{ij}^{(t)}$ refers to the pairwise distance at the end of iteration t. As to dimensionality parameter k, the constant value P can be used for noisefree data, whereas a schedule such as $k^{(t)} = P + (M - P)t/T$ can be adopted for noisy data.

The initialization of the algorithm can be achieved with a P-dimensional PCA projection. During the stochastic gradient descent, it is advised to randomly permute the order of the updates with respect to index i in (7). A reinitialization of the random number generator with always the same seed makes the optimization fully deterministic, provided the data vectors are not permuted in $\mathbf{\Xi}$ from run to run. These permutations can be avoided by computing $c = \arg\min_i \sum_{j=1}^{N} \delta_{ij}^2$ and by sorting the vectors in $\mathbf{\Xi}$ according to δ_{cj}.

4 Connection with Other Techniques

Simbed can be related to several other methods described in the literature, such as SNE, t-SNE, CCA, and SOMs. SNE and t-SNE follow the same paradigm as Simbed, that is, similarity matching with a stochastic optimization scheme. However several important differences can be pointed out.

First, the pairwise similarities used in Simbed involve cumulative distribution functions whose value depends only on the corresponding distances. In contrast, SNE and t-SNE rely on empirical probability density functions defined as

$$\sigma_{ij}(\lambda) \doteq \frac{g(\delta_{ij}/\lambda)}{\sum_{m \neq n} g(\delta_{mn})} \quad \text{and} \quad s_{ij} \doteq \frac{h(d_{ij})}{\sum_{m \neq n} h(d_{mn})} \quad , \tag{10}$$

where $g(u) = \exp(-u^2/2)$ and either $h(u) = g(u)$ for SNE or $h(u) = 1/(1 + u^2)$ for t-SNE ($h(u)$ is proportional to a Student's t pdf with one degree of freedom). These similarity functions involve a softmax denominator; it ensures that $\sum_{i \neq j} \sigma_{ij}(\lambda) = \sum_{i \neq j} s_{ij} = 1$. Such a normalization makes all similarities interdependent and leads to paradoxical situations. For instance, equalities $g = h$

and $\delta_{ij} = \lambda d_{ij}$ do not imply $\sigma_{ij}(\lambda) = s_{ij}$. On the other hand, similarities in Simbed are such that $\delta_{ij} = d_{ij} \Leftrightarrow \sigma_{ij}(\lambda) = s_{ij}$, provided $k = l$ or $\delta_{ij} = 0$.

The second difference between Simbed and SNE/t-SNE results from the first one. Simbed estimates the similarity matching between the high- and low-dimensional spaces with a mean square error. In comparison, similarities in SNE and t-SNE are pdfs and their matching is thus computed by sums of Kullback-Leibler divergences in the cost function (see [20] for details and variants). This is a key point knowing that the particular choice of the similarity measure deeply impacts the form of the stochastic gradient update. The Gaussian functions in SNE lead to simplifications in the KL divergences whereas this does not happen in t-SNE. In (symmetric) SNE, the stochastic gradient update is proportional to $(\sigma_{ij}(\lambda) - s_{ij})(\mathbf{x}_j - \mathbf{x}_i)$ whereas it is proportional to $(\sigma_{ij}(\lambda) - s_{ij})(1/(1 + d_{ij}))(\mathbf{x}_j - \mathbf{x}_i)$ in t-SNE. This additional factor in t-SNE explains why it outperforms SNE [20]. Such a damping factor that decreases with respect to d_{ij} can be also found in Simbed as well as in CCA. This factor accounts for the capability of these methods to 'tear' manifolds [1,4,9].

A third difference concerns the absence of multiscale approach in SNE and t-SNE, although such a strategy has proved to be useful in methods such as CCA, SOMs and their variants.

