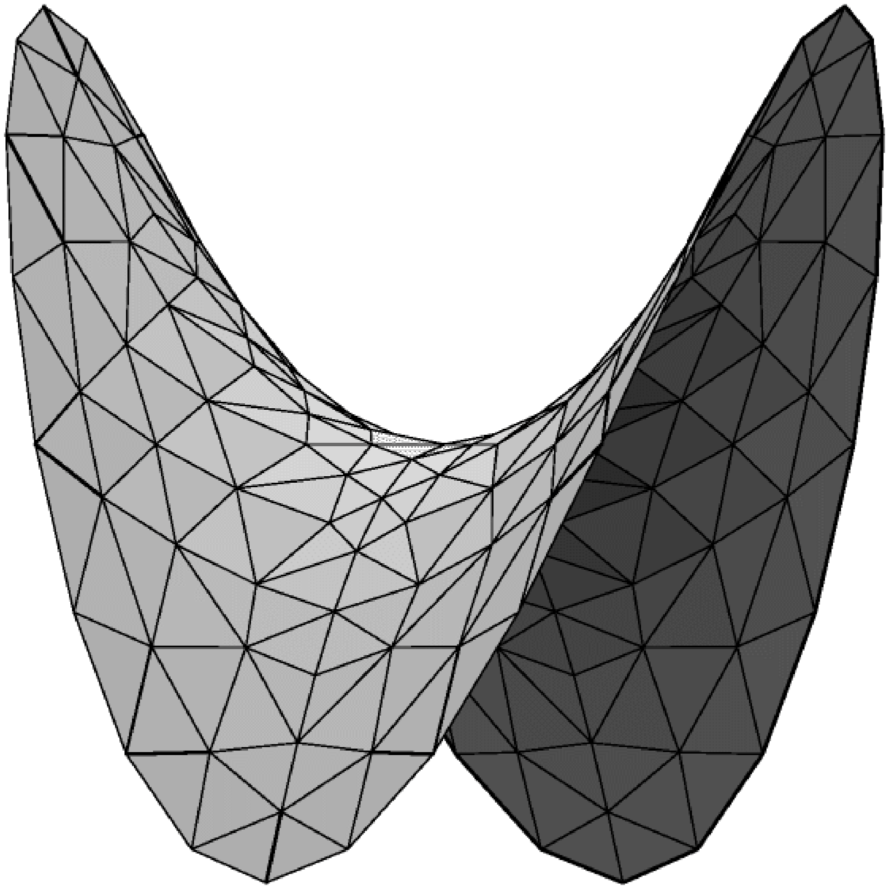
manifold learning
with applications to object recognition



Advanced Perception
David R. Thompson

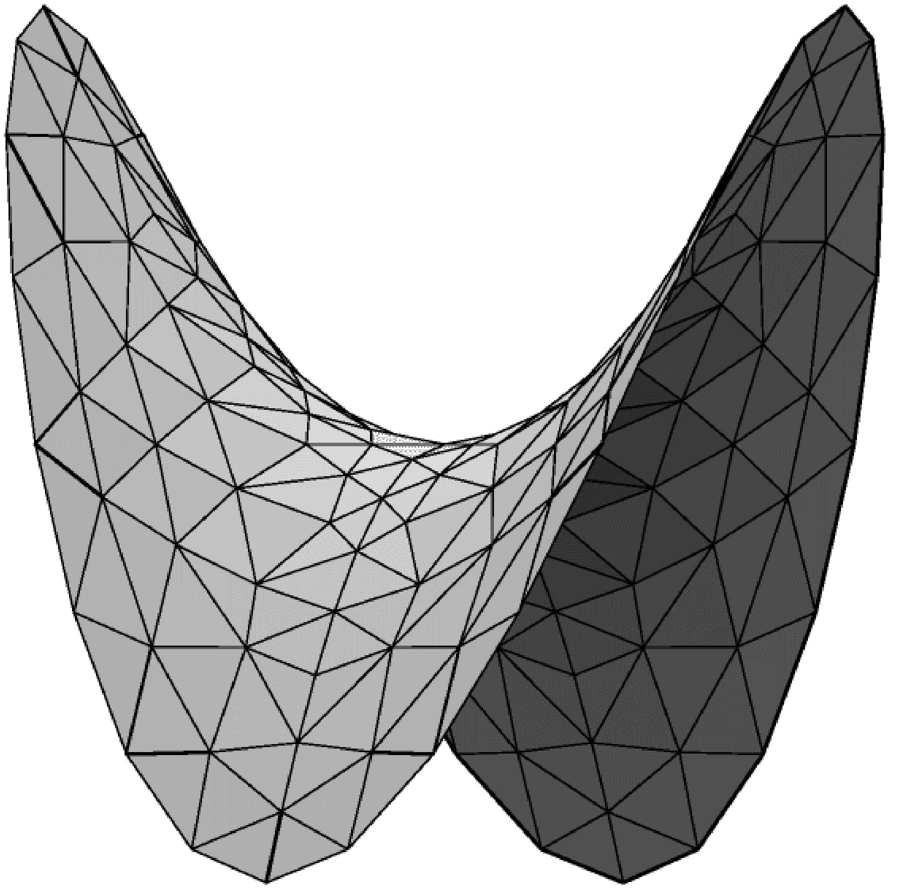
agenda

1. why learn manifolds?

