ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 185-192

Curvilinear Distance Analysis *versus* Isomap

John Aldo Lee*, Amaury Lendasse, Michel Verleysen†

Université catholique de Louvain
Place du Levant, 3, B-1348 Louvain-la-Neuve, Belgium
{lee,verleysen}@dice.ucl.ac.be, lendasse@auto.ucl.ac.be

Abstract. Dimension reduction techniques are widely used for the analysis and visualization of complex sets of data. This paper compares two nonlinear projection methods: Isomap and Curvilinear Distance Analysis. Contrarily to the traditional linear PCA, these methods work like multidimensional scaling, by reproducing in the projection space the pairwise distances measured in the data space. They differ from the classical linear MDS by the metrics they use and by the way they build the mapping (algebraic or neural). While Isomap relies directly on the traditional MDS, CDA is based on a nonlinear variant of MDS, called CCA (Curvilinear Component Analysis). Although Isomap and CDA share the same metrics, the comparison highlights their respective strengths and weaknesses.

1 Introduction

Data coming from the real world are often difficult to understand because of their high dimensionality. Several dimension reduction techniques address this issue and allow the user to better analyze or visualize complex data sets. These techniques may be distinguished into two classes. In the first one are linear methods like the Principal Component Analysis (PCA, [8]) or the original metric multidimensional scaling (MDS, [14]). In the second class are nonlinear algorithms like Kohonen's Self-Organizing Map (SOM, [9, 10]) or nonlinear variants of the MDS. Contrarily to the linear PCA, the last ones do not use a criterion based on variance preservation. Instead, they try to reproduce in the projection space the pairwise distances measured in the data space. This paper compares two of these nonlinear projection methods that derive more or less directly from the MDS. The first one is called Isomap [13] and is described in Section 2. Isomap differs from the linear MDS by the innovative metrics [2] used to measure the pairwise distances in the data. The second method is

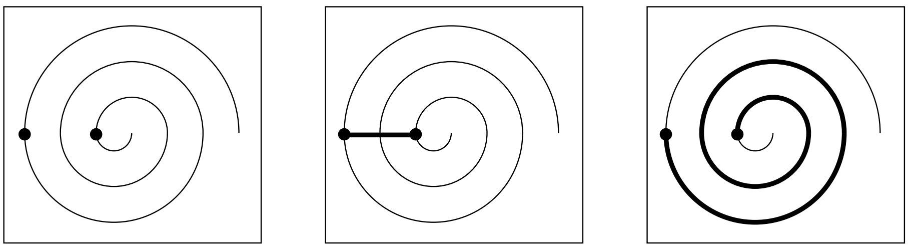
*This work was realized with the support of the 'Ministère de la Région wallonne', under the 'Programme de Formation et d'Impulsion à la Recherche Scientifique et Technologique'.
†M.V. works as a senior research associate of the Belgian FNRS.

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 185-192

called Curvilinear Distances Analysis (CDA, [11], see Section 3) and shares the same metrics as Isomap. Both methods have been developed simultaneously and independently. The comparison (Section 4) highlights the theoretical and practical differences brought by their individual use of their common metrics. Section 5 gives some details about interpolation, which is a specific feature of CDA. Finally, section 6 draws the general conclusions.

2 Isomap

The basic idea behind Isomap consists in overcoming the limitations of the traditional metric MDS, which is linear, by replacing the Euclidean distance by another metrics. Indeed, the MDS encounters difficulties when projecting nonlinear structures like the spiral illustrated in Fig. 1a. Actually, the spiral is embedded in a two-dimensional space, but clearly its intrinsic dimension [5, 6] does not exceed one: only one parameter suffices to describe the spiral. Unfortunately, the projection from two dimensions to only one dimension is not easy because the spiral needs to be unrolled onto a straight line. This unfolding is difficult for MDS because the pairwise Euclidean distances after projection are much larger than in the embedding space: they cannot go through shortcuts like in Fig. 1b. Instead, they have to be measured like in Fig. 1c, along the spiral.



- a - - b - - c -

Figure 1: (-a-) two points in a spiral, (-b-) the Euclidian distance between the two same points and (-c-) the curvilinear or geodesic distance

This so-called geodesic distance is approximated [2, 13] by Isomap in the following way. First, the neighborhood for each point is calculated. For example, the neighborhood of a point may be the k nearest points or the set of points closer than a radius ϵ (k and ϵ being predetermined constants). Once the neighborhoods are known, a graph is build, by linking all neighboring points. Next, each arc of the graph is labelled with the Euclidean distance between the corresponding linked points. Finally, the geodesic distance between two points is approximated by the sum of the arc lengths along the shortest path linking both points. Practically, the shortest path is computed by Dijkstra's algorithm [4].

