NONLINEAR DIMENSIONALITY REDUCTION

Dimensionality Reduction

Recall that the goal of dimensionality reduction is to represent a higher dimensional data set by a lower dimensional data set while preserving as much information as possible. PCA is a linear method, but many data sets possess nonlinear structure.

Kernel PCA

Similar to previous kernel methods, the idea to transform the data by a nonlinear feature map $\underline{\Phi}(x)$ where $\underline{\Phi}$ is chosen such that

$$k(x,x') = \langle \underline{\Phi}(x), \underline{\Phi}(x') \rangle$$

can be computed easily. Then PCA is applied to $\Phi(x_1), \dots, \Phi(x_n)$. PCA can be kernelized, meaning the algorithm can be expressed entirely in terms of k and not Φ. Here is the final algorithm:

Input: $x_1, \dots, x_n$, dimension p

1. Construct kernel matrix $K = [K_{ij}]$, $K_{ij} = k(x_i, x_j)$
2. Form centered kernel matrix

$$\tilde{K} = K - OK - KO + OKO$$

where O is $n \times n$ and has entries equal to $\frac{1}{n}$

3. Compute the eigenvalue decomposition $\tilde{K} = U \Lambda U^T$,

$$U = \begin{bmatrix} u_1 & \dots & u_n \end{bmatrix}, \quad \Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_m, 0, \dots, 0)$$

4. Set $\alpha_j = (\alpha_j^{(1)} \dots \alpha_j^{(n)})^T := \frac{1}{\sqrt{\lambda_j}} u_j \in \mathbb{R}^n$

Output: Dimensionality reduction mapping

$$x \longmapsto y = (y^{(1)}, \dots, y^{(p)})^T \in \mathbb{R}^p$$

where

$$y^{(j)} = \sum_{i=1}^{n} \alpha_j^{(i)} k(x, x_i)$$

where

$$y^{(j)} = \sum_{i=1}^{n} \alpha_j^{(i)} k(x, x_i)$$

KPCA is a good general-purpose method for NLDR.

Isomap

One limitation of KPCA is that it does not try to exploit the intrinsic dimensionality of a data set.

One method that does is called isometric feature mapping, or Isomap.

Isomap is the following procedure:

Input $x_1, \ldots, x_n$

1. Construct a similarity graph, such as a k-nearest neighbor graph
2. Form a dissimilarity matrix $D = [d_{ij}]$ where d_{ij} = length of shortest path connecting x_i and x_j
3. Apply MDS to D to get an embedding $y_1, \ldots, y_n$

$y_1, \ldots, y_n$

Output: $y_1, \ldots, y_n$

d_{ij} is viewed as an approximation to the geodesic distance on the underlying data manifold.

Example: Swiss roll data

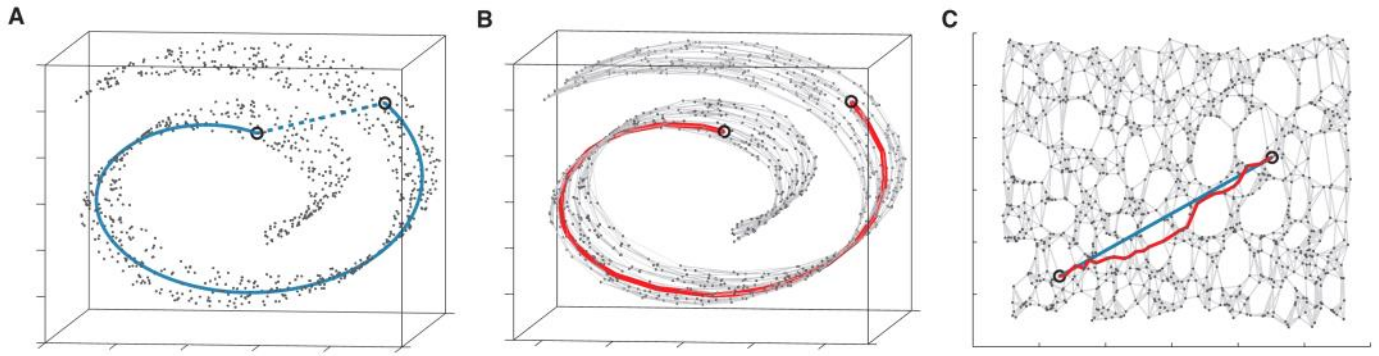

Fig. 3. The "Swiss roll" data set, illustrating how Isomap exploits geodesic paths for nonlinear dimensionality reduction. (**A**) For two arbitrary points (circled) on a nonlinear manifold, their Euclidean distance in the high-dimensional input space (length of dashed line) may not accurately reflect their intrinsic similarity, as measured by geodesic distance along the low-dimensional manifold (length of solid curve). (**B**) The neighborhood graph G constructed in step one of Isomap (with $K = 7$ and $N = 1000$ data points) allows an approximation (red segments) to the true geodesic path to be computed efficiently in step two, as the shortest path in G. (**C**) The two-dimensional embedding recovered by Isomap in step three, which best preserves the shortest path distances in the neighborhood graph (overlaid). Straight lines in the embedding (blue) now represent simpler and cleaner approximations to the true geodesic paths than do the corresponding graph paths (red).