2. Isomap

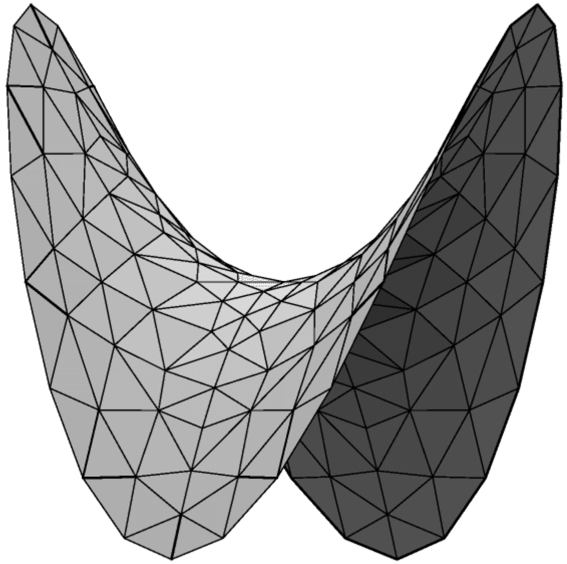
3. LLE

4. applications

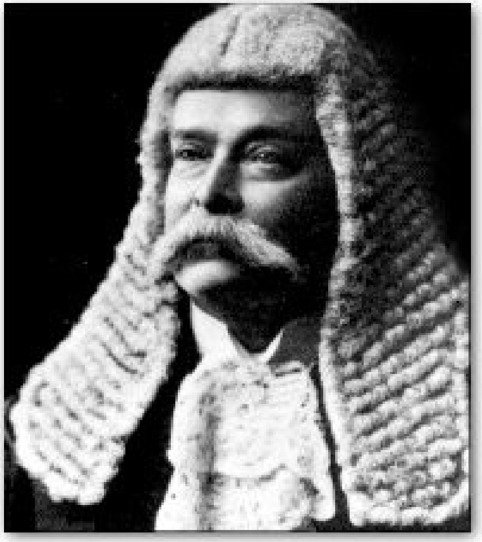


types of manifolds

exhaust
manifold



low-D surface
embedded in
high-D space



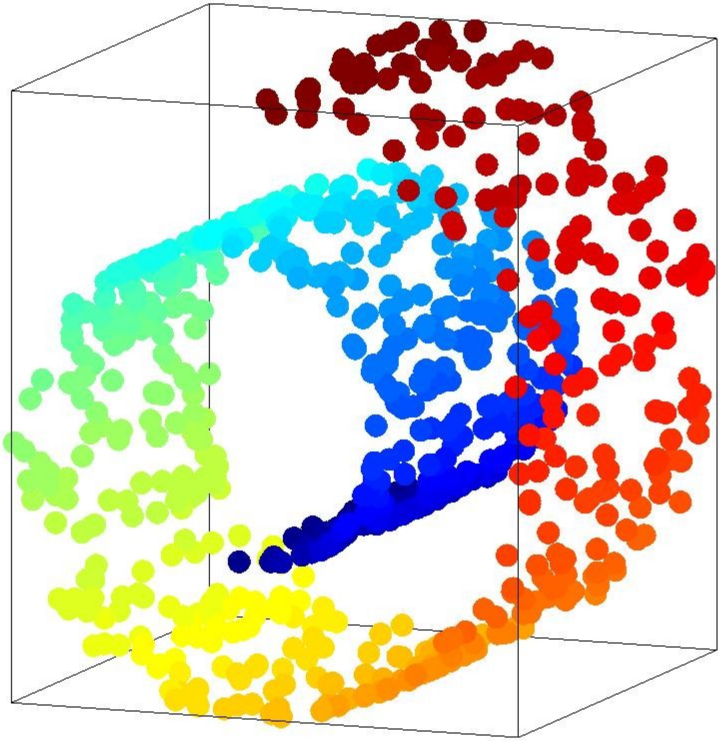
Sir Walter
Synnot Manifold
1849-1928

manifold learning

Find a low-D basis for describing high-D data.

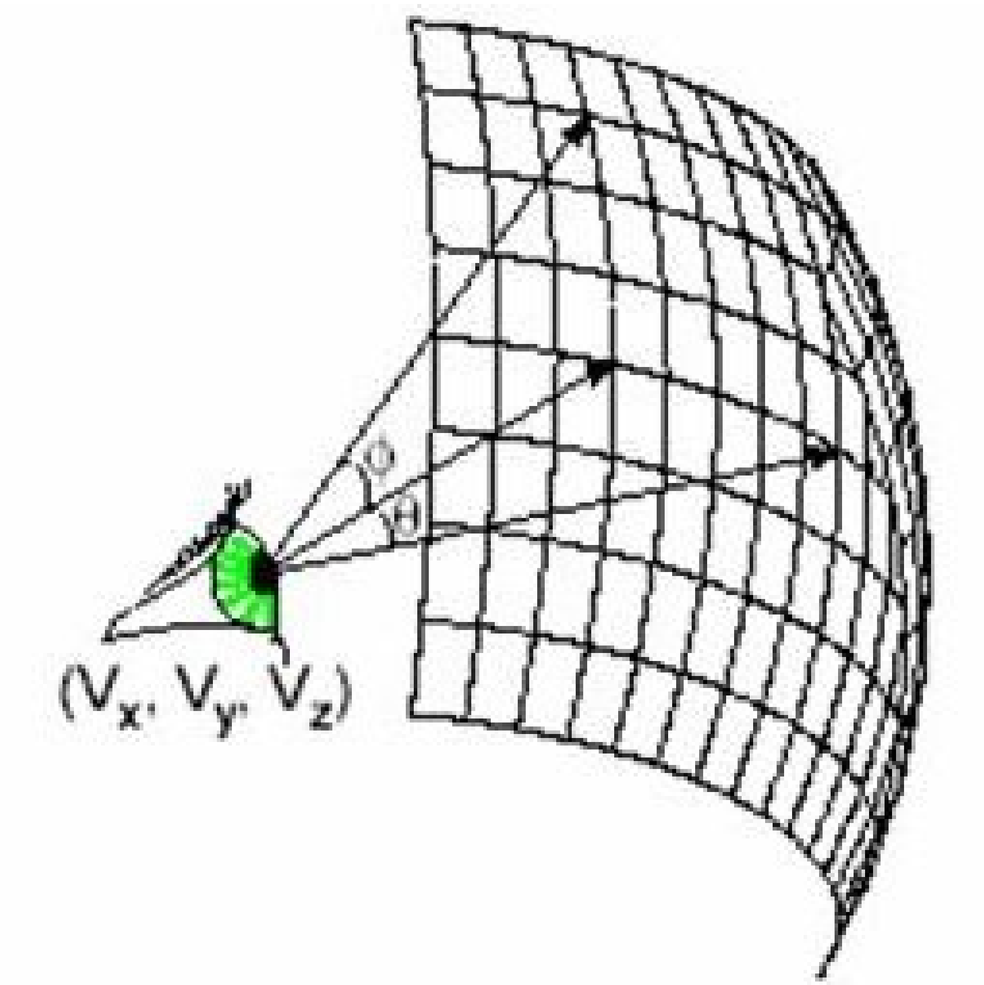
X → X' S.T.
dim(X') << dim(X)

uncovers the intrinsic dimensionality (invertible)



manifolds in vision

plenoptic function / motion / occlusion



manifolds in vision

appearance variation

images from hormel corp.

manifolds in vision

deformation

images from www.golfswingphotos.com

why do manifold learning?

1. data compression

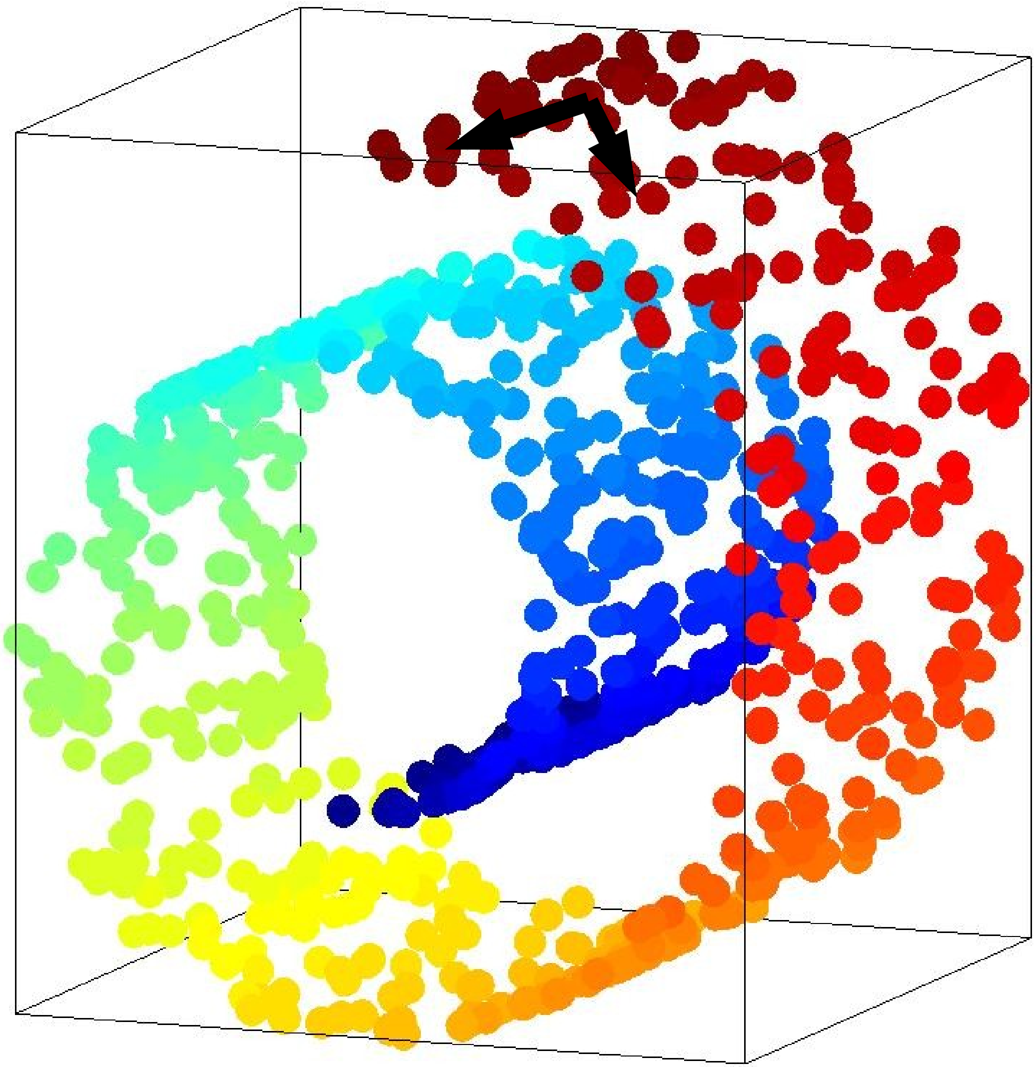
2. “curse of dimensionality”

