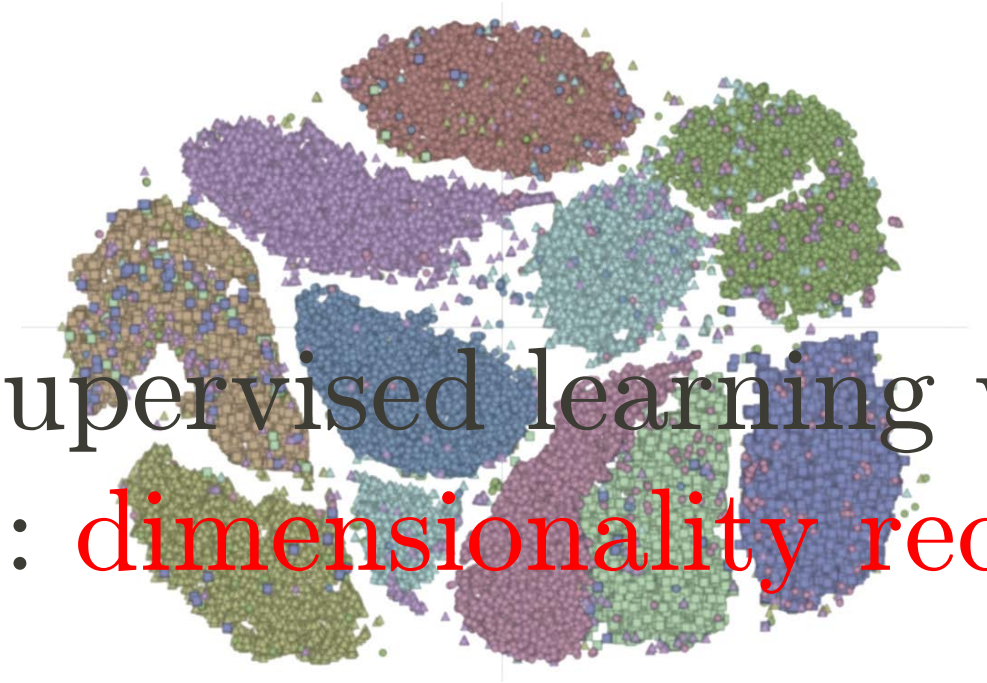


Unsupervised learning with graphs: dimensionality reduction

Andreas Loukas

Signal Processing Laboratory (LTS2), EPFL

Adapted from the slides of Daniel Steinberg, as well as from those of Raykar, Silva and Tenenbaum.
The method comparisons were taken from Nik Melchior.

EPFL

Outline of lecture

- Why dimensionality reduction?
- Linear methods
  - PCA
  - MDS
- Nonlinear methods (with graphs)
  - Isomap
  - Laplacian Eigenmaps
  - LLE
- Comparison

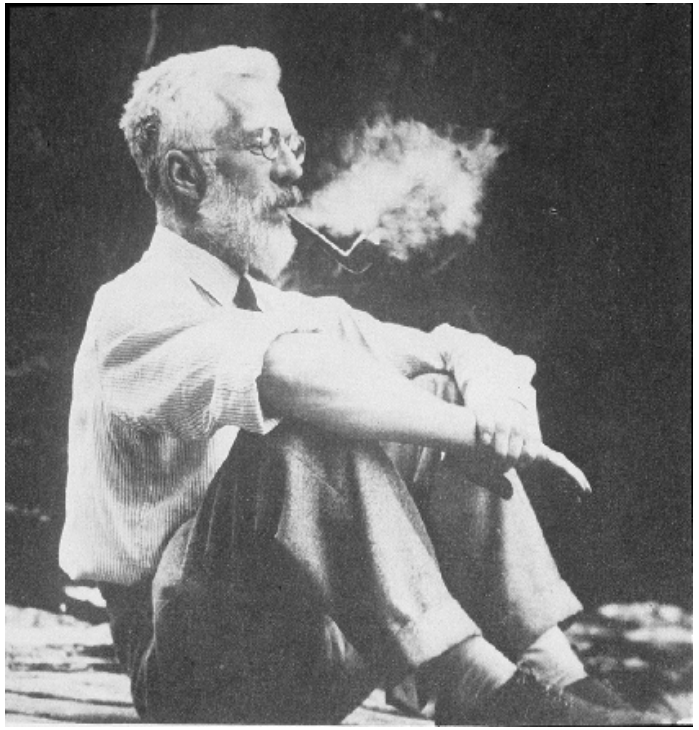
Why dimensionality reduction?

R. A. Fisher

"The objective of statistical methods is the reduction of data.

A quantity of data... is to be replaced by relatively few quantities which shall adequately represent (...) the relevant information contained in the original data.

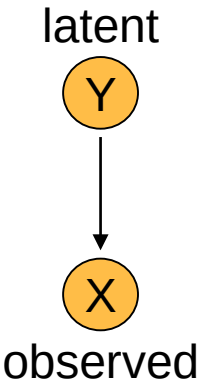
Since the number of independent facts supplied in the data is usually far greater than the number of facts sought, much of the information supplied by an actual sample is irrelevant. It is the object of the statistical process employed in the reduction of data to exclude this irrelevant information, and to isolate the whole of the relevant information contained in the data."



Why dimensionality reduction?

The objective of any dimensionality reduction algorithm:

> Map points $x_1, x_2, \ldots, x_N \in \mathbb{R}^d$ into $y_1, y_2, \ldots, y_N \in \mathbb{R}^k$ such that $k \ll d$ and the mapping preserves relations between points.



- Y is also called an embedding of X.

Benefits

- Discard irrelevant information (discover relevant features).
- Data visualization and exploratory data analysis

Two common approaches:

- Manually select features by some ad-hoc criteria / intuition
- Systematically identify a reduced representation by optimizing some desired criterion

How to store these images?

If we can recognize perceptually meaningful structure

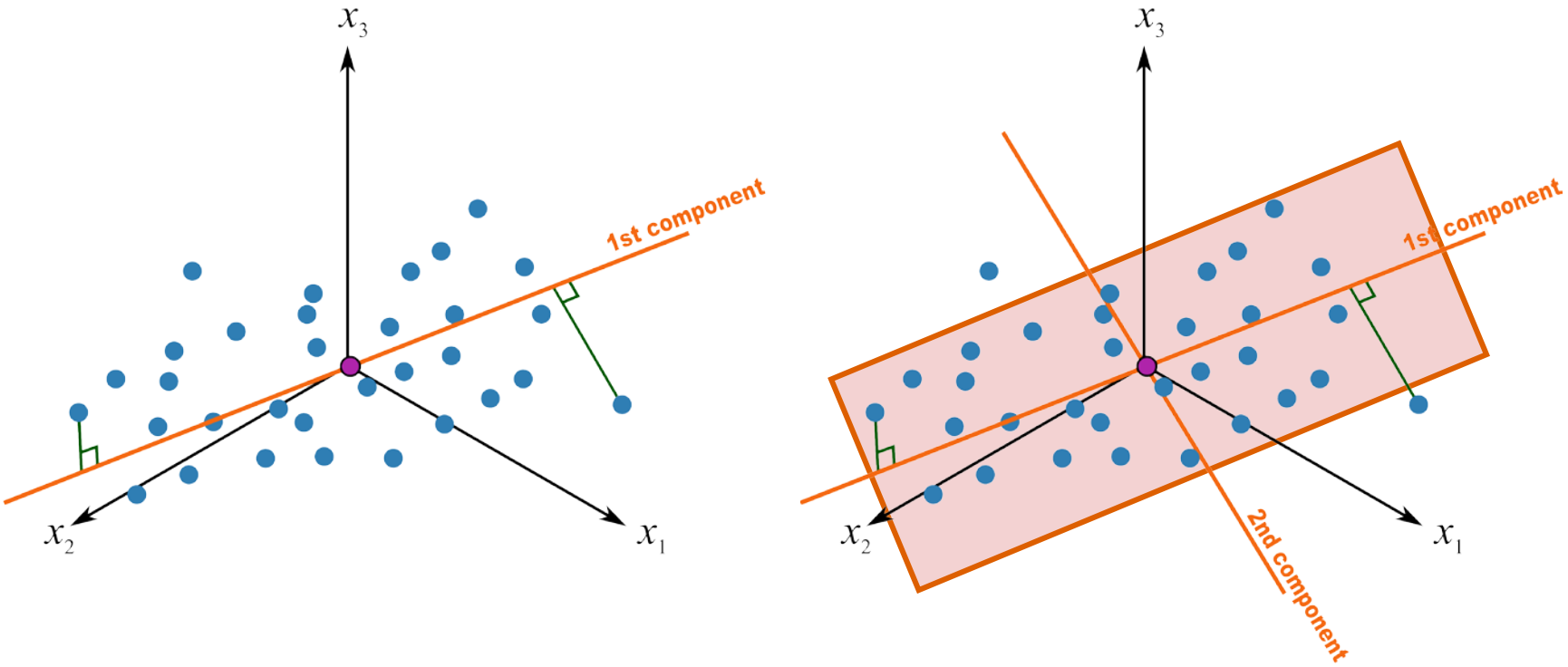
- i.e., geometry of head, position of camera, lighting direction
- we can achieve good compression & interpretability.

Outline of lecture

- Why dimensionality reduction?
- Linear methods
  - PCA
  - MDS
- Nonlinear methods (with graphs)
  - Isomap
  - Laplacian Eigenmaps
  - LLE
- Comparison

Principal component analysis (PCA)

Project data over best linear subspace.



- PC correspond to eigenvectors of the correlation matrix

Reminder: PCA for dim. reduction

For $X \in \mathbb{R}^{N \times d}$, the problem

$$\text{minimize}_Y \|XX^\top - YY^\top\|_F \quad \text{subject to} \quad Y \in \mathbb{R}^{N \times k}$$

has analytic solution in terms of the SVD: $X = U\Sigma V^\top$.

Partition the above as follows:

$$U = \begin{bmatrix} U_1 & U_2 \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}, \quad \text{and} \quad V = \begin{bmatrix} V_1 & V_2 \end{bmatrix}.$$

where Σ_1 is $k \times k$. Then

$$Y^* = XV_1 = U_1\Sigma_1,$$

is such that

$$\|XX^\top - Y^*(Y^*)^\top\|_F = \min_{Y \in \mathbb{R}^{N \times k}} \|XX^\top - YY^\top\|_F.$$

Multi-dimensional scaling (MDS)

Also called Principal Coordinate Analysis

Given a matrix D of distances between points such that $D(i,j) = \text{dist}(x_i, x_j)$, minimize

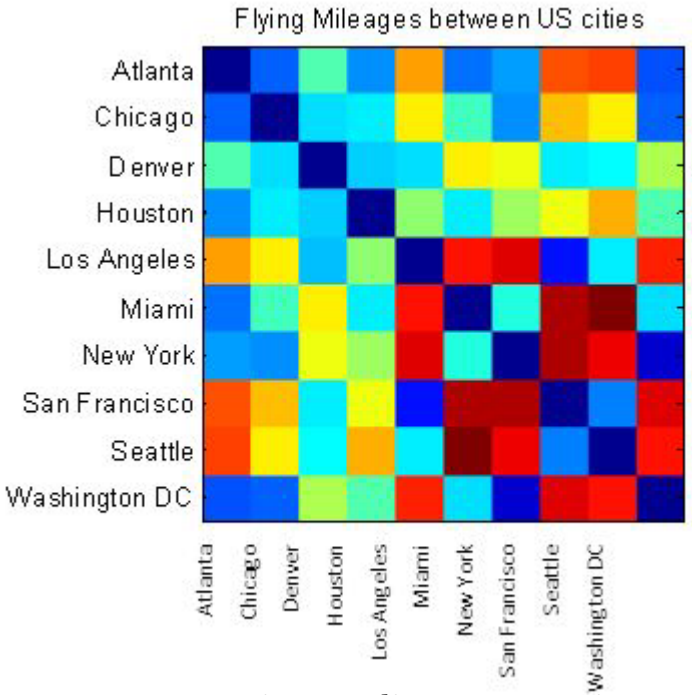
$$Y^* = \underset{y_1, y_2, \ldots, y_N \in \mathbb{R}^k}{\arg\min} \sum_{i,j} (D(i,j) - \text{dist}(y_i, y_j))^2.$$

Classical MDS:

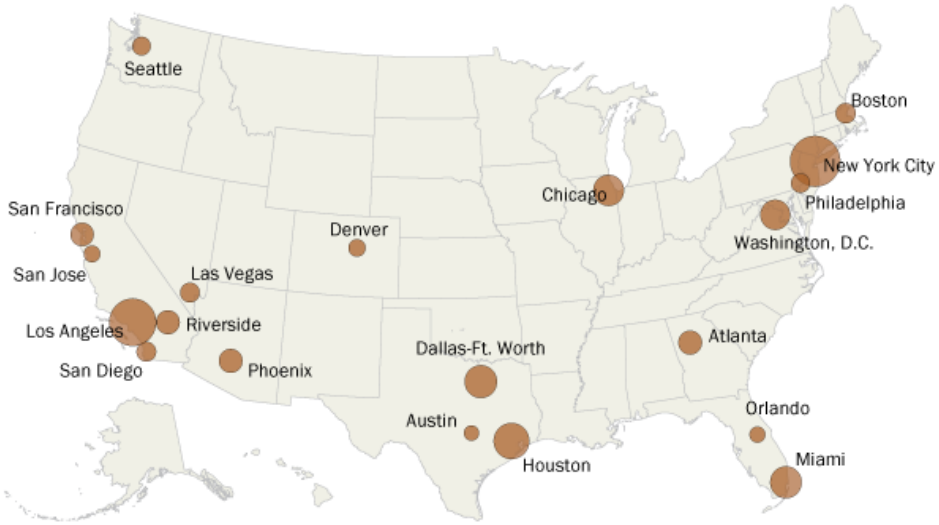
- Use the Euclidean distance

Kruskal and Wish. Multidimensional scaling. 1978

MDS illustrated



input: distance matrix



output: points

Question.

