Analysis of Non-linear Dimensionality Reduction Techniques

Debabrota Basu
Department of Computer Science
School of Computing, National University of Singapore

Overview

Motivation

Problem Description

State-of-art techniques:

- Isomap
- Local Linear Embedding (LLE)
- Laplacian Eigenmap

Limitations:

- Non-convexity
- Complex geometry
- Noise

Discussion: Part I

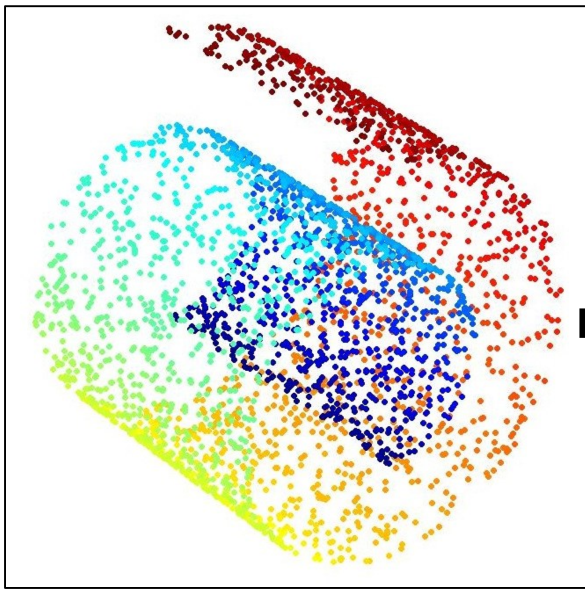
Curvature Adaptive LLE

- Theory
- Experiments

Discussion: Part II

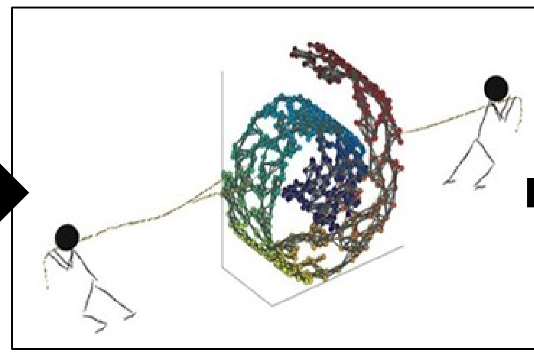
MOTIVATION

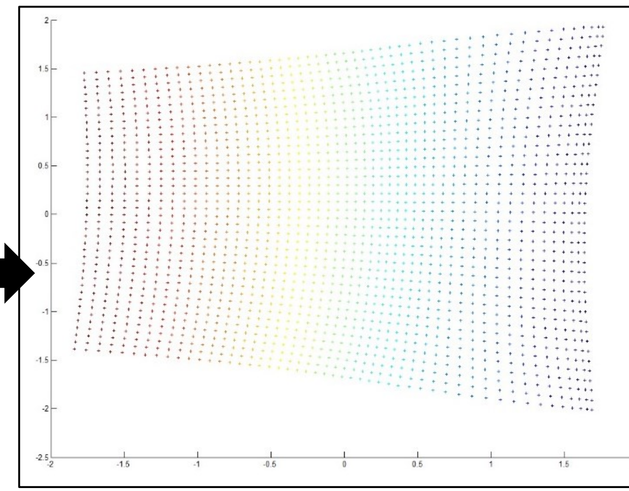
Why Dimensionality Reduction?



3-dimensional data points
on Swiss roll

Dimensionality Reducing
Algorithm

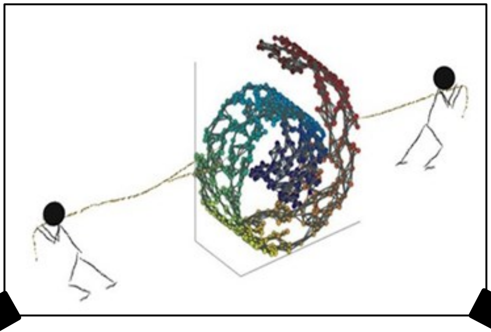




2-dimensional embedding
of Swiss roll

Dimensionality Reduction Techniques

Dimensionality Reducing
Algorithm



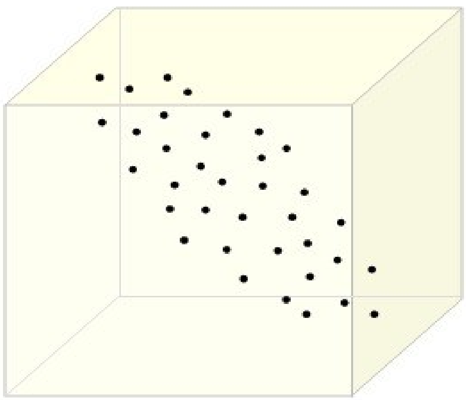
Linear Methods

- PCA
- MDS

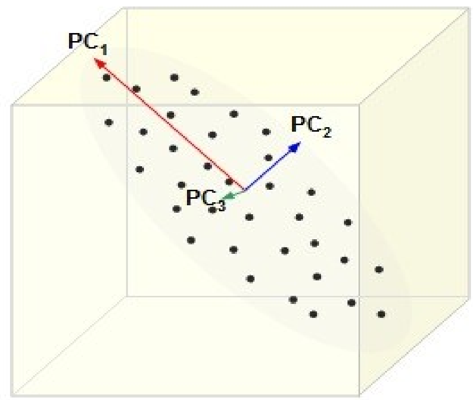
Non-linear Methods

- Isomap
- LLE
- Laplacian Eigenmap

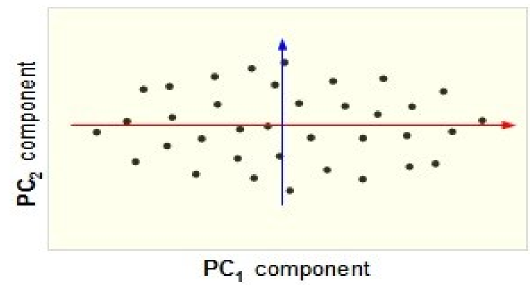
Why Non-linear Dimensionality Reduction?



