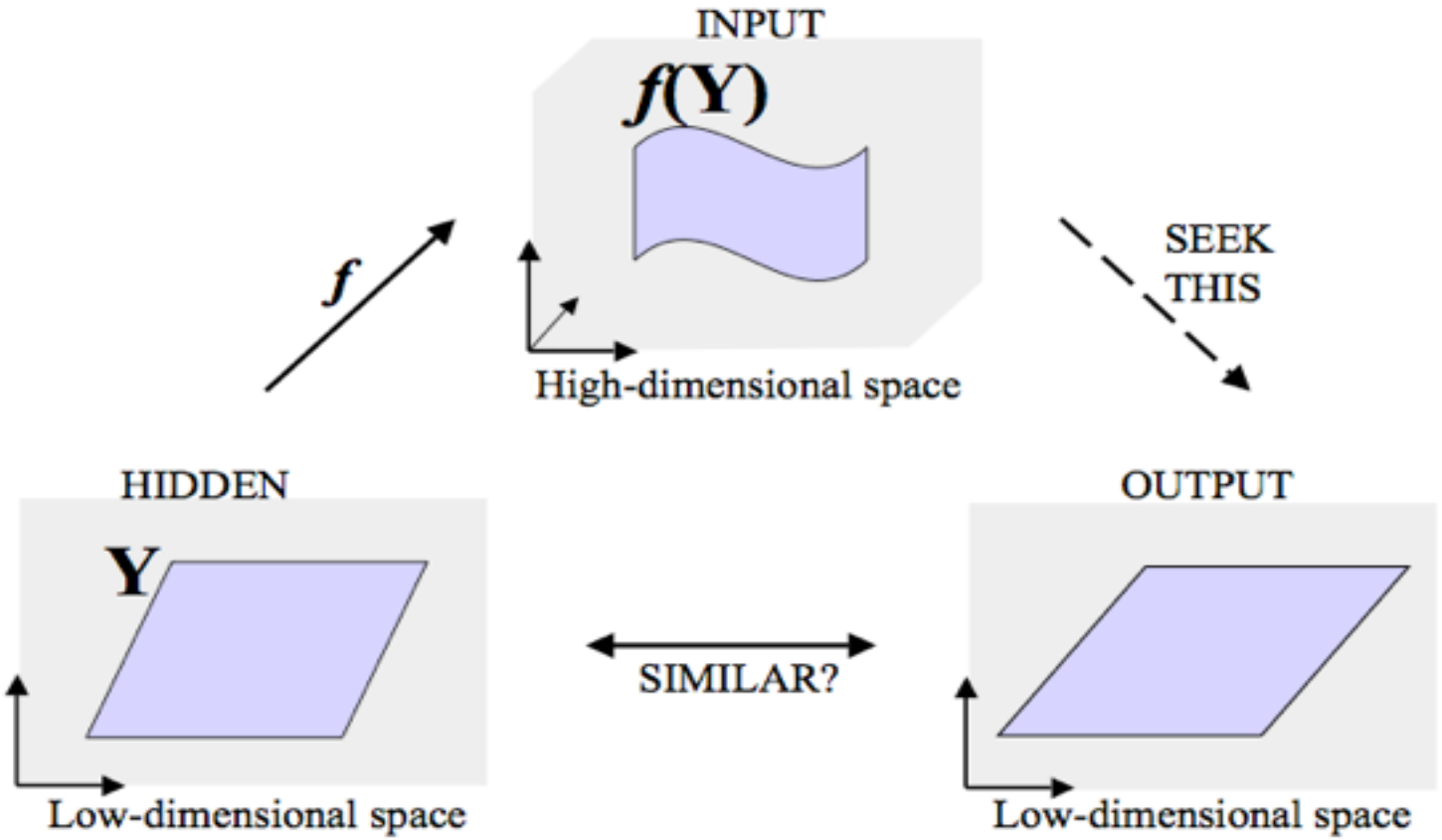
Manifold Learning II: Extended Locally Linear Embedding