- How many dimensions k are necessary to embed X perfectly?
- What if X contains only noise?

Classic MDS algorithm

- Y can be derived from $B = XX^\top$ (matrix square root).
- B can be computed from D (double centering).

Algorithm (Classic MDS).

1. Let $D^{(2)} = [d_{ij}^2]$
2. Apply double centering: $B = -\frac{1}{2} J D^{(2)} J$ using $J = I - \frac{1}{N} 1 1^\top$.
3. Let U_1 and Σ_1 be the matrices of k singular vectors and singular values of B (akin to PCA)
4. Return $Y = U_1 \Sigma_1^{1/2}$ (akin to PCA).

Wickelmaier, Florian. "An introduction to MDS." Aalborg University, Denmark, 2003

Linear methods

Classical MDS is similar to PCA, but

- MDS uses the distance matrix,
- whereas PCA uses the points themselves.

- PCA properties:
  - equivariant to rotation, re-scaling
  - depends on translation (unless data is centered)

- MDS properties:
  - equivariant to re-scaling
  - invariant to rotation, translation

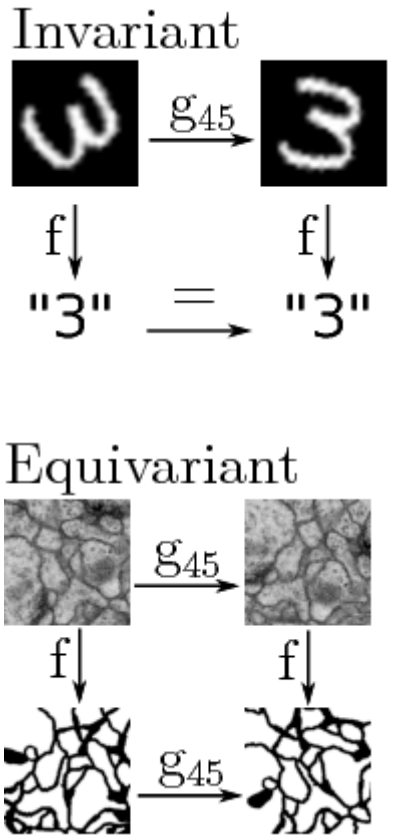
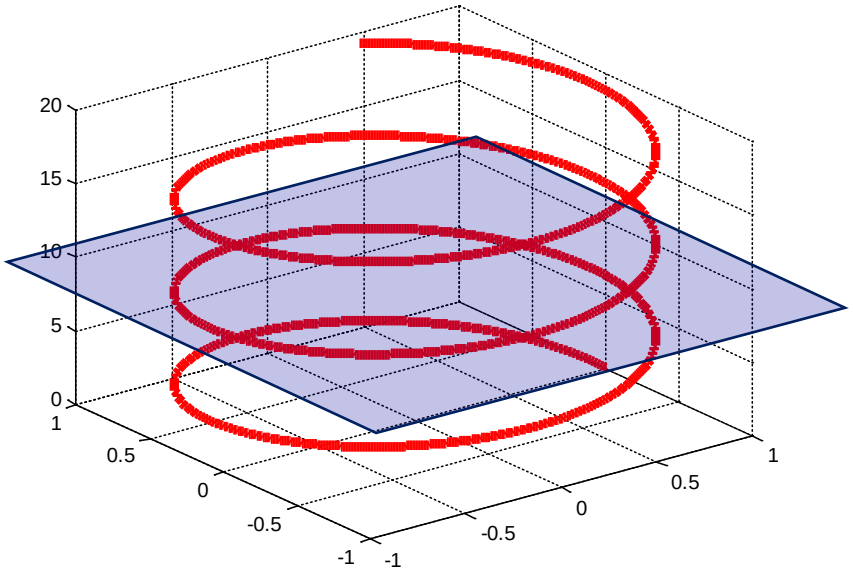


Figure by Marcos, Volpi, Komodakis,and Tuia

Deficiencies of linear methods

- Data may not be best summarized by linear combination of features



PCA/MDS cannot discover 1D structure of a helix.

Outline of lecture

- Why dimensionality reduction?
- Linear methods
  - PCA
  - MDS
- Nonlinear methods (with graphs)
  - Isomap
  - Laplacian Eigenmaps
  - LLE
- Comparison

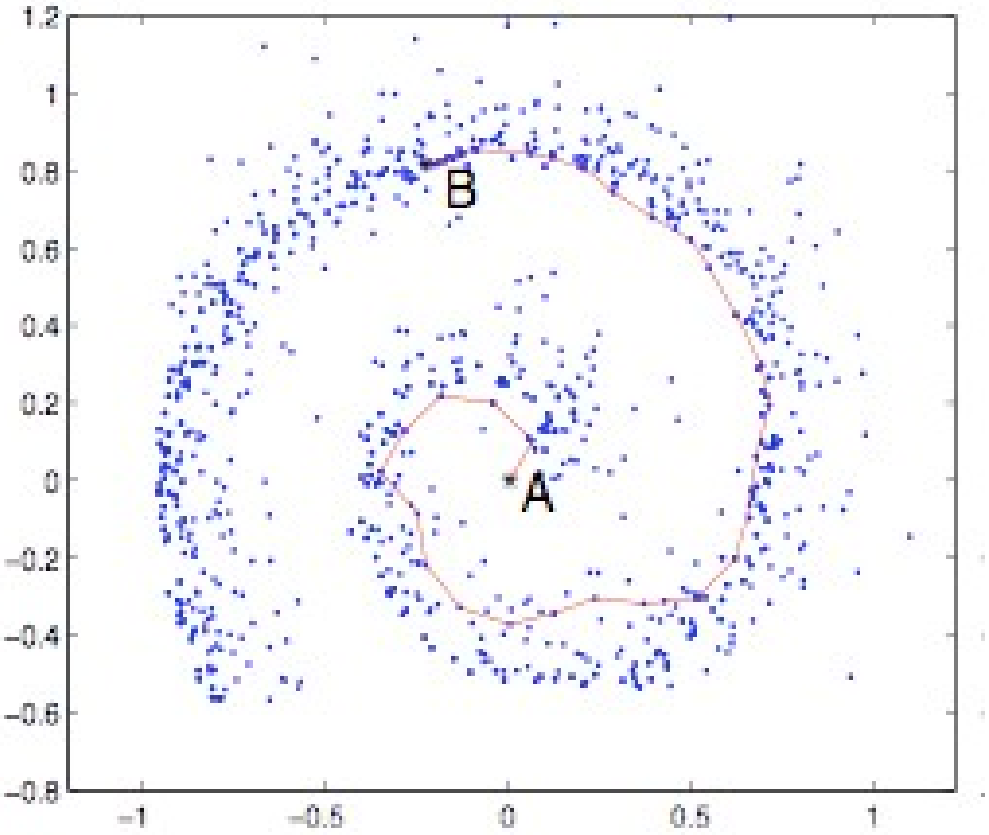
Non-linear dimensionality reduction

Data lying on a low-dimensional manifold within a high-dimensional space.

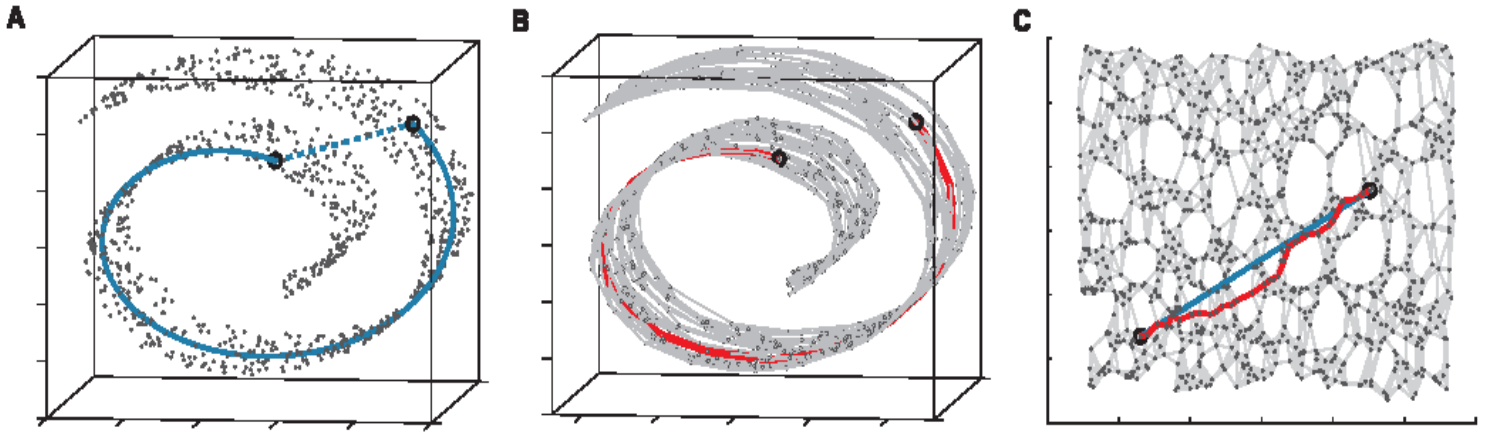
Euclidean distance underestimates geodesics.

Recipe common to non-linear methods

- preserve only local distances



Isomap



Classical MDS – uses Euclidean distance

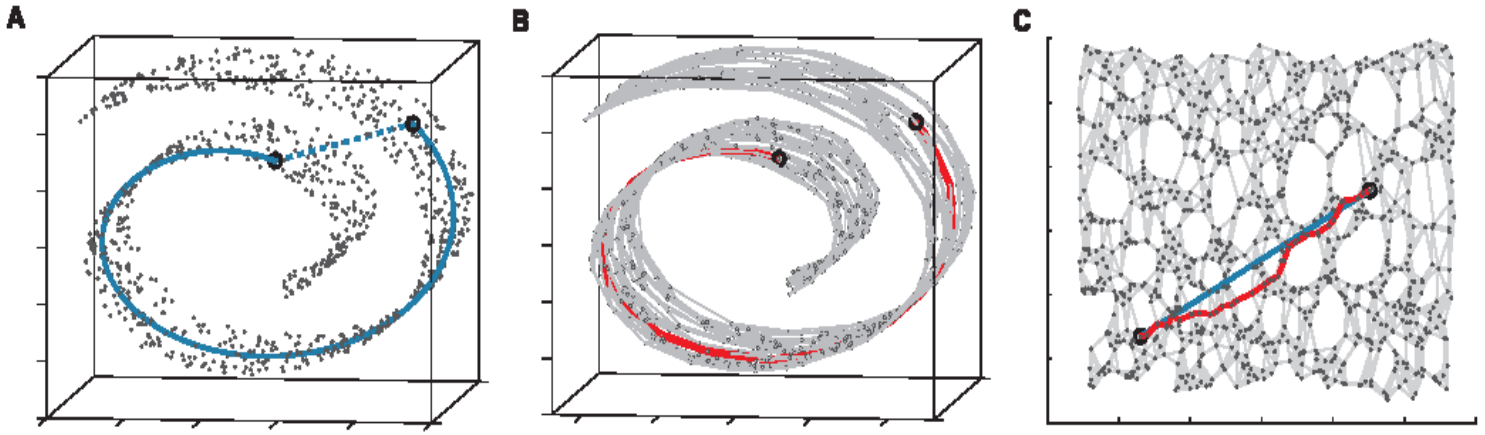
What we really want:

- Distance measurements along manifold (geodesic distance)
- Find low dimensional reconstruction which also has these geodesic distances

Isomap: Classical MDS with approximate geodesic distances.

Tenenbaum, Silva and Langford. "A global geometric framework for nonlinear dimensionality reduction". 2000

Isomap



Algorithm (Isomap).