Original data points

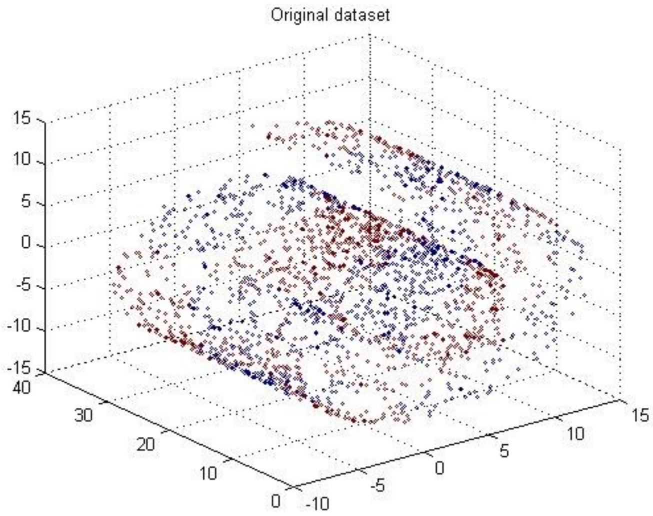


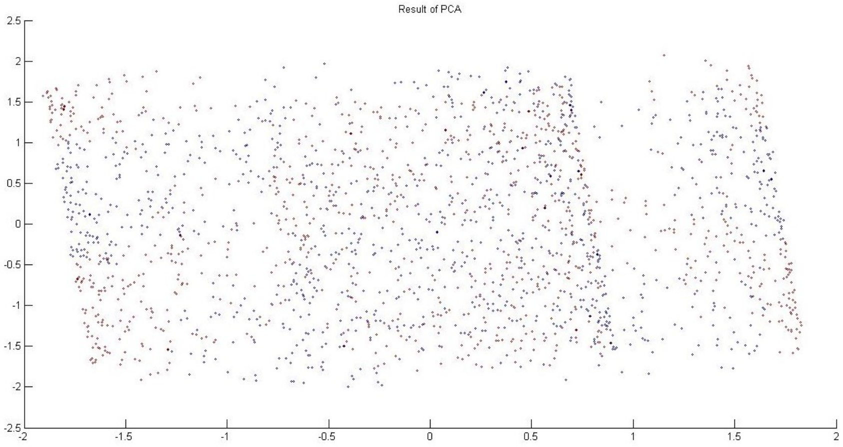
Principal Components



Low dimensional embedding

Why Non-linear Dimensionality Reduction?





PROBLEM DESCRIPTION

Non-linear dimensionality reduction (NLDR)

NLDR

Input

- Collection of *N* data points in $\mathbb{R}$

$$\mathbf{X} = \left[\overrightarrow{x_1}, \overrightarrow{x_2}, \ldots\ldots, \overrightarrow{x_N}\right]$$

Axiom

- Data points are sampled from some underlying non-linear manifold given by

$$\overrightarrow{\mathrm{x}_{\mathrm{i}}} = f\left(\overrightarrow{\mathrm{y}_{\mathrm{i}}}\right) + \overrightarrow{\mathrm{u}_{\mathrm{i}}} \qquad \forall\, i = 1, \ldots\ldots, N$$

Output

- Find *d*(<< *D*) dimensional embedding $\mathbf{Y} = \left[\overrightarrow{y_1}, \overrightarrow{y_2}, \ldots\ldots, \overrightarrow{y_N}\right]$ that gives a good approximation of *f*

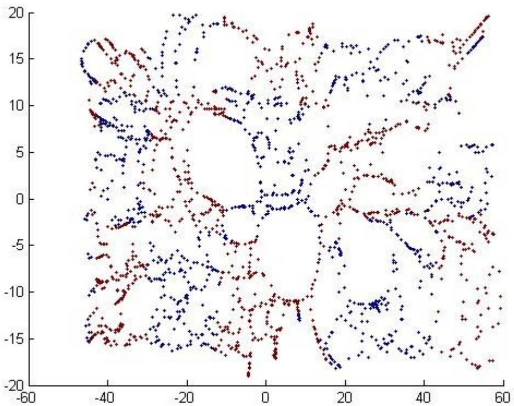
STATE-OF-ART TECHNIQUES

ISOMAP, LOCAL LINEAR EMBEDDING (LLE) AND LAPLACIAN EIGENMAP

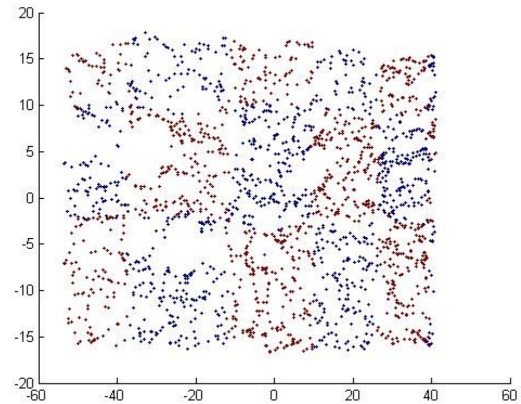
ISOMAP

- "Curves can give the shortest distance."
- Steps:
  - Build graph from K Nearest Neighbors.
  - Estimate geodesic distances between points using shortest path algorithm.
  - Run MDS on the data.
- Problems:
  - Assumes dataset is convex and uniformly sampled.
  - Faces problem with noise.
  - Sensitive to k.
  - Slow.

