Nonlinear Manifold Learning

Part One: Background, LLE, IsoMap

6.454 Area One Seminar

October 8th 2003

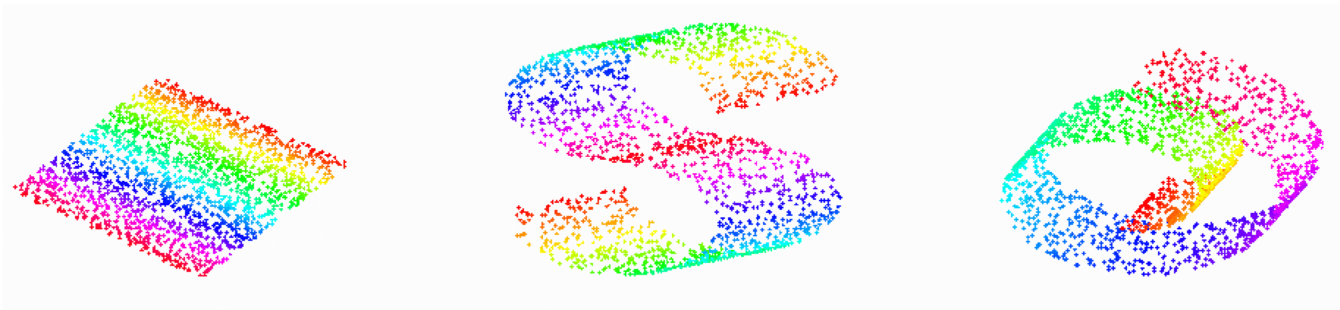
Alexander Ihler

Introduction

Motivation

- Observe high-dimensional data
- Hopefully, a low-dimensional (simple) underlying process
  - Few degrees of freedom
  - Relatively little noise (in observation space)
  - Complex (nonlinear) observation process
- Low-dim process lends structure to the high-dim data
  - how can we access that structure?
- Multivariate examples
  - Image data, spectral coefficients, word co-appearance, gene co-regulation, many more…

Introduction (cont'd)



- Three (simple) examples of manifolds
- All three are two-dim. data embedded in 3D
  - Linear, "S"-shape, "Swiss roll"
- For all three, we would like to recover:
  - That the data is only two-dimensional
  - "Consistent" locations for the data in 2D

Outline

Background

- Principal Component Analysis
- Multidimensional Scaling
  - Principal Coordinate Analysis

Locally Linear Embedding *(Roweis and Saul)*

IsoMap *(Tenenbaum, de Silva, and Langford)*

- Original version
- Landmark and Conformal versions

Comparisons

PCA I

- Principal Component Analysis
- Find linear subspace projection *P* which preserves the data locations (under quadratic error)
- Equivalent: find linear subspace projection *P* which leaves largest variance for *PX*

$$
\begin{aligned}
\min & E[\|XJ - PXJ\|^2] \\
&= E[XJJ'X' - 2PXJJ'X - PXJJ'X'P'] \\
&= const - E[PXJJ'X'P]
\end{aligned}
$$

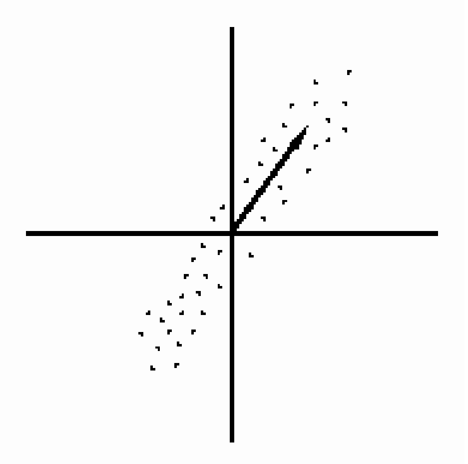
- *J* is the "centering matrix" (*XJ* is zero-mean)
- Simple eigenvector solution

PCA II

- Eigenvectors =

  directions of principal variation

$$\max_P PXJJ'X'P'$$



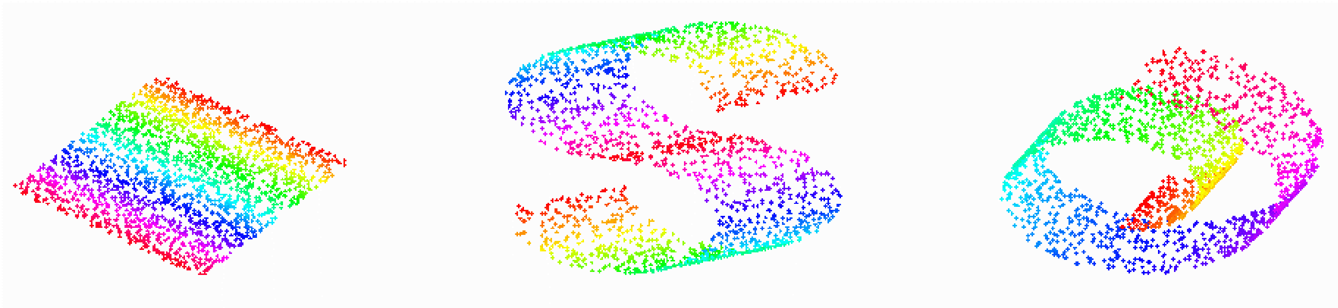
- Top q eigenvectors of $XJJ'X'$

$$V_q = [v_1, \ldots, v_q]$$

  is a basis for the q-dim subspace

- Locations given by $V_q' XJ$

Manifolds



(a) (b) (c)

