CS434b/ 654b: Pattern Recognition
Prof. Olga Veksler

Lecture 16

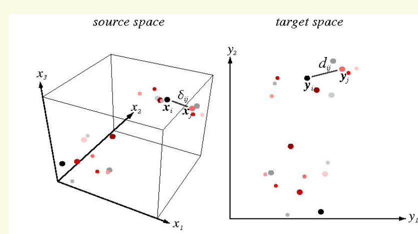
Low-dimensional Representations

- Humans are good at analyzing data in 2D or 3D
- Most datasets scientists have to deal with are multidimensional
- It would help if we could visualize structure of the data in 2D or 3D
- Although data is usually presented is in high dimensions, intrinsic dimension is much lower
  - for faces, it is estimated that there are 30 intrinsic dimensions

Today

- Low-dimensional Representations of high dimensional data
  - MDS (multidimensional scaling)
  - Isomap
  - LLE (locally linear embedding)
  - Kohonen Maps

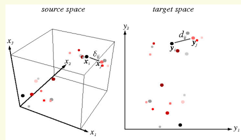
Multidimensional Scaling (MDS)



- Multidimensional Scaling
  - find a configuration of points in a low dimensional space whose interpoint distances correspond to similarities (dissimilarities) in higher dimensions

Multidimensional Scaling (MDS)

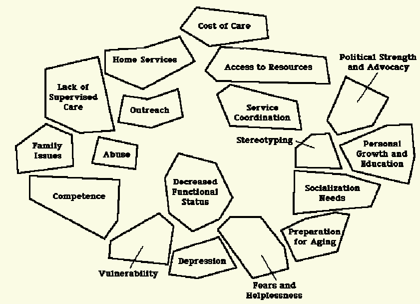
- Given:
  - points $x_1,\ldots,x_n$ in k dimensions
  - distance between points x_i and x_j is δ_{ij}



- Find
  - points $y_1,\ldots,y_n$ in 2 (or 3) dimensions s.t. distance d_{ij}, the distance between y_i and y_j is close to δ_{ij}
- In general, it's not possible to find lower dimensional representation s.t. $d_{ij} = \delta_{ij}$
- Can look for δ_{ij} which minimize an objective function

Multidimensional Scaling

- Document Clustering



Example from John Canny

Multidimensional Scaling

- Possible objective function:

$$J_{ee}(y) = \frac{\sum_{i<j} (d_{ij} - \delta_{ij})^2}{\sum_{i<j} \delta_{ij}^2}$$

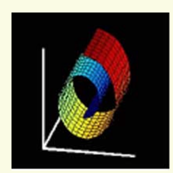
- Not trivial to optimize, have to use gradient descent

$$\nabla_{y_k} J_{ee}(\delta) = \frac{2}{\sum_{i<j} \delta_{ij}^2} \sum_{j \neq k} (d_{kj} - \delta_{kj}) \frac{(y_k - y_j)}{d_{kj}}$$

- Good initialization choice
  - Select the 2 (or 3) coordinates of $x_1,\ldots,x_n$ which have the largest variance

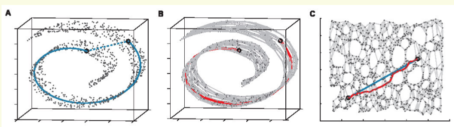
Multidimensional Scaling

- MDS is equivalent to PCA under Eucledian distance
  - Fails for nonlinear data
- Often data lies on a low dimensional manifold in a high dimensions
  - manifold is locally "flat"
  - For example, the earth (sphere) is locally flat, that's why in ancient times people believed that the earth is flat