1. Construct nearest neighbor graph (k-nn, ε-neighborhood graph) setting edge lengths equal to dist(i,j).
2. Compute the shortest path matrix D_G
3. Apply classical MDS on D_G to find k-dimensional embedding

Isomap

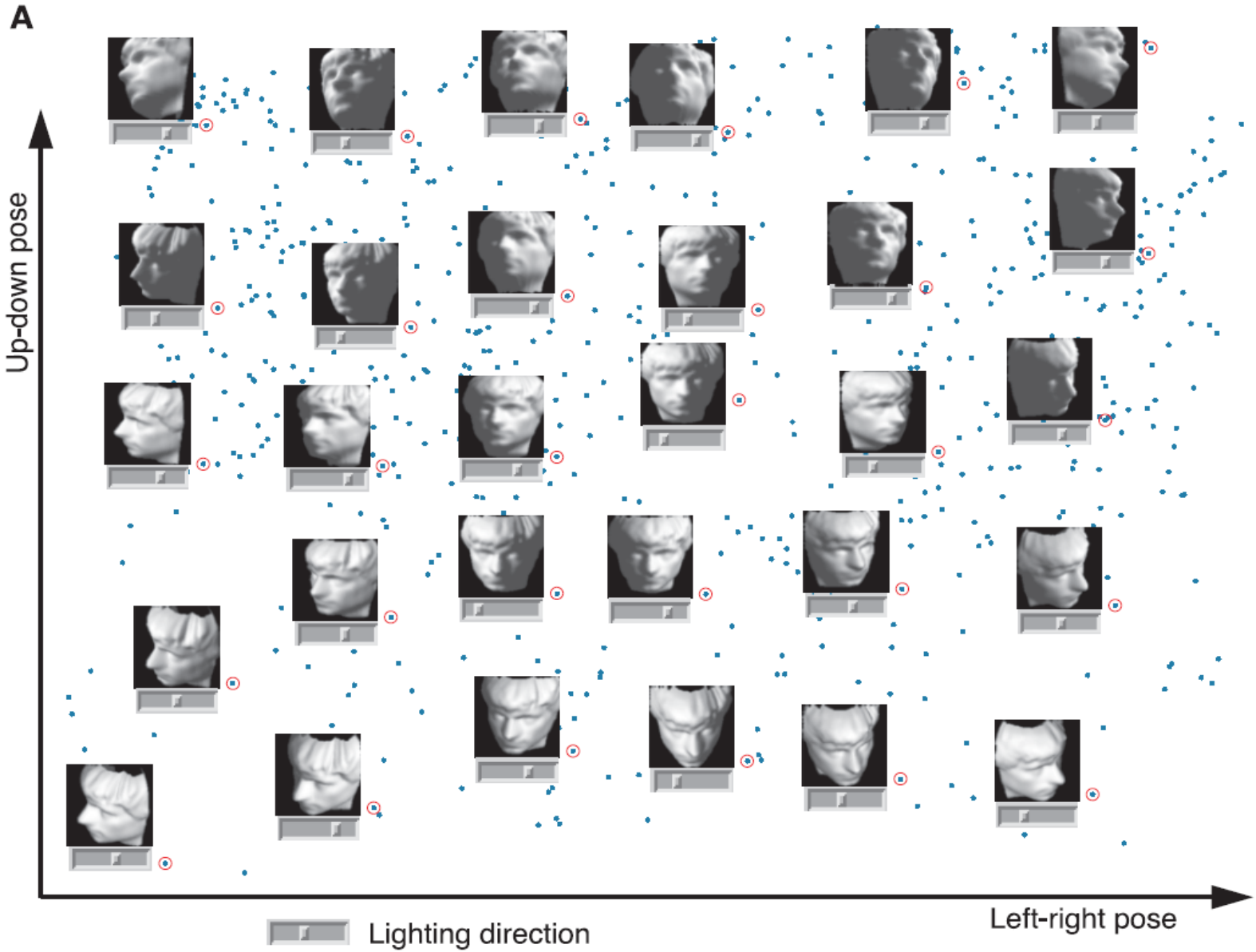
Advantages

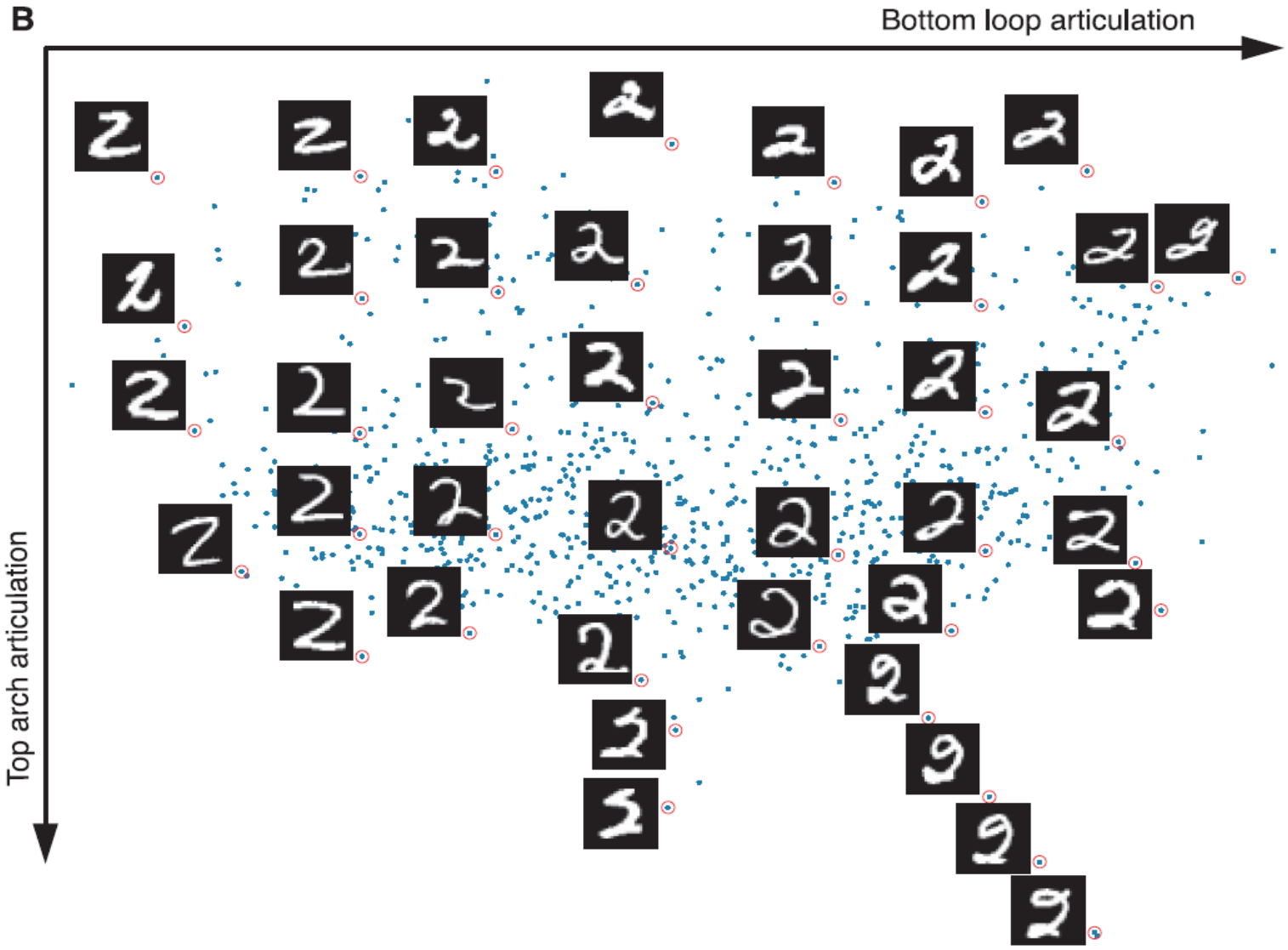
- Non-iterative polynomial algorithm – O(N M), Floyd-Warshall has $O(N^3)$.
- Guarantee of global optimality
  - For intrinsically Euclidean manifolds, a guarantee of asymptotic convergence to the true structure
  - the ability to discover manifolds of arbitrary dimensionality

Disadvantages

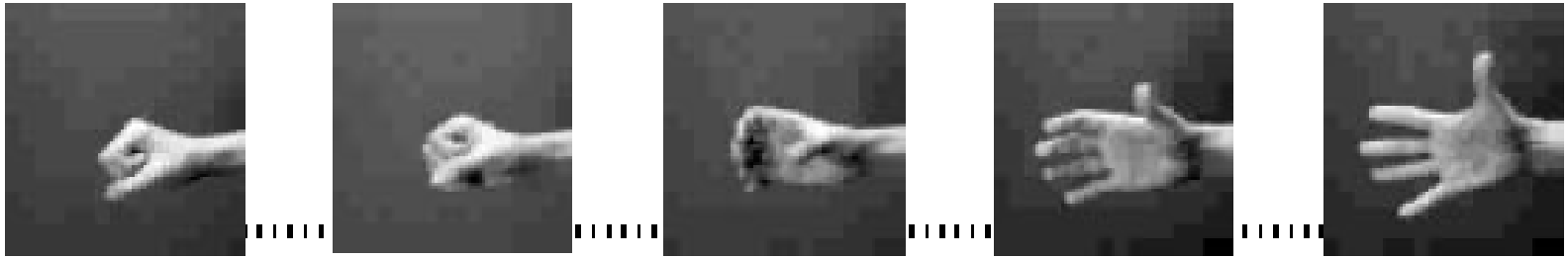
- short-circuit errors from noise or if k is too large.

  "Even a single short-circuit error can alter many entries in the geodesic distance matrix, which in turn can lead to a drastically different (and incorrect) low-dimensional embedding." -- Wikipedia