- PCA : works for (a)
- Doesn't do much good for (b) or (c)
  - Linear subspace doesn't explain it well
- What do we mean by "consistent locations"?
  - Preserve local relationships and structure
  - One possibility : preserve distances

Multidimensional Scaling

- Multidimensional scaling (MDS)
  - Given “pre-distances” D (possibly non-Euclidean)
  - Find Euclidean q-dim space which preserves those relationships
  - We’ll just concentrate on Euclidean pre-distances; (possibly unknown) locations X in p-dim space
- “preserves” : use $\widehat{D}$ = distance in the q-dim space
  - Need to define a cost function
  - STRAIN $\rho(D, \widehat{D}) = \|J'(D^2 - \widehat{D}^2)J\|_F^2$
  - STRESS $\rho(D, \widehat{D}) = \|D - \widehat{D}\|_F^2$
  - SSTRESS $\rho(D, \widehat{D}) = \|D^2 - \widehat{D}^2\|_F^2$

Classical MDS

- STRAIN : $\rho(D, \hat{D}) = \|J'(D^2 - \hat{D}^2)J\|_F^2$
  - Solution is given by the eigenstructure of

$$-\frac{1}{2}J'D^2J = J'X'XJ$$

  - Top q eigenvectors $W_q = [w_1, \ldots, w_q]$

    give locations $Y = W_q'$

  - This is *exactly* the same solution as PCA:

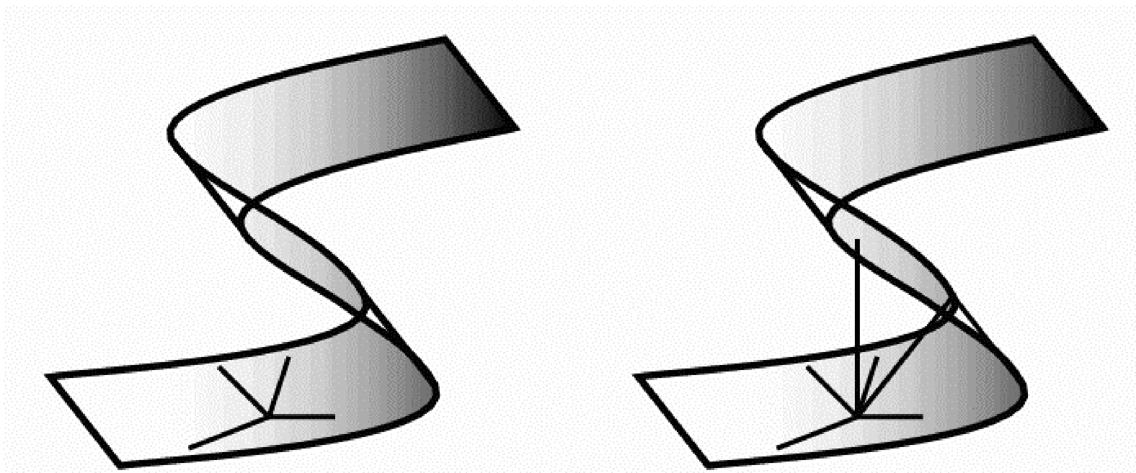
$$XJJ'X'v_i = \lambda_i v_i \Rightarrow J'X'XJ(J'X'v_i) = \lambda_i(J'X'v_i)$$

$$\Rightarrow \qquad Y = W_q' = V_q'XJ$$

  - So, we didn't really get anywhere?

“Local” relationships

- MDS – still produced a linear embedding – why?
    - Preserved *all* pairwise distances
- Let’s look at one of our examples:



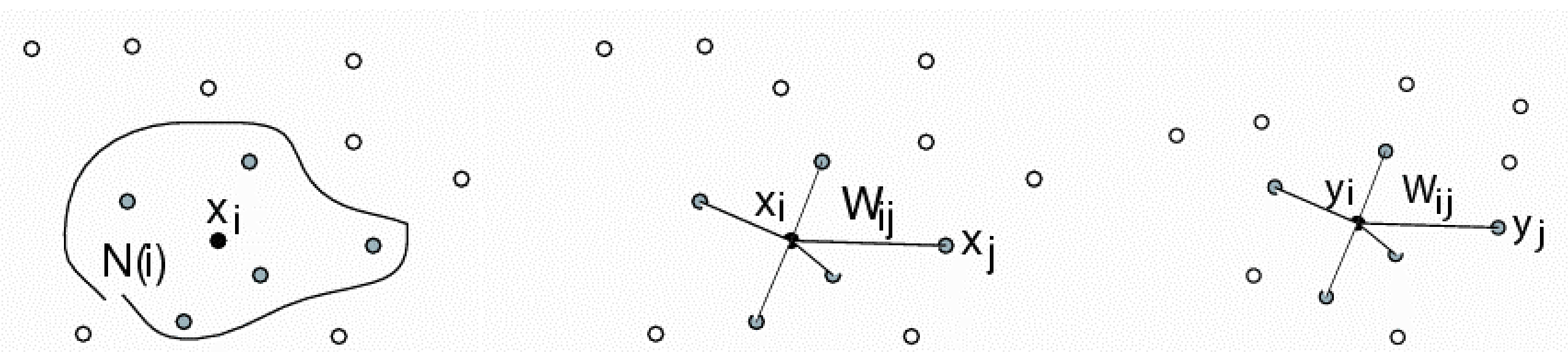
- Nonlinear manifold:
    - local distances (a) make sense
    - but, global distances (b) don’t respect the geometry