Experiments: Swiss Roll

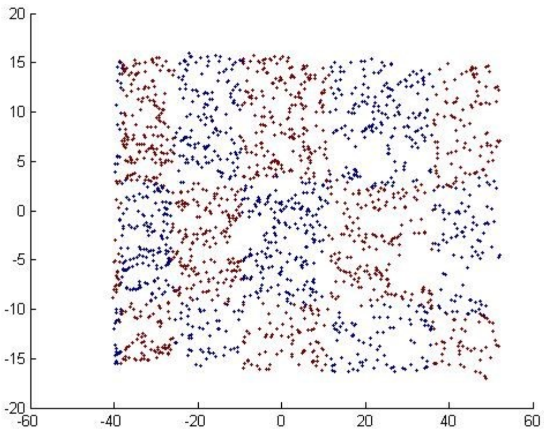


k=4

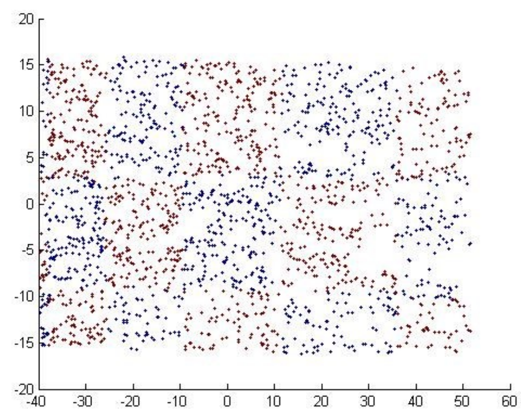


k=7

N=2000



k=10

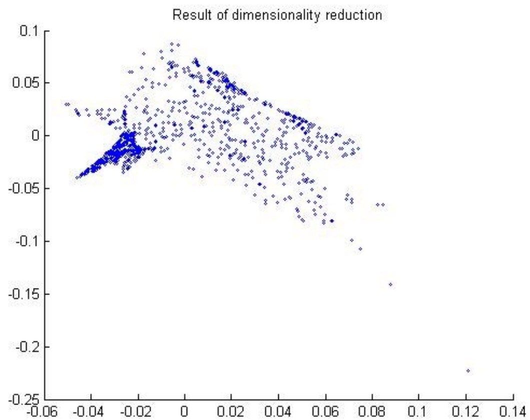


k=12

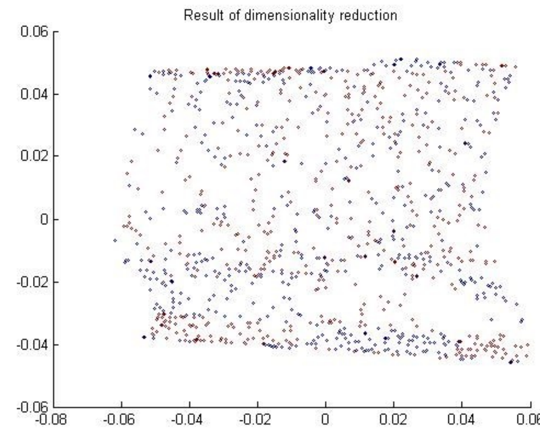
LLE

- "Think locally, Fit globally."
- Steps:
  - Build graph from k Nearest Neighbors.
  - Determine reconstruction weights by assuming neighborhoods are locally linear and insuring invariance.
  - Determine embedding by minimizing the reconstruction error term using eigen-solver.
- Problems:
  - Sensitive to noise and k.
  - Fails abruptly with complex geometry and noise.

Experiments: Swiss Roll

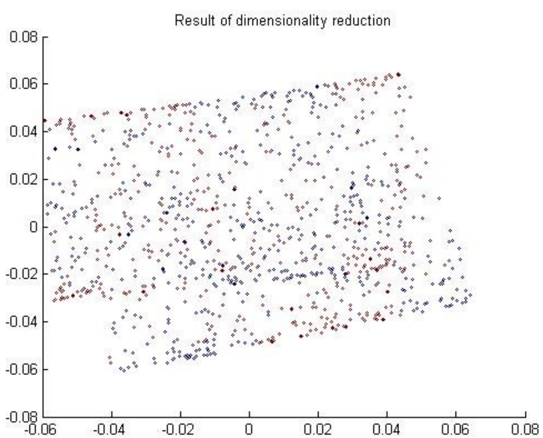


k=5

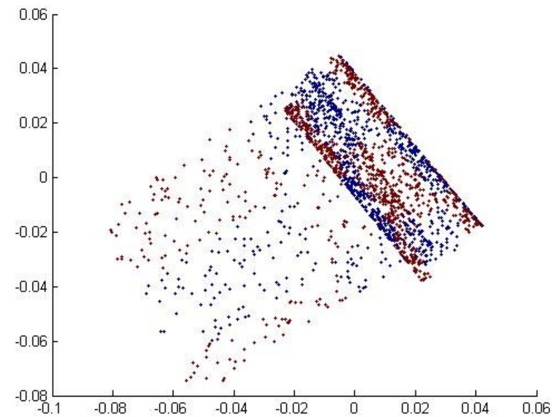


k=8

N=2000



k=11

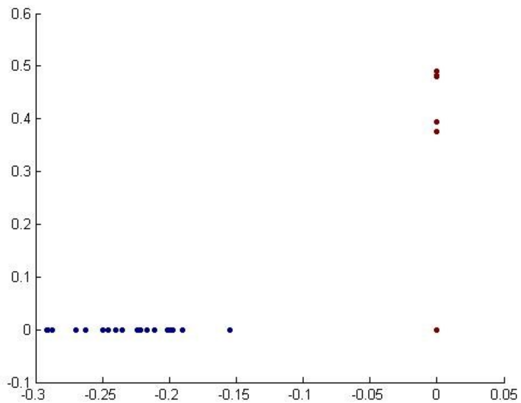


k=14

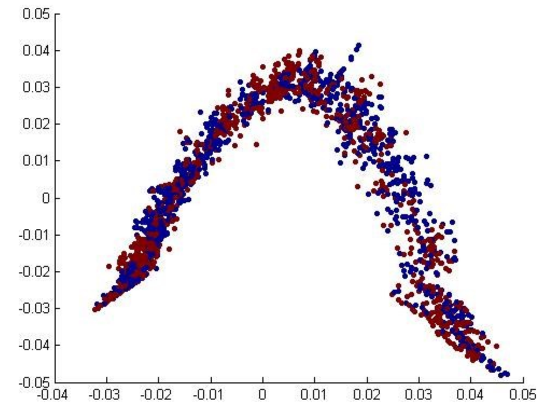
Laplacian Eigenmap

- "Neighbors are normally distributed around you."
- Steps:
  - Build graph from k Nearest Neighbors.
  - Construct weighted adjacency matrix with Gaussian kernel.
  - Compute embedding from by minimizing of normalized graph Laplacian *f.*
- Problems:
  - Assumes convexity of neighborhood and uniform sampling.
  - Sensitive to k.