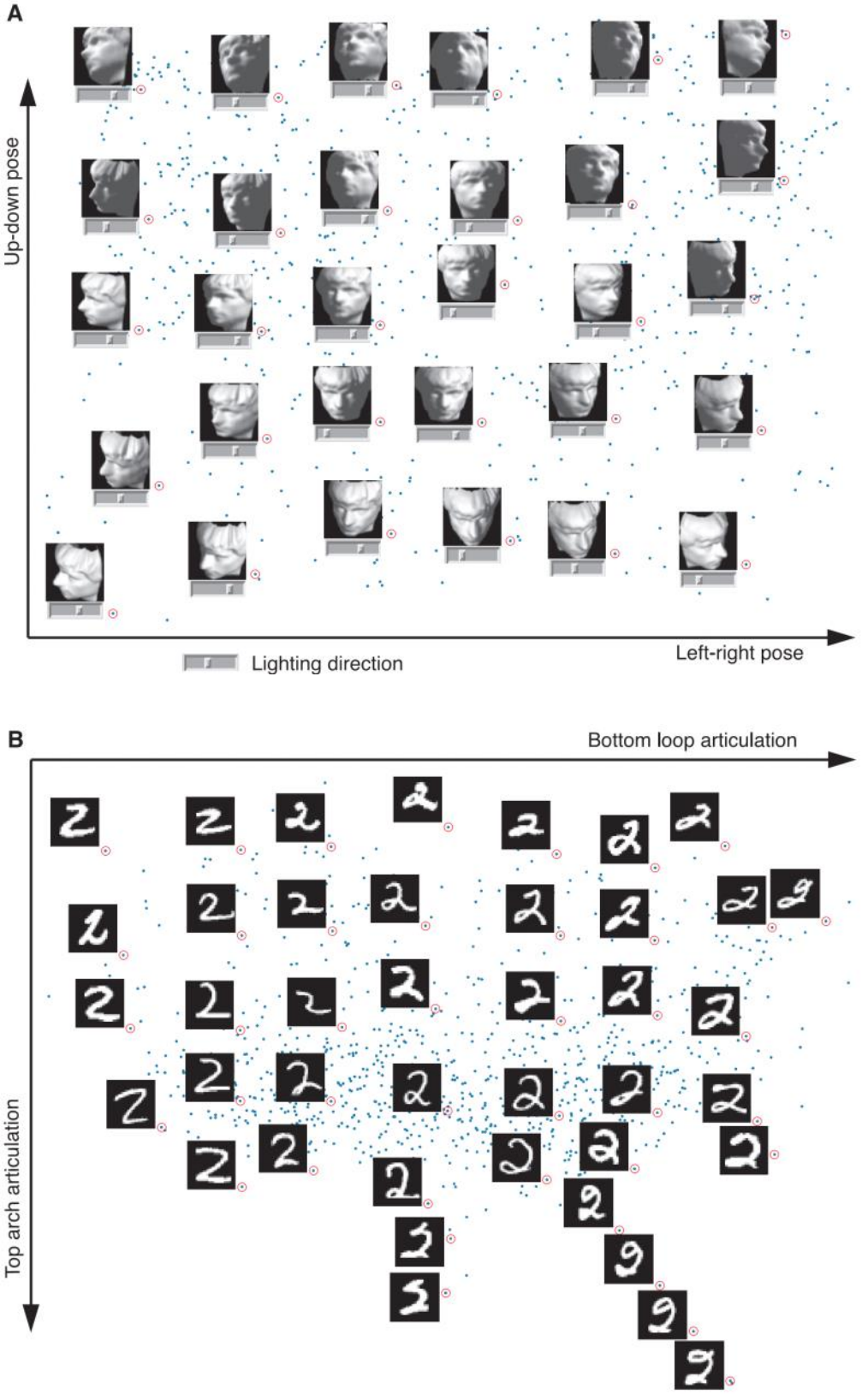

Fig. 1. (**A**) A canonical dimensionality reduction problem from visual perception. The input consists of a sequence of 4096-dimensional vectors, representing the brightness values of 64 pixel by 64 pixel images of a face rendered with different poses and lighting directions. Applied to $N = 698$ raw images, Isomap ($K = 6$) learns a three-dimensional embedding of the data's intrinsic geometric structure. A two-dimensional projection is shown, with a sample of the original input images (red circles) superimposed on all the data points (blue) and horizontal sliders (under the images) representing the third dimension. Each coordinate axis of the embedding correlates highly with one degree of freedom underlying the original data: left-right pose (x axis, $R = 0.99$), up-down pose (y axis, $R = 0.90$), and lighting direction (slider position, $R = 0.92$). The input-space distances $d_X(i,j)$ given to Isomap were Euclidean distances between the 4096-dimensional image vectors. (**B**) Isomap applied to $N = 1000$ handwritten "2"s from the MNIST database (*40*). The two most significant dimensions in the Isomap embedding, shown here, articulate the major features of the "2": bottom loop (x axis) and top arch (y axis). Input-space distances $d_X(i,j)$ were measured by tangent distance, a metric designed to capture the invariances relevant in handwriting recognition (*41*). Here we used ϵ-Isomap (with $\epsilon = 4.2$) because we did not expect a constant dimensionality to hold over the whole data set; consistent with this, Isomap finds several tendrils projecting from the higher dimensional mass of data and representing successive exaggerations of an extra stroke or ornament in the digit.

Reference: Tennenbaum, de Silva, and Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction,"

Science, 290, 2319-2323 (2000).

Laplacian Eigenmaps

Laplacian Eigenmaps (LEM) is the method of dimensionality reduction underlying spectral clustering. It can be derived from the following perspective. Let W be a weighted similarity graph for $x_1, \dots, x_n$. Let $y_1, \dots, y_n \in \mathbb{R}^p$ be the reduced dimensionality versions to be learned. LEM is obtained by minimizing