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 185-192

From a technical point of view, Isomap processes a d-dimensional set of n data points as follows:

1. randomly select l landmarks points in the n data vectors (this optional step reduces the size of distance matrix and the computational load);
2. compute the k- or ϵ-neighborhoods and link neighboring landmarks;
3. run Dijkstra's algorithm to get the square matrix D (l^2 entries) that contains all pairwise geodesic distances;
4. apply the traditional metric MDS on matrix D, i.e. compute the eigenvectors of D and keep the ones associated with the p largest eigenvalues, giving the coordinates of the landmarks points in the p-dimensional projection space.

Actually, if Isomap had used Euclidean distances, step 4 computed with D would have given the same results as PCA directly applied to the d-dimensional coordinates. However, the use of geodesic distances introduces an implicit nonlinear transform of the coordinates and forbids the use of PCA.

Step 1, unless skipped, implies that Isomap projects only the landmarks: contrarily to CDA, no interpolation procedure is available.

3 Curvilinear Distance Analysis

As already mentioned in Section 1, Isomap and CDA share the same metrics. The only difference relates to the name: curvilinear instead of geodesic distance. On the contrary, Isomap and CDA completely differ in the way they use the measured distances. In Isomap, the distance reproduction from the d-dimensional space to the p-dimensional space is achieved algebraically by the traditional MDS, while CDA works with neural methods. Actually, CDA derives from the Curvilinear Component Analysis (CCA, [3, 7]) and from Sammon's Nonlinear Mapping (NLM, [12]). Those two techniques act by preserving distances, like the MDS, but they proceed with an energy function which is minimized by gradient descent (NLM) or by stochastic gradient descent (CCA). Formally, the error function of CDA is written as:

$$E_{CDA} = \sum_{i=1}^{n} \sum_{j=1}^{n} (\delta^d_{i,j} - d^p_{i,j})^2 F(d^p_{i,j}) \ , \tag{1}$$

where $\delta^d_{i,j}$ and $d^p_{i,j}$ are respectively the curvilinear distance in the data space and the Euclidean distance in the projection space between the i-th and j-th points. The factor $F(d^p_{i,j})$ weighing each term of the error function decreases as its argument grows and gives values between 0 and 1. Actually, this factor allows the algorithm to focus on the preservation of small distances rather than of large ones. Section 4 shows the advantages of factor F.

Five steps implement the whole CDA procedure:

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 185-192

1. apply vector quantization [1] on the raw data (this optional step yields prototypes which are very similar to the landmarks points of Isomap);
2. compute the k- or ϵ-neighborhoods and link neighboring prototypes;
3. run Dijkstra's algorithm to get the square matrix D (l^2 entries) that contains all pairwise geodesic distances;
4. optimize E_{CDA} by stochastic gradient descent, in order to get coordinates for the prototypes in the projection space;
5. run a piecewise linear interpolator to compute the projection of the original data points (this step is unnecessary if step 1 was skipped).

The combination of step 1 and 5 is very useful when projecting large data sets, in order to avoid noise influence and to keep reasonable computation times.

4 Comparison between Isomap and CDA

From a theoretical point of view, two differences distinguish Isomap from CDA: firstly, the way they determine landmark points or prototypes and, secondly, the way they compute low-dimensional coordinates.

The vector quantization, for example with Competitive Learning [1] finds prototypes that are more representative of the data distribution than the randomly selected landmark points. The difference appears clearly on Fig. 2 where the Swiss roll has been projected on a plane with Isomap (Fig. 3 left) and CDA (Fig. 3 right). While CDA detects the uniform distribution of the manifold, Isomap produces strange holes like in a slice of ... Swiss cheese! Fortunately, this issue is easily solved by adding to Isomap a VQ step like in CDA. Moreover, the good properties of VQ would allow Isomap to interpolate the whole data set like it is done in CDA.

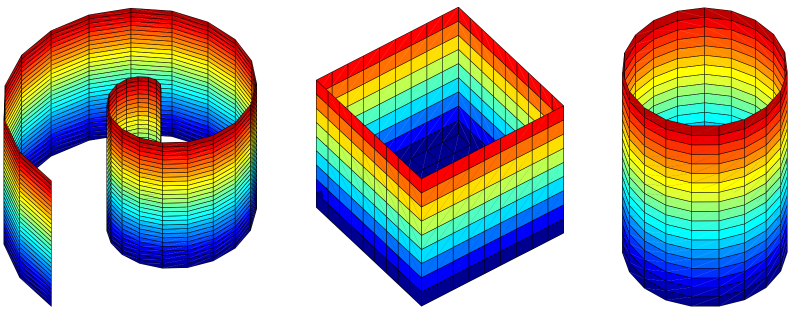


Figure 2: Three typical manifolds: the Swiss roll (left), the open box (middle) and the cylinder (right)