Experiments: Swiss Roll

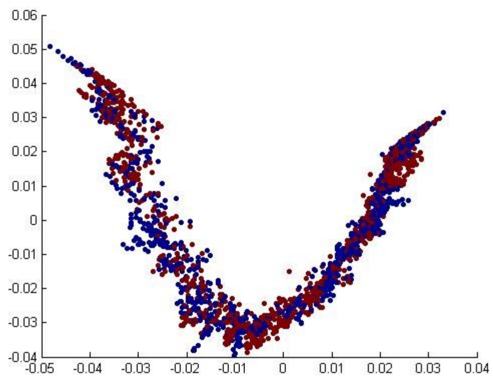


k=3

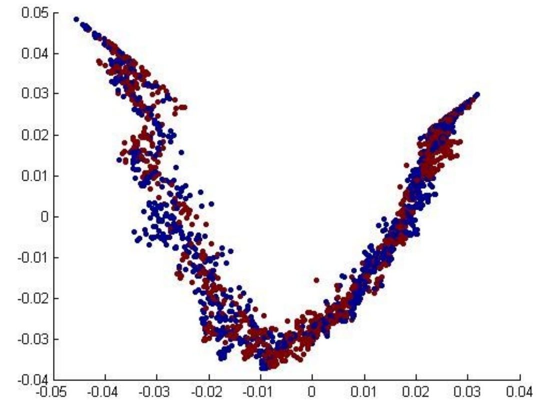


k=7

N=2000
sigma=1



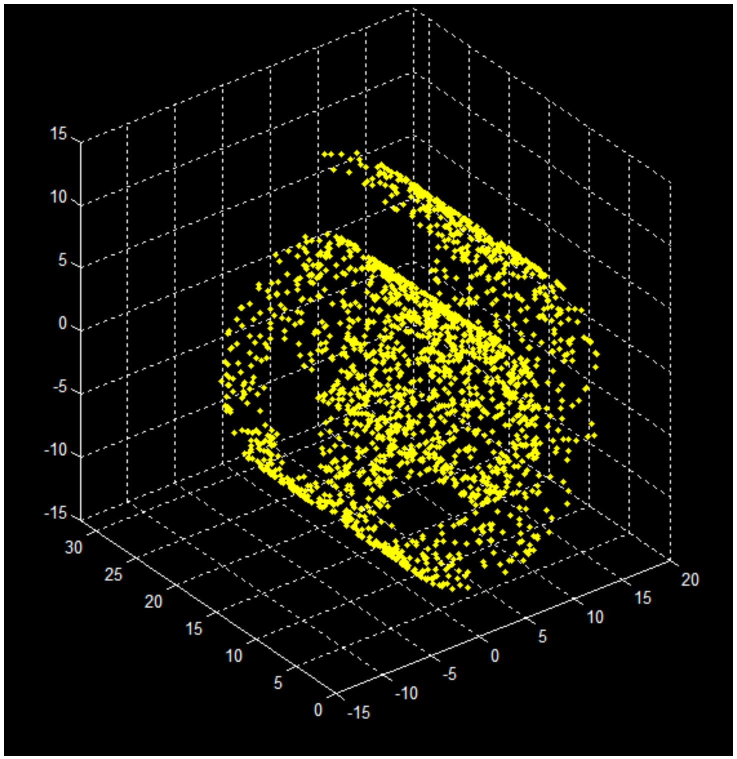
k=11



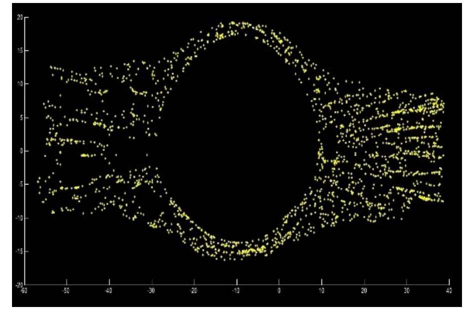
k=15

LIMITATIONS

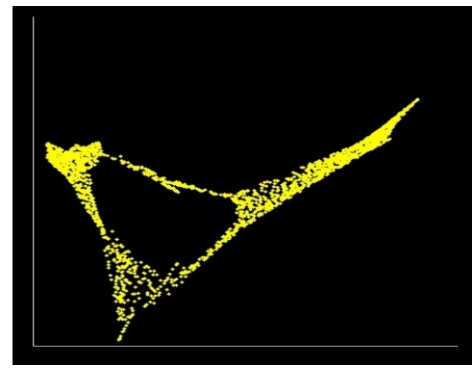
Non-convexity: Swiss hole



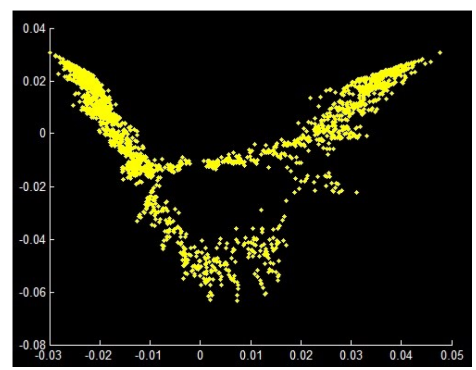
Swiss hole



ISOMAP (k =7)

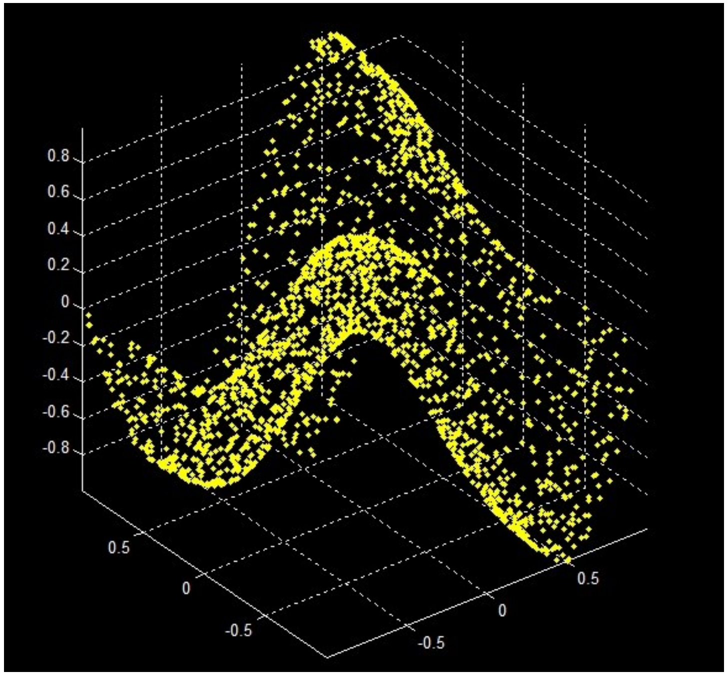


LLE (k = 9)

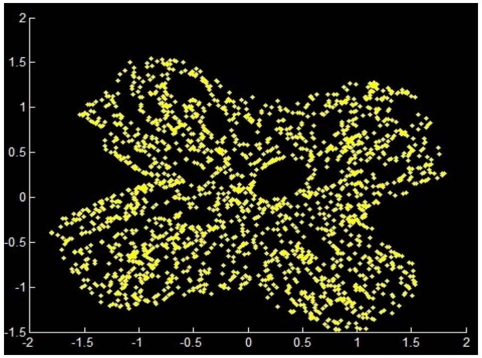


Laplacian (k = 8)

