SELECTION OF THE OPTIMAL PARAMETER VALUE FOR THE LOCALLY LINEAR EMBEDDING ALGORITHM

Olga Kouropteva, Oleg Okun and Matti Pietikäinen

Machine Vision Group, Infotech Oulu and Department of Electrical and Information Engineering
P.O.Box 4500, FIN-90014 University of Oulu, FINLAND
email: {kouropte,oleg,mkp}@ee.oulu.fi

ABSTRACT

The locally linear embedding (LLE) algorithm has recently emerged as a promising technique for nonlinear dimensionality reduction of high-dimensional data. One of its advantages over many similar methods is that only one parameter has to be defined, but no guidance was yet given how to choose it. We propose a hierarchical method for automatic selection of an optimal parameter value. Our approach is experimentally verified on two large data sets of real-world images and applied to visualization of multidimensional data.

1. INTRODUCTION

Dimensionality reduction is a useful operation for data clustering and pattern recognition. Because high-dimensional data can bear a lot of redundancies and correlations hiding important relationships, the purpose of this operation is to eliminate the redundancies and to lower the amount of data to be processed.

There are many methods for dimensionality reduction, but we will only concentrate on nonlinear methods because they analyze a more complex case (it is necessary to unfold a nonlinear manifold) than their linear counterparts. A key purpose of such methods is to preserve distances when mapping data to a low-dimensional space, that is, the points close in the input space must be also close in the output space.

Methods meeting this requirement have been recently proposed [1, 2, 3]. While Isomap [1] tries to preserve global geometry of the manifold, the other two methods, namely LLE [2] and Laplacian Eigenmap [3], aim at preserving local geometry. Since both LLE and Laplacian Eigenmap are based on similar principles, we concentrate on LLE without loss of generality.

Because LLE is a new method, it is not well studied, yet, and its real applications are still rare [4]. Among its advantages are 1) only one parameter to be predefined and 2) optimization avoiding the problem with local minima. In this article, we propose an automatic method for selecting of the optimal parameter value of LLE and apply LLE to multidimensional data visualization.

2. LLE ALGORITHM

As an input, the LLE algorithm requires N points X_i, $X_i \in R^D$, $i \in [1, N]$. As an output, it gives N points Y_i, $Y_i \in R^d$, $i \in [1, N]$, where $d \ll D$.

The algorithm consists of three steps:

Step 1. For each X_i find its K nearest neighbors $X_{i_1}, \ldots, X_{i_K}$.

Step 2. Measure reconstruction error resulting from the approximation of each X_i by its nearest neighbors and compute reconstruction weights minimizing this error.

Step 3. Compute low-dimensional embeddings best preserving the local geometry represented by the reconstruction weights.

In Step 1 the Euclidean distances are used in order to determine a neighborhood around each X_i, though other definitions of "closeness" are possible as well.

Step 2 assumes that the manifold is well-sampled, i.e., there are enough data, each data point and its nearest neighbors lie on or close to a *locally linear* patch of the manifold. Hence, we can approximate X_i by a linear combination of its neighbors. This is equivalent to approximating the nonlinear manifold in the vicinity of X_i by the linear hyperplane passing through $X_{i_1}, \ldots, X_{i_K}$. To do so, we need to minimize the reconstruction error ε:

$$\varepsilon = \sum_{i=1}^{N} \parallel X_i - \sum_{j=1}^{N} W_{ij} X_{i_j} \parallel^2 \qquad (1)$$

subject to two constraints: $\sum_{j=1}^{N} W_{ij} = 1$ and $W_{ij} = 0$ for points which are not neighbors of X_i.

Eq. 1 has a closed-form solution (see [2]) which determines the optimal (for reconstruction) weights W_{ij}, where each W_{ij} stands for the contribution of the jth point to the ith reconstruction.

In Step 3 the low-dimensional embeddings are found which best preserve high-dimensional neighborhood geometry represented by the weights W_{ij}. That is, the weights are fixed and we need to minimize the following cost function:

$$\Phi = \sum_{i=1}^{N} \parallel Y_i - \sum_{j=1}^{N} W_{ij} Y_j \parallel^2 \qquad (2)$$

subject to two constraints: $\sum_{i=1}^{N} Y_i = 0$, $\frac{1}{N}\sum_{i=1}^{N} Y_i Y_i^T = I$, where I is the dxd identity matrix. The bottom d (nonzero) eigenvectors of the matrix $(I-W)^T(I-W)$ provide the solution of Eq. 2 as shown in [2]. These eigenvectors form rows of the matrix Y.