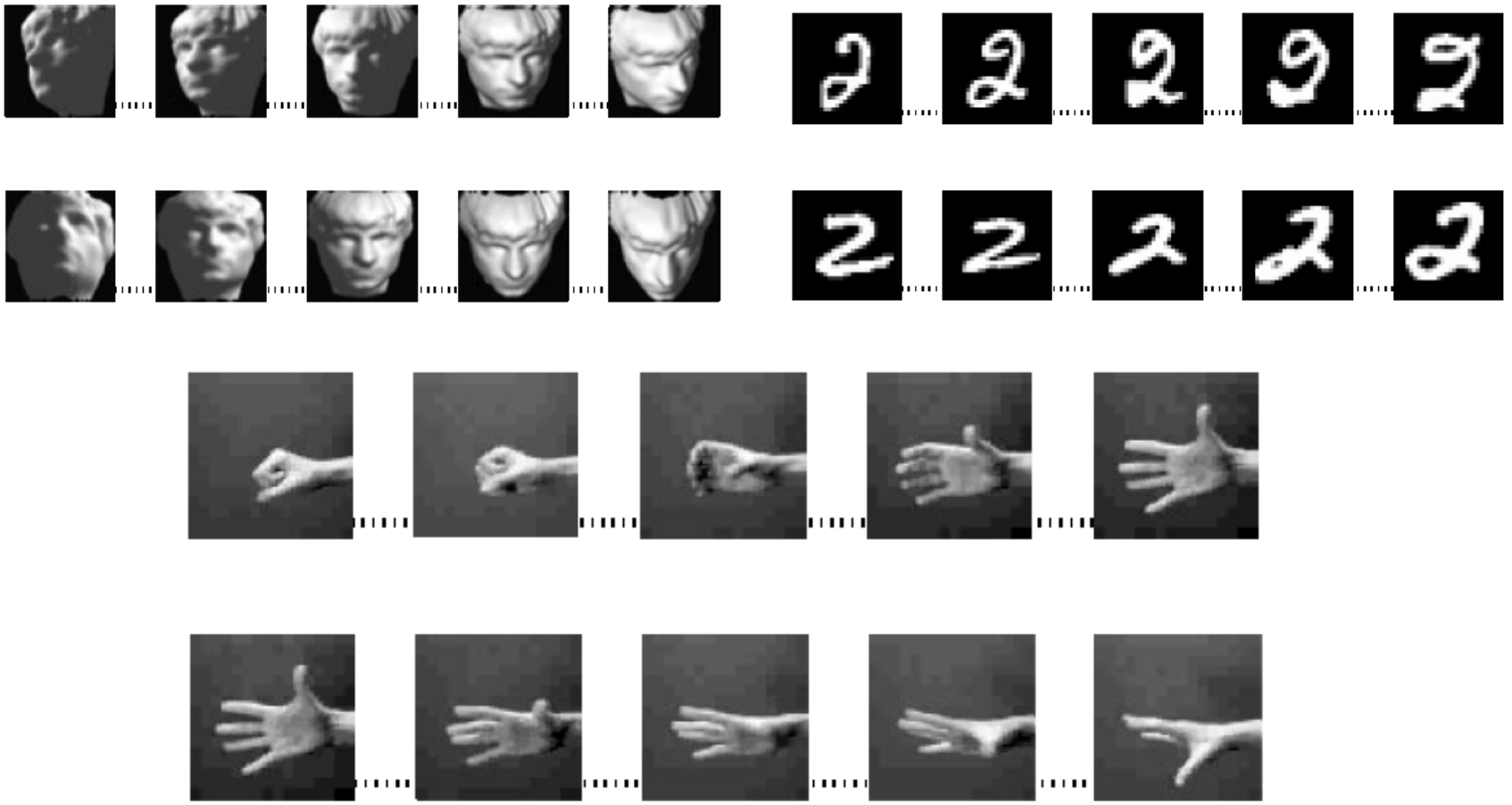

A
Up-down pose
Left-right pose
Lighting direction


B
Bottom loop articulation
Top arch articulation

Interpolating between images

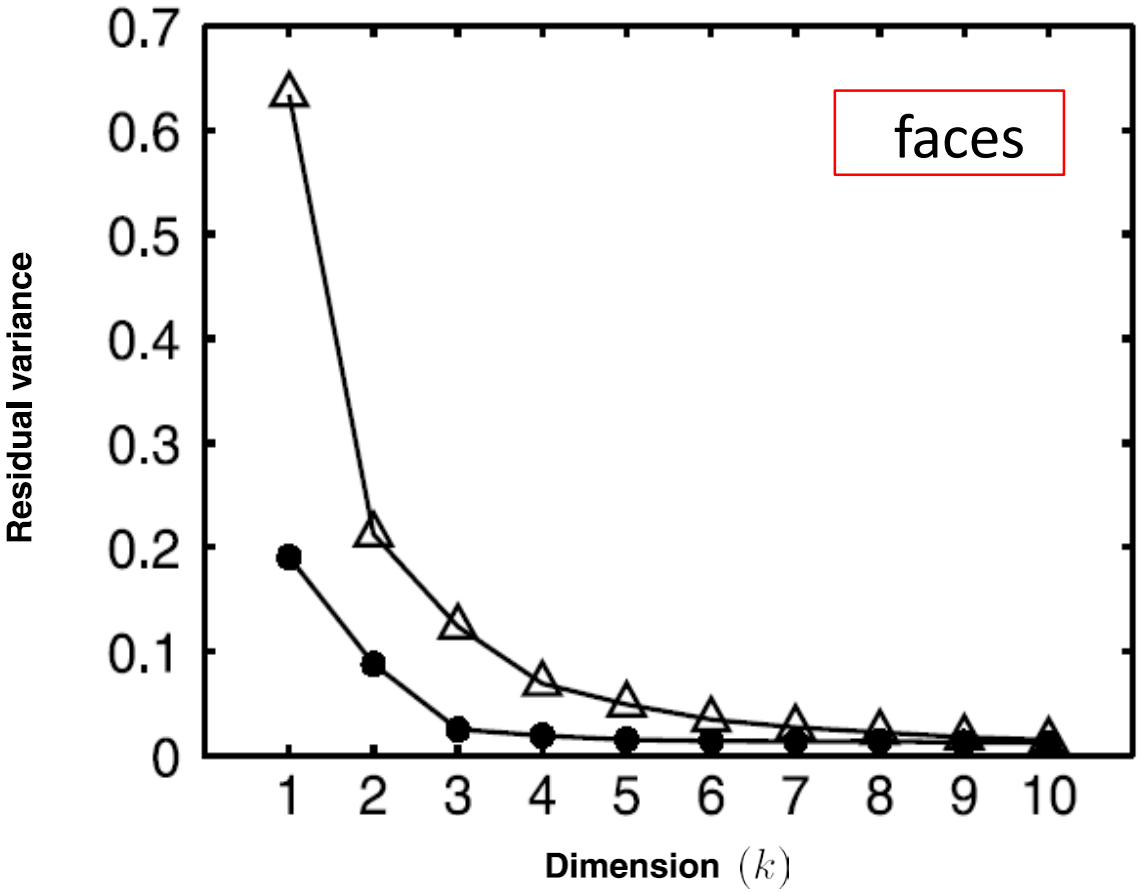


Interpolating between images



Residual Variance

$$\rho(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \in [-1, 1]$$



Residual Variance $1 - \rho^2(D_M, D_Y)$

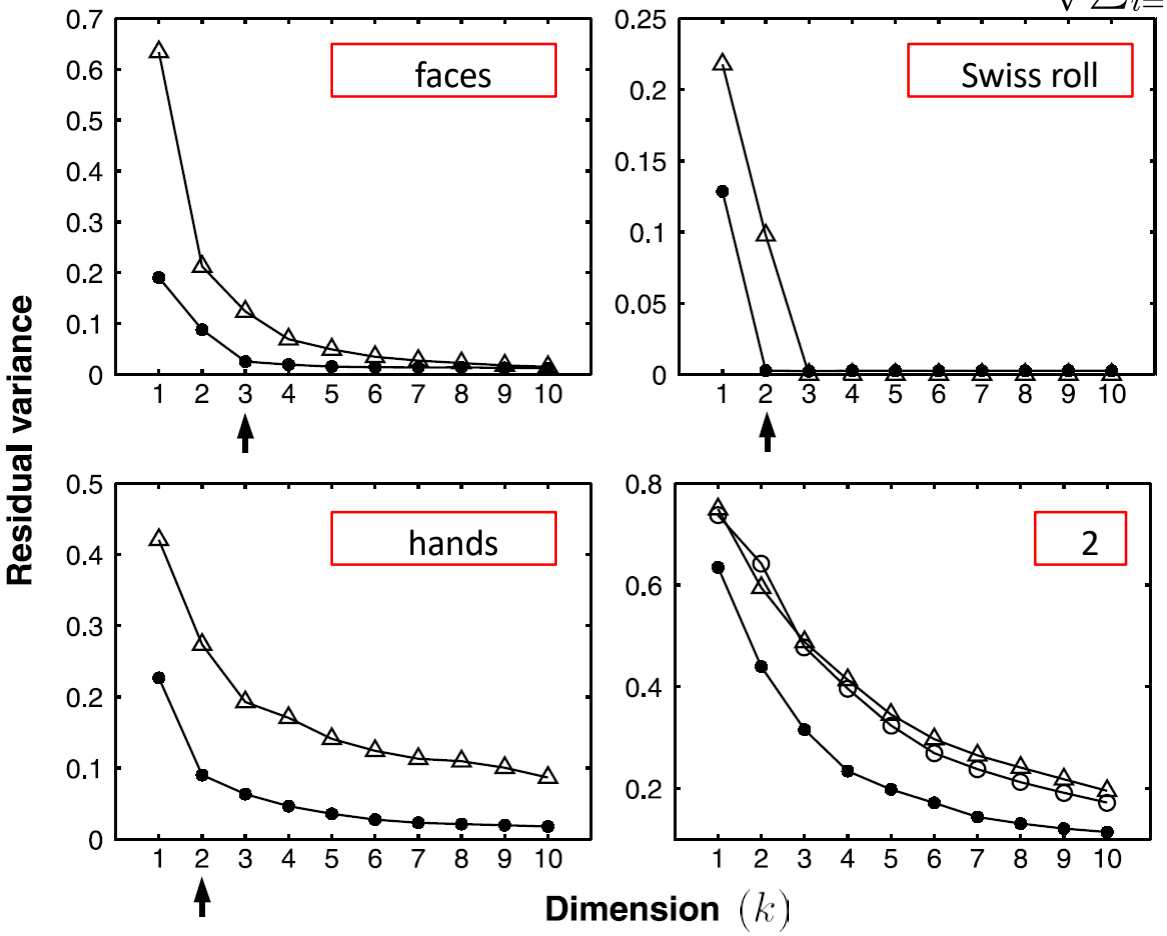
- ρ is the correlation coefficient
- D_M : (estimated) real distances
- D_Y : "reduced" distances

Residual Variance $= 0$:

- real/reduced distances perfectly correlated.

Residual Variance

$$\rho(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \in [-1, 1]$$



Residual Variance $1 - \rho^2(D_M, D_Y)$

- ρ is the correlation coefficient
- D_M : (estimated) real distances
- D_Y : "reduced" distances

Residual Variance $= 0$:

- real/reduced distances perfectly correlated.

Outline of lecture

- Why dimensionality reduction?
- Linear methods
  - PCA
  - MDS
- Nonlinear methods (with graphs)
  - Isomap
  - Laplacian Eigenmaps
  - LLE
- Comparison

Laplacian Eigenmaps

Intuition

- Matrix W captures similarities of points $x_i \in \mathbb{R}^d$
- Similar points should be embedded close to each other.

Suppose we embed in $k = 1$ dimension:

$$\min_{y_1,\dots,y_N \in \mathbb{R}} \sum_{(i,j)\in E} W(i,j)(y_i - y_j)^2 = \min_{y \in \mathbb{R}^N} y^\top L y$$

subject to:

- to avoid collaplse ($y = 0$): $y^\top D y = 1$
- to avoid trivial solution: $y^\top D 1 = 0$

Solution is the second eigenvector u_2 of the generalized eigenproblem $Ly = \lambda Dy$ (equivalently $D^{-1/2} L D^{-1/2} y = L_n y = \lambda y$).

Laplacian Eigenmaps

To embed in k dimensions, change the objective to:

$$\min_{y_1,\dots,y_N \in \mathbb{R}^k} \sum_{(i,j)\in E} W(i,j)\|y_i - y_j\|_2^2 = \operatorname{tr}(Y^\top L Y)$$

Algorithm (Laplacian Eigenmaps)

Collect the coordinates of embedded points as rows of matrix Y^*, where

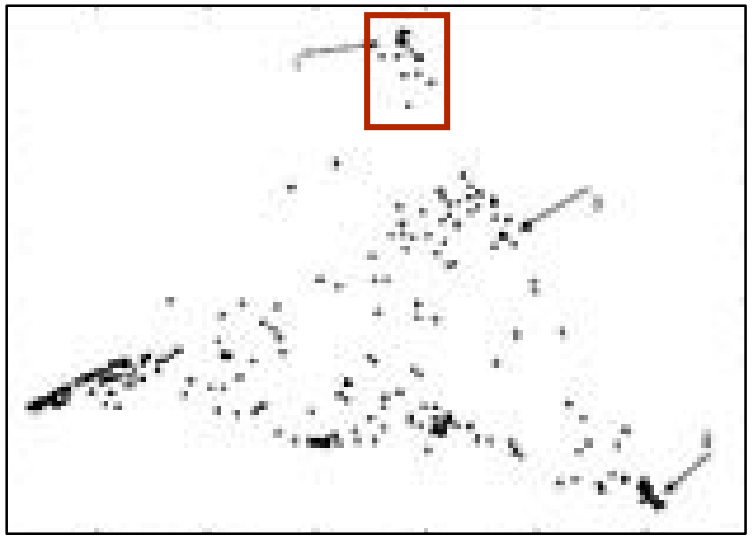
$$Y^* = \underset{Y \in \mathbb{R}^{N\times k}}{\arg\min} \operatorname{tr}(Y^\top L Y), \quad \text{subject to} \quad Y^\top D Y = I$$

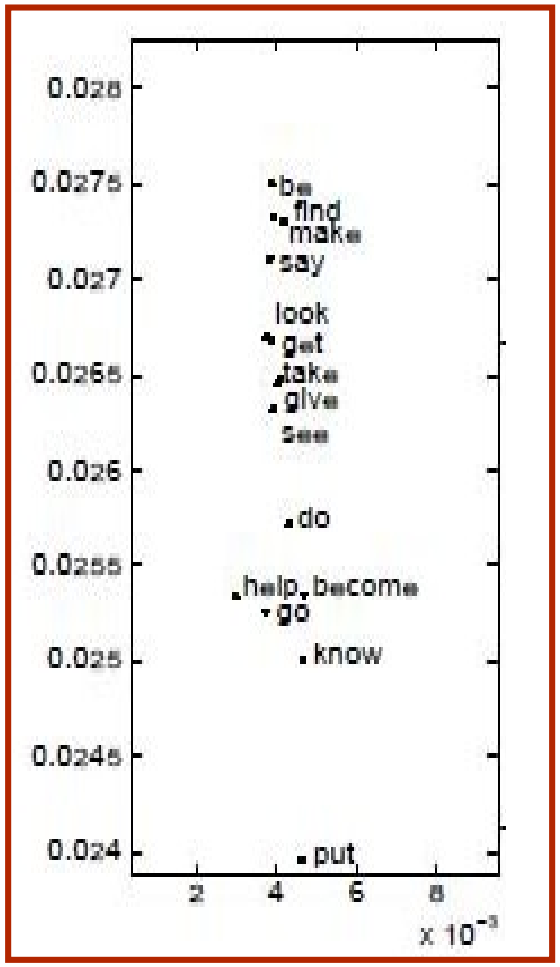
- Solutions are the k first non-trivial eigenvectors of L_n
- Coordinate maps are smooth functions over the original graph. Note similarity with clustering!