3. de-noising

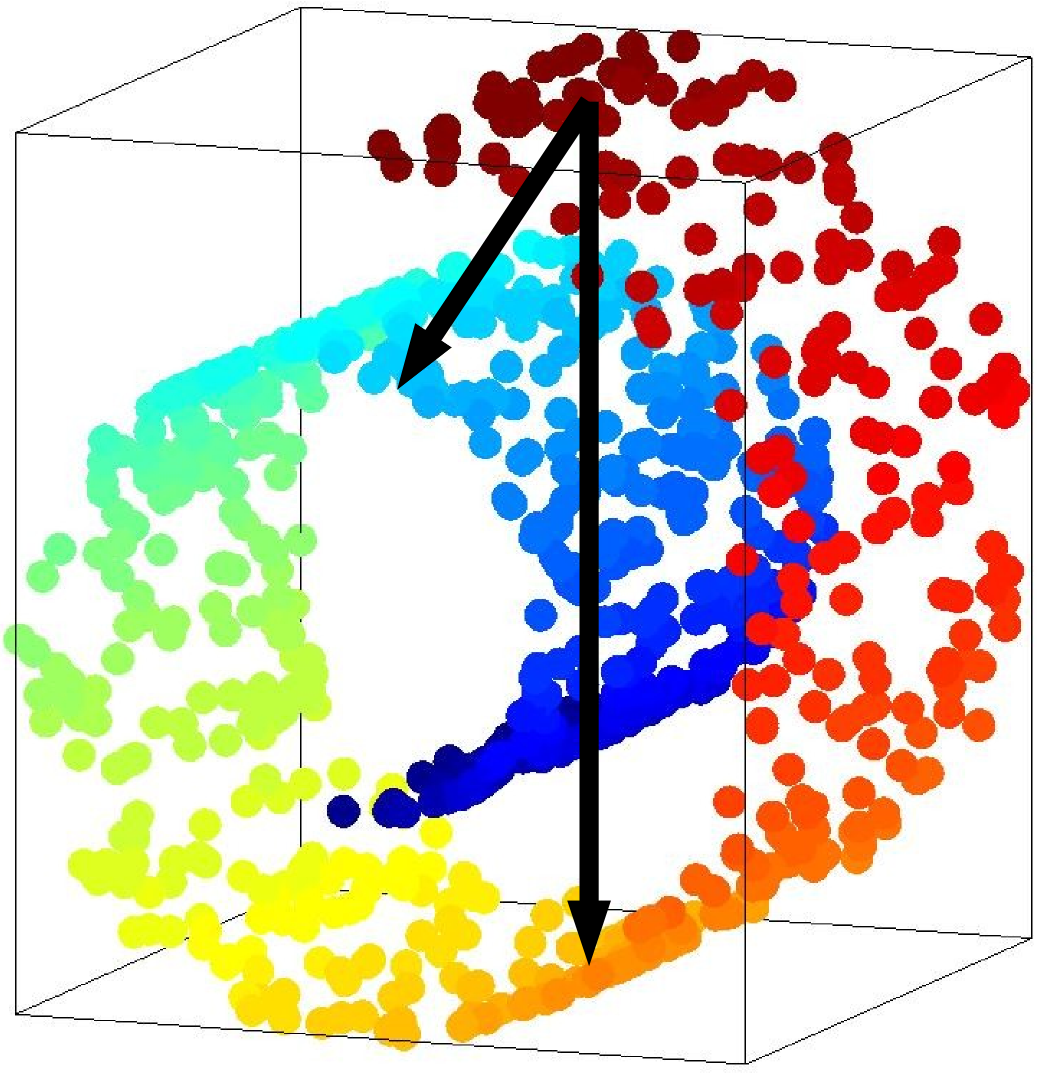
4. visualization

5. reasonable distance metrics *

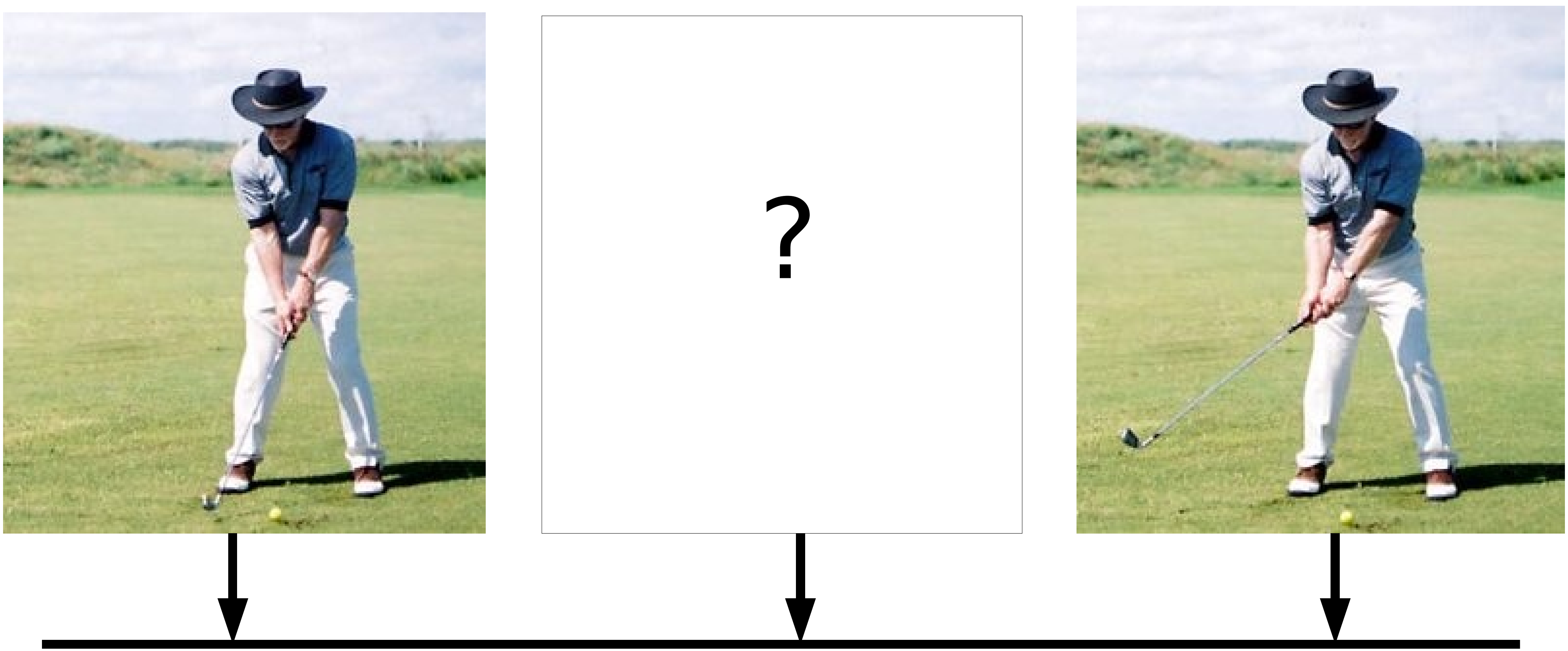
reasonable distance metrics



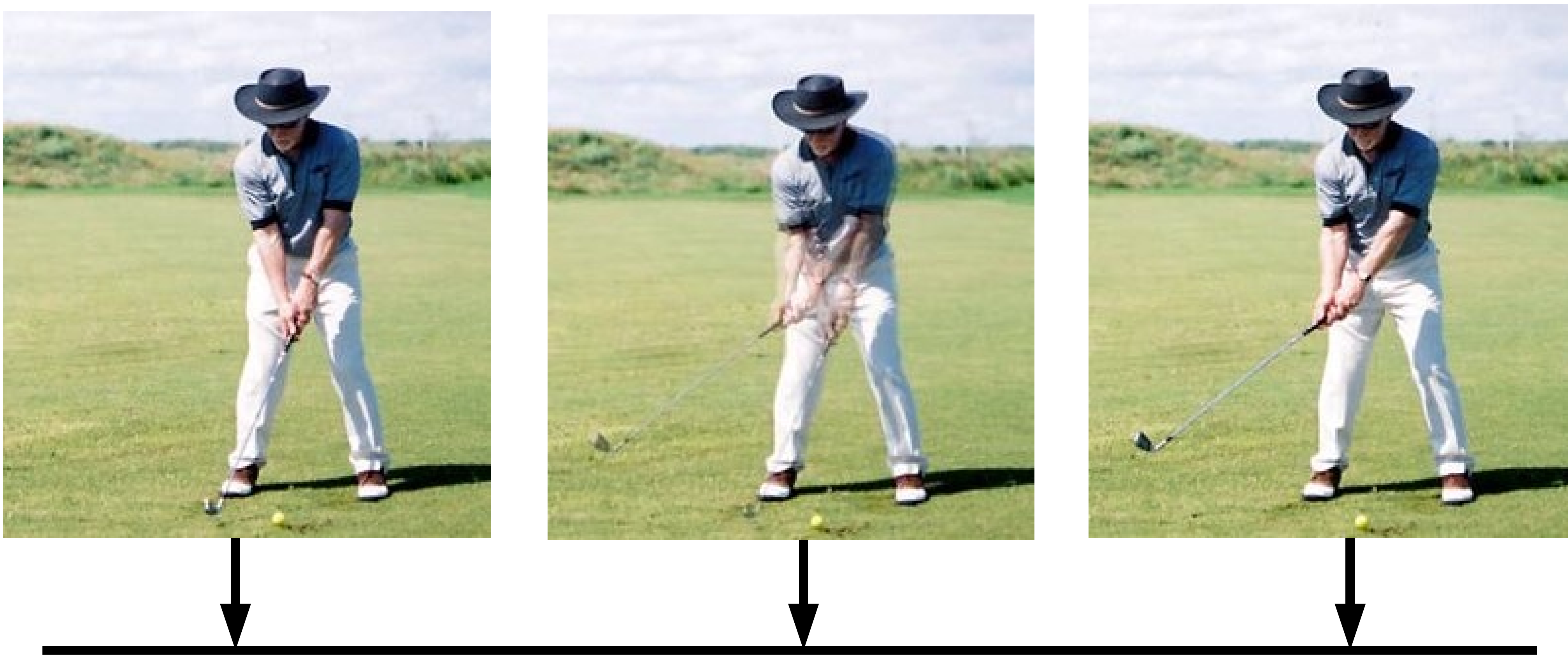
reasonable distance metrics



reasonable distance metrics

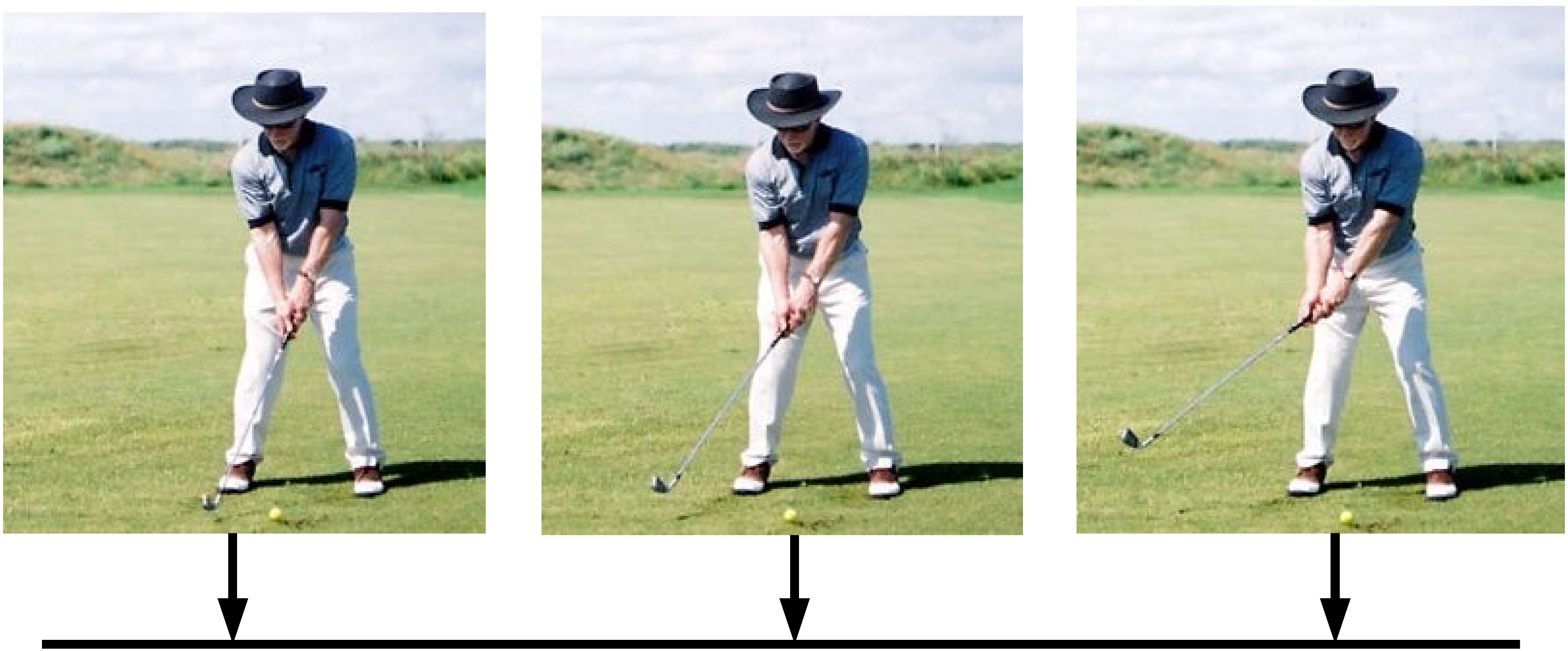


reasonable distance metrics