“Local” relationships

- Two solutions which preserve local structure:
- Locally Linear Embedding (LLE)
  - Change to a local representation (at each point)
  - Base the local rep. on position of neighboring points
- IsoMap
  - Estimate actual (geodesic) distances in p-dim. space
  - Find q-dim representation preserving those distances
- Both rely on the locally flat nature of the manifold
  - How do we find a locality in which this is true?
  - (At least) two possibilities
    - *k*-nearest-neighbors
    - ε-ball

Locally Linear Embedding

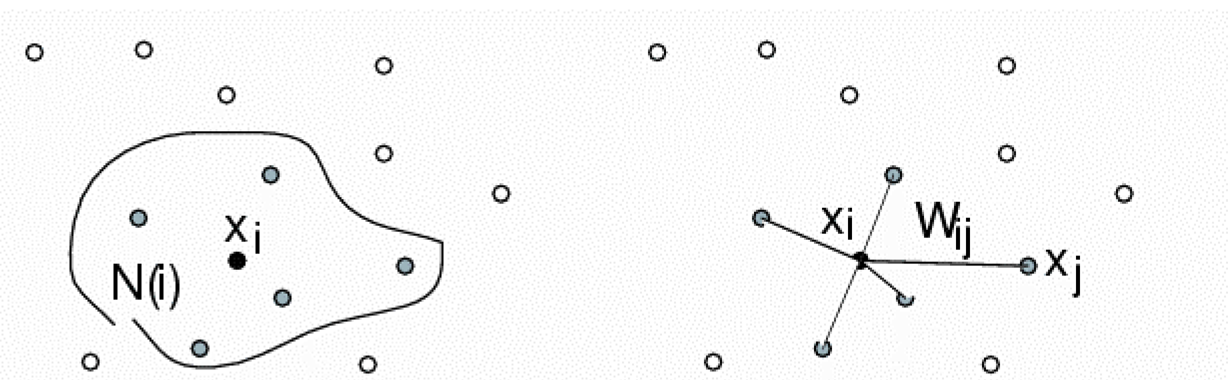
- Overview
  - Select a local neighborhood
  - Change each point into a coordinate system based on its neighbors
  - Find new (q-dim) coordinates which reproduce these local relationships



Locally Linear Embedding

$$\tilde{W} = \arg\min_{W} \|x_i - \sum_{j \in \Gamma(i)} W_{ij} x_j\|^2 \quad \text{s.t.} \quad \forall_i \sum_j W_{ij} = 1$$

- This has several nice properties
  - Invariant to (local) rotation of all points in $\Gamma(i)$
  - Invariant to (local) scale…
  - Invariant to (local) translations (due to norm. of W)



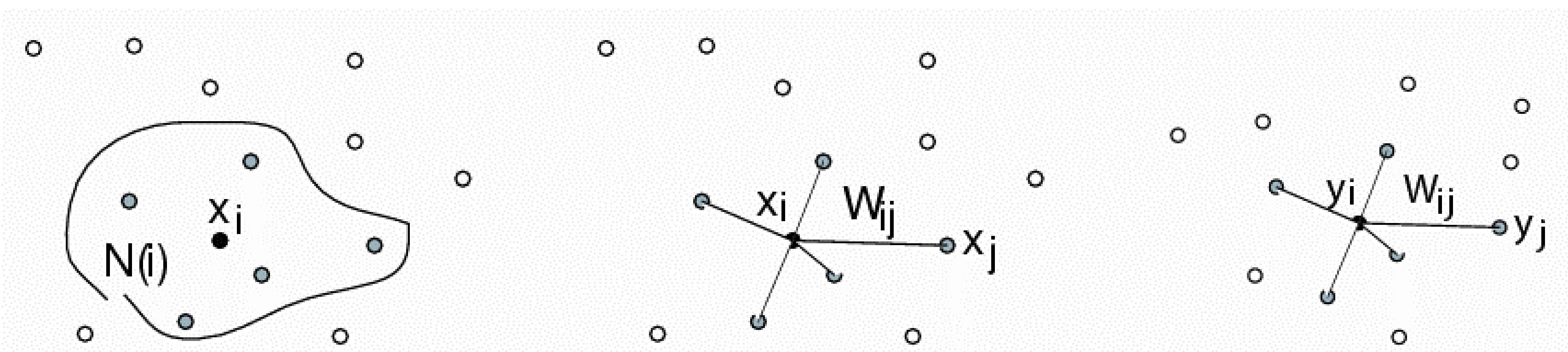
Locally Linear Embedding

Find new (q-dim) coordinates which reproduce these local coordinates

$$\tilde{Y} = \arg\min_{Y} \|y_i - \sum_{j \in \Gamma(i)} W_{ij} y_j\|^2 \quad \text{s.t.} \quad Y\mathbf{1} = 0, \quad YY' = I$$

Or, as the quadratic form

$$\arg\min_{Y} Y'(I - \tilde{W})'(I - \tilde{W})Y$$



Locally Linear Embedding

Find new (q-dim) coordinates which reproduce these local coordinates

$$\tilde{Y} = \arg\min_{Y} \|y_i - \sum_{j \in \Gamma(i)} W_{ij} y_j\|^2 \quad \text{s.t.} \quad Y\mathbf{1} = 0, \quad YY' = I$$

Or, as the quadratic form

$$\arg\min_{Y} Y'(I - \tilde{W})'(I - \tilde{W})Y$$

This can be solved using the eigenstructure as well:

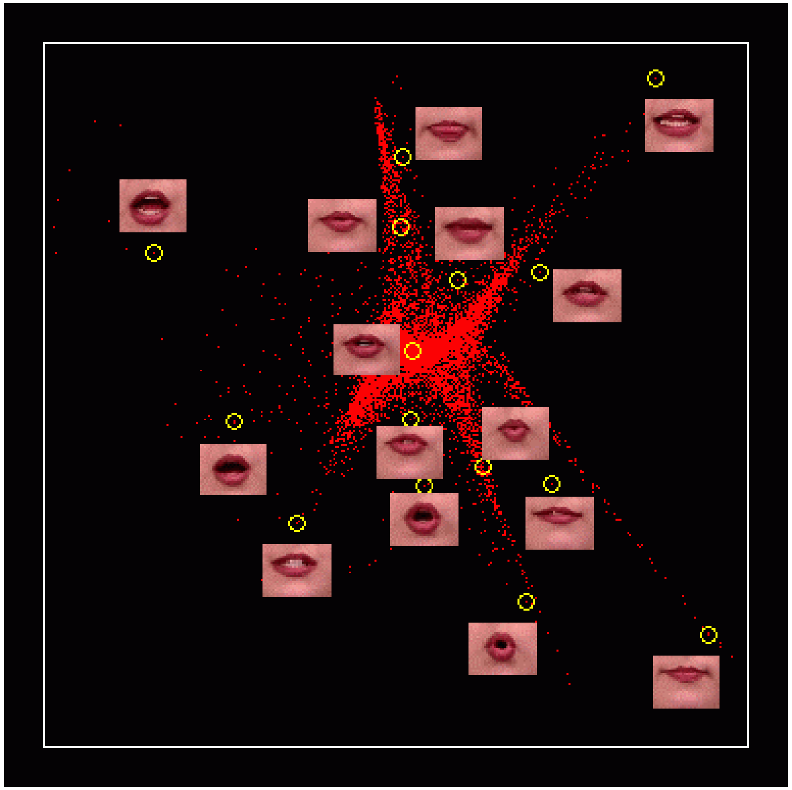
We want the min. variance directions of $(I - \tilde{W})'(I - \tilde{W})$

1 is an eigenvector with eigenvalue 0 (translational invar);

The next *q* smallest eigenvectors form the coordinates *Y*

Application

- Does it work?
  - Yes, often
  - When does it fail? Hard to answer this…
- Another method (IsoMap) will be easier to analyze
  - Makes a clear set of assumptions
  - Will help quantify what LLE lacks…



(From LLE homepage)

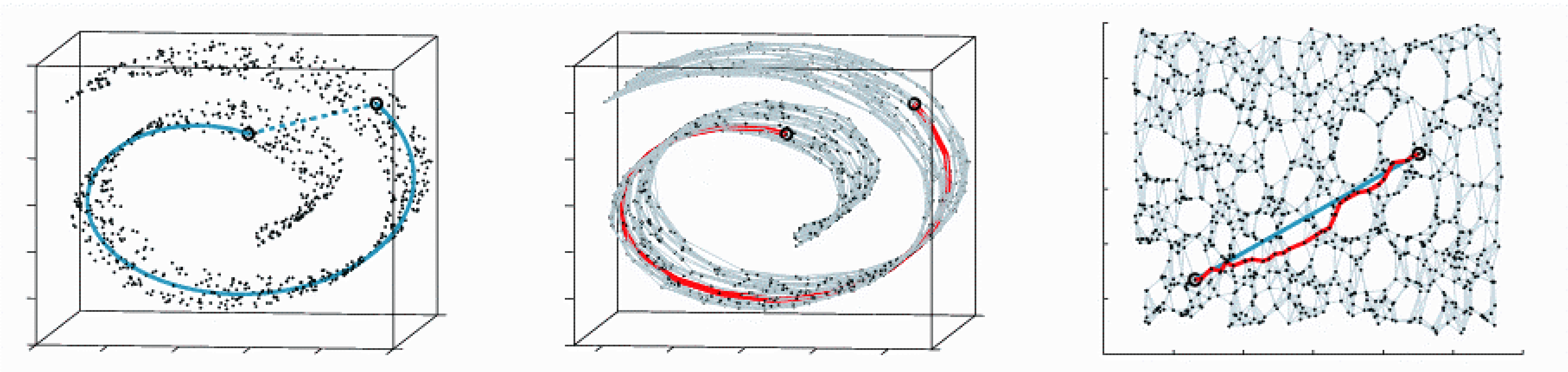
IsoMap

- Recall classical MDS (principal coordinate analysis)
  - Given a set of (all) distance measurements
  - Finds optimal Euclidean-distance reconstruction

    (assuming cost criterion ρ)
- What we really want:
  - Find distance measurements along manifold (geodesics)
  - Find low-dim reconstruction which also has these geodesic distances
- Under certain conditions, we can obtain this from MDS!
  - Need low-dim geodesics = low-dim Euclidean dist.