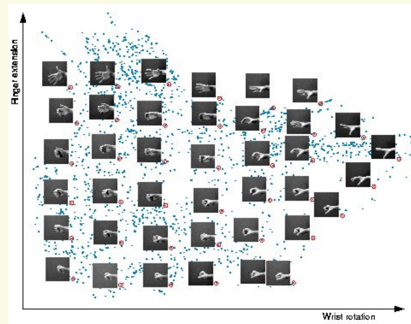
Isomap

- Josh. Tenenbaum, Vin de Silva, John Langford 2000
- Algorithm for nonlinear dimensionality reduction, works well for some types of manifolds
- Idea: instead of measuring Euclidean distance between points, measure the distance along the inherent geometric surface

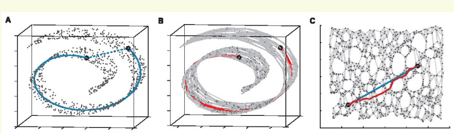


Isomap

- Two-dimensional embedding of hand images (from Josh. Tenenbaum, Vin de Silva, John Langford 2000)



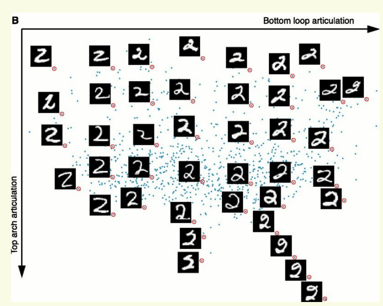
Isomap



- Construct a graph by connecting each data point to its k (***7*** in this example) nearest neighbors.
- Measure the distance between any 2 samples as the shortest path in the graph between these 2 samples
- After all pairwise distances are computed, use MDS or any other linear dimensionality reduction method

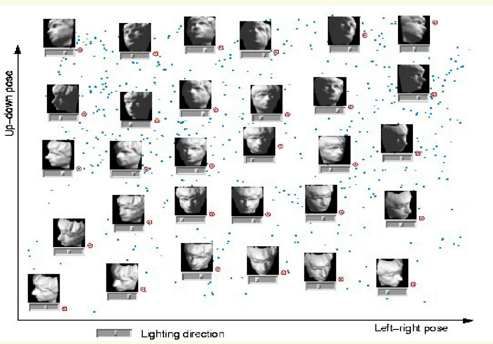
Isomap

- two-dimensional embedding of hand-written ‘2’ (from Josh. Tenenbaum, Vin de Silva, John Langford 2000)



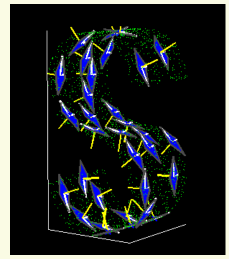
Isomap

- three-dimensional embedding of faces (from Josh. Tenenbaum, Vin de Silva, John Langford 2000)



Locally Linear Embedding (LLE)

- S. Roweis and L.K. Saul, 2000
- Assume that data on a manifold
  - That is each sample and its neighbors lie on approximately linear subspace
- Idea:
  1. approximate data by a bunch of linear patches
  2. Glue these patches together on a low dimensional subspace s.t. neighborhood relationships between patches are preserved. This step is done by global optimization.

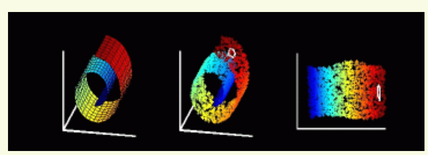


Isomap

- Advantages:
  - Works for nonlinear data
  - Preserves the global data structure
  - Performs global optimization
- Disadvantages
  - Works best for swiss-roll type of structures
  - Not stable, sensitive to “noise” examples
  - Computationally very expensive

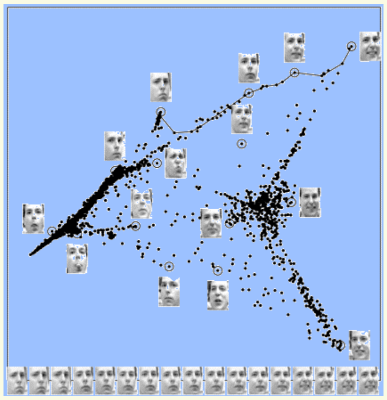
Locally Linear Embedding (LLE)