linear interpolation

reasonable distance metrics



manifold interpolation

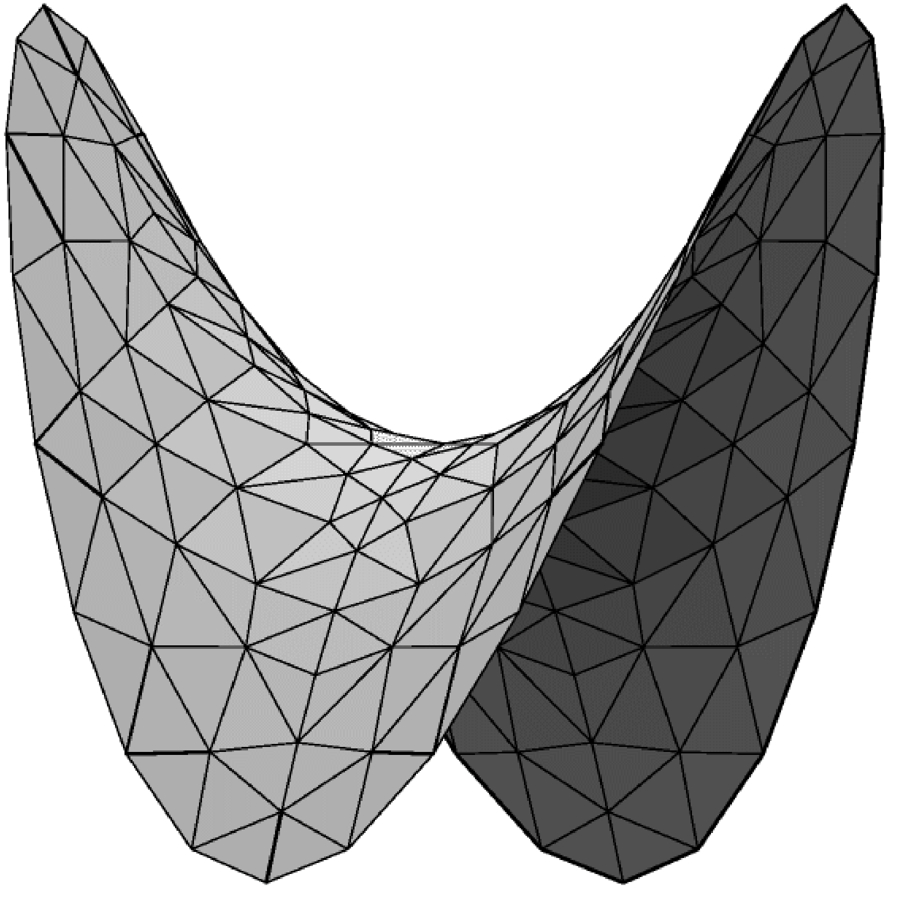
agenda

1. why learn manifolds?

2. Isomap

3. LLE

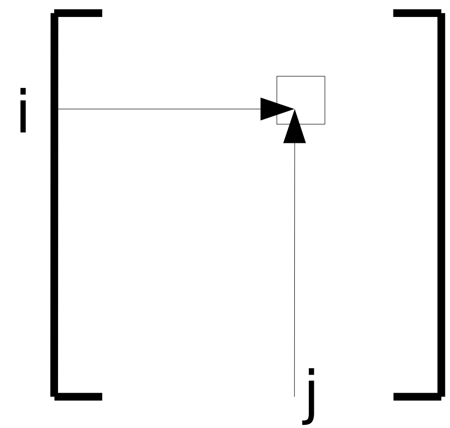
4. applications



Isomap

For *n* data points, and a distance matrix D,

$D_{ij} =$