The connection between Simbed and CCA can be investigated by looking at the terms of CCA's cost function, which are written as $E_i(\mathbf{X}, \boldsymbol{\Xi}, \lambda) = \sum_{j=1}^{N} (\delta_{ij} - d_{ij})^2 H(\lambda - d_{ij})$, where H denotes Heaviside's step function. The stochastic gradient update is proportional to $(1 - \delta_{ij}/d_{ij}) H(\lambda - d_{ij})(\mathbf{x}_j - \mathbf{x}_i)$. Although CCA is often related to Sammon's nonlinear mapping and other methods based on distance preservation, it can easily be cast within the framework of similarity preservation. For this purpose, we can equivalently rewrite CCA's cost function as $E_i(\mathbf{X}, \boldsymbol{\Xi}, \lambda) = \sum_{j=1}^{N} (\sigma_{ij}(\lambda) - s_{ij}(\lambda))^2$, where $\sigma_{ij}(\lambda) \doteq (\lambda - \delta_{ij}) H(\lambda - d_{ij})$ and $s_{ij}(\lambda) \doteq (\lambda - d_{ij}) H(\lambda - d_{ij})$. While $s_{ij}(\lambda)$ satisfies all conditions to be a similarity function, positivity of $\sigma_{ij}(\lambda)$ can be enforced by the approximation $\sigma_{ij}(\lambda) \approx (\lambda - \delta_{ij}) H(\lambda - \delta_{ij})$, which leads to

$$\frac{\partial E_i(\mathbf{X}, \boldsymbol{\Xi}, \lambda)}{\partial \mathbf{x}_j} = \left(1 - \frac{\delta_{ij}}{d_{ij}} - \frac{(\delta_{ij} - \lambda) H(\delta_{ij} - \lambda)}{d_{ij}}\right) H(\lambda - d_{ij})(\mathbf{x}_j - \mathbf{x}_i) \ . \quad (11)$$

The only difference with CCA's genuine gradient is the additional term in the first factor, which is non-zero only if $d_{ij} < \lambda < \delta_{ij}$. This shows that CCA is closely related to similarity-based embedding and that the multiplication by a Heaviside function in its cost function plays a much more important role than a simple weighting of the cost function terms, such as in SNLM.

5 Experiments

The experiments involve several data sets as well as several NLDR techniques. The first data set contains 750 vectors that sample a Swiss roll with uniform distribution. Its equation is written as $\boldsymbol{\xi} = [\sqrt{u}\cos(3\pi\sqrt{u}),\ \sqrt{u}\sin(3\pi\sqrt{u}),\ v]^T$,

where random parameters u and v have uniform distributions between 0 and 1. The second data set stems from the same manifold and includes 750 vectors as well, but these have three additional coordinates that are kept constant. Gaussian noise with standard deviation 0.05 is added to all six dimensions. The third data set contains 1965 pictures of B.J. Frey's face [14]. Each face is 20 pixels wide and 28 pixels high. After concatenation into 560-dimensional vectors, PCA achieves a first dimensionality reduction that leaves 20 coordinates.

Simbed is compared to t-SNE, CCA, SNLM, and PCA, whose result serves as baseline. Two versions of Simbed are used, one with constant and equal values for k and l, the second with the adaptive schedule for k. The implementation of t-SNE is provided by the authors of [20]; the 'perplexity' (i.e. the scale parameter) is left to its default value. CCA is implemented as in [4], with a constant step size that is equal to 0.2. The scale parameter follows a similar schedule as in Simbed, except that $\lambda^{(1)}$ is doubled; the stopping criterion is the same. SNLM is implemented as in [15] with a step size equal to 0.3. All methods are fed with pairwise Euclidean distances, no geodesic distances are used.

