Nonlinear dimensionality reduction viewed as information retrieval

Jarkko Venna **Samuel Kaski**
Helsinki Institute for Information Technology
Laboratory of Computer and Information Science
Helsinki University of Technology
P.O. Box 5400, FI-02015 TKK, FINLAND
jarkko.venna@tkk.fi, samuel.kaski@tkk.fi

1 Introduction

Nonlinear dimensionality reduction methods are commonly used for two purposes: (i) as preprocessing methods to reduce the number of input variables or to represent the inputs in terms of more natural variables describing the embedded data manifold, or (ii) for making the data set more understandable, by making the similarity relationships between data points explicit through visualizations. The visualizations are commonly needed in exploratory data analysis, and in interfaces to high-dimensional data. In this abstract we will focus on the latter types of applications and call them *information visualization*, with the understanding that the goal is to visualize neighborhood or proximity relationships within a set of high-dimensional data samples. The introduced methods are expected to be useful for other kinds of dimensionality reduction tasks as well, however.

In information visualization applications, a problem with most of the existing dimensionality reduction methods is that they do not optimize the performance in the task of visualizing similarity relationships. The cost functions measure preservation of pairwise distances for instance, but that is only indirectly related to the goodness of the resulting visualization. Manifold search methods, on the other hand, have been designed to find the "true" manifold which may be higher than two-dimensional, which is the upper limit for visualization in practice. Hence, evaluating goodness of visualizations seems to require usability studies which would be laborious and slow.

We view information visualization from the user perspective, as an information retrieval problem. Assuming that the task of the user is to understand the proximity relationships in the original high-dimensional data set, the task of the visualization algorithm is to construct a display that helps in this task. For a given data point, the user wants to know which other data points are its neighbors, and the visualization should reveal this for all data points, as well as possible.

1.1 Retrieval of Neighbors

The SNE algorithm [4] was originally motivated as a method for placing a set of objects into a low-dimensional space in a way that preserves neighbor identities. Such a projection does not try to preserve pairwise distances as such, as multidimensional scaling (MDS) does, but instead the probabilities of points being neighbors.

We have shown (to be published) that SNE can be seen as an information retrieval algorithm; it optimizes a smoothed form of recall, a traditional goodness measure. To show the connection we need to define neighborhoods as step functions instead of Gaussians as in the original SNE. The user is studying r neighbors in the output space, and her goal is to find a large proportion of the k "true" neighbors, that is, neighbors in the input space. Technically, we assume the closest points to be neighbors with a high probability and the rest with a very low probability.

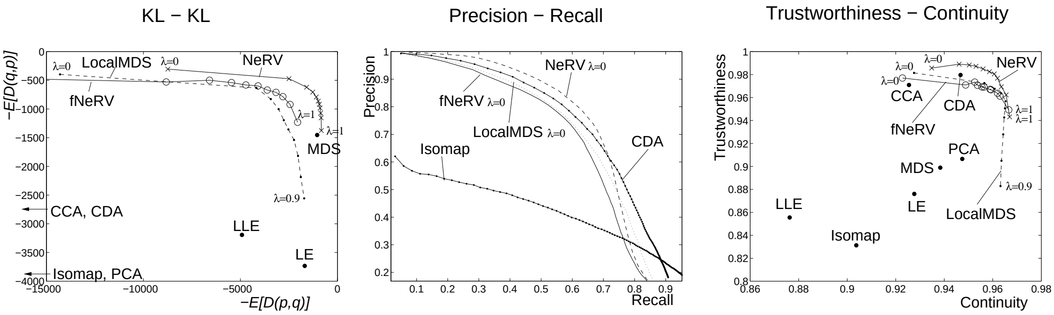

Figure 1: Results on projecting the face dataset [9] to two dimensions. KL-KL curves (left), precision–recall curves (middle) and trustworthiness–continuity [6] curves (right) as a function of λ. Other nonlinear projection methods have been added for reference. The precision–recall curves have been calculated with 20 nearest neighbors in the input space as the set of relevant items; the number of retrieved items (neighbors) is varied from 1 to 100. Only the reference method that achieved the highest precision and the highest recall, and the λ values that had the largest area under the curve are included for clarity. The KL–KL curve and the trustworthiness-continuity curve are calculated using 20 nearest neighbors. On each plot the best performance is in the top right corner. Methods; NeRV, LocalMDS [10], fNeRV: a faster approximative version of NeRV, PCA: Principal Component Analysis [5], MDS: metric Multidimensional Scaling [2], LLE: Locally Linear Embedding [8], LE: Laplacian Eigenmap [1], CCA: Curvilinear Component Analysis [3], CDA: CCA using geodesic distances [7] and Isomap [9].

.

The Kullback-Leibler divergence in the SNE cost function can be divided in four parts, and it is straightforward to check that the part corresponding to misses dominates the cost function and, moreover,

$$D_{KL}(p_i, q_i) \approx \frac{N_{MISS}}{k} C, \tag{1}$$

where C is a constant and N_{MISS} the number of misses. The p_i is the probability distribution of point being a neighbor of i in the input space, and q_i is the corresponding probability distribution in the output space. SNE tries to minimize this cost function, and hence it would maximize recall

It is well known that maximizing recall typically leads to low precision. If we want to maximize precision, we can reverse the direction of the KL divergence in the SNE cost function. It can be shown that minimizing this would correspond to maximizing precision.

In practice it would be best to optimize a compromise. If we assign a relative cost λ to misses and $(1-\lambda)$ to false positives, then the total cost function to be optimized is

$$\begin{aligned} E_{\text{NeRV}} &= \lambda E_i[D(p_i, q_i)] + (1-\lambda) E_i[D(q_i, p_i)] \\ &= \lambda \sum_i \sum_{j \neq i} p_{ij} \log \frac{p_{ij}}{q_{ij}} + (1-\lambda) \sum_i \sum_{j \neq i} q_{ij} \log \frac{q_{ij}}{p_{ij}}. \end{aligned} \tag{2}$$

We call the new method that optimizes (2) *Neighbor Retrieval Visualizer (NeRV)*. In Figure 1 NeRV is shown to outperform other dimensionality reduction methods on three slightly different pairs of performance measures.

References

[1] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In Thomas G. Dietterich, Suzanna Becker, and Zoubin Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 585–591, Cambridge, MA, 2002. MIT Press.

[2] Ingwer Borg and Patrick Groenen. *Modern Multidimensional Scaling*. Springer, New York, 1997.

[3] Pierre Demartines and Jeanny Hérault. Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on Neural Networks*, 8:148–154, 1997.

[4] Geoffrey Hinton and Sam Roweis. Stochastic neighbor embedding. In *Advances in Neural Information Processing Systems 15*, pages 833–840. MIT Press, 2002.

[5] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441,498–520, 1933.

[6] Samuel Kaski, Janne Nikkilä, Merja Oja, Jarkko Venna, Petri Törönen, and Eero Castrén. Trustworthiness and metrics in visualizing similarity of gene expression. *BMC Bioinformatics*, 4:48, 2003.

[7] John Aldo Lee, Amaury Lendasse, and Michel Verleysen. Nonlinear projection with curvilinear distances: Isomap versus curvilinear distance analysis. *Neurocomputing*, 57:49–76, 2004.

[8] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.

[9] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000.

[10] Jarkko Venna and Samuel Kaski. Local multidimensional scaling. *Neural Networks*, 19:889–899, 2006.