…we can construct a *m*-dimensional space to preserve inter-point distances by using the top eigenvectors of D scaled by their eigenvalues.

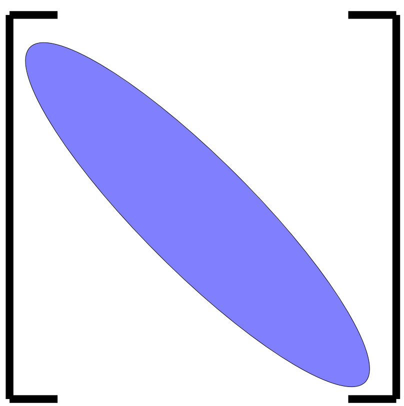
$$y_i = [\ \sqrt{\lambda_1} v_1^i\ ,\ \sqrt{\lambda_2} v_2^i\ ,\ \ldots\ ,\ \sqrt{\lambda_m} v_m^i\ ]$$

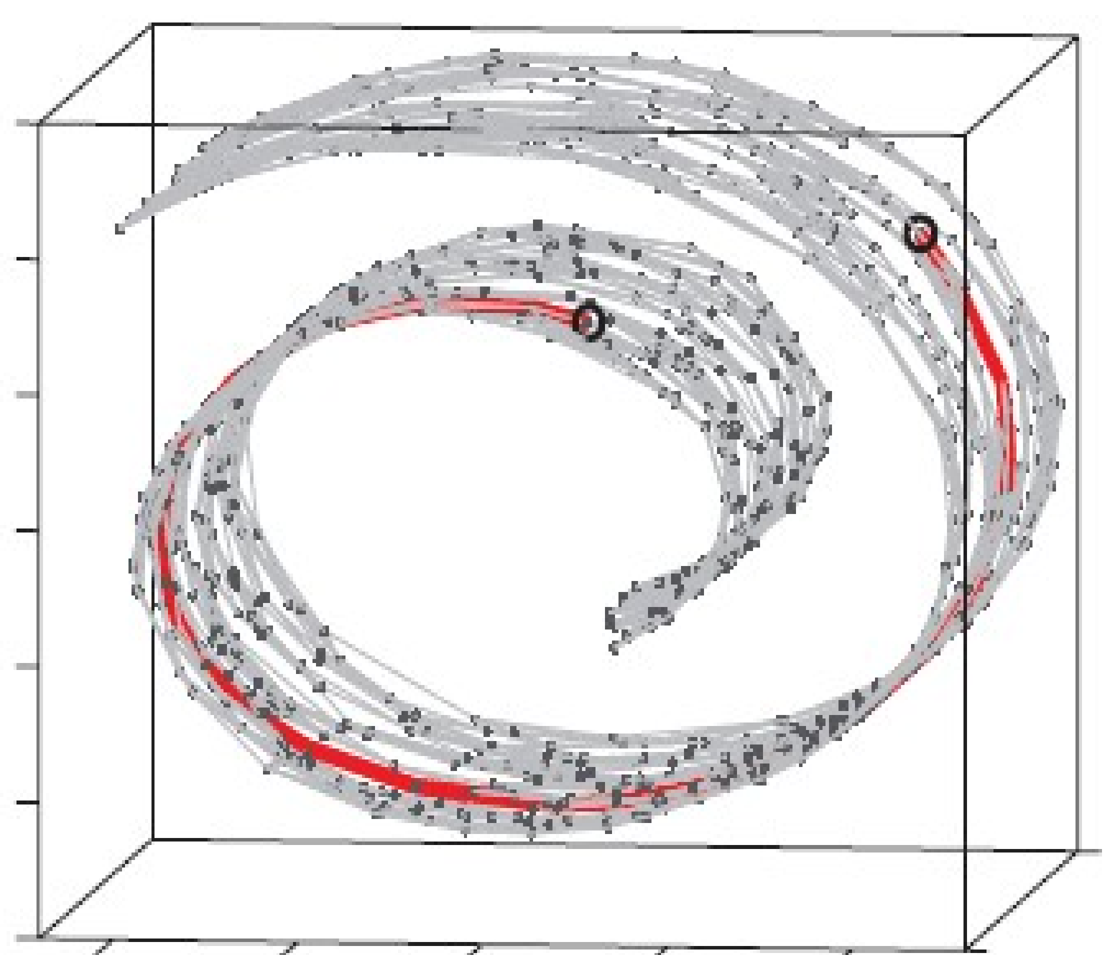
Isomap

Infer a distance matrix using distances along the manifold.