$$\frac{1}{2} \sum_{i,j} w_{ij} \| y_i - y_j \|^2 = \operatorname{tr}(YLY^T)$$

↖ algebra

where $Y = \begin{bmatrix} y_1 & \cdots & y_n \end{bmatrix} \in \mathbb{R}^{p \times n}$, and L is the unnormalized graph Laplacian. To obtain a unique solution, an energy constraint is added.

- $YY^T = I$

The solution to

$$\begin{aligned} \min_{Y} \quad & \operatorname{tr}(YLY^T) \\ \text{s.t.} \quad & YY^T = I \end{aligned}$$

This is the same problem at the heart of PCA

is

$$Y^T = \left[u_1 \cdots u_p \right]$$

where $L = U \Lambda U^T$. This is the DR method underlying unnormalized spectral clustering.

- $YDY^T = I$

The solution to

$$\min_{Y} \; tr(YLY^T) \quad \text{s.t.} \quad YDY^T = I$$

is

$$Y^T = \left[\tilde{u}_1 \cdots \tilde{u}_p \right]$$

where $\tilde{L} = D^{-1}L = \tilde{U} \tilde{\Lambda} \tilde{U}^T$ is the normalized graph Laplacian. This can be shown by substituting $Z = YD^{\frac{1}{2}}$, reducing the problem to the PCA problem.

The latter method is the one known as LEM. The

above methods actually gives another perspective and justification for spectral clustering.

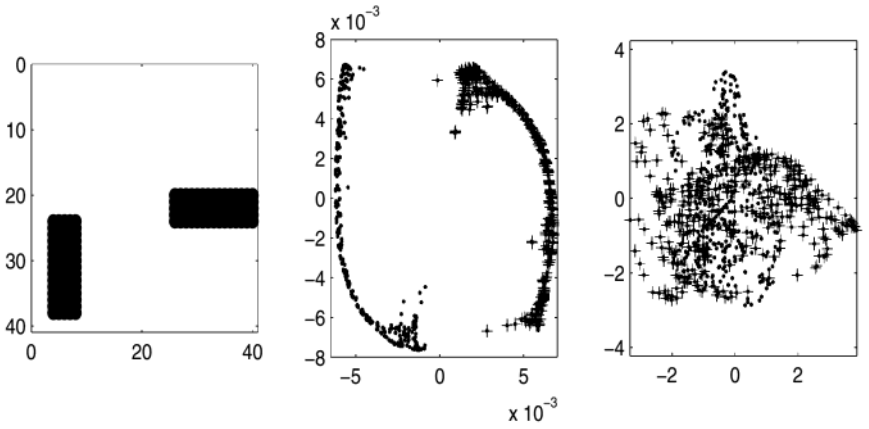

Figure 3: (Left) A horizontal and a vertical bar. (Middle) A two-dimensional representation of the set of all images using the Laplacian eigenmaps. (Right) The result of PCA using the first two principal directions to represent the data. Blue dots correspond to images of vertical bars, and plus signs correspond to images of horizontal bars.

Reference: Belkin and Nyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," Neural Computation 15, 1373-1396 (2003).

Final Thoughts

Isomap emphasizes preservation of global (geodesic) distances, while LEM emphasizes preservation of local information.

Both Isomap and LEM can be viewed (approximately)

as KPCA with a particular kernel.

KPCA has the advantage that the DR mapping readily extends to new feature vectors x. Isomap and LEM do not have a straightforward out-of sample extension. A method that combines LEM and KPCA and has the strengths of both methods (adapting to low-dimensional structure, out-of-sample extension) is Locality Preserving Projections.