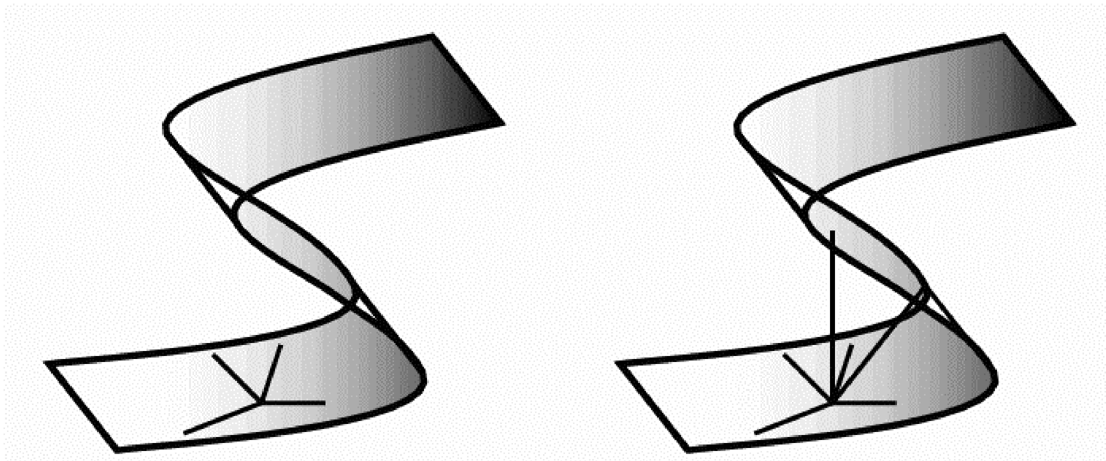
IsoMap

- Overview
  - Select a local neighborhood
  - Find estimated geodesic distances between all pairs in *X*
  - use classical MDS to find the best *q*-dim. space with these (Euclidean) distances



IsoMap

Find estimated geodesic distances between all pairs in X:



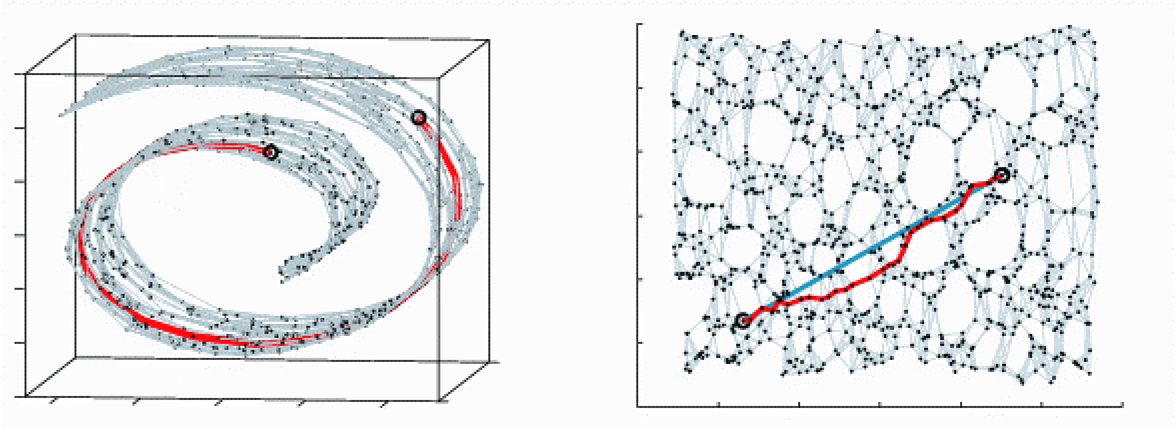
Keep local distances (close to geodesic)

Discard far distances

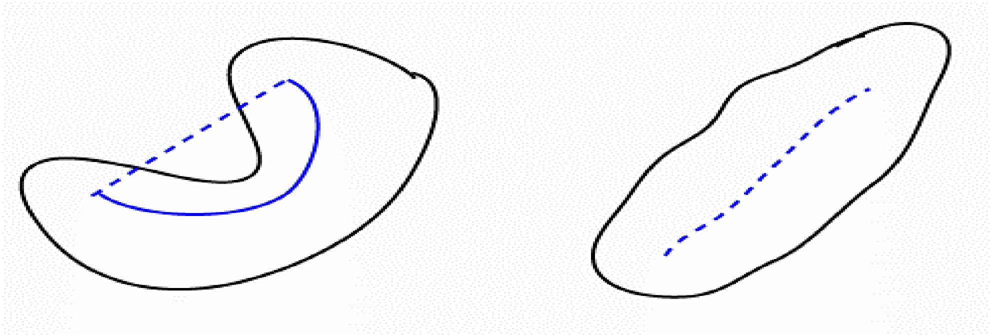
For far points, we can approximate the geodesic by the shortest path along retained distances: (found e.g. via dynamic programming)

IsoMap

Use classical MDS to find an equivalent low-dim Euclidean space



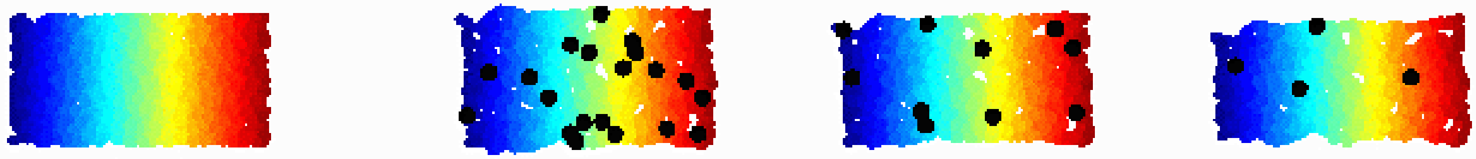
If the true data comes from a convex set of **R^q** this will recover the true geometry (since geodesic length = Euclidean distance); otherwise it will introduce distortions