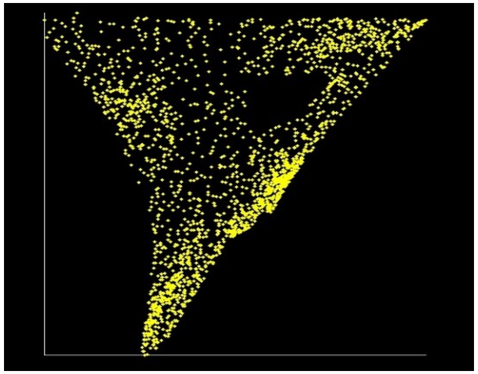
Non-uniform Geometry: Twin peaks



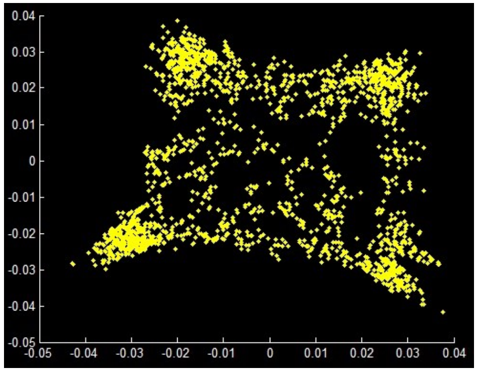
Twin Peaks



ISOMAP (k =7)

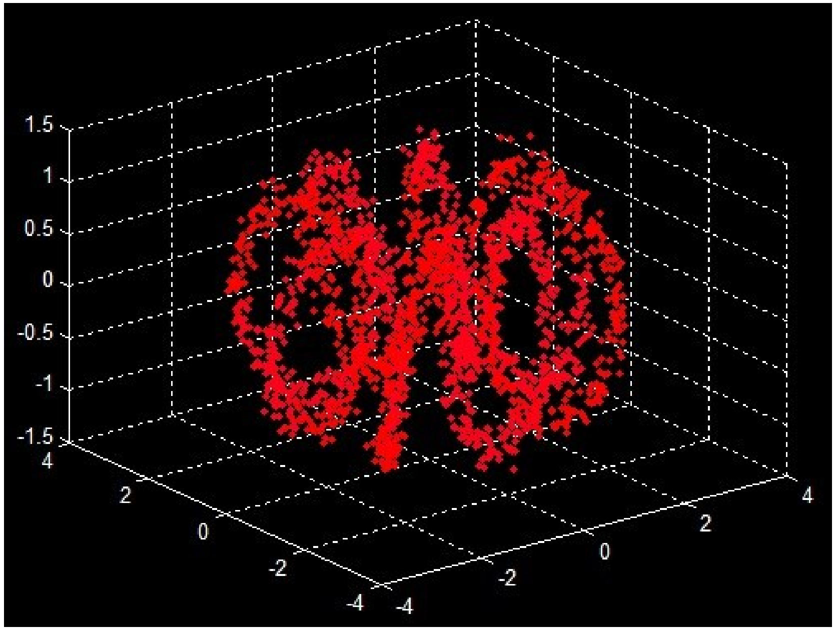


LLE (k =7)

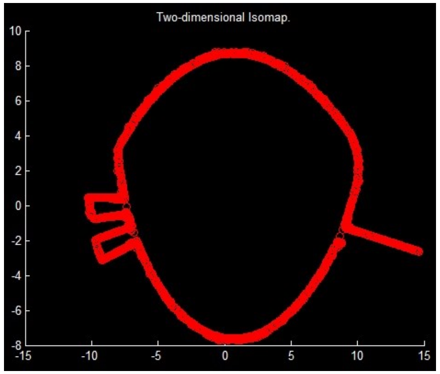


Laplacian (k =7)

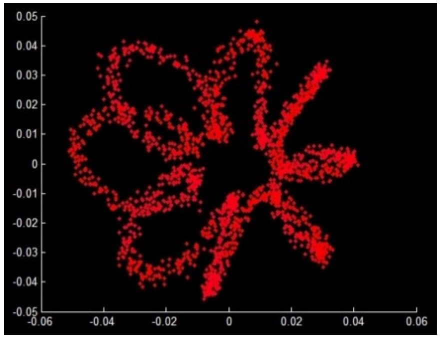
Noise: Helix



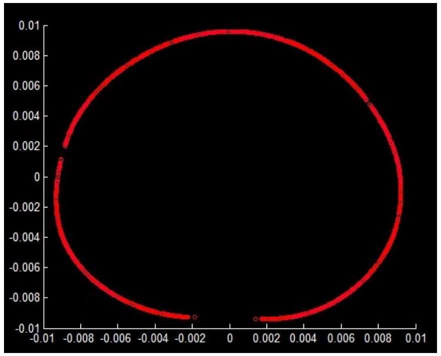
Helix with Gaussian noise
(Noise=0.10)



ISOMAP (k =10)



LLE (k =10)



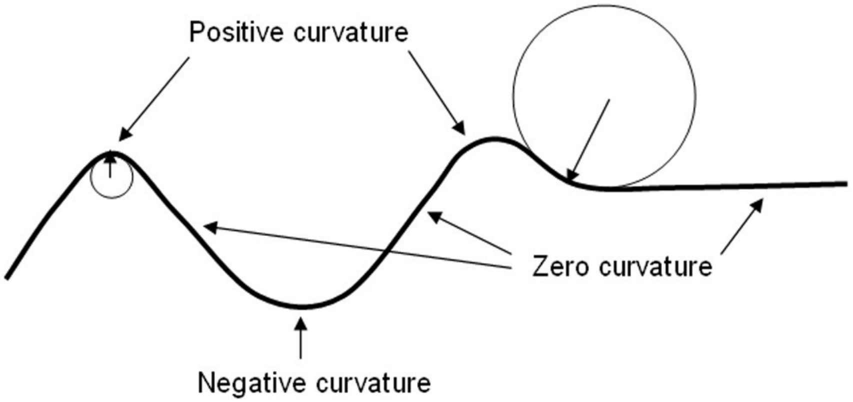
Laplacian (k =9)