Word embedding example

Experiment setup:

- 300 most freq. words from Brown corpus
- d = 600
- k = 14, σ = infty



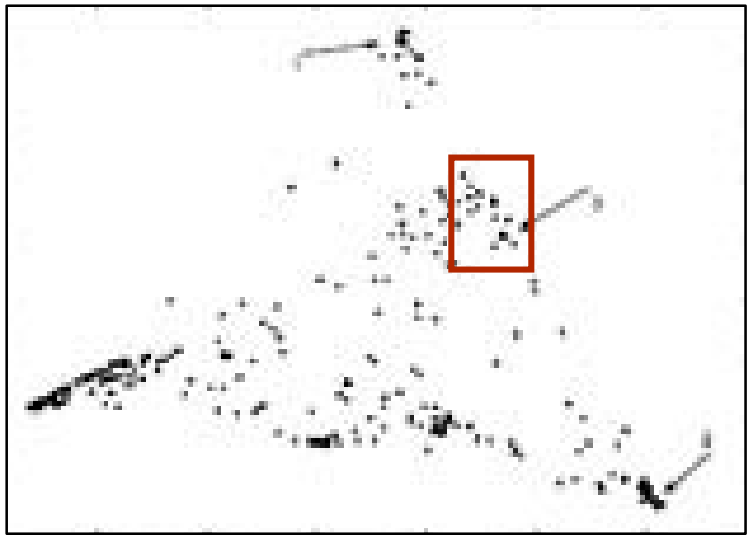


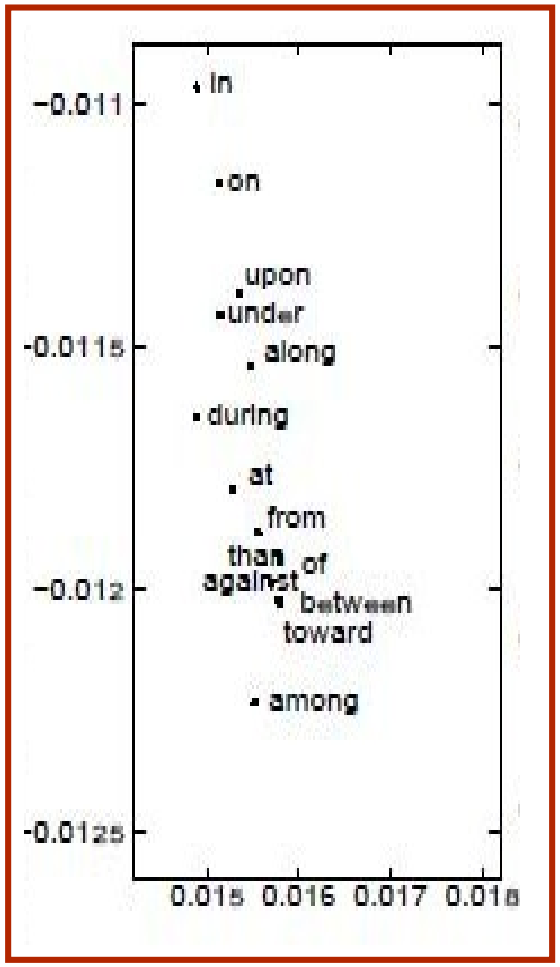
Embedding sub-region contains infinites of verbs.

Word embedding example

Experiment setup:

- 300 most freq. words from Brown corpus
- d = 600
- k = 14, σ = infty



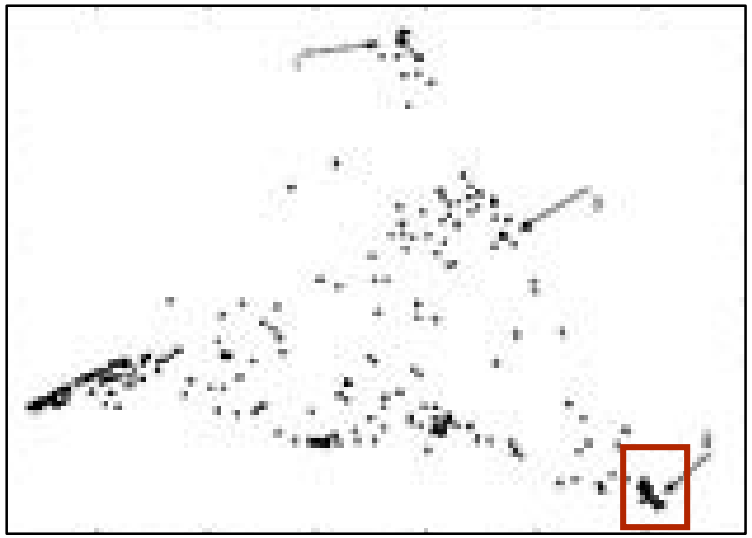


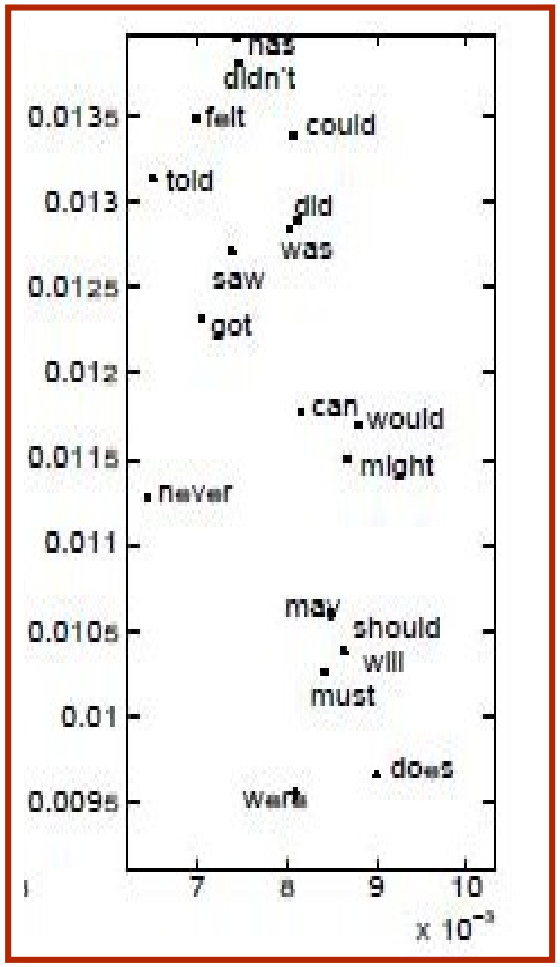
Embedding sub-region contains prepositions. modal and auxiliary verbs

Word embedding example

Experiment setup:

- 300 most freq. words from Brown corpus
- d = 600
- k = 14, σ = infty



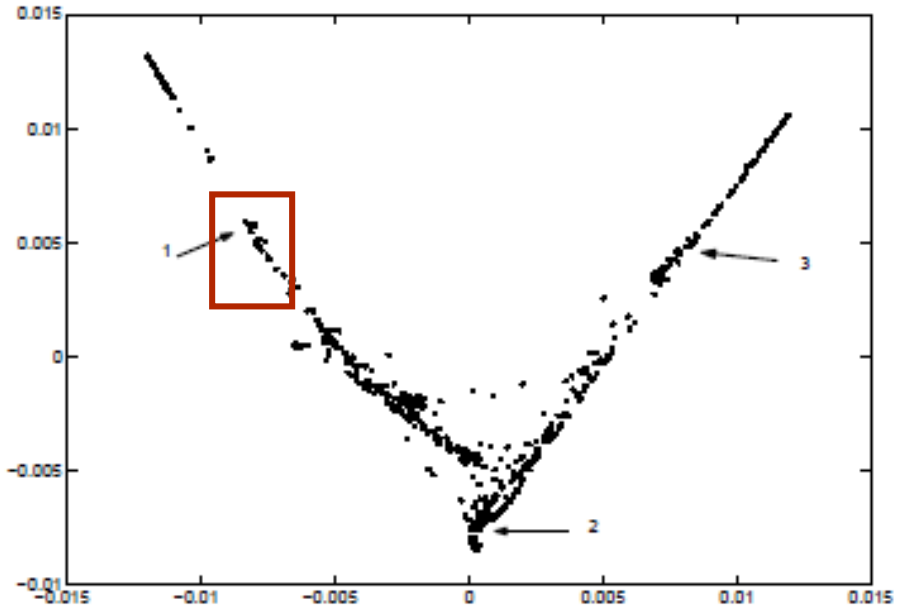


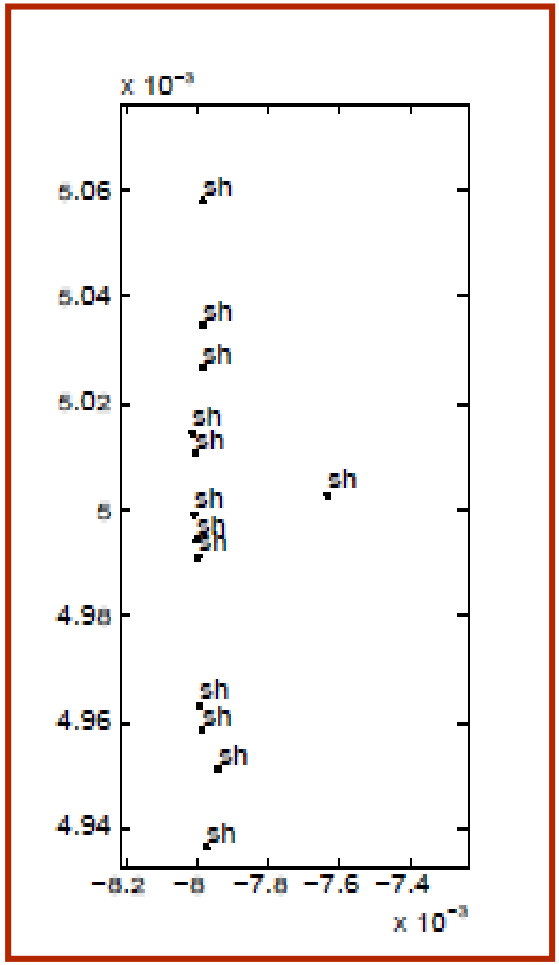
Embedding sub-region contains modal and auxiliary verbs.

Speech embedding example

Experiment setup:

- 30ms window at 5ms interval
- 256 Fourier coefficients for each 30ms chunk
- 685 such vectors



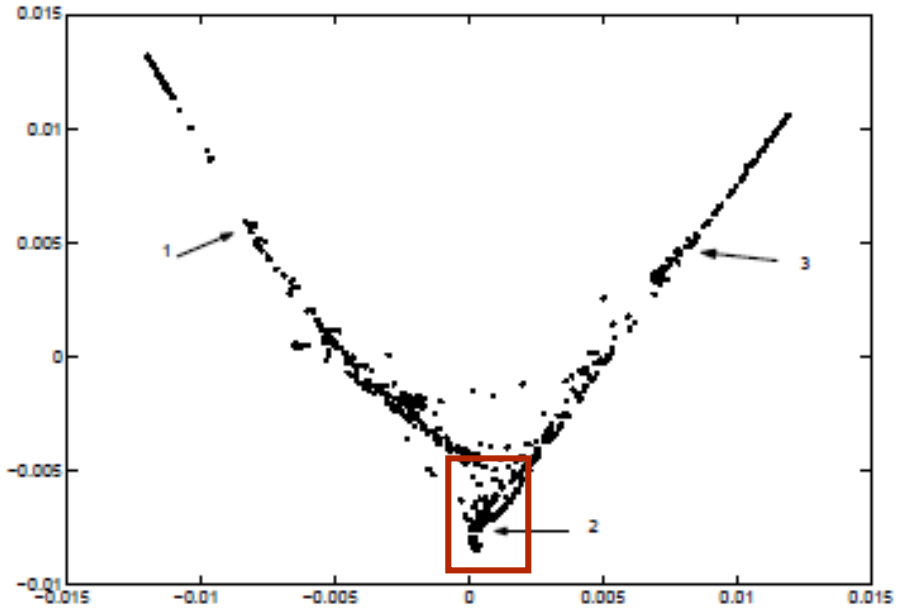


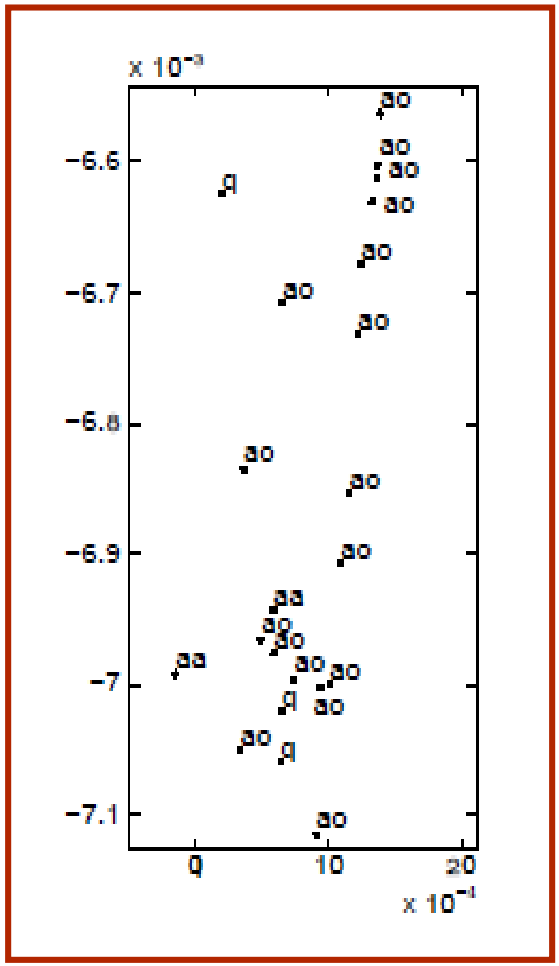
Embedding sub-regions have similar phonetic identities.