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 185-192

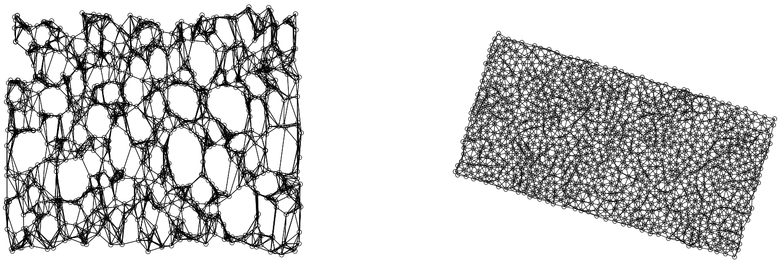


Figure 3: Projection of 20000 samples drawn from the Swiss roll by Isomap (left) and CDA (right), with 1000 landmarks or prototypes

Unlike VQ, the computation of the low-dimensional coordinates is very specific to each algorithm and cannot be easily grafted on each other. At first glance, Isomap shows the advantage of working with a well established procedure, with a strong algebraical foundation. Moreover, the traditional MDS yields simultaneously the coordinates for spaces of dimension 1, 2, 3, etc. However, the matrix of pairwise distances is sometimes too large to get accurate numerical estimations of the eigenvectors. Instead of computing the solution globally, CDA works in a more neural way, by a stochastic gradient descent. Another neural feature of CDA is the factor F: it acts like a neighborhood width and allows the CDA to converge locally. Of course, the decrease of E_{CDA} can get stuck in a local minimum and the convergence requires to parameterize F and the learning rate. Nevertheless, when both are well adjusted (see [11]), CDA find useful projections. For example, CDA can stretch or tear non Euclidean manifold like the open box (Fig. 4) or the cylinder (Fig. 5). The corresponding projections given by Isomap do not unfold the structures as well as CDA.

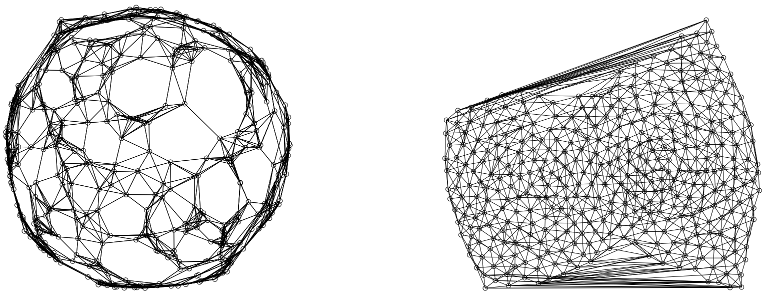


Figure 4: Projection of 20000 samples drawn from the open box by Isomap (left) and CDA (right), with 400 landmarks or prototypes

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 185-192

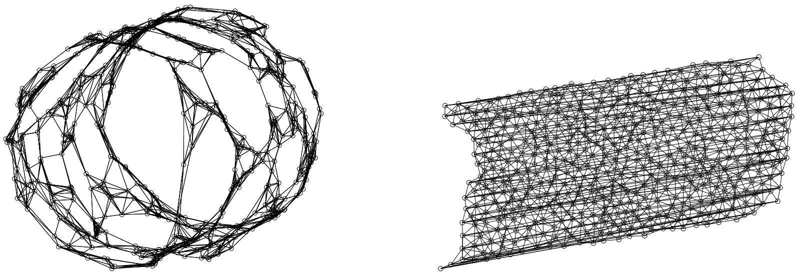


Figure 5: Projection of 10000 samples drawn from the cylinder by Isomap (left) and CDA (right), with 500 landmarks or prototypes

Finally remains the question of speed. Obviously, CDA works more slowly than Isomap: the vector quantization and the gradient descent take more time than a quick random selection and an optimized search of eigenvectors. The problem is still more alarming when CDA has to be run several times to guess the intrinsic dimensionality of a manifold.

5 Interpolation

Contrarily to Isomap, CDA comes with an interpolation procedure. The advantage of interpolation is its ability to generalize: it can project not only the learning set but also new data. Moreover, the new points can either be projected from d to p dimensions or 'deprojected' from p to d dimensions. The last feature is used to estimate the quality of the mapping, by computing a reconstruction error like in PCA. Indeed, for any point x, it suffices to interpolate it back and forth ($d \to p \to d$ dimensions) to get the reconstructed point x' that is next subtracted to x to estimate a quadratic error (for n points):

$$E_{IP} = \frac{1}{n}\sum_{i=1}^{n}(x_i - x_i')^2 \quad . \tag{2}$$

The value of the error brings useful information on the mapping quality. For example, a perfect mapping produces an error close to the variance of the noise present in the data, while bad unfolding, observed as the superposition of several parts of the manifold at the same place in the low-dimensional space, compromises the reconstruction and increases the error.