Discussion: Part I

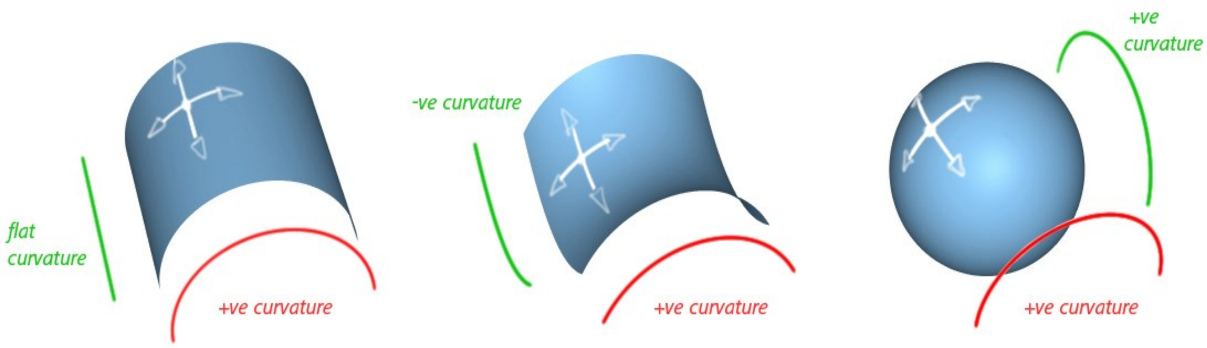
ISOMAP	LLE	Laplacian
Very slow	Fast	Faster
Distance preserving	Topology preserving	Topology preserving
Very poor in handling non-convexity	Very poor in handling non-convexity	Poor in handling non-convexity
May deal with curvature variation	Deals poorly with curvature variation	Deals with curvature variation better
Medium sensitivity to noise	Very sensitive to noise	Less sensitive to noise
Average level of dependence on k	Too much dependent on k	Less dependent on k

CURVATURE ADAPTIVE LLE

Theory: Curvature

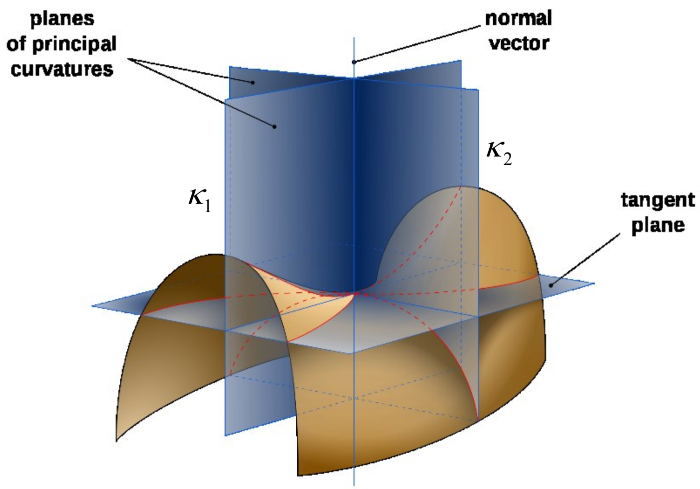


In 2 dimension for curves



In 3 dimension for surfaces

Theory: Gaussian & Mean Curvature

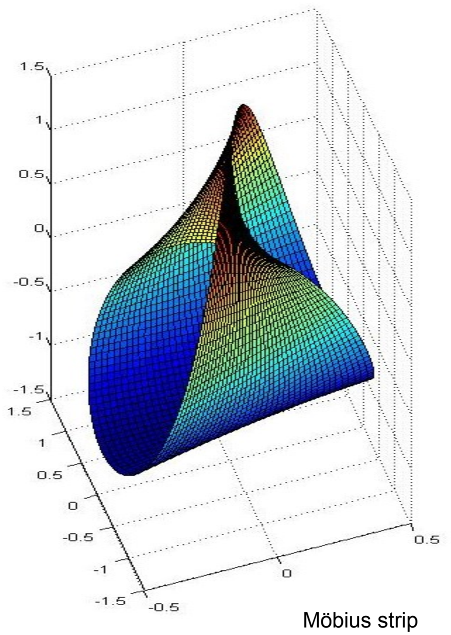


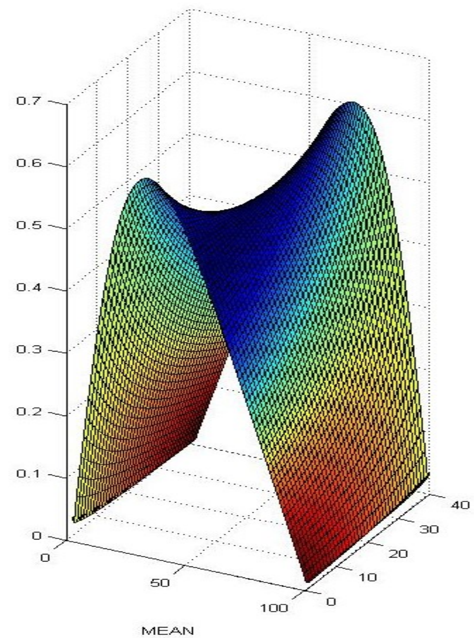
- Gaussian Curvature

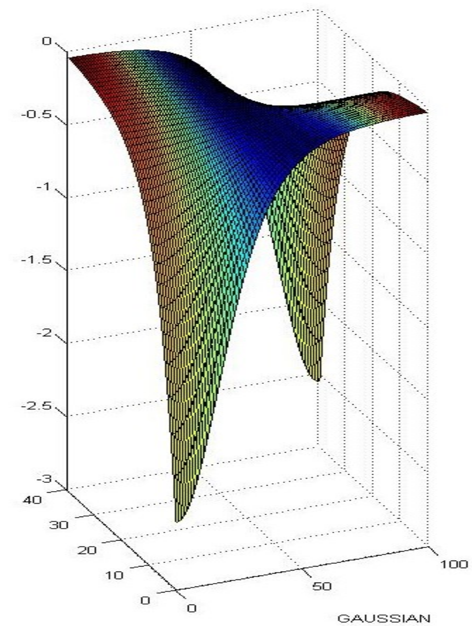
$$H = \frac{1}{2}(\kappa_1 + \kappa_2)$$

- Mean Curvature

$$K = \kappa_1 \kappa_2$$







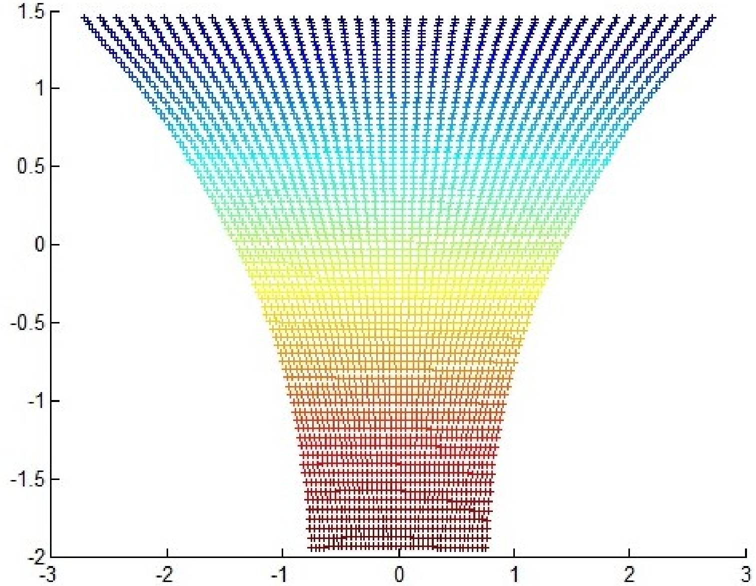
Theory: Algorithm

- Capturing local geometry:
  - Calculate principal and mean curvature, H and K, using the area and angle of neighboring triangles obtained from data mesh