Performance assessment is achieved by means of the criteria proposed in [11]. These criteria look at K-ary neighborhoods around each vector in the data space as well as in the embedding space. The first criterion is denoted by $Q_{\mathrm{NX}}(K)$ and reflects the overall quality of the embedding; its value corresponds to the average percentage of identical neighbors in both spaces. The second criterion is denoted by $B_{\mathrm{NX}}(K)$ and reveals the 'behavior' of a NLDR method. A positive value indicates that distant points are embedded close to each other whereas a negative one indicates that close neighbors are embedded far away. Results for the three data sets are shown in Figs. 1 to 3. Each figure includes three panels; the first one spans the interval $1 \leq K \leq N-1$, whereas the small ones on the right focus on small values of K, for each criterion separately.

As to the noisefree Swiss roll, CCA slightly outperforms all other methods for small values of K. Simbed closely follows, whereas t-SNE comes third and precedes both SNLM and PCA. On the other hand, the global shape of the manifold is best preserved by PCA and SNLM, followed by Simbed, CCA, and t-SNE. Simbed thus reaches the best 'global-local' tradeoff. The multiscale optimization of CCA and Simbed actually leads to flatter curves than for the other methods. For this noisefree data set, Simbed with $k = l = 2$ performs slightly better than the variant with an increasing value of k, as expected.

The situation gets reversed in Fig. 2 for the noisy Swiss roll. Thanks to its use of similarity kernels with heavier tails in the embedding space than in the data space, t-SNE unfolds the Swiss roll better than the other methods and achieves the best performance for small values of K. Simbed is second and its version with an increasing k takes advantage of its more appropriate noise model. CCA comes next and precedes both SNLM and PCA.

The results for the face bank are shown in Fig. 3. Simbed with increasing values of k clearly outperforms the version with constant k. CCA climbs on the third step, followed by SNLM and PCA. Despite numerous attempts, t-SNE has never converged. As can be seen, similarity matching proves to be very