Speech embedding example

Experiment setup:

- 30ms window at 5ms interval
- 256 Fourier coefficients for each 30ms chunk
- 685 such vectors



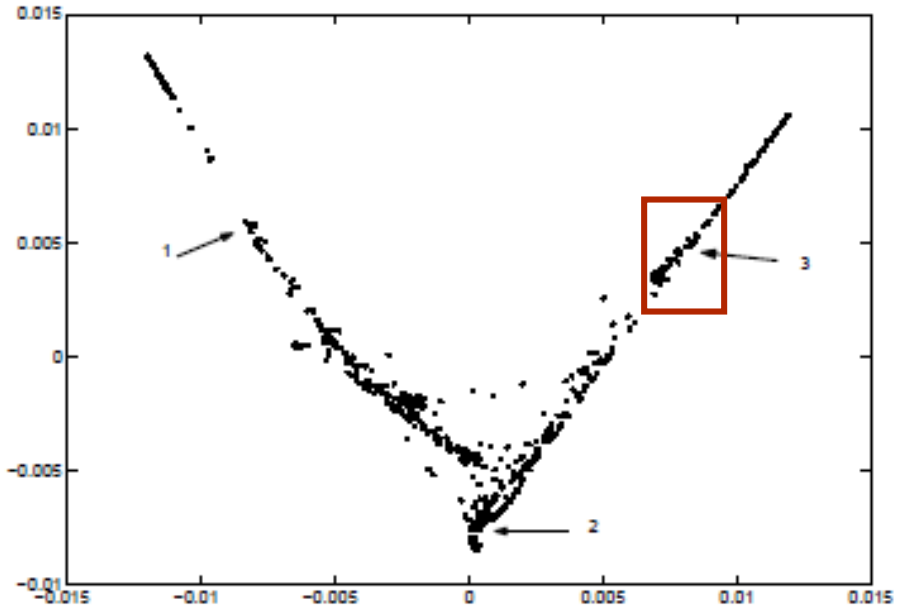


Embedding sub-regions have similar phonetic identities.

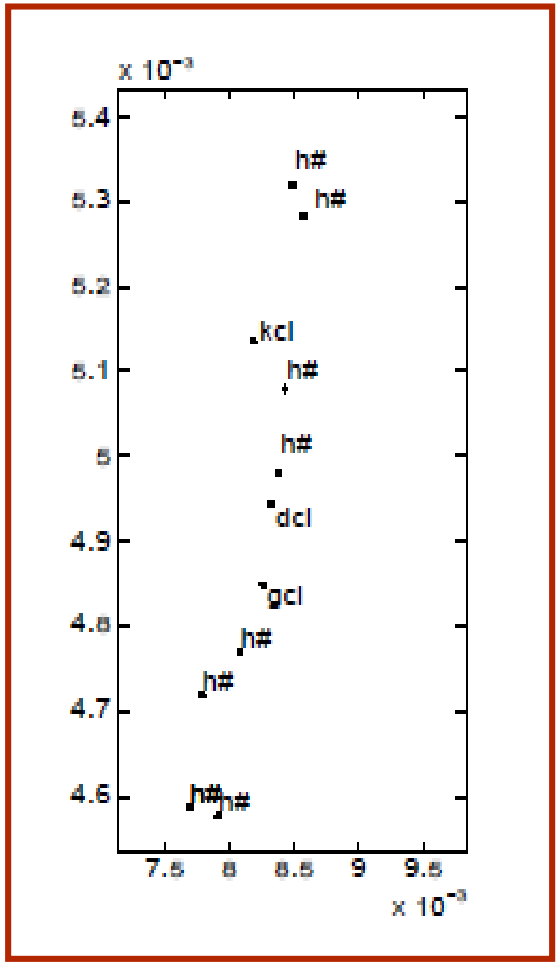
Speech embedding example

Experiment setup:

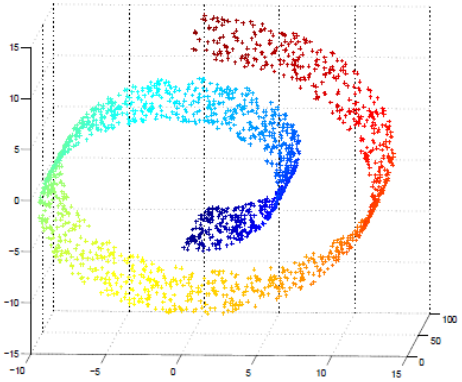
- 30ms window at 5ms interval
- 256 Fourier coefficients for each 30ms chunk
- 685 such vectors



Embedding sub-regions have similar phonetic identities.

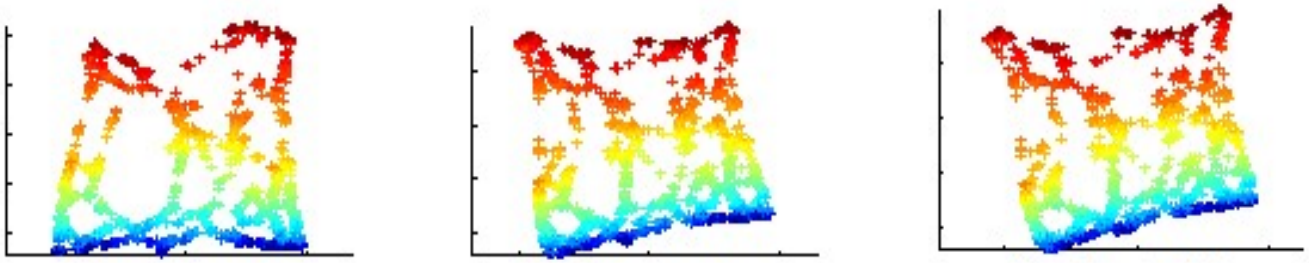


Parameter sensitivity

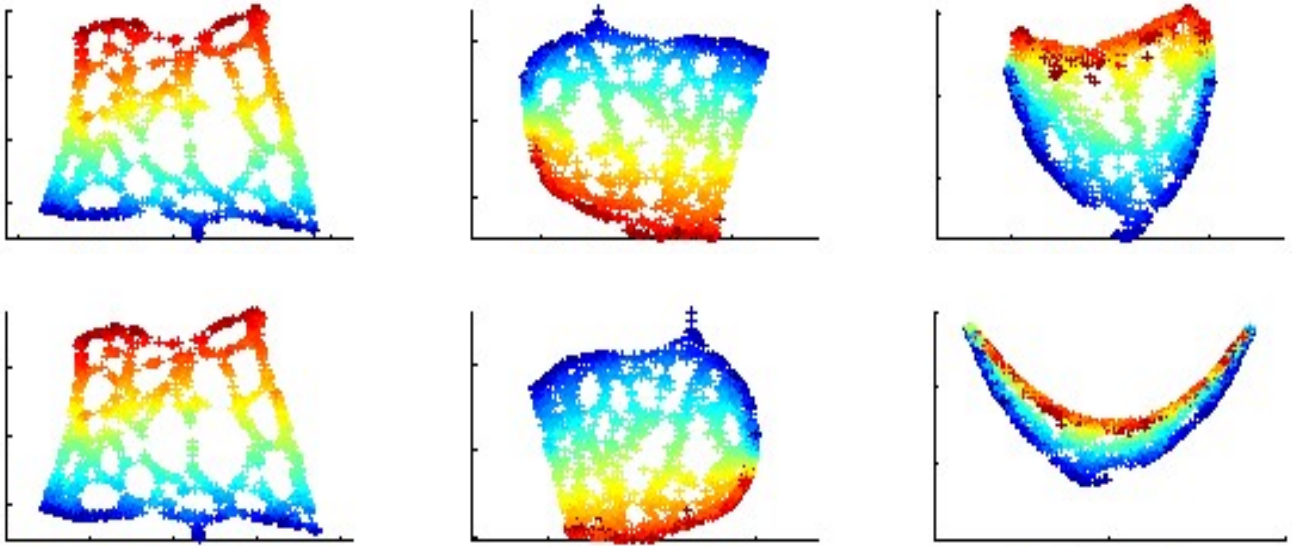


$\sigma = 5^{1/2}$

$k = 5$ $k = 10$ $k = 15$



$\sigma = 5$



$\sigma = \text{infty}$

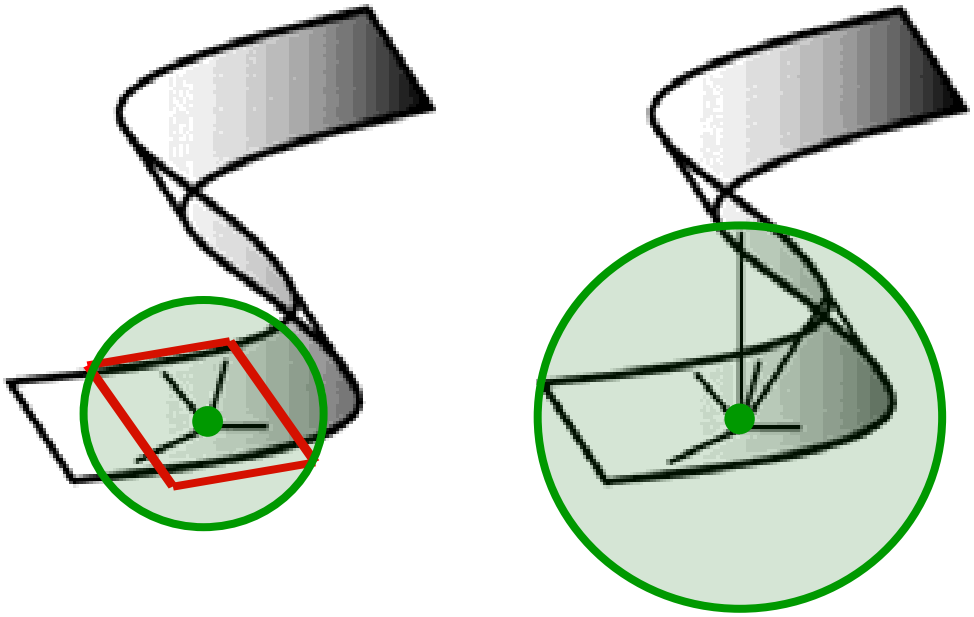
Outline of lecture

- Why dimensionality reduction?
- Linear methods
  - PCA
  - MDS
- Nonlinear methods (with graphs)
  - Isomap
  - Laplacian Eigenmaps
  - LLE
- Comparison

Locally Linear Embedding (LLE)

Key Assumption

- Provided there is sufficient data, we expect each data point and its neighbors to lie on or close to a locally linear patch



Roweis and Saul. "Nonlinear Dimensionality Reduction by Locally Linear Embedding". Science, 2323-2326, 2000

Locally Linear Embedding (LLE)

Algorithm (LLE).

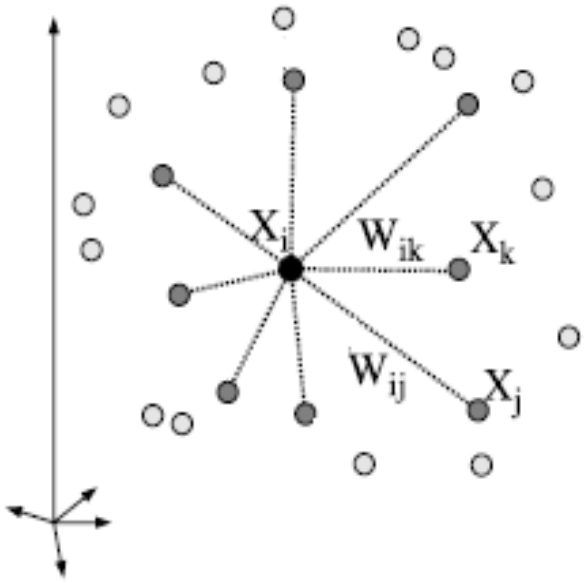
1. Construct unweighted graph G
2. Find weights W that reconstruct each point from its neighbors

$$W^* = \arg\min_{W \in \mathbb{R}^{N \times N}} \sum_{i=1}^{N} \|x_i - \sum_{(i,j) \in E} W(i,j) x_j\|_2^2$$

subject to $\sum_{(i,j) \in E} W(i,j) = 1$.