Isomap

1. Build a sparse graph with K-nearest neighbors

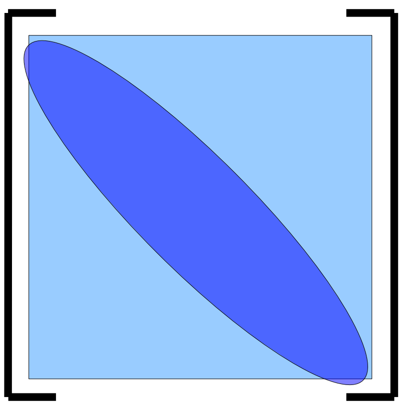
$D_g =$ 

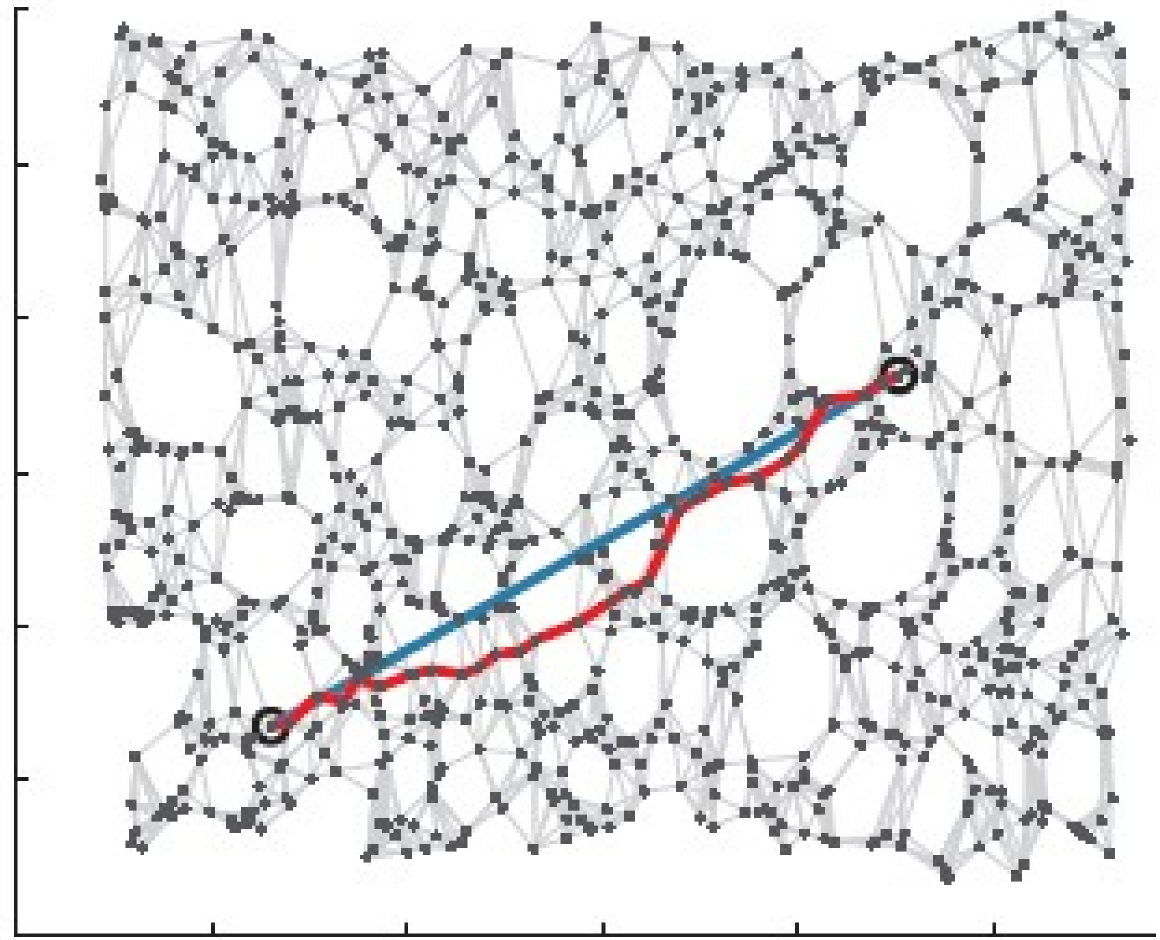


(distance matrix is sparse)

Isomap

2. Infer other interpoint distances by finding shortest paths on the graph (Dijkstra's algorithm).

$D_g =$ 



Isomap

3. Build a low-D embedded space to best preserve the complete distance matrix.

Error function:

inner product distances in graph

inner product distances in new coordinate system

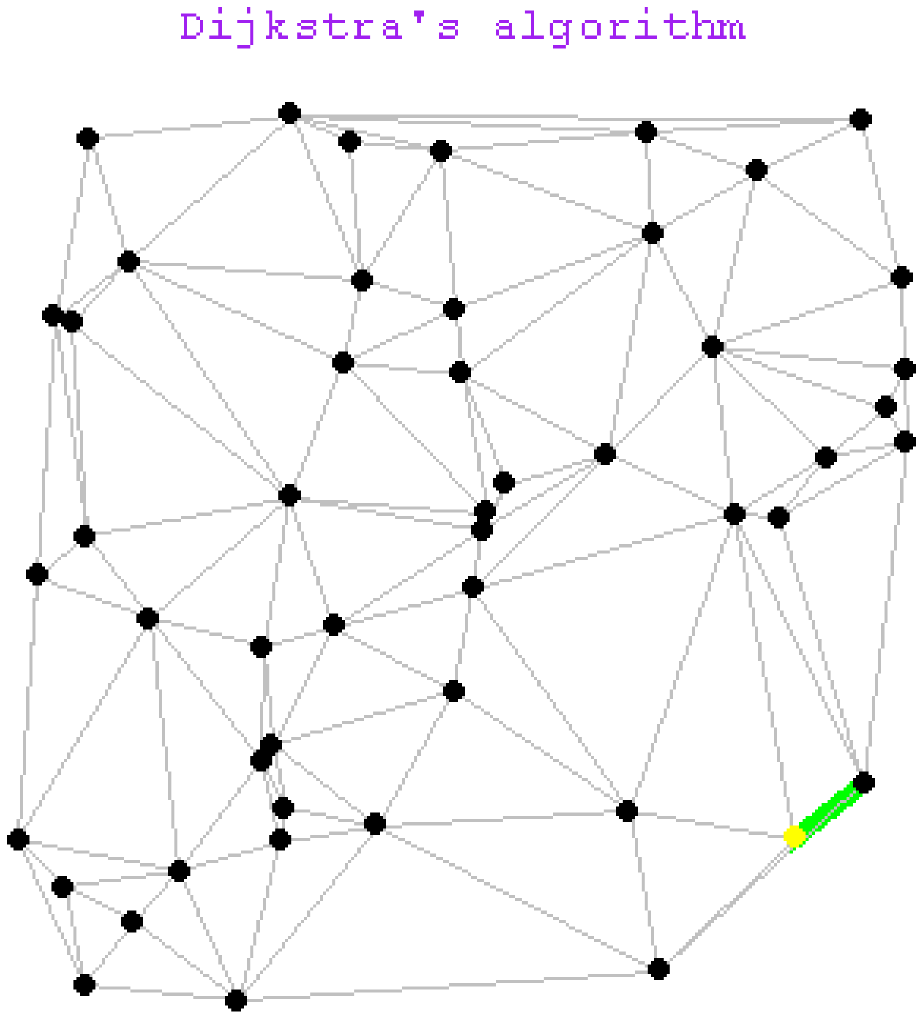
L2 norm

$$E = \|\tau(D_G) - \tau(D_Y)\|_{L^2}$$

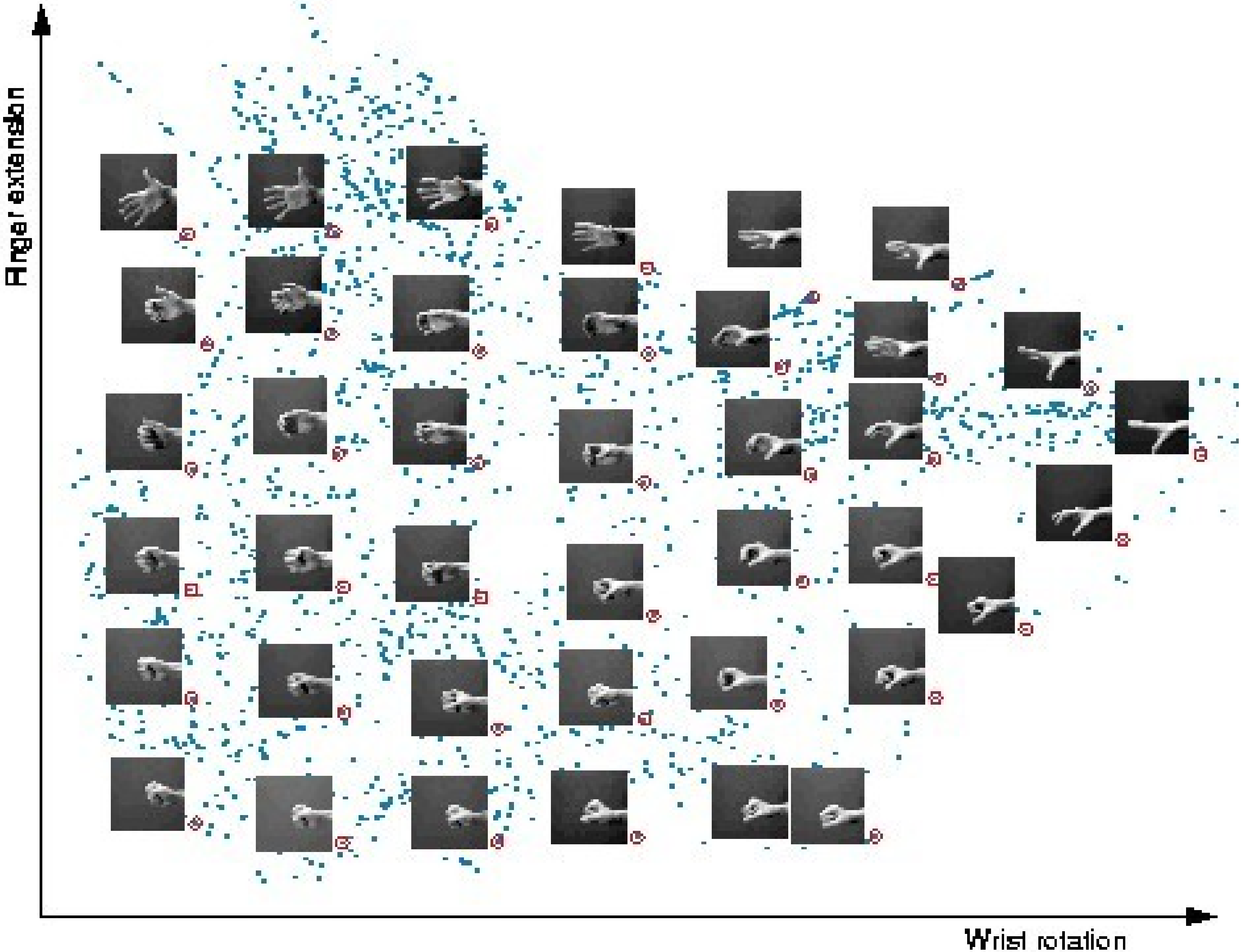
Solution – set points Y to top eigenvectors of D_g

Isomap

- shortest-distance on a graph is easy to compute



Isomap results: hands



Isomap: pro and con

- preserves global structure

- few free parameters

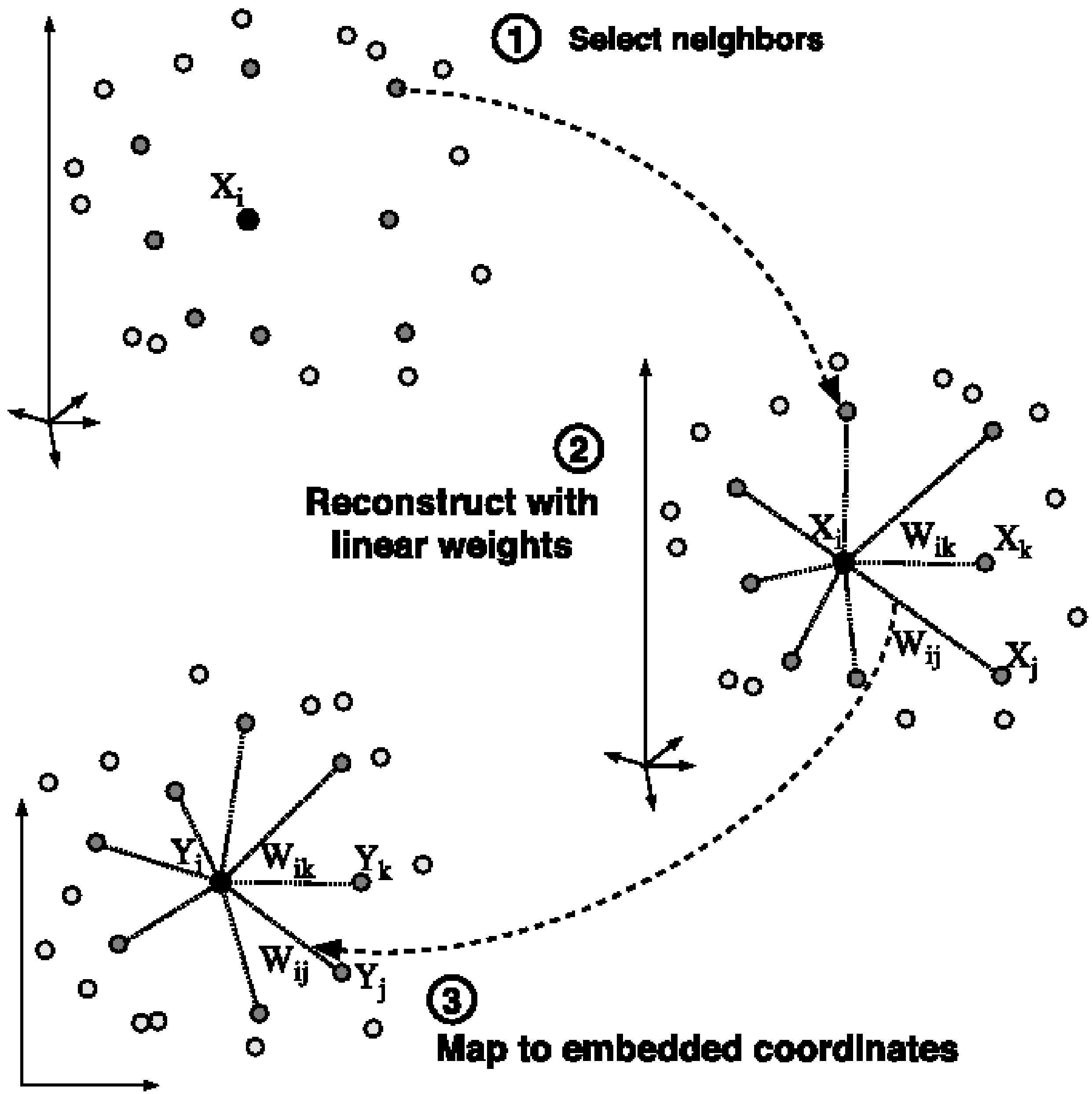
- sensitive to noise, noise edges

- computationally expensive (dense matrix eigen-reduction)

Locally Linear Embedding

Find a mapping to preserve local linear relationships between neighbors

Locally Linear Embedding



LLE: Two key steps

1. Find weight matrix W of linear coefficients:

$$\mathcal{E}(W) = \sum_i \left| \vec{X}_i - \sum_j W_{ij} \vec{X}_j \right|^2$$

Enforce sum-to-one constraint with the Lagrange Multiplier:

$$\frac{1 - \sum_{jk} C_{jk}^{-1} (\vec{x} \cdot \vec{\eta}_k)}{\sum_{jk} C_{jk}^{-1}}$$

$$w_j = \sum_k C_{jk}^{-1} (\vec{x} \cdot \vec{\eta}_k + \lambda)$$

LLE: Two key steps

2. Find projected vectors Y to minimize reconstruction error

$$\Phi(Y) = \sum_i \left| \vec{Y}_i - \sum_j W_{ij} \vec{Y}_j \right|^2$$

must solve for whole dataset simultaneously

LLE: Two key steps

$$\Phi(Y) = \sum_i \left| \vec{Y}_i - \sum\nolimits_j W_{ij} \vec{Y}_j \right|^2$$