3. AUTOMATIC SELECTION OF THE OPTIMAL NUMBER OF NEAREST NEIGHBORS

Why does one need to care about this problem? The reason is that a large number of nearest neighbors causes smoothing or eliminating of small-scale structures in the manifold. In contrast, too small neighborhoods can falsely divide the continuous manifold into disjoint sub-manifolds.

In general, there can be different definitions of "optimality". We rely on a quantitative measure introduced below to characterize this term in order to avoid a subjective evaluation often accompanying a human visual check used in many cases.

This measure has to estimate the "quality" of input-output mapping, that is, how well the high-dimensional structure is represented in the embedded space. The residual variance [2] seems to be suitable for this purpose. It is defined as $1 - \rho^2_{D_X D_Y}$, where ρ is the standard linear correlation coefficient, taken over all entries of D_X and D_Y; D_X and D_Y are the matrices of Euclidean distances (between pairs of points) in X and Y (Y was computed in Step 3 above), respectively. The lower the residual variance is, the better high-dimensional data are represented in the embedded space. Hence, the optimal value for K, K_{opt}, can be determined as

$$K_{opt} = \arg\min_{K}(1 - \rho^2_{D_X D_Y}). \qquad (3)$$

3.1. Straightforward method

Having such a measure, a straightforward method to determine K_{opt} is to run LLE with every possible K ($K \in [1, K_{max}]$, where K_{max} is the maximal possible value for K_{opt}) and select K_{opt} according to Eq. 3. Such an approach is, however, computationally demanding.

3.2. Hierarchical method

We propose another method called hierarchical when a set of potential candidates to become K_{opt} is first selected, however, without proceeding through all steps of LLE, followed by computing the residual variance for each candidate and picking that candidate for which this measure is minimal. As a result, the most time-consuming operation of LLE - eigenvector computation - is carried out only few times compared to the straightforward method. The essence of the hierarchical method consists of the following.

Given K, Eq. 1 defines the reconstruction error ε resulted from the approximation of a small part of the nonlinear manifold by the linear hyperplane. The smaller ε, the better the approximation. When varying K, ε can be considered as a function of K (though, of course, it is also a function of W since the weights alter as K changes). As a result, K_{opt} can correspond to the smallest value of $\varepsilon(K)$.

However, our experiments demonstrate that $\varepsilon(K)$ has more than one minimum (see Figs. 2 and 5) by leading to a set S of potential candidates $K_1^{'}, \ldots, K_{N_S}^{'}$ for K_{opt}. It implies that the residual variance must be computed for each $K_i^{'} \in S$, i = 1,...,N_S.

In summary, the hierarchical method for finding K_{opt} can be described as follows:

- Select K_{max} - the maximal possible value of K_{opt}.
- Calculate ε for each K, $K \in [1, K_{max}]$ according to Eq. 1.
- Find all minima of $\varepsilon(K)$ and corresponding K's which compose the set S of initial candidates.
- For each $K \in S$, run LLE and compute the residual variance.
- Select K_{opt} according to Eq. 3.

4. EXPERIMENTAL RESULTS

We experimented with two data sets of real-world images. The first data set was composed of face images of one person with different (but smoothly varying) head orientations. The second data set contained wood surface images from the database of the University of Oulu [5]. It included examples of both clean and defective surfaces.

The purpose of experiments was to evaluate two methods presented in Section 3 for automatic selection of K_{opt} when visualizing high-dimensional data in a 2-D space. The LLE code implemented in MATLAB running under Windows was utilized [6].

4.1. Experiments with face images

The face data set comprised of 500 120x100 pixels grayscale images. Each image was first transformed into a column vector (D=12000). All vectors were then concatenated to form the input matrix X so that the original space was composed of vectors of intensity values. Typical samples from this data set are shown in Fig. 1.

Fig. 2 shows a plot of $\varepsilon(K)$ for K ranging from 1 to 50. There are several minima of this function with K_{opt} equal to 22 according to the hierarchical method. The existence of a global minimum for $K < 5$ is caused by the fact that for many points their first few neighbors are all indeed close to them and adding a new neighbor decreases the reconstruction error every time. However, as K grows, this error begins to alternate (it rises and falls), because the Euclidean distance becomes an unreliable indicator for proximity.