姚 遠

2017

Generative Models in Manifold Learning



Spectral Geometric Embedding

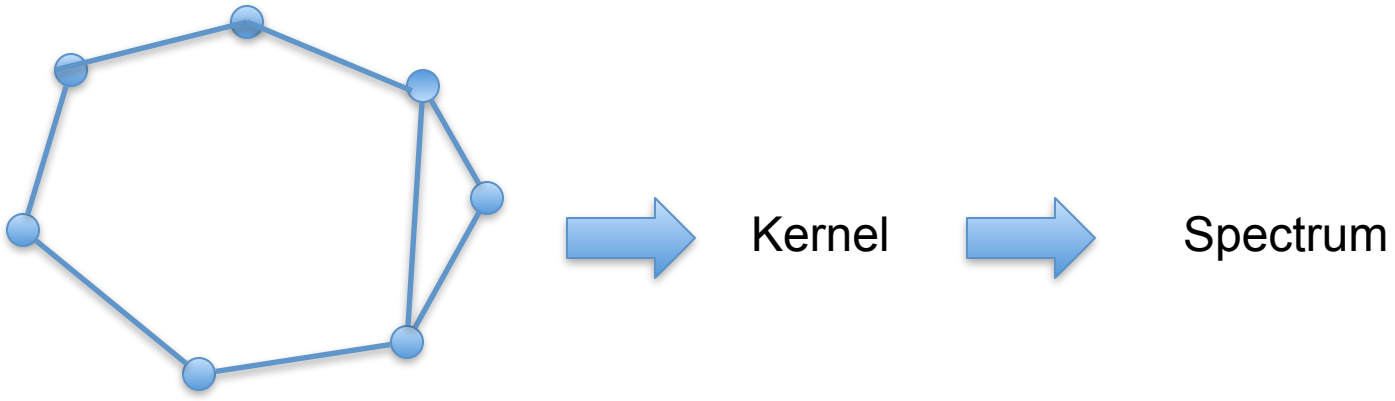
Given $x_1, \ldots, x_n \in \mathcal{M} \subset \mathbb{R}^N$,
Find $y_1, \ldots, y_n \in \mathbb{R}^d$ where $d << N$

- ISOMAP (Tenenbaum, et al, 00)
- LLE (Roweis, Saul, 00)
- Laplacian Eigenmaps (Belkin, Niyogi, 01)
- Local Tangent Space Alignment (Zhang, Zha, 02)
- Hessian Eigenmaps (Donoho, Grimes, 02)
- Diffusion Maps (Coifman, Lafon, et al, 04)

Related: Kernel PCA (Schoelkopf, et al, 98)

Meta-Algorithm

- Construct a neighborhood graph
- Construct a positive semi-definite kernel
- Find the spectrum decomposition



Recall: ISOMAP

1. Construct Neighborhood Graph.
2. Find shortest path (geodesic) distances.

D_{ij} is $n \times n$

3. Embed using Multidimensional Scaling.

Recall: LLE

- Construct a neighborhood Graph G=(V,E)
- Solve weights

$$\min_{\sum_j w_{ij}=1} \|X_i - \sum_{j \in \mathcal{N}(i)} w_{ij} \bar{X}_j\|^2, \quad \bar{X}_j = X_j - X_i.$$

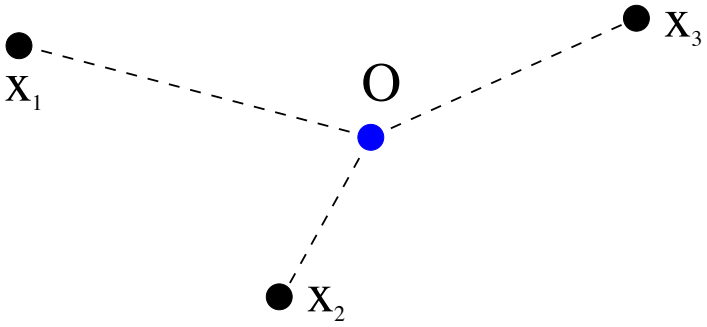
- Compute Embedding

$$\min_Y \sum_{i=1}^{n} \|Y_i - \sum_{j=1}^{n} W_{ij} Y_j\|^2 = \text{trace}((I-W)Y^T Y(I-W)^T).$$

$$W_{ij}^{n \times n} = \begin{cases} w_{ij} & j \in \mathcal{N}(i), \\ 0 & \text{other's.} \end{cases}$$

This is equivalent to find smallest eigenvectors of $K = (I-W)^T(I-W)$.

Laplacian and LLE



$$\sum w_i x_i = 0$$

$$\sum w_i = 1$$

Hessian H. Taylor expansion :

$$f(x_i) = f(0) + x_i^t \nabla f + \frac{1}{2} x_i^t H x_i + o(\|x_i\|^2)$$

$$(I - W)f(0) = f(0) - \sum w_i f(x_i) \approx f(0) - \sum w_i f(0) - \sum_i w_i x_i^t \nabla f - \frac{1}{2} \sum_i x_i^t H x_i =$$