IsoMap

Landmark Points to improve efficiency

- Naïve implementation of IsoMap
  - Shortest Path – $O(n^3)$ (slightly less)
  - Find eigenvectors – $O(n^3)$
- Use only a subset of points (m) for transformation
  - Shortest path – $< O(m\ n^2)$
  - Eigenvectors – $O(m^2\ n)$



Original points and reconstruction using landmark points (black)

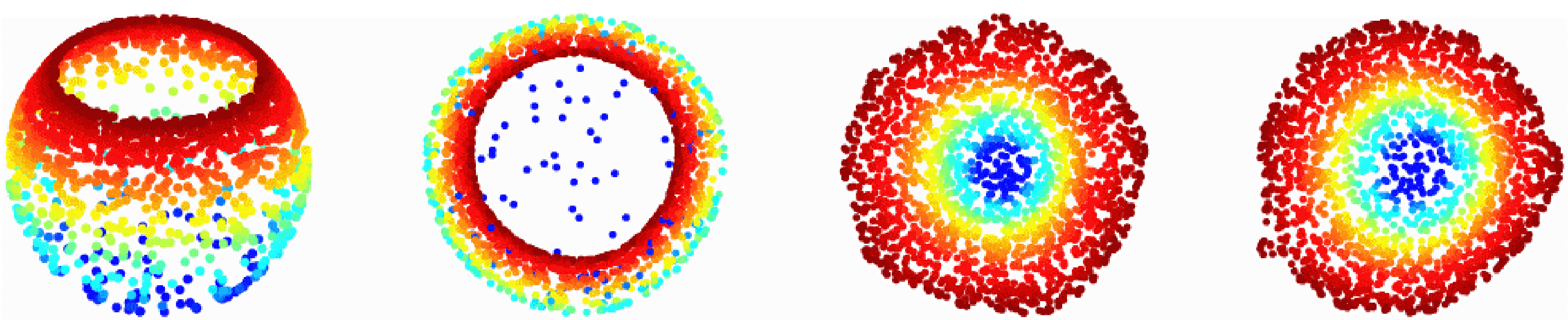
Conformal IsoMap

Extend to non-isomorphic mappings

- Conformal mappings: preserve orientation but not distance; distance can warp (locally)

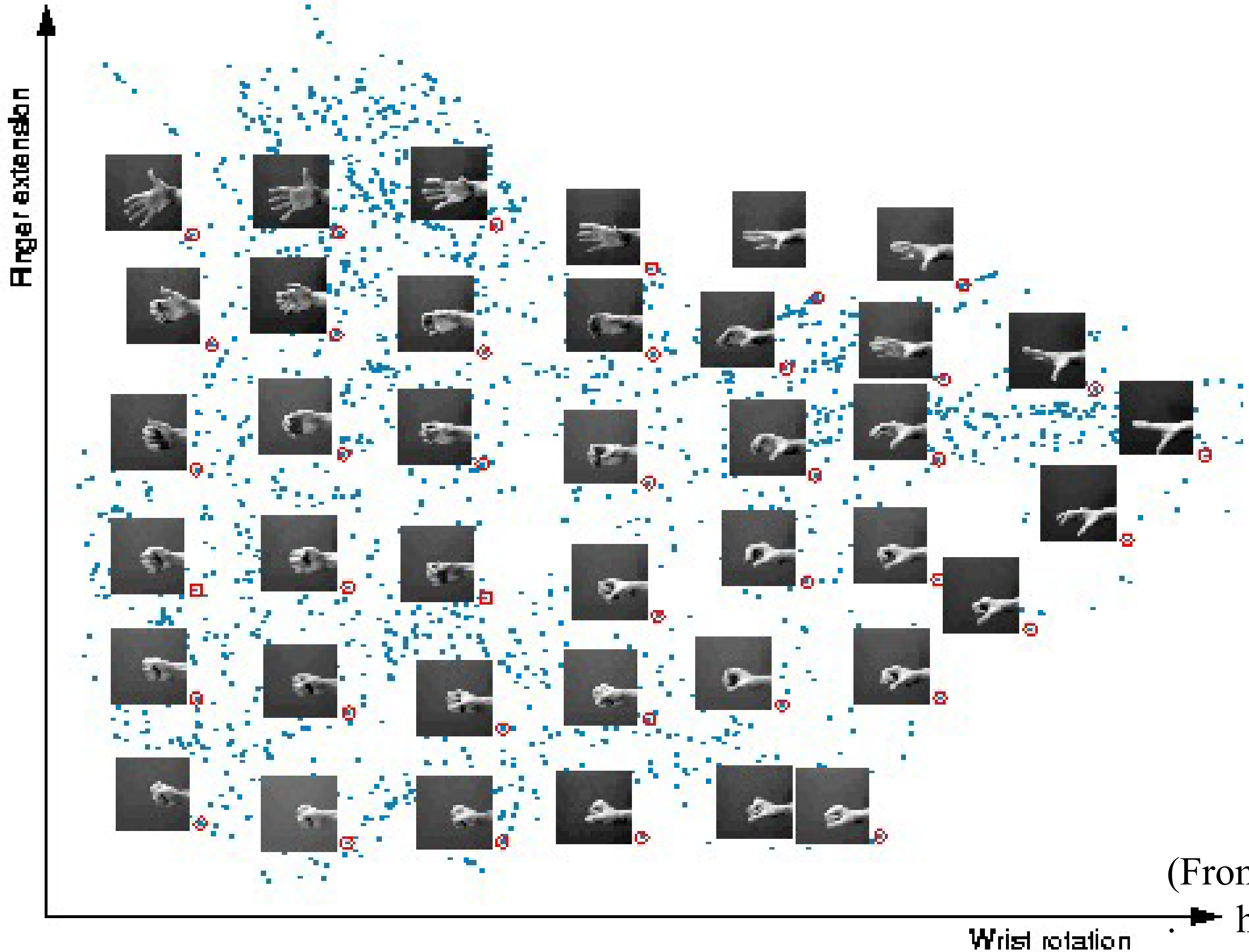
    (LLE already tries to allow for this)