The embedded space formed by the two bottom non-zero eigenvectors is presented in Fig. 3 together with some representative samples and their projections in this space. The LLE algorithm preserved neighborhoods as expected: nearby samples remain close after embedding. Smooth changes in samples are clearly visible when examing images in Fig. 3. It is possible to observe a curve when traversing from the left upwards and then to the right downwards. This corresponds to the head movement from the left to the right.

K_{opt} found with the straightforward method was 23 and the embedded space looked much the same as that in Fig. 3. This result demonstrates that the hierarchical method for finding K_{opt} is quite accurate.

Figure 1: Samples of face images used in experiments.

4.2. Experiments with wood images

The wood surface data set consisted of 1,900 32x32 pixels RGB images. For each image, values from different color channels were concatenated into one vector as done in [7] (D=3072). The matrix X was formed by such vectors similarly as in case of face images. Fig. 4 shows several examples from this data set displayed as grayscale images.

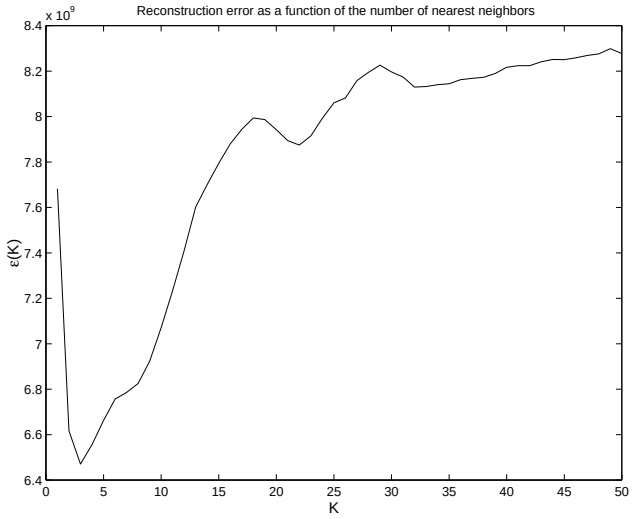

Figure 2: Reconstruction error for K from 1 to 50 for face images.

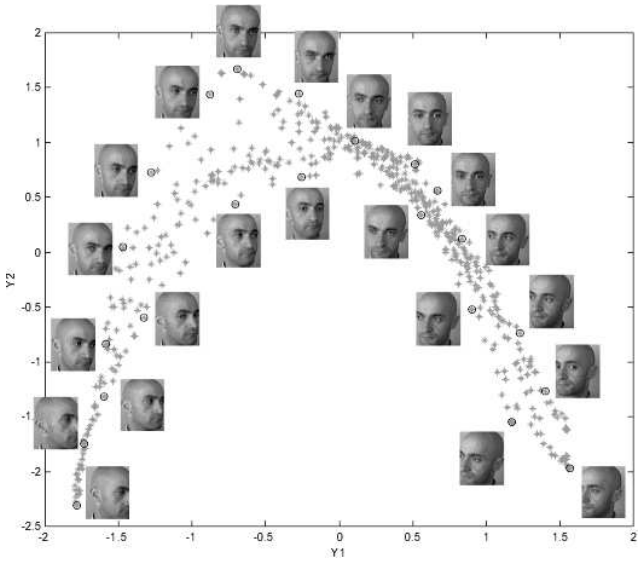

Figure 3: 2-D embedded space of face images. Circles indicate places where the images nearby are mapped into by LLE.

The reconstruction error was computed for $K \in$ [1,50] (see Fig. 5). There were more minima than in case of face data, which can be attributed to a larger data variability, though the plot of ε vs. K looks quite similar to that in Fig. 2. The hierarchical method was nevertheless able to select K_{opt} as 11, which led to the embedded space shown in Fig. 6 and spanned by the two bottom non-zero eigenvectors. One can again observe a quite smooth transition from

sample to sample by moving along a curve in this space.

The straightforward method for computing K_{opt} was applied and it gave 11, which coincides with the value obtained with the hierarchical method. This fact again confirms the accuracy of the hierarchical method.