$$= -\frac{1}{2} \sum_i x_i^t H x_i \approx -trH = \Delta f$$

Discrete Laplacian

Find $y_1, \ldots, y_n \in R$

$$\min \sum_{i,j} (y_i - y_j)^2 W_{ij}$$

Tries to preserve locality

A Fundamental Identity

But

$$\frac{1}{2}\sum_{i,j}(y_i - y_j)^2 W_{ij} = \mathbf{y}^T L \mathbf{y}$$

$$\sum_{i,j}(y_i - y_j)^2 W_{ij} = \sum_{i,j}(y_i^2 + y_j^2 - 2y_i y_j) W_{ij}$$

$$= \sum_i y_i^2 D_{ii} + \sum_j y_j^2 D_{jj} - 2\sum_{i,j} y_i y_j W_{ij}$$

$$= 2\mathbf{y}^T L \mathbf{y}$$

Embedding of Unnormalized Laplacian Eigenmap

$$\lambda = 0 \rightarrow \mathbf{y} = \mathbf{1}$$

$$\min_{\mathbf{y}^T\mathbf{1}=0} \mathbf{y}^T L\mathbf{y}$$

Let $Y = [\mathbf{y}_1\mathbf{y}_2 \ldots \mathbf{y}_m]$

$$\sum_{i,j} ||Y_i - Y_j||^2 W_{ij} = \text{trace}(Y^T LY)$$

subject to $Y^T Y = I$.

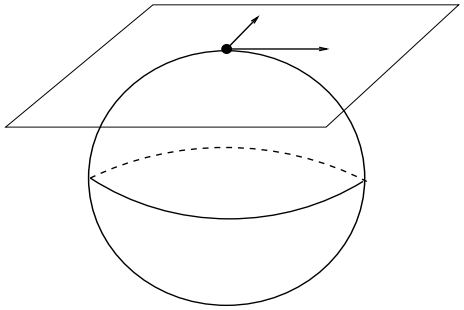
Use eigenvectors of L to embed.

Manifold Laplacian

Recall ordinary Laplacian in $\mathbb{R}^k$
This maps

$$f(x_1, \ldots, x_k) \to \left(-\sum_{i=1}^{k} \frac{\partial^2 f}{\partial x_i^2} \right)$$

Manifold Laplacian is the same on the tangent space.



On the Manifold

smooth map $f : \mathcal{M} \to R$

$$\int_{\mathcal{M}} \|\nabla_{\mathcal{M}} f\|^2 \approx \sum_{i \sim j} W_{ij}(f_i - f_j)^2$$

Recall standard gradient in $\mathbb{R}^k$ of $f(z_1, \ldots, z_k)$

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial z_1} \\ \frac{\partial f}{\partial z_2} \\ \cdot \\ \\ \cdot \\ \\ \frac{\partial f}{\partial z_k} \end{bmatrix}$$

Stokes Theorem

A Basic Fact

$$\int_{\mathcal{M}} \|\nabla_{\mathcal{M}} f\|^2 = \int f \cdot \Delta_{\mathcal{M}} f$$

This is like

$$\sum_{i,j} W_{ij}(f_i - f_j)^2 = \mathbf{f}^T \mathbf{L} \mathbf{f}$$

where

$\Delta_{\mathcal{M}} f$ is the manifold Laplacian

Manifold Laplacian Eigenvectors

Eigensystem

$$\Delta_{\mathcal{M}} f = \lambda_i \phi_i$$

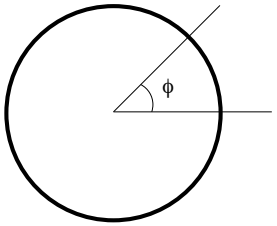
$\lambda_i \geq 0$ and $\lambda_i \to \infty$

$\{\phi_i\}$ form an orthonormal basis for $L^2(\mathcal{M})$

$$\int \|\nabla_{\mathcal{M}} \phi_i\|^2 = \lambda_i$$

Manifold Laplacian is non-compact!

Example: Circle



$$-\frac{d^2u}{dt^2} = \lambda u \text{ where } u(0) = u(2\pi)$$

Eigenvalues are

$$\lambda_n = n^2$$

Eigenfunctions are

$$\sin(nt), \cos(nt)$$

Spherical Harmonics in high-D sphere!

Spectral Growth

$$\lambda_1 \leq \lambda_2 \ldots \leq \lambda_j \leq \ldots$$

Then

$$A + \frac{2}{d}\log(j) \leq \log(\lambda_j) \leq B + \frac{2}{d}\log(j+1)$$

Example: on S^1

$$\lambda_j = j^2 \implies \log(\lambda_j) = \frac{2}{1}\log(j)$$

(Li and Yau; Weyl's asymptotics)

Solution of Heat Equations

Heat equation in $\mathbb{R}^n$:

$u(x,t)$ – heat distribution at time t.
$u(x,0) = f(x)$ – initial distribution. $x \in \mathbb{R}^n, t \in \mathbb{R}$.

$$\Delta_{\mathbb{R}^n} u(x,t) = \frac{du}{dt}(x,t)$$

Solution – convolution with the heat kernel:

$$u(x,t) = (4\pi t)^{-\frac{n}{2}} \int_{\mathbb{R}^n} f(y) e^{-\frac{\|x-y\|^2}{4t}} dy$$

Discretization of Heat Eq.