3. Map to embedded coordinates

$$Y^* = \arg\min_{Y \in \mathbb{R}^{N \times k}} \sum_{i=1}^{N} \|y_i - \sum_{(i,j) \in E} W^*(i,j) y_j\|_2^2$$



LLE properties

For each point, the weights are invariant to

- rotation,
- rescaling and
- translation of the data point and its neighbors.

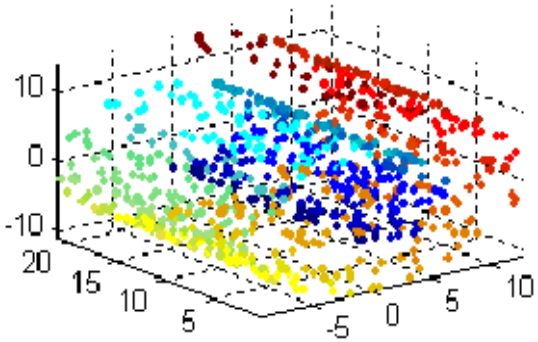
Algorithmically,

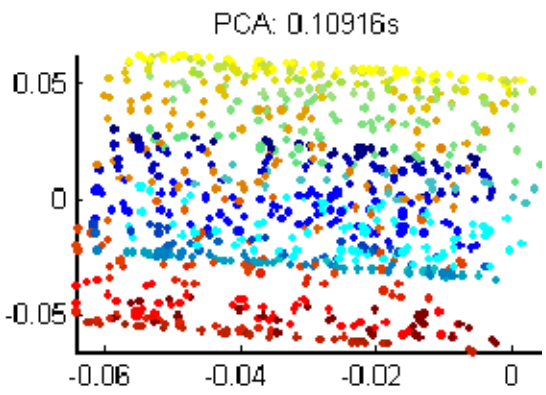
- 1st step corresponds to k-nn graph construction
- 2nd step is a constrained least squares problem
- 3rd step is a sparse eigenvalue problem (of matrix I-W)

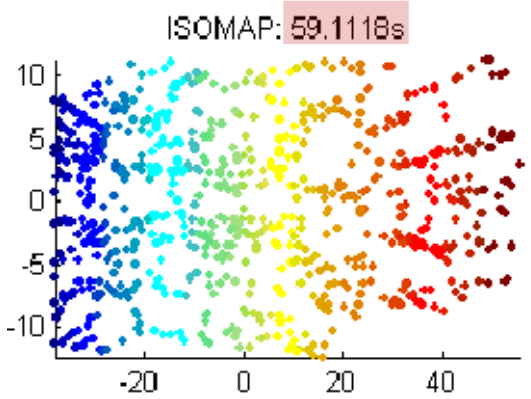
Outline of lecture

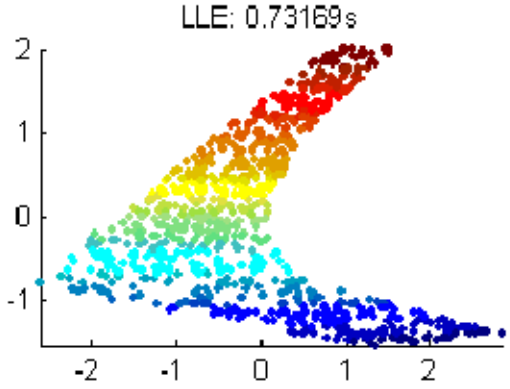
- Why dimensionality reduction?
- Linear methods
  - PCA
  - MDS
- Nonlinear methods (with graphs)
  - Isomap
  - Laplacian Eigenmaps
  - LLE
- Comparison

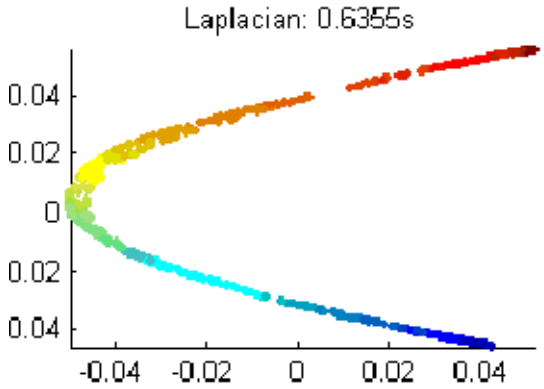
Comparison – Swiss roll



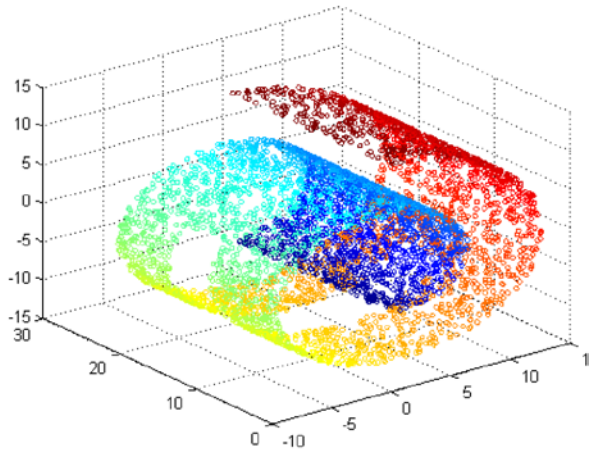


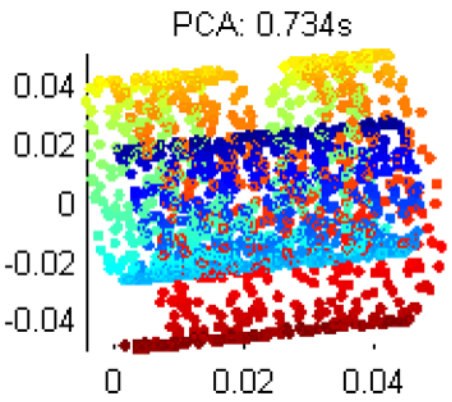


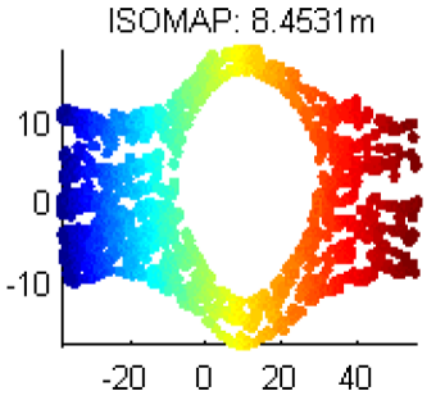


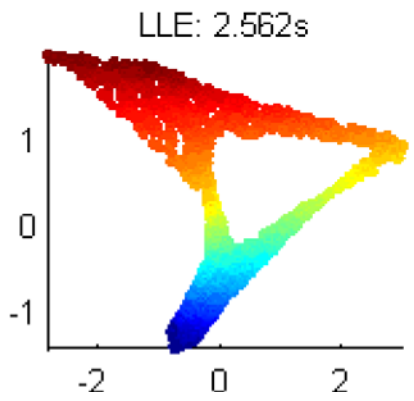


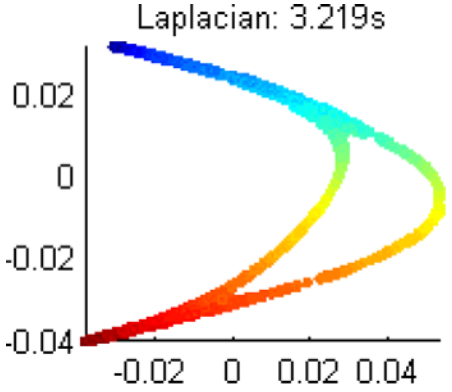
Comparison – non-convex roll





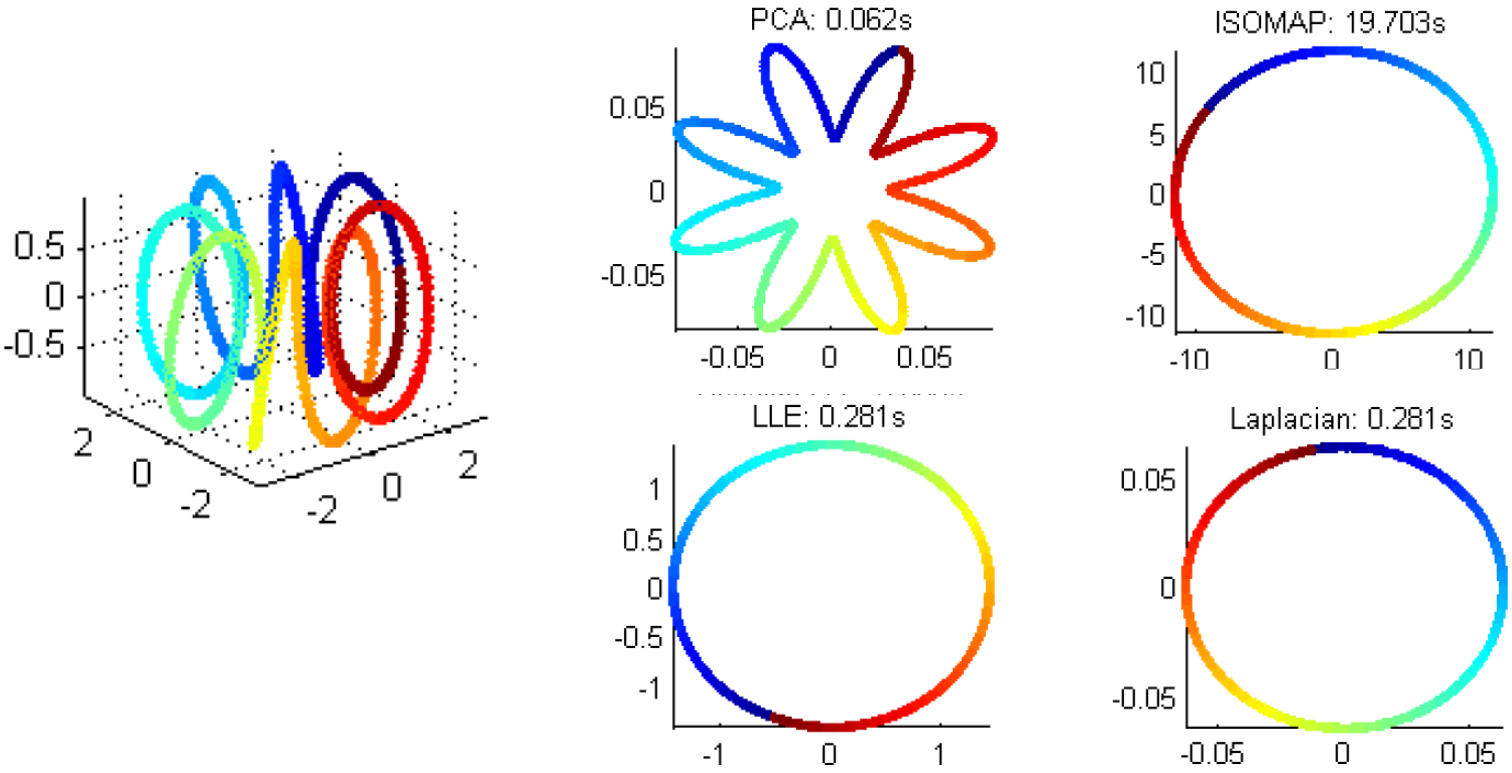






Comparison – noise

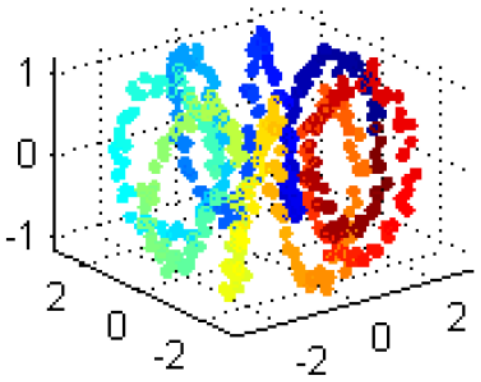
With no noise added, ISOMAP, LLE, and Laplacian Eigenmaps do well.