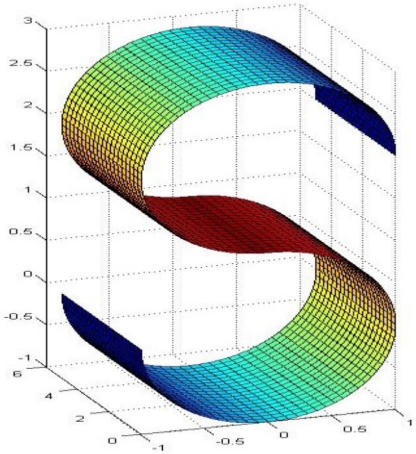
- Adapting Neighborhood:
  - For each data point consider other data points as neighboring points if their H and K values are comparable with the data point.
  - H and K values are comparable if they are inside certain deviation from the corresponding value. This deviation is decided from H and K variation of total curve.
  - Also check $\frac{\partial K}{\partial x}, \frac{\partial H}{\partial x}$ i.e, variation of K and H in different directions to finally identify the region where the two considered data points might occur in manifold.

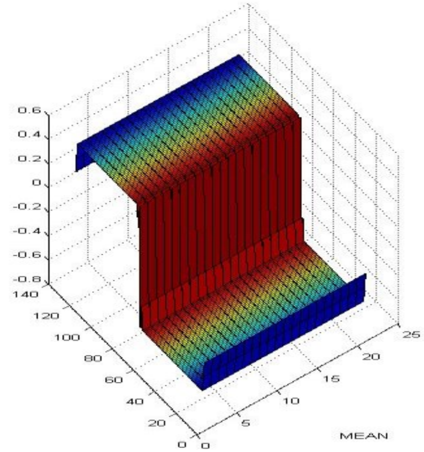
- Apply LLE on this structure.