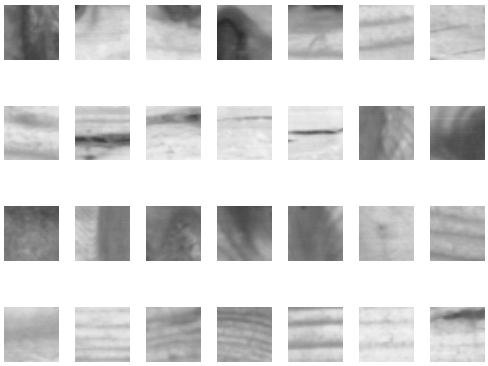

Figure 4: Samples of wood surface images used in experiments.

4.3. Estimation of the computational complexity

The cost of individual steps for the hierarchical method:

- Finding nearest neighbors - $O(DN^2)$.
- Computing reconstruction weights - $O(N_t DNK^3)$, where $N_S = card(S)$, $N_t = K_{max} + N_S$.
- Computing bottom eigenvectors - $O(N_S dN^2)$.

The cost of steps for the straightforward method:

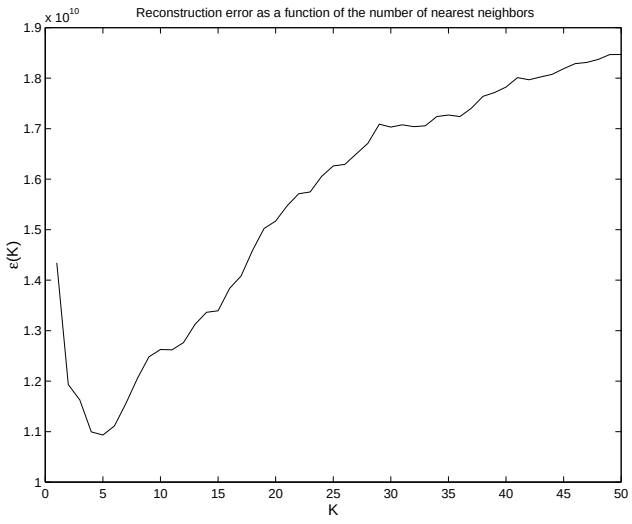

Figure 5: Reconstruction error for K from 1 to 50 for wood surface images.

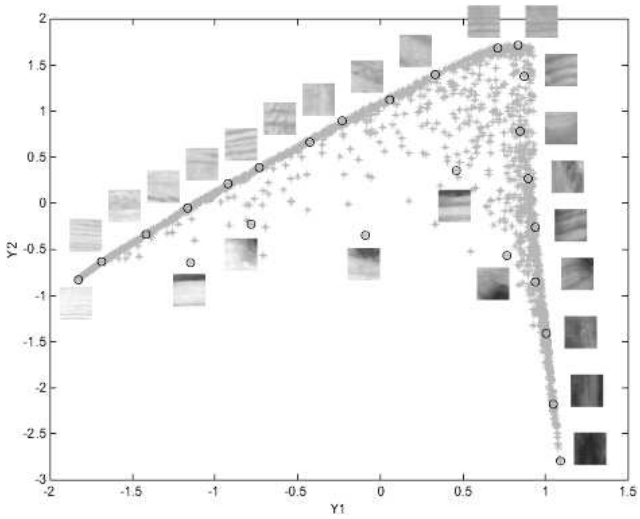

Figure 6: 2-D embedded space of wood surface images. Circles indicate places where the images nearby are mapped into by LLE.

- Finding nearest neighbors - $O(DN^2)$.
- Computing reconstruction weights-$O(K_{max} DNK^3)$.
- Computing bottom eigenvectors - $O(K_{max} dN^2)$.

By comparing the costs for the two methods, one can ask whether one of them will outperform another in terms of time. The answer indeed depends on values of D and N.

When $D \gg N$ (few points of high dimension), the hierarchical method will hardly outperform the straightforward one. The reason is that the eigenvector computation operates on an NxN matrix and for small N's (up to 500 in our experiments) and with unlimited memory space the time spent on this operation is relatevely small so that the cost of the last two steps in the straightforward method is balanced by the cost of the similar steps in the hierarchical method if D is large (say, 10,000).

The situation is changed when D is less than or comparable to N. In this case, the cost of computing the reconstruction weights is increased for the hierarchical method, but this increase is not dramatic since $N_S \ll K_{max}$ in practice. Therefore the cost growth attained for the hierarchical method is compensated by far less expensive (than in the straightforward method) eigenvector computation. As a result, we obtain the total decreased cost. A particular gain is problem-dependent. For example, for wood data it was more than four-fold.