Functional approximation:

Taking limit as $t \to 0$ and writing the derivative:

$$\Delta_{\mathbb{R}^n} f(x) = \frac{d}{dt}\left[(4\pi t)^{-\frac{n}{2}} \int_{\mathbb{R}^n} f(y) e^{-\frac{\|x-y\|^2}{4t}} dy\right]_0$$

$$\Delta_{\mathbb{R}^n} f(x) \approx -\frac{1}{t}(4\pi t)^{-\frac{n}{2}} \left(f(x) - \int_{\mathbb{R}^n} f(y) e^{-\frac{\|x-y\|^2}{4t}} dy\right)$$

Empirical approximation:

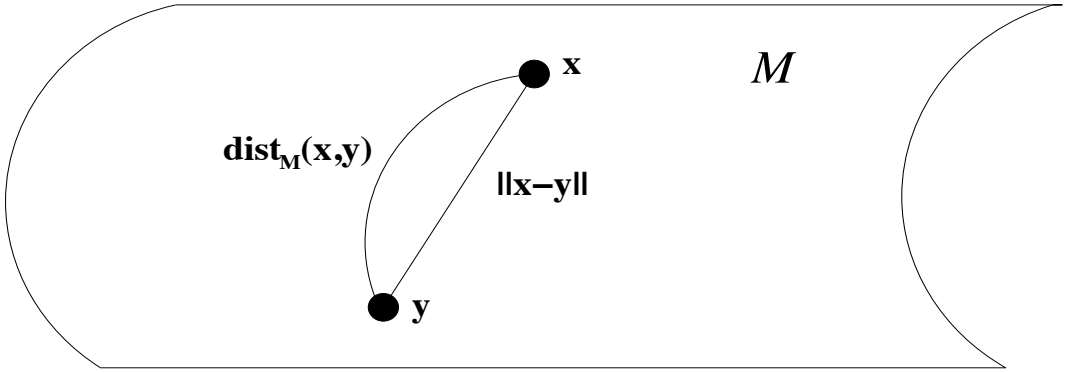
Integral can be estimated from empirical data.

$$\Delta_{\mathbb{R}^n} f(x) \approx -\frac{1}{t}(4\pi t)^{-\frac{n}{2}} \left(f(x) - \sum_{x_i} f(x_i) e^{-\frac{\|x-x_i\|^2}{4t}}\right)$$

Some Difficulties for Manifolds

Some difficulties arise for manifolds:

- Do not know distances.
- Do not know the heat kernel.



Careful analysis needed.

The Heat Kernel Approximation

- $H_t(x,y) = \sum_i e^{-\lambda_i t}\phi_i(x)\phi_i(y)$
- in $\mathbb{R}^d$, closed form expression

$$H_t(x,y) = \frac{1}{(4\pi t)^{d/2}} e^{-\frac{\|x-y\|^2}{4t}}$$

- Goodness of approximation depends on the gap

$$\left| H_t(x,y) - \frac{1}{(4\pi t)^{d/2}} e^{-\frac{\|x-y\|^2}{4t}} \right|$$

- H_t is a Mercer kernel intrinsically defined on manifold. Leads to SVMs on manifolds.

Pointwise Convergence

$$f : \mathcal{M} \to \mathbb{R} \quad x \in \mathcal{M} \quad x_1, \ldots, x_n \in \mathcal{M}$$

Graph Laplacian:

$$L_n^t(f)(x) = f(x) \sum_j e^{-\frac{\|x-x_j\|^2}{t}} - \sum_j f(x_j) e^{-\frac{\|x-x_j\|^2}{t}}$$

Theorem [pointwise convergence] $t_n = n^{-\frac{1}{k+2+\alpha}}$

$$\lim_{n\to\infty} \frac{(4\pi t_n)^{-\frac{k+2}{2}}}{n} L_n^{t_n} f(x) = \Delta_{\mathcal{M}} f(x)$$

Belkin 03, Lafon Coifman 04, Belkin Niyogi 05, Hein et al 05

Convergence of Eigenfunctions

Theorem [convergence of eigenfunctions]

$$\lim_{t \to 0, n \to \infty} Eig[L_n^{t_n}] \to Eig[\Delta_{\mathcal{M}}]$$

Laplacian Eigenmaps (I) [Belkin-Niyogi]

Step 1 [*Constructing the Graph*]

$$e_{ij} = 1 \Leftrightarrow \mathbf{x}_i \text{ “close to” } \mathbf{x}_j$$

1. ϵ-neighborhoods. [*parameter* $\epsilon \in \mathbb{R}$] Nodes i and j are connected by an edge if

$$||\mathbf{x}_i - \mathbf{x}_j||^2 < \epsilon$$

2. n nearest neighbors. [*parameter* $n \in \mathbb{N}$] Nodes i and j are connected by an edge if i is among n nearest neighbors of j or j is among n nearest neighbors of i.