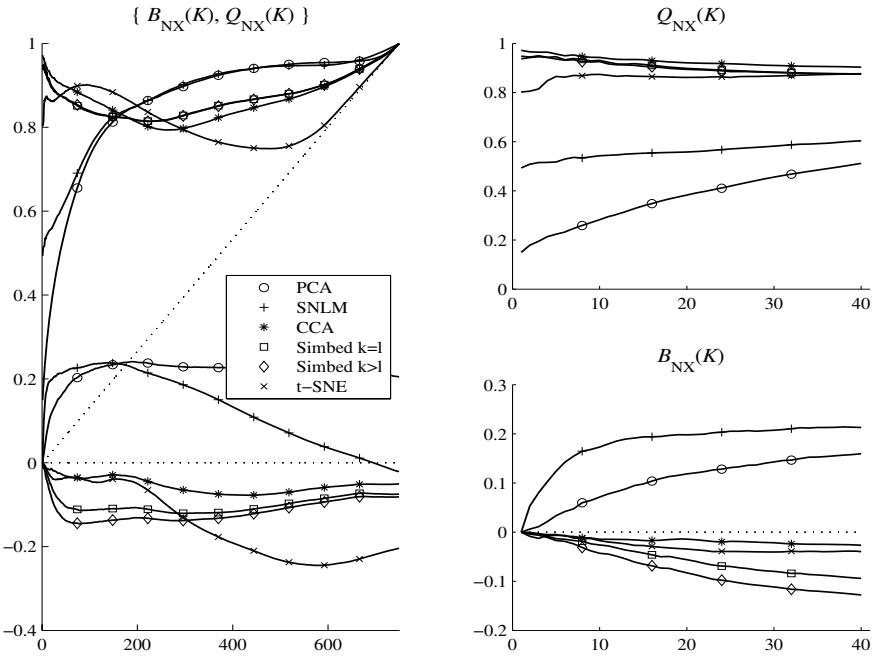

Fig. 1. Quality assessment for the embeddings of the noisefree Swiss roll

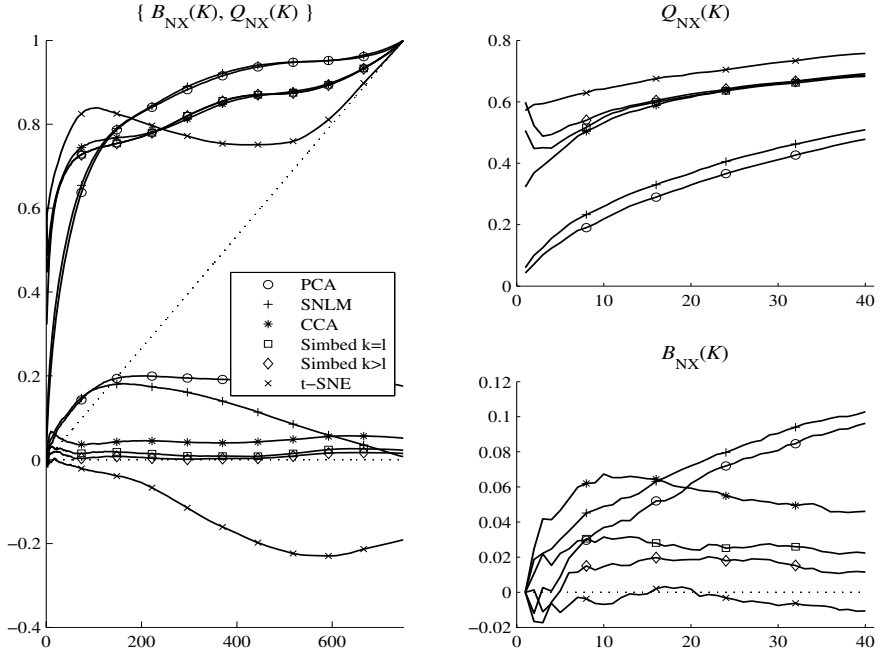

Fig. 2. Quality assessment for the embeddings of the noisy Swiss roll

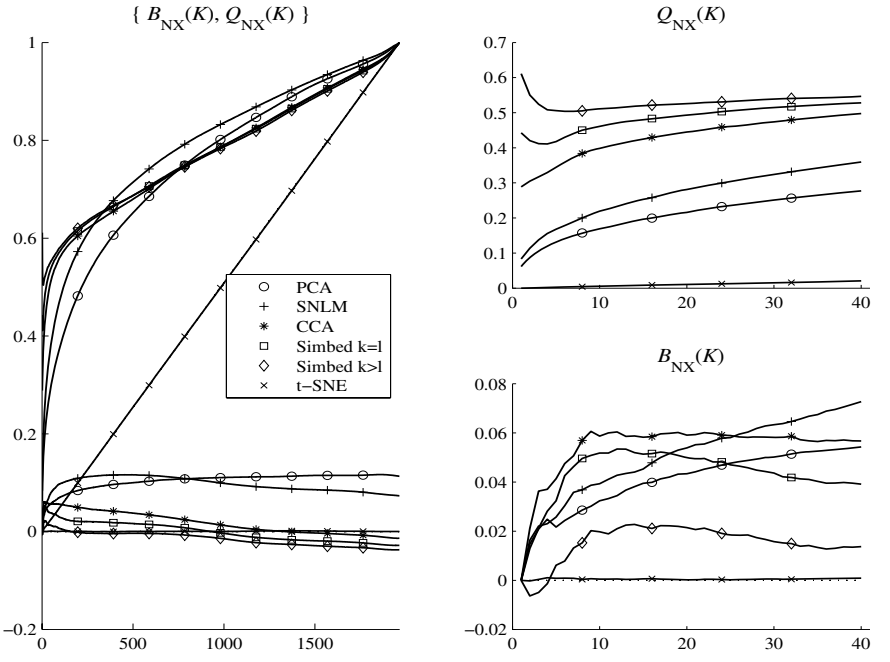

Fig. 3. Quality assessment for the embeddings of the face bank

efficient for this very high-dimensional data set. More importantly, the successive performance leaps between CCA and the two versions of Simbed indicate that the definition of the similarity kernel plays a key role as to the quality of the results. In the case of this data set, shifting from the piecewise linear kernel of CCA to smooth kernels that take into account the properties of high-dimensional spaces proves to be decisive.