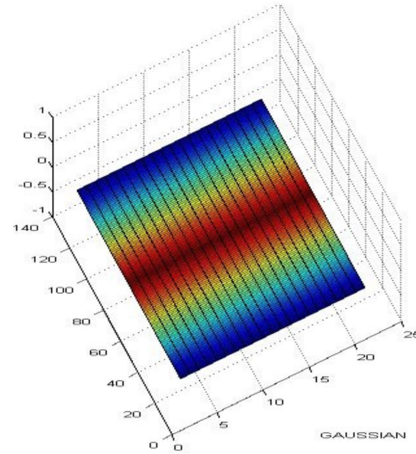
Experiments: Swiss Roll



Experiments: S curve

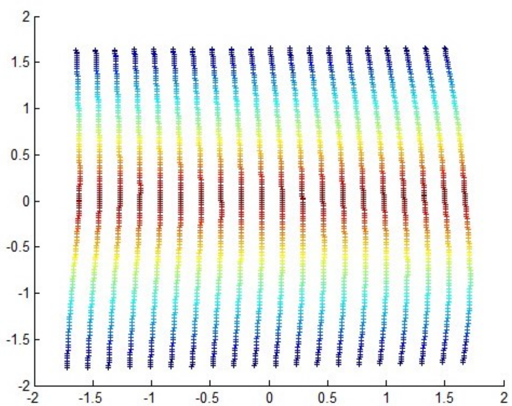




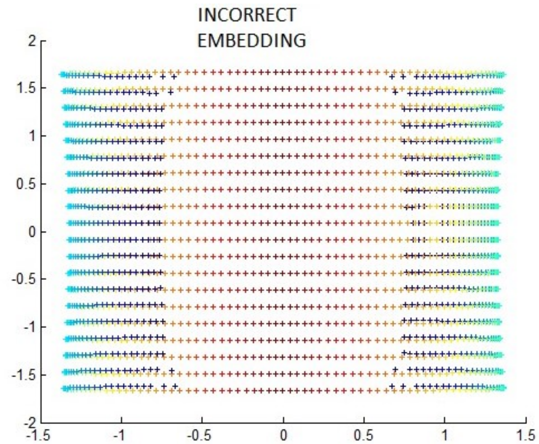


Original LLE

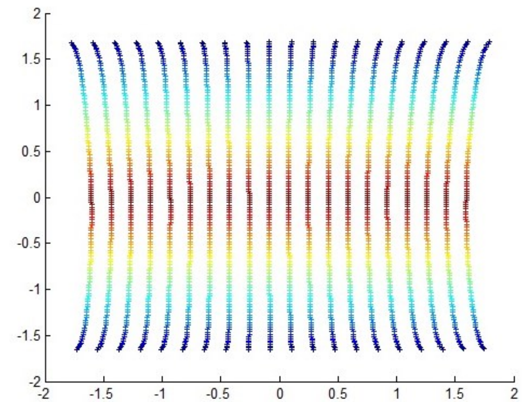
Curvature Adaptive LLE



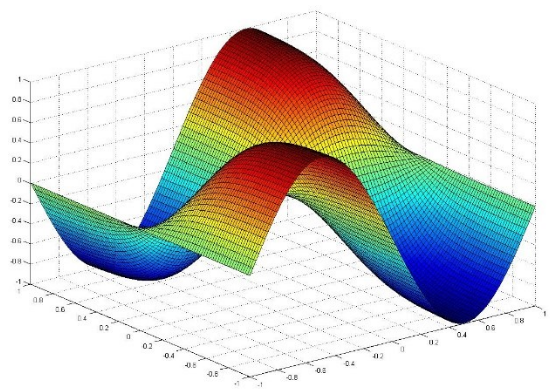
k=10

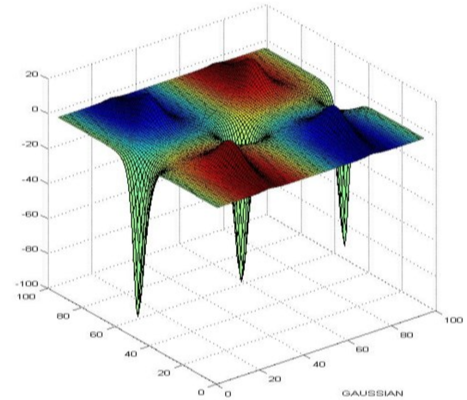


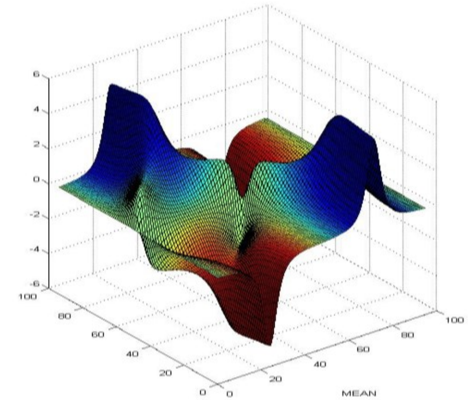
k=12

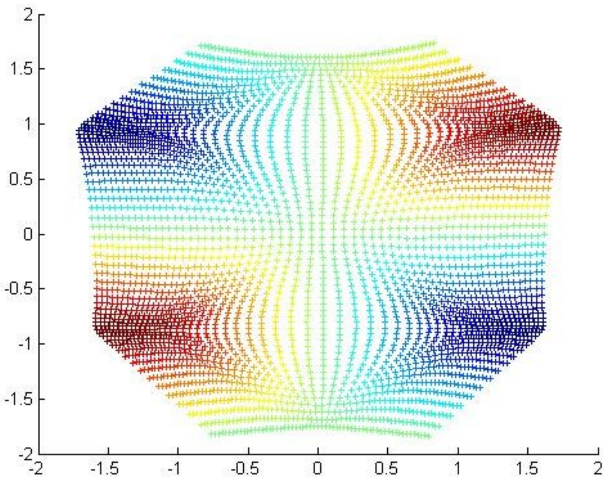


Experiments: Twin peaks





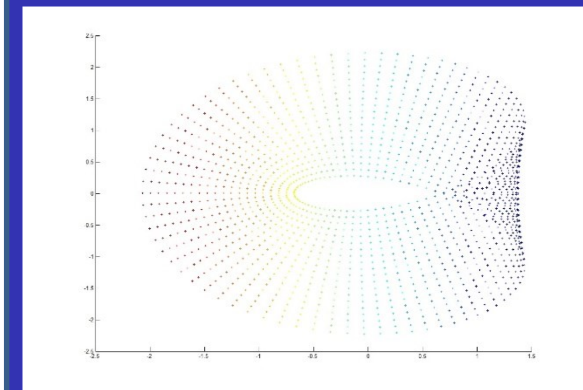




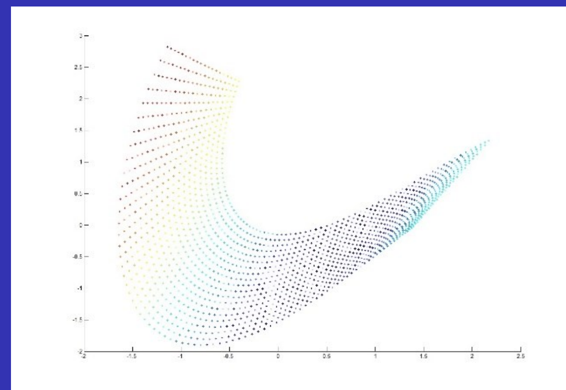
Curvature Adaptive LLE

Experiments: Möbius Strip

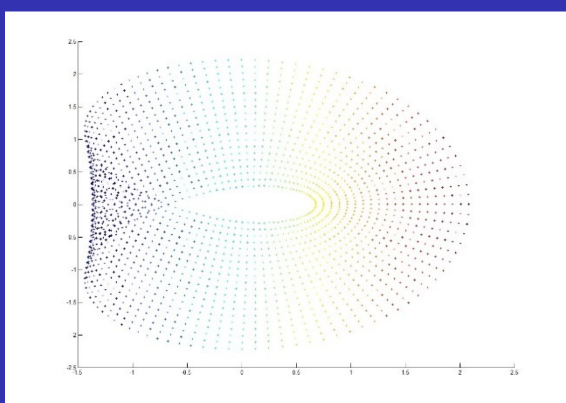
Original LLE



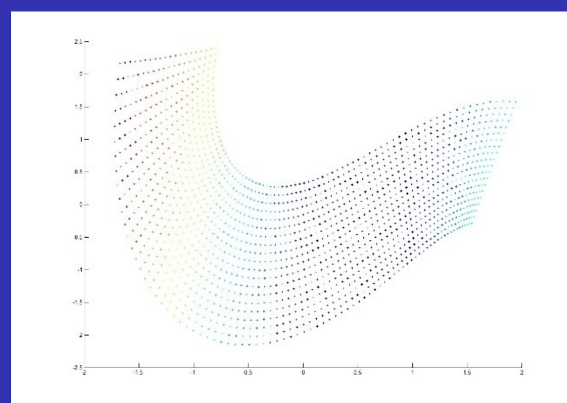
k=12



k=13



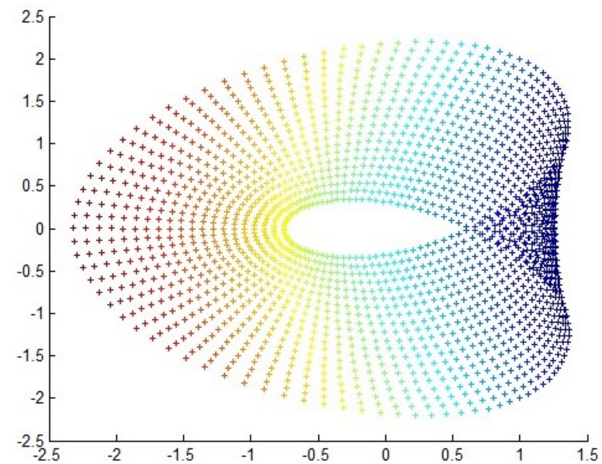
k=14



k=15

UNSTABLE

Curvature Adaptive LLE



Discussion: Part II

Hopes

- Gives better embedding in the test cases.
- Takes competitive operation time.
- Gives unique stable solution where LLE fails or generate unstable solutions.

Worries

- No theoretical guarantee, yet.
- Heterogeneous neighborhood size may create challenge in the path of proof.
- Probably would not work for non-convex surfaces like Swiss hole.

Questions?

Thank you…