Laplacian Eigenmaps (II)

Step 2. [*Choosing the weights*].

1. Heat kernel. [*parameter* $t \in \mathbb{R}$]. If nodes i and j are connected, put

$$W_{ij} = e^{-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{t}}$$

2. Simple-minded. [*No parameters*]. $W_{ij} = 1$ if and only if vertices i and j are connected by an edge.

Laplacian Eigenmaps (III)

Step 3. [*Eigenmaps*] Compute eigenvalues and eigenvectors for the generalized eigenvector problem:

$$Lf = \lambda Df$$

D is diagonal matrix where

$$D_{ii} = \sum_j W_{ij}$$

$$L = D - W$$

Let $\mathbf{f}_0, \ldots, \mathbf{f}_{k-1}$ be eigenvectors.

Leave out the eigenvector $\mathbf{f}_0$ and use the next m lowest eigenvectors for embedding in an m-dimensional Euclidean space.

Connection to Markov Chain

- $L = D - W$
- $P = I - D^{-1}L = D^{-1}W$ is a markov matrix
- v is generalized eigenvector of L: $L\,v = \lambda\, D\, v$
- v is also a right eigenvector of P with eigenvalue $1-\lambda$
- P is lumpable iff v is piece-wise constant
- So Laplacian eigenmaps have Markov Chain interpretations (Diffusion Map)

Lumpability of Markov Chains

- Let P be the transition matrix of a Markov chain defined on n states S={1,…,n}.
- $\Gamma=\{S_1,\ldots,S_k\}$ is a partition of S into k macrostates.
- Sequences $\{x_0,\ldots,x_t,\ldots\}$ generated by P, i.e.

$$\text{Prob}(x_t=j\ ;\ x_{t-1}=i)= P_{ij}$$

- Induced dynamics: relabel x_t by y_t from corresponding states in partition Γ
- [Kemeny-Snell'76] P is called *lumpable* if

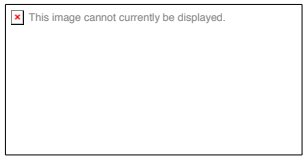
$$\text{Prob}(y_t=k_0;\ y_{t-1}=k_1,\ \ldots,y_{t-m}=k_m) = \text{Prob}(y_t=k_0;\ y_{t-1}=k_1)$$

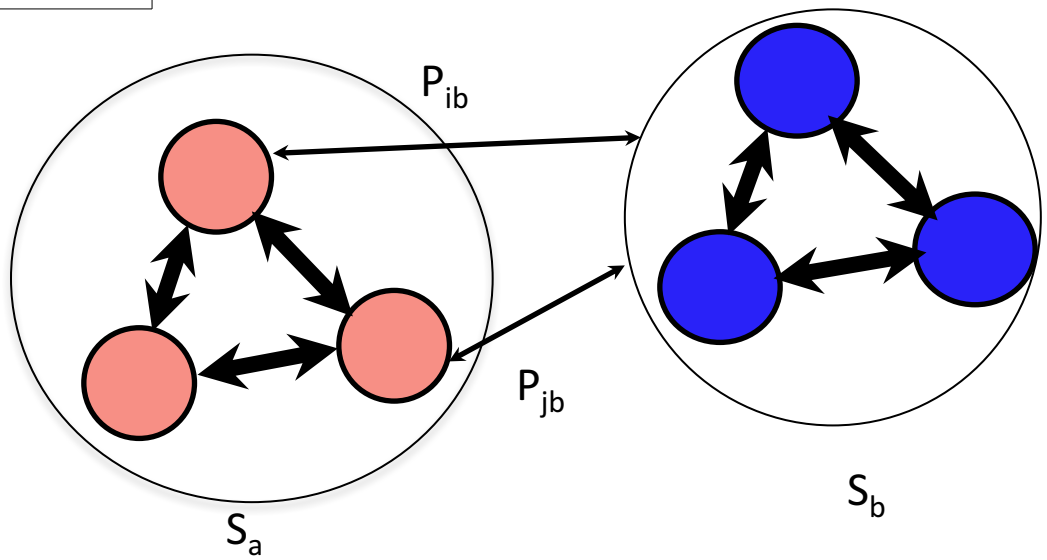
i.e. the induced dynamics is Markovian.