- S. Roweis and L.K. Saul, 2000



- This is similar to flattening out the map of the earth on a globe into a flat map

LLE: Face expressions

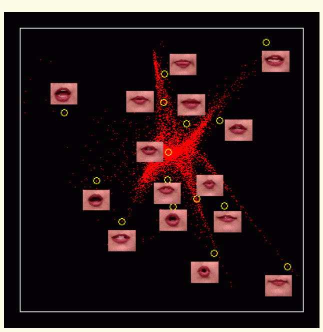


From S. Roweis and L.K. Saul, 2000

Isomap vs. LLE

Tenenbaum: “Our approach [Isomap], based on estimating and preserving global geometry, may distort the local structure of the data. Their technique [LLE], based only on local geometry, may distort the global structure," he said.

LLE: Face expressions



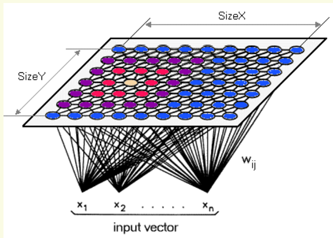
From S. Roweis and L.K. Saul, 2000

Kohonen Self-Organizing Maps

- The goal, again, is to map samples to a lower dimensional space s.t. inter-sample distances are preserved as much as possible
- Kohonen maps produce a mapping from multidimensional input onto a 1D or 2D grid of nodes (neurons)
- This mapping is topology preserving, that is similar samples are mapped to nearby neurons
- Kohonen maps learn without teacher
- Kohonen maps have connection to biology
  - Similar perception input lead to excitation in nearby parts of the brain

Kohonen Self-Organizing Maps (SOM)

- Interconnected structure of units (neurons) which compete for the signal. Usually neurons arranged on 1D or 2D grid



- SOM algorithm learns a mapping from input samples to the 2D (1D) grid of neurons
- Each neuron is represented by weights w_{ij}, the number of weights = dimensionality of an input sample

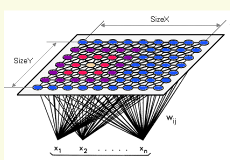
Kohonen SOM World Poverty Map

- Example from Helsinki University of Technology Finland
- World Bank statistics of countries in 1992
  - 39 features describing various quality-of-life factors, such as state of health, nutrition, educational services
  - countries that had similar values of the indicators found a place near each other on the map
  - different clusters on the map were automatically encoded with different bright colors, nevertheless so that colors change smoothly on the map display
  - As a result of this process, each country was in fact automatically assigned a color describing its poverty type in relation to other countries
  - The poverty structures of the world can then be visualized in a straightforward manner: each country on the geographic map has been colored according to its poverty type.

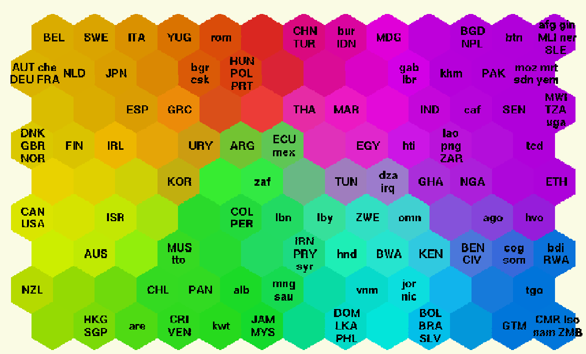
Kohonen Self-Organizing Maps (SOM)

- Training
  - Repeat steps 1,2,3 until convergence or maximum number of iterations

  1. Select sample x_i
  2. Find the neuron n closest to x_i (i.e. the distance between x_i and the neuron weights w_{ij} is minimum
  3. Adjust the weight of n and the weights of neurons around n so that they move even closer to sample x_i
     - The neighborhood size is initially large, but shrinks with time



Kohonen SOM World Poverty Map



Kohonen SOM World Poverty Map