We add constraints to prevent multiple / degenerate solutions:

$$\sum\nolimits_i \vec{Y}_i = \vec{0}$$

$$\frac{1}{N} \sum\nolimits_i \vec{Y}_i \otimes \vec{Y}_i = I$$

LLE: Two key steps

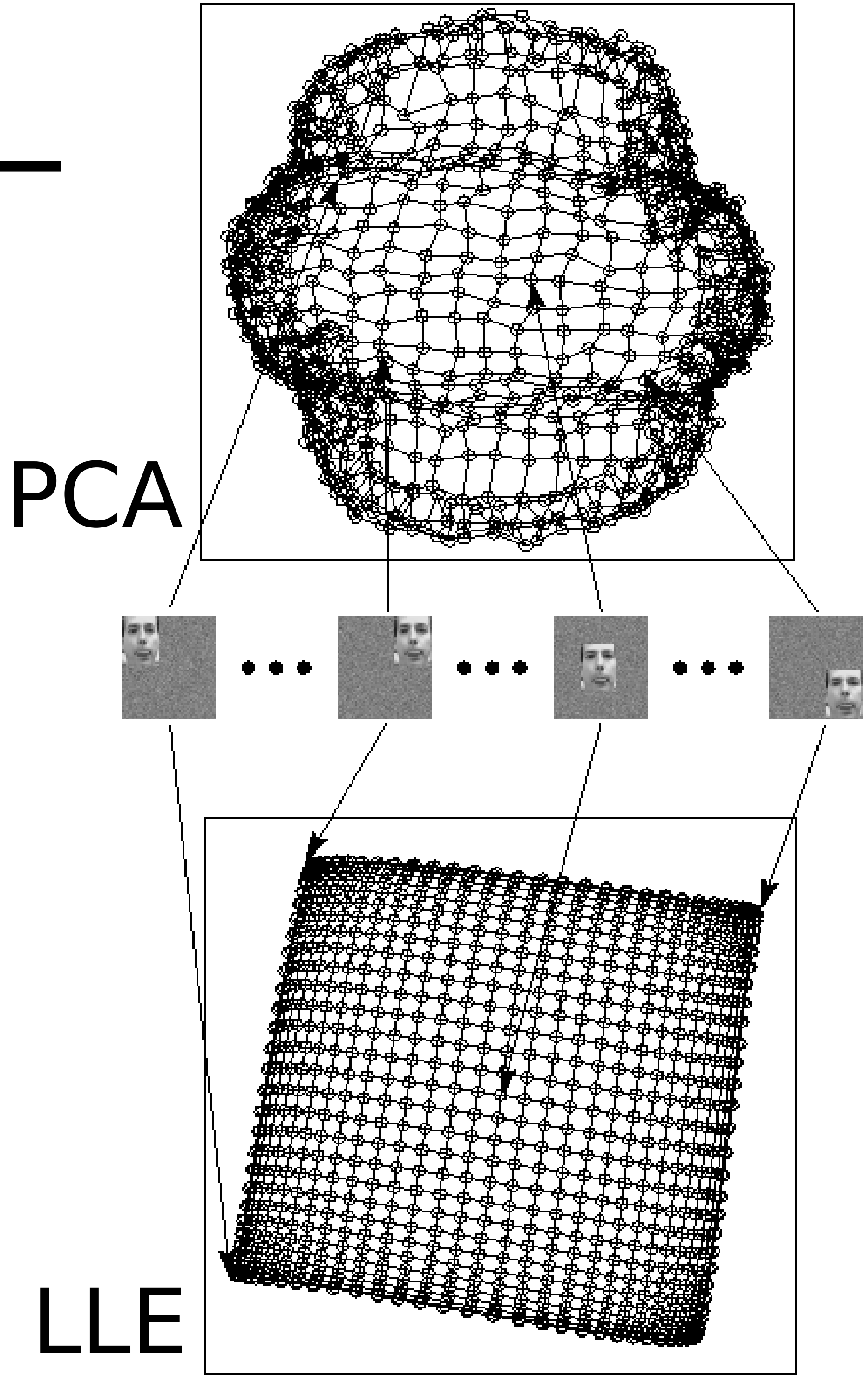
cost function becomes:

$$M_{ij} = \delta_{ij} - W_{ij} - W_{ji} + \sum_k W_{ki} W_{kj}$$

the optimal embedded coordinates are given by bottom *m*+1 eigenvectors of the matrix M

LLE: Result

preserves local topology



LLE: pro and con

- no local minima, one free parameter

- incremental & fast

- simple linear algebra operations

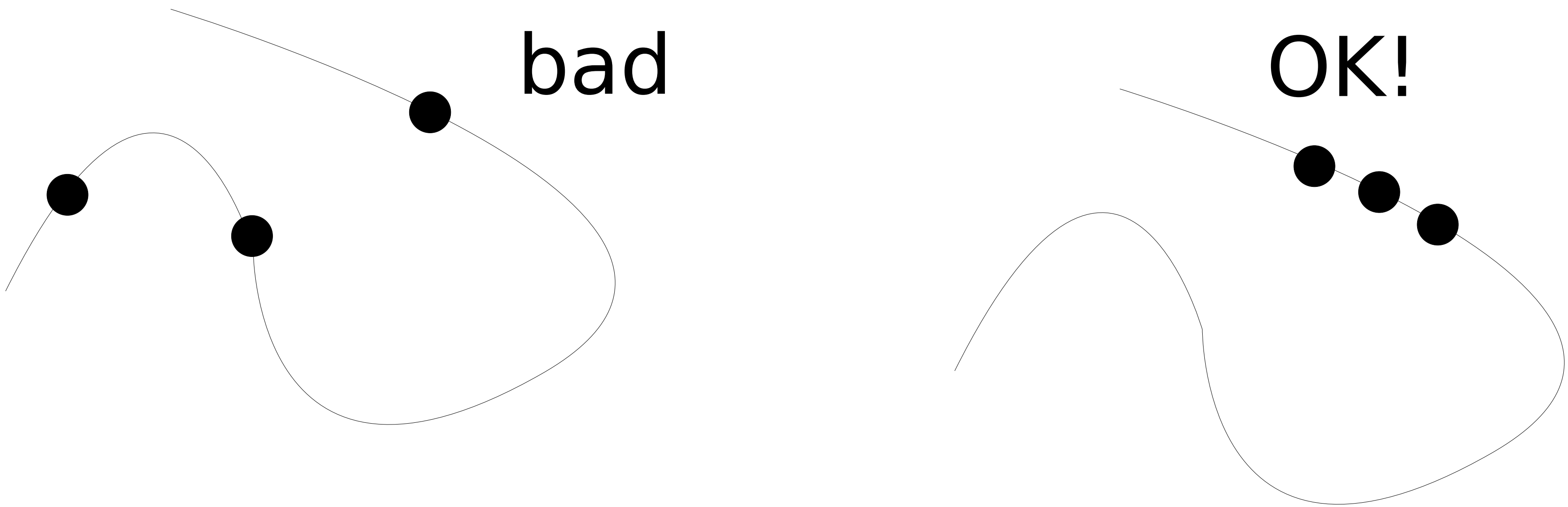
- can distort global structure

Others you may encounter

- Laplacian Eigenmaps (Belkin 2001)
  - spectral method similar to LLE
  - better preserves clusters in data
- Kernel PCA
- Kohonen Self-Organizing Map (Kohonen, 1990)
  - iterative algorithm fits a network of pre-defined connectivity
  - simple, fast for on-line learning
  - local minima
  - lacking theoretical justification

No Free Lunch

the “curvier” your manifold, the denser your data must be



conclusions

Manifold learning is a key tool in your object recognition toolbox

A formal framework for many different ad-hoc object recognition techniques