A Necessary and Sufficient Condition for Lumpability

- [Kemeny-Snell'76] P is *lumpable* w.r.t. partition $\Gamma=\{S_1,\ldots,S_k\}$ iff for any s, t chosen from P, and for any i, j lying in S_a, the following holds

$$P_{ib} = P_{jb}$$

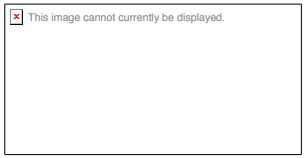
where 

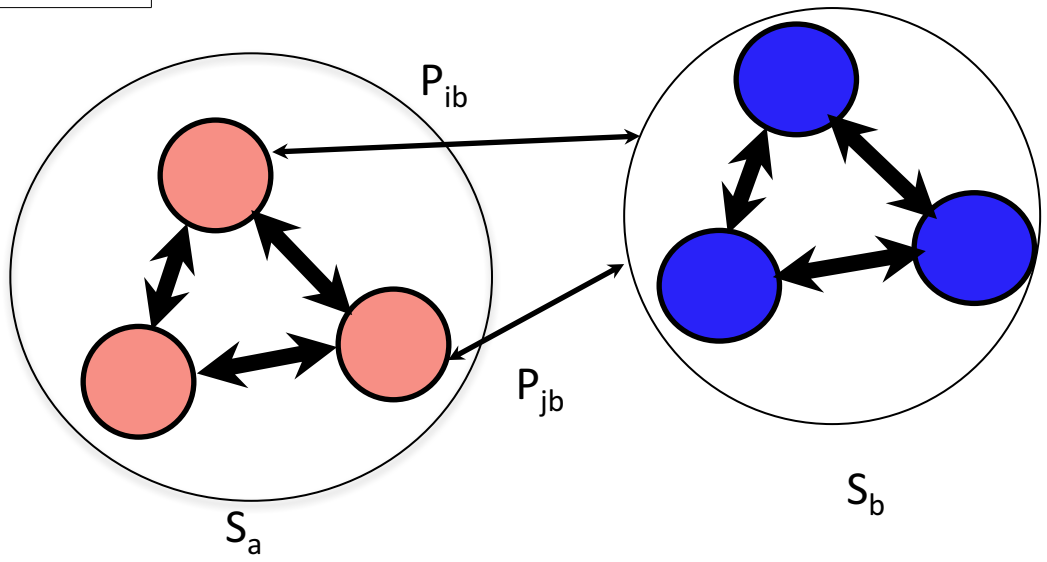


A Necessary and Sufficient Condition for Lumpability

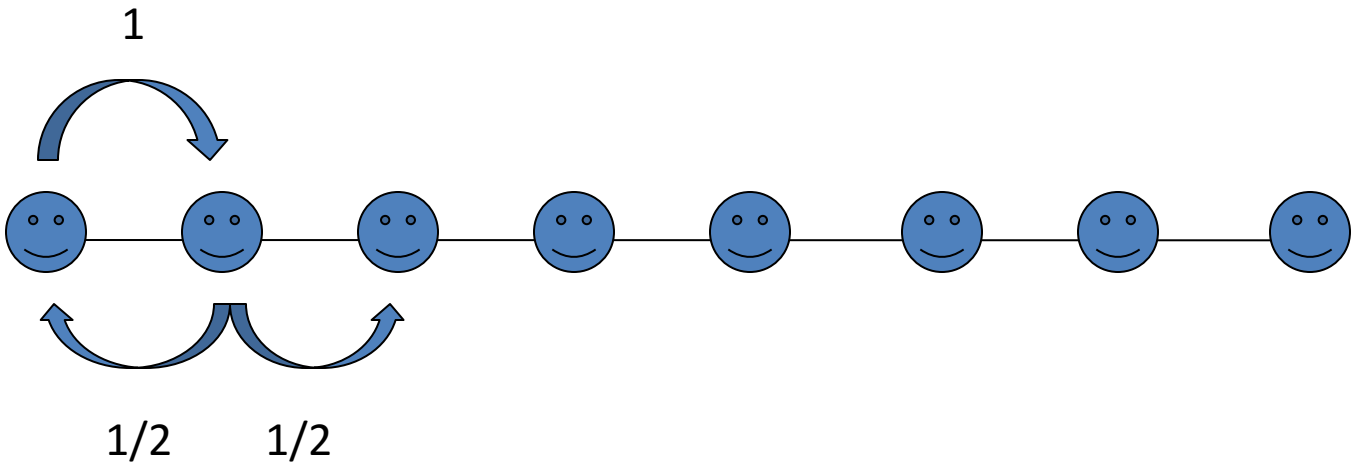
- [Kemeny-Snell'76] P is *lumpable* w.r.t. partition $\Gamma=\{S_1,\ldots,S_k\}$ iff for any s, t chosen from P, and for any i, j lying in S_a, the following holds