- Example: fishbowl – no isomorphic map to plane
- Solution: a different assumption
    - Assume that data is uniformly distributed in low-dimensional space
    - Use distribution to estimate local distance warp



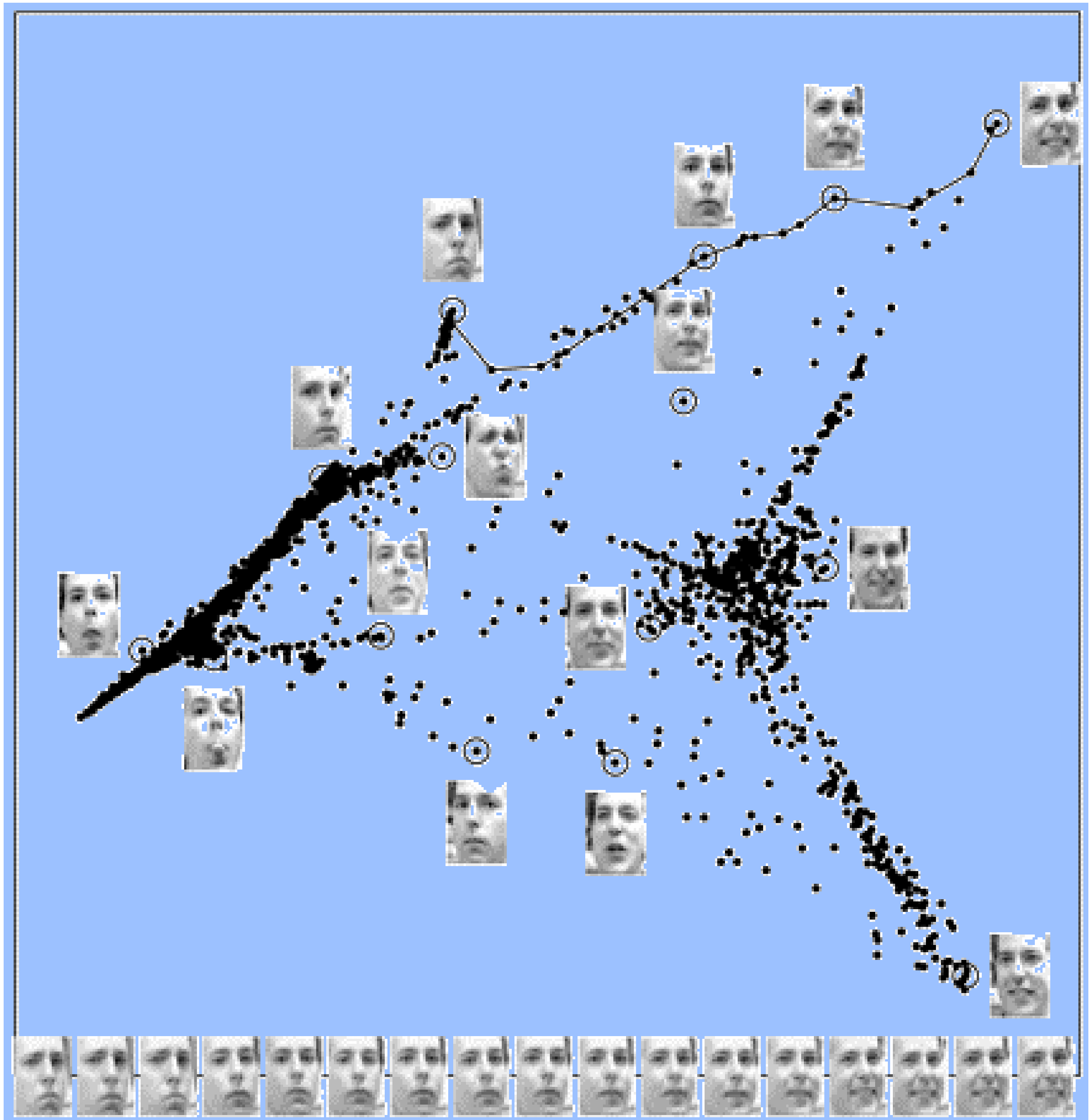
3D data *IsoMap* *Conformal IsoMap* *LLE*