More precisely, the interpolation works as follows:

1. in the d-dimensional space, find the closest prototype from x;
2. around this prototype, find p other prototypes, among the nearest ones, in order to define a *suitable* p-dimensional local subspace (suitable means that the axis base is not close to degeneration);

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 185-192

3. compute for x its best linear projection z in the local subspace, in the least-square sense;

4. translate the coordinates z in the corresponding subspace defined by the *mapped* prototypes.

The last step gives y, the interpolated coordinates of x in the low-dimensional space. Starting from y, the inverse procedure yields x'.

Figure 6 illustrates the interpolation procedure on the Swiss roll. The learning set (20000 points) is interpolated after building a mapping with only 1000 prototypes. The reconstruction error E_{IP} is equal to 0.064, ten times lower than the quantization error (0.75).

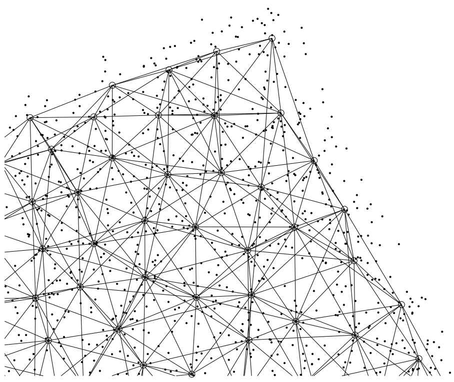


Figure 6: Interpolation procedure of CDA: zoom on a corner of the projected Swiss roll, after interpolation of the 20000 points

6 Conclusion

Isomap and Cuvilinear Distance Analysis are two nonlinear projection algorithms useful to explore data sets. Although they have been developed simultaneously and independently, they share common innovative ideas like an alternative metrics in order to overcome the difficulties encountered when working with more traditional algorithms like metric multidimensional scaling and Sammon's nonlinear mapping. Isomap has the advantage of speed and algebraical solidity. CDA relies on more complicated techniques like vector quantization and stochastic gradient. Though, when these features are well parameterized, they allow CDA to give results which preserves better some characteristics of the projected data sets. Moreover, the combination of VQ and interpolation, respectively before and after mapping, can project large data sets with minimal error and low computational cost.

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 185-192

References

[1] A. Ahalt, A. K. Krishnamurthy, P. Chen, and D. E. Melton. Competitive learning algorithms for vector quantization. *Neural Networks*, 3:277–290, 1990.

[2] M. Bernstein, V. de Silva, J. C. Langford, and J. B. Tenenbaum. Graph approximations to geodesics on embedded manifolds. Technical report, Stanford University, Stanford, December 2000.

[3] P. Demartines and J. Hérault. Curvilinear Component Analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Transaction on Neural Networks*, 8(1):148–154, January 1997.

[4] E. W. Dijkstra. A note on two problems in connection with graphs. *Numerical Mathematics*, (1):269–271, 1959.

[5] K. Fukunaga. Intrinsic dimensionality extraction. In P. R. Krishnaiah and L. N. Kanal, editors, *Classification, Pattern Recognition and Reduction of Dimensionality, Volume 2 of Handbook of Statistics*. North Holland, Amsterdam, 1982.

[6] P. Grassberger and I. Procaccia. Measuring the strangeness of strange attractors. *Physica*, D9:189–208, 1983.

[7] J. Hérault, C. Jaussions-Picaud, and A. Guérin-Dugué. Curvilinear Component Analysis for high dimensional data representation: I. Theoretical aspects and practical use in the presence of noise. In J. Mira and J. V. Sánchez, editors, *Proceedings of IWANN'99*, volume II, pages 635–644. Springer, Alicante (Spain), June 1999.

[8] I. T. Jollife. *Principal Component Analysis*. Springer-Verlag, New York, 1986.

[9] T. Kohonen. Self-organization of topologically correct feature maps. *Biological Cybernetics*, 43:59–69, 1982.

[10] T. Kohonen. *Self-Organizing Maps*. Springer, Heidelberg, 2nd edition, 1995.

[11] J. A. Lee, A. Lendasse, N. Donckers, and M. Verleysen. A Robust Nonlinear Projection Method. In M. Verleysen, editor, *Proceedings of ESANN'2000, 8th European Symposium on Artificial Neural Networks*, pages 13–20. D-Facto public., Bruges (Belgium), April 2000.

[12] W. Sammon, J. A nonlinear mapping algorithm for data structure analysis. *IEEE Transactions on Computers*, CC-18(5):401–409, 1969.

[13] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, December 2000.

[14] W. S. Torgerson. Multidimensional Scaling, I: Theory and Method. *Psychometrika*, 17:401–419, 1952.