The plot in Fig. 7, displaying processing time in seconds versus N, supports our arguments. To generate it, we ran both methods for artificial 3-D data, where N varied from 1000 to 4000 and K_{max} was equal to 50.

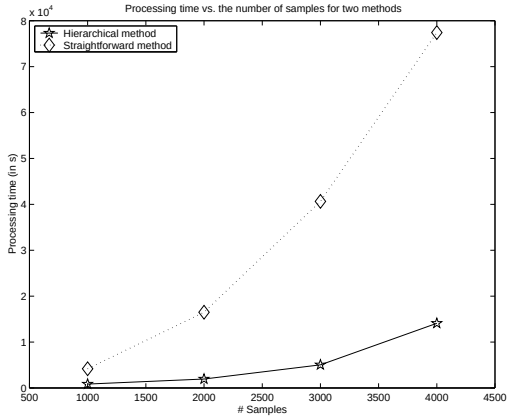

Figure 7: Processing time versus N for the two methods.

5. DISCUSSION

In this paper, we explored the LLE algorithm for nonlinear dimensionality reduction. This algorithm chops a curved surface, as that of a sphere, into small patches, each of which does not involve a lot of curve so that each patch can be considered to be almost flat. These patches are then flattened and "stitched" together in a lower dimensional space in such a way that global high-dimensional nonlinear structures are sufficiently well preserved. It is worthwhile to mention that the idea of dealing with the nonlinearity by means of dividing the whole manifold into local patches that can be approximated by hyperplanes is not completely new (see, for example, [8]). In that paper, VQ was first applied in order to extract patches, followed by PCA applied to each patch separately. This resulted in as many coordinate systems as patches. In contrast, LLE produces one global coordinate system, which greatly simplifies further analysis compared to the approach in [8]. Both approaches however share a common problem: how to define an optimal partition of the manifold into patches. In case of LLE, the partition is determined by the parameter K.

We proposed a hierarchical method for automatic selection of the optimal K and applied it to the visualization of high-dimensional data in a 2-D embedded space. Results obtained indicate that the hierarchical method is feasible and can be used in combination with a suitable cluster detection technique at the next level of analysis. Compared to the straightforward method, the hierarchical one is generally less time-consuming, while yielding very similar results.

One of the drawbacks of LLE is a lack of generalization when new points are to be added, i.e., in this case, we have to rerun LLE on pooled (both old and new) data in order to obtain the embeddings. However, LLE shares this drawback with other popular techniques for nonlinear dimensionality reduction, such as Sammon's mapping [9], for example. It will be therefore interesting to provide LLE with generalization abilities. Although there are few methods capable of generalization (see, for example, [10, 11]), they are still very time-consuming when applied to such high-dimensional data that we used.

6. ACKNOWLEDGMENTS

This work was partly supported by the Academy of Finland and the Infotech Oulu graduate school.

7. REFERENCES

[1] J.B. Tenenbaum, V. de Silva, and J.C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[2] S.T. Roweis and L.K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[3] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," Tech. Rep. TR-2002-01, Dept. of Computer Science, University of Chicago, 2002.

[4] T.L. Ainsworth and J.S. Lee, "Optimal polarimetric decomposition variables - non-linear dimensionality reduction," in *Proc. of IEEE Int. Geoscience and Remote Sensing Symposium, Sydney, Australia*, 2001, pp. 928–930.

[5] ftp://ftp.ee.oulu.fi/pub/tklab/.

[6] http://www.cs.toronto.edu/ roweis/lle/code.html.

[7] O. Kouropteva, "Unsupervised learning with locally linear embedding algorithm: an experimental study," M.S. thesis, University of Joensuu, 2001.

[8] N. Kambhatla and T.K. Leen, "Fast non-linear dimension reduction," in *Proc. of IEEE Int. Conf. on Neural Networks, Nagoya, Japan*, 1993, pp. 1213–1218.

[9] J.W. Sammon Jr., "A nonlinear mapping for data structure analysis," *IEEE Trans. on Computers*, vol. 18, no. 5, pp. 401–409, 1969.

[10] J. Mao and A.K. Jain, "Artificial neural networks for feature extraction and multivariate data projection," *IEEE Trans. on Neural Networks*, vol. 6, no. 2, pp. 296–317, 1995.

[11] N. Pal and V.K. Eluri, "Two efficient connectionist schemes for structure preserving dimensionality reduction," *IEEE Trans. on Neural Networks*, vol. 9, no. 6, pp. 1142–1154, 1998.