Examples



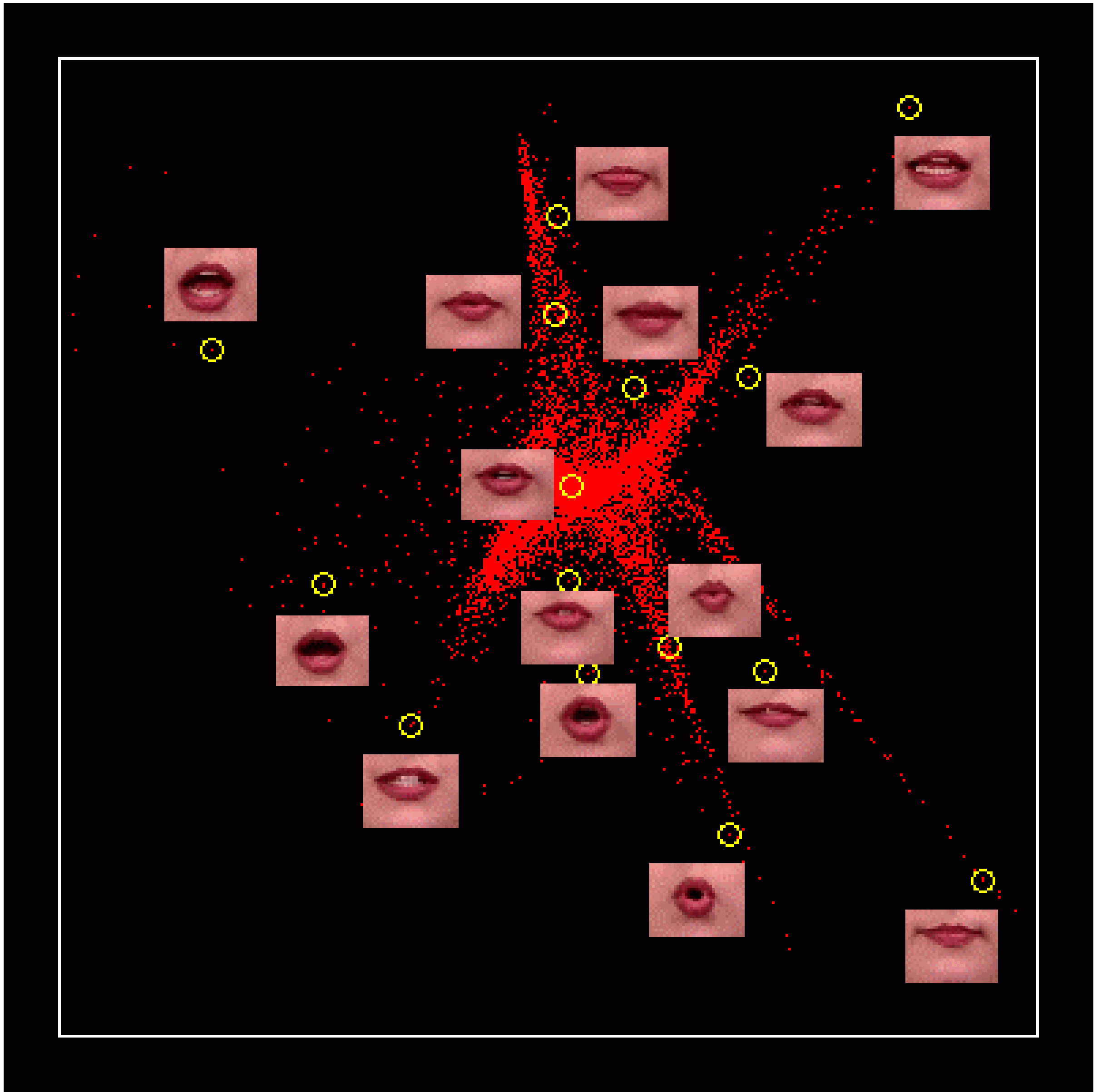
(From IsoMap homepage)

Examples



(From LLE homepage)

Examples



(From LLE homepage)

Examples

(From LLE homepage)

Difficulties

IsoMap

- When assumptions are violated:
  - Non-convex sets in $\mathbf{R}^q$
  - Non-isomorphic mappings (standard version)
  - Non-uniform distributions (conformal version)

LLE

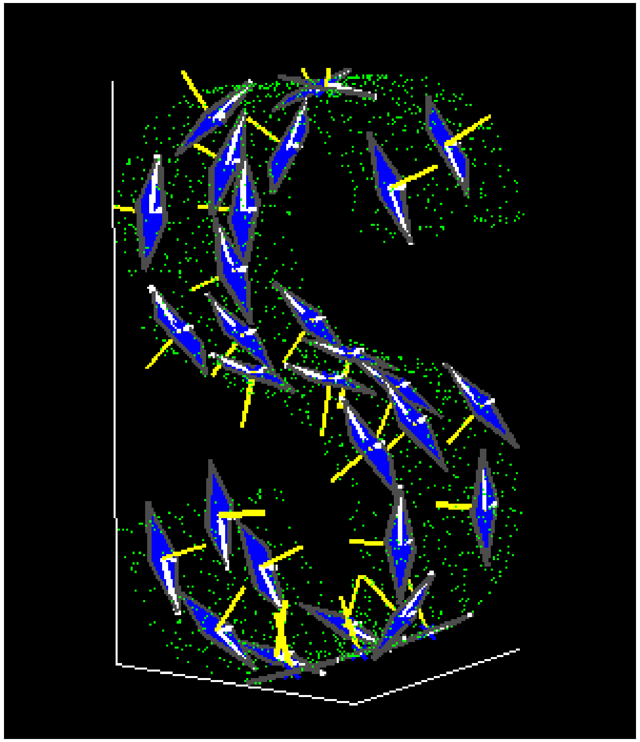
- Much more difficult to say…
  - No requirement that faraway points stay far
  - Susceptible to “folding”
  - Can see “spider-web” like behavior
  - Hard to tell if this is an artifact or not…

More recent work

- Lots of "LLE-like" solutions that try to fix this:
  - Penalties to align multiple local coordinate systems
  - Adding ideas from (and for) density estimation
- Next week…
- Also: finding mappings

  X to Y, Y to X

  - Supervised learning
  - Re-solve optimization



(From LLE homepage)