$$P_{ib} = P_{jb}$$

where 

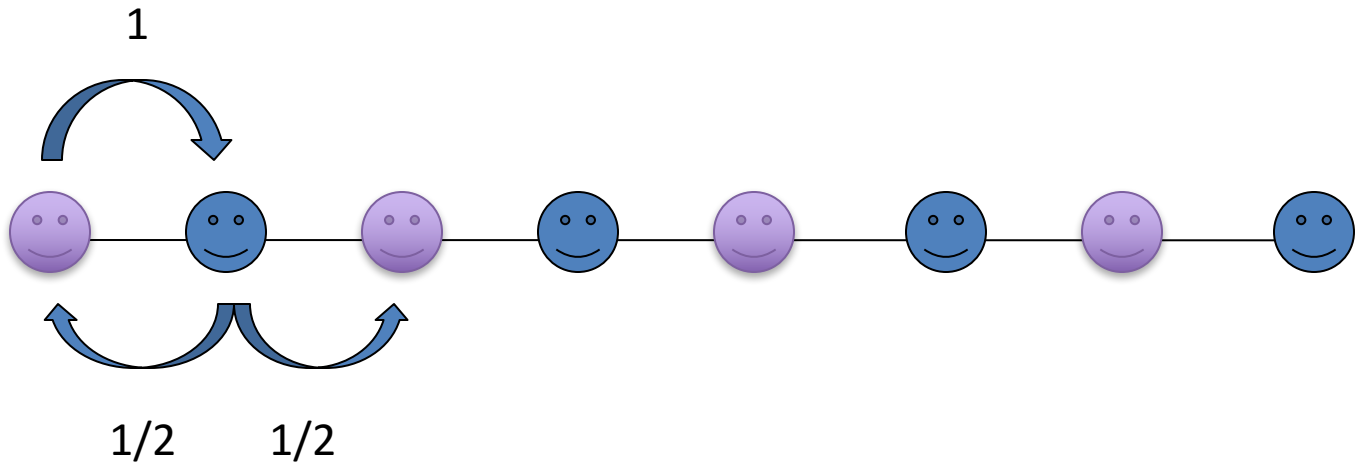


Example I



- Consider 2n nodes on a linear chain
- Markov Chain: a node will jump to its neighbors with equal probability
  - P(i, i-1) = P(i, i+1) = ½, for 2n>i>1
  - P(1,2) = P(2n,2n-1) = 1

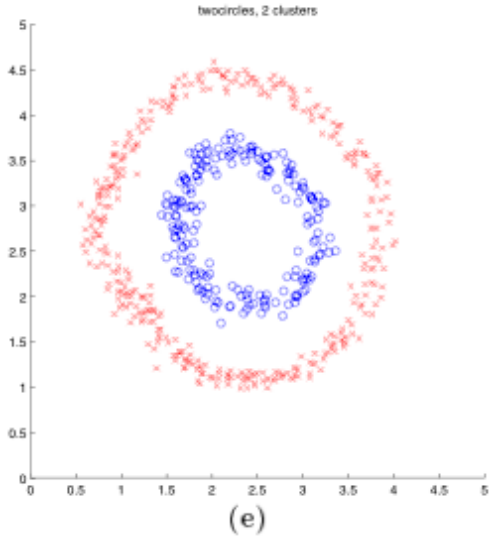
Example I



- P is lumpable w.r.t. $\Gamma^*=(S_{even},S_{odd})$
  - S_{even}: even nodes
  - S_{odd}: odd nodes
- Γ^* corresponds to eigenvector with eigenvalue -1

Spectral Clustering Algorithm

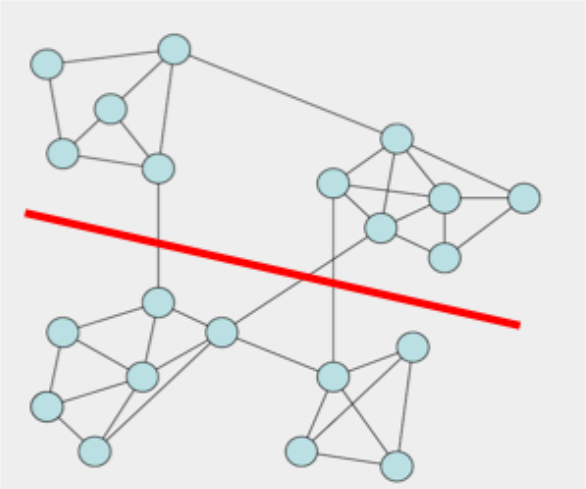
- Typical spectral algorithm to find lumpable states in nearly uncoupled systems [Ng-Jordan-Weiss NIPS'01]:
  1) Find top k right eigenvectors of P where a large spectral gap occurs, $v_1,\ldots,v_k$
  2) Embed the data into R^k by those eigenvectors
  3) Use k-means (or alternatives) to find k clusters in R^k



Graph Partition Problem

- goal: find a cut with the smallest Cheeger ratio (conductance)
  - For $S \subset V$, volume of S: $vol(S) = \sum_{v \in S} d_v$
  - $\partial S = \{(u, v) \in E : u \in S \& v \in S\}$
  - Cheeger ratio of S, $h(S) = \frac{|\partial S|}{\min\{vol(S), vol(G) - vol(S)\}}$