6 Conclusion

Simbed is a new method of nonlinear dimensionality reduction that relies on similarity matching. It has two prominent features. First, it involves a principled similarity measure that can cope with the phenomenon of norm concentration in high-dimensional spaces. Second, its cost function can be optimized with a multiscale approach, which diminishes the probability of getting stuck in a local optimum. Simbed can be related to other methods such as SNE and t-SNE, and it also extends CCA.

Experiments with both artificial and real data show that Simbed compares to some of the best NLDR methods. It can provide excellent quantitative results in the case of noisy and very high-dimensional data, such as face images.

References

1. Demartines, P., Hérault, J.: Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. IEEE Transactions on Neural Networks 8(1), 148–154 (1997)
2. Evans, M., Hastings, N., Peacock, B.: Statistical Distributions, 3rd edn., New York (2000)
3. François, D., Wertz, V., Verleysen, M.: The concentration of fractional distances. IEEE Transactions on Knwoledge and Data Engineering 19(7), 873–886 (2007)
4. Hérault, J., Jaussions-Picaud, C., Guérin-Dugué, A.: Curvilinear component analysis for high dimensional data representation: I. Theoretical aspects and practical use in the presence of noise. In: Mira, J., Sánchez, J.V. (eds.) Proceedings of IWANN 1999, vol. II, pp. 635–644. Springer, Alicante (1999)
5. Hinton, G., Roweis, S.T.: Stochastic neighbor embedding. In: Becker, S., Thrun, S., Obermayer, K. (eds.) Advances in Neural Information Processing Systems (NIPS 2002), vol. 15, pp. 833–840. MIT Press, Cambridge (2003)
6. Kohonen, T.: Self-organization of topologically correct feature maps. Biological Cybernetics 43, 59–69 (1982)
7. Kramer, M.: Nonlinear principal component analysis using autoassociative neural networks. AIChE Journal 37(2), 233–243 (1991)
8. Kruskal, J.B.: Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. Psychometrika 29, 1–28 (1964)
9. Lee, J.A., Verleysen, M.: Curvilinear distance analysis versus isomap. Neurocomputing 57, 49–76 (2004)
10. Lee, J.A., Verleysen, M.: Nonlinear dimensionality reduction. Springer, Heidelberg (2007)
11. Lee, J.A., Verleysen, M.: Quality assessment of dimensionality reduction: Rank-based criteria. Neurocomputing (2009)
12. Pearson, K.: On lines and planes of closest fit to systems of points in space. Philosophical Magazine 2, 559–572 (1901)
13. Robbins, H., Monro, S.: A stochastic approximation method. Annals of Mathematical Statistics 22, 400–407 (1951)
14. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500), 2323–2326 (2000)
15. Sammon, J.W.: A nonlinear mapping algorithm for data structure analysis. IEEE Transactions on Computers CC-18(5), 401–409 (1969)
16. Saul, L.K., Weinberger, K.Q., Ham, J.H., Sha, F., Lee, D.D.: Spectral methods for dimensionality reduction. In: Chapelle, O., Schoelkopf, B., Zien, A. (eds.) Semisupervised Learning. MIT Press, Cambridge (2006)
17. Schölkopf, B., Smola, A., Müller, K.-R.: Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation 10, 1299–1319 (1998)
18. Shepard, R.N.: The analysis of proximities: Multidimensional scaling with an unknown distance function (parts 1 and 2). Psychometrika 27, 125–140, 219–249 (1962)
19. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science 290(5500), 2319–2323 (2000)
20. van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. Journal of Machine Learning Research 9, 2579–2605 (2008)
21. Young, G., Householder, A.S.: Discussion of a set of points in terms of their mutual distances. Psychometrika 3, 19–22 (1938)