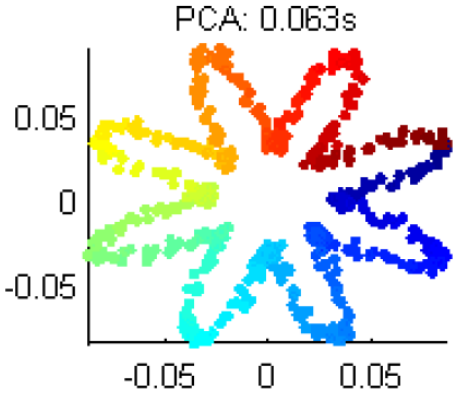


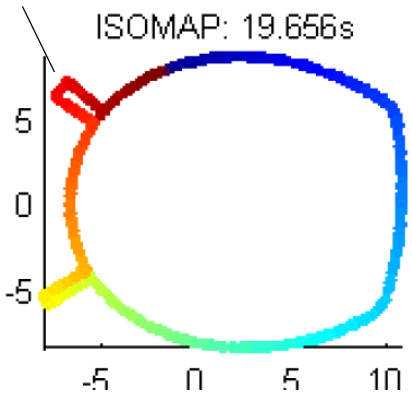
Comparison – noise

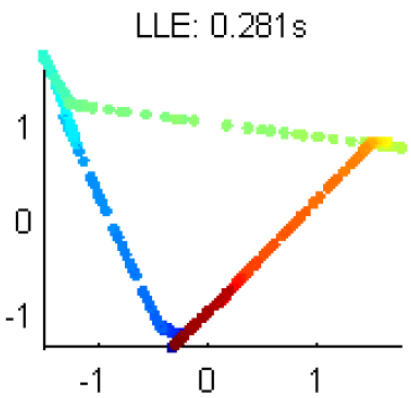
With noise, LLE cannot recover the circle. ISOMAP emphasizes outliers.

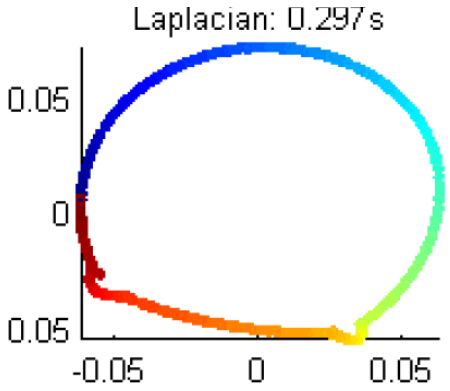
Q: Why does this happen?





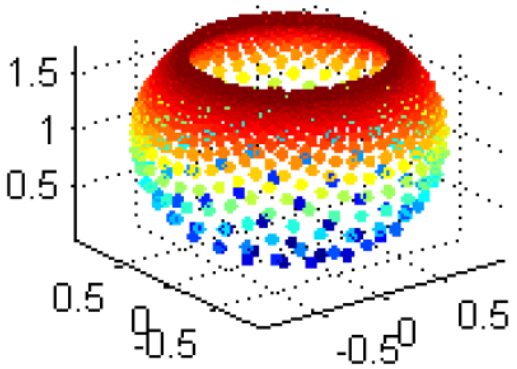


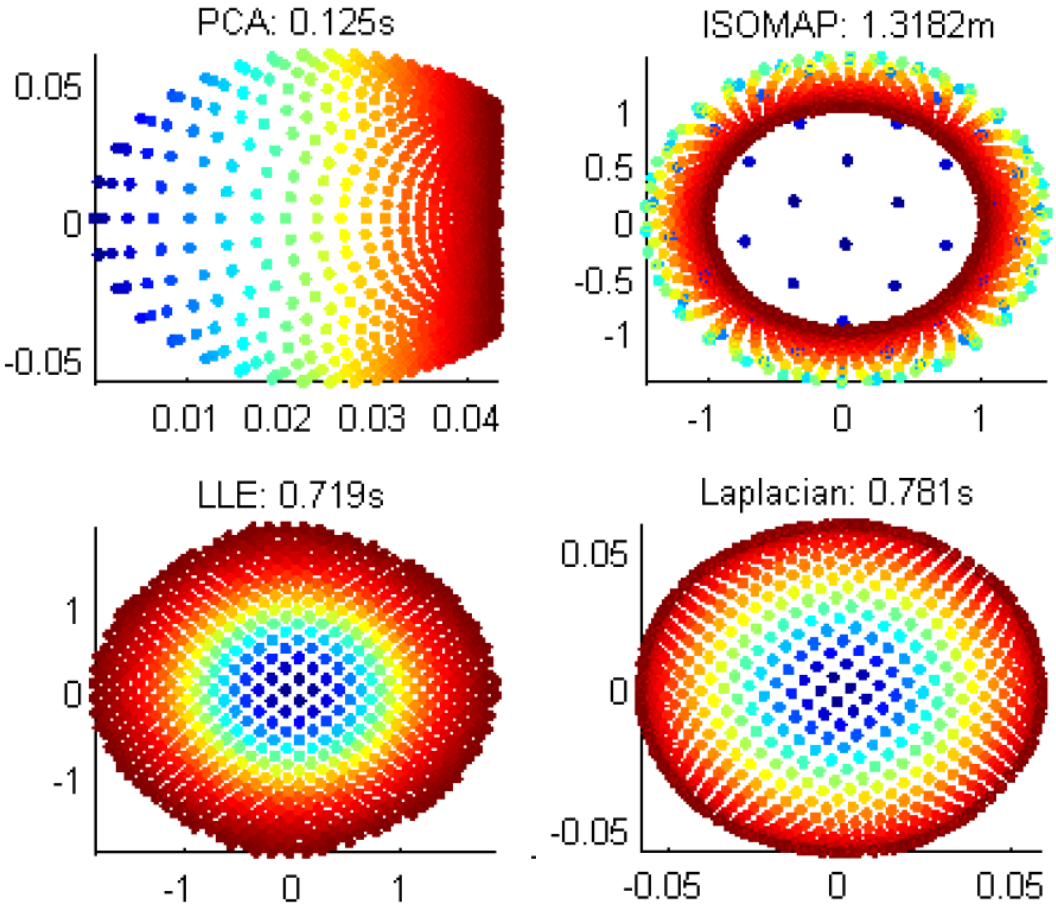




Comparison – non-uniform sampling

Punctured sphere: LLE and Laplacian achieve decent results.





Summary

Dimensionality reduction (unsupervised learning problem):

> Map points $x_1, x_2, \ldots, x_N \in \mathbb{R}^d$ into $y_1, y_2, \ldots, y_N \in \mathbb{R}^k$ such that $k \ll d$ and the mapping preserves distances between points.

- Linear methods (PCA, MDS) focus on all distances.
- Non-linear methods (Isomap, Laplacian Eigenmaps, LLE) preserve local distances by constructing a graph that approximates the manifold

As always, graph construction is a challenge..

Crowding problem

In high-dimension:

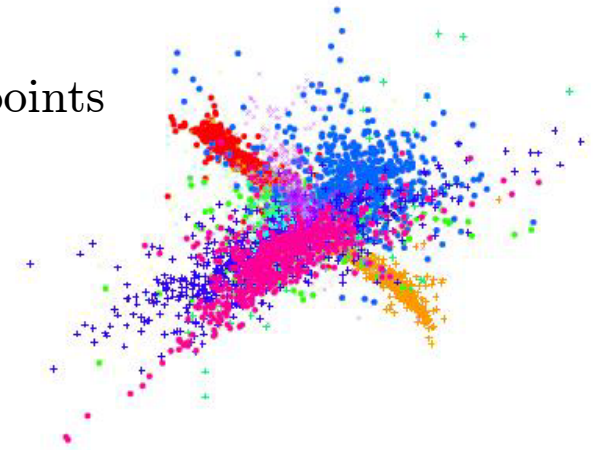
- Are a lot of points in moderate distance from each other

In low-dimension:

- Simply not enough space to represent all distances accurately
- Area accommodating moderately distant points is not large enough compared to area accommodating nearby points

Consequence:

- Small attractive forces between moderate-distance points accumulate
- Algorithms tend to crowd points together
- Prohibits separation of clusters



The t-SNE solution

t-SNE objective:

- Try to reflect the symmetrized similarities $p_{ij} = (p_{j|i} + p_{i|j})/(2N)$ as well as possible, where

$$p_{j|i} = \frac{\exp\left(-\|x_i - x_j\|^2/2\sigma_i^2\right)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2/2\sigma_i^2)},$$

Van der Maaten and Hinton: "The similarity of x_j to x_i is the conditional probability, $p_{j|i}$, that x_i would pick x_j as its neighbor if neighbors were picked in proportion to their probability density under a Gaussian centered at x_i."

But, measure similarities q_{ij} between y_i and y_j as:

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l}(1 + \|y_k - y_l\|^2)^{-1}}$$

Heavy-tailed Student-t distribution allows dissimilar objects to be modeled far apart in the map.

Laurens van der Maaten, and Geoffrey Hinton. Visualizing Data using t-SNE. JMLR 2008

t-SNE (part 2)

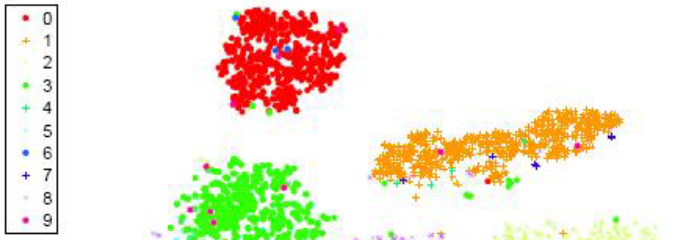
Points y_i are determined by minimizing the (non-symmetric) KullbackLeibler divergence of the distribution Q from the distribution P, that is:

$$KL(P||Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

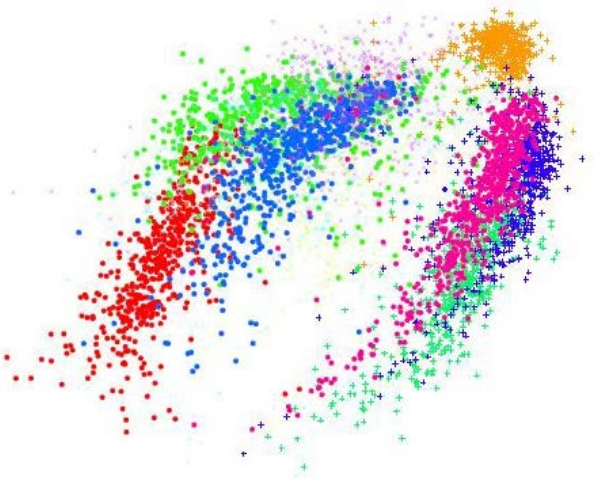
Measure of surprise when going from Q (prior) to P (posterior)

The minimization is done with gradient descent.

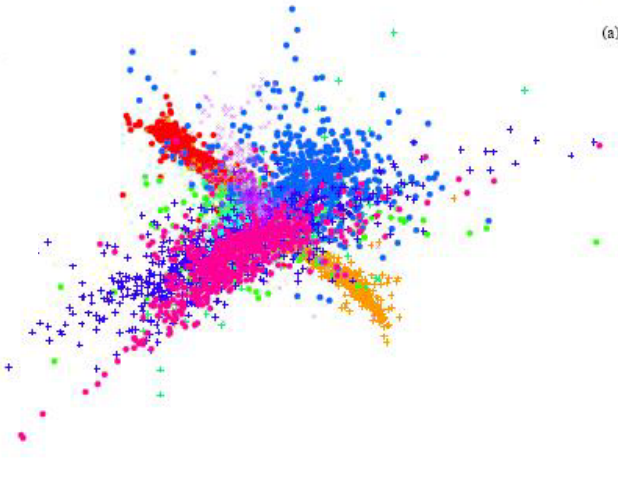
MNIST comparison



(a) Visualization by t-SNE.



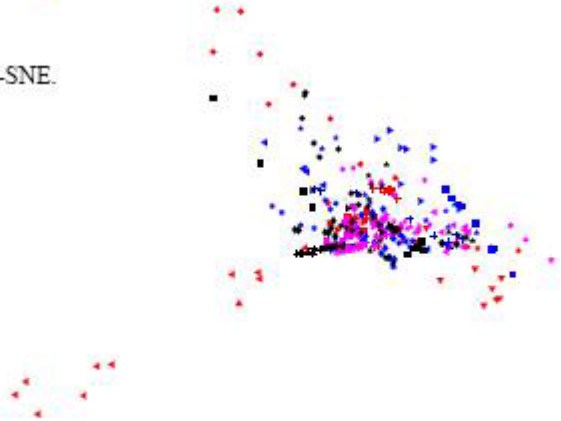
(a) Visualization by Isomap.



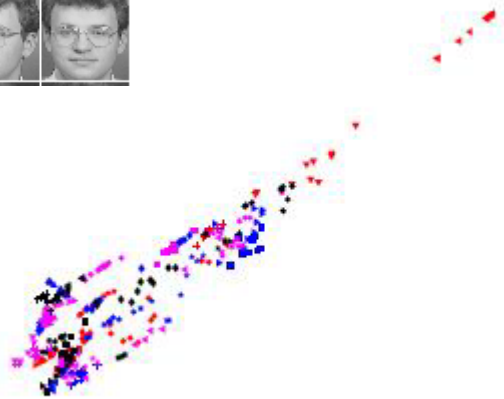
(b) Visualization by LLE.

Olivetti faces comparison

(a) Visualization by t-SNE.



(d) Visualization by LLE.



(c) Visualization by Isomap.

t-SNE on MNIST