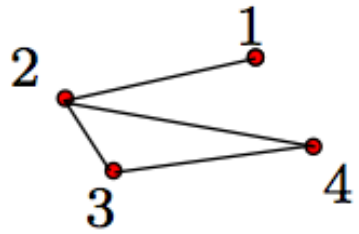
- applications
  - clustering
  - segmentation
  - task partitioning for parallel processing
  - a preprocessing step to divide-and-conquer algorithms



Graph Laplacian Operator

- given an undirected graph G=(V, E),
  - Adjacency matrix A:

$$A(u,v) = \left\{ \begin{array}{ll} 1 & \text{if } u \sim v \\ 0 & \text{o.w.} \end{array} \right\}$$



  - Diagonal degree matrix $D = diag(d_{v_1}, \cdots, d_{v_n})$
  - Graph Laplace Operator $L = D^{-1}(D - A)$
  - Tranistion probability matrix $W = D^{-1}A = I - L$,
  - $Wv = \lambda v$ implies $Lv = (1 - \lambda)v$
  - 1 is the largest eigenvalue for W; 0 is the smallest eigenvalue for L.

Graph Partition Problem

- Rayleigh quotient $R(f) = \frac{\sum_{u \sim v}(f(u)-f(v))^2}{\sum_u f^2(u) d_u}$ for $f \neq 0$
  - find a boolean function f minimizing $R(f)$ $\Leftarrow$ NP-complete
  - RELAXATION: find a real valued function f minimizing $R(f)$
  - $R(f) = \frac{<f,(D-A)f>}{<f,Df>}$
  - $\lambda_1 = \inf_f R(f) \Rightarrow \lambda_1$ and f are the first nonzero eigenvalue and eigenvector of L.

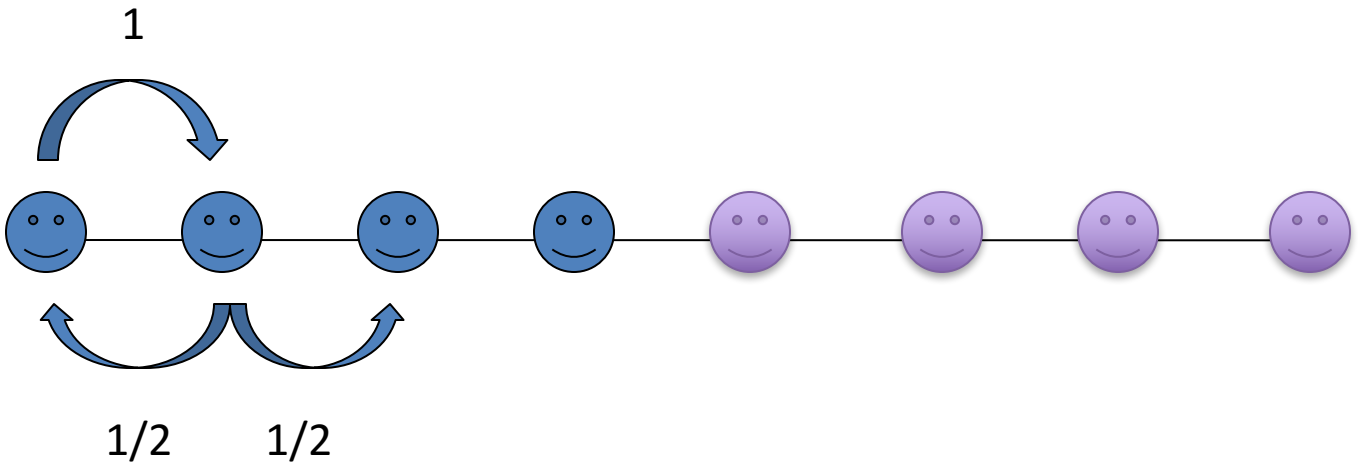
How good is this relaxation? Cheeger inequality

Cheeger Inequality

$$2h_G \geq \lambda_1 \geq \frac{h_f^2}{2} \geq \frac{h_G^2}{2}.$$

- f is the eigenvector of L corresponding to λ_1
- h_G is the smallest conductance (Cheeger ratio) of graph G
- h_f: the minimum Cheeger ratio determinded by a sweep of f
  - order the vertices: $f(v_1) \geq f(v_2) \geq \cdots \geq f(v_n)$.
  - $S_i = \{v_1, \cdots, v_i\}$
  - $h_f = \min_i h_{S_i}$
- find a partition whose conductance is within $2\sqrt{h_G}$

Example I



- One graph min-cut given by second largest right eigenvector of T
- n=8,
  - v_2=[0.4714 0.4247 0.2939 0.1049 -0.1049 -0.2939 -0.4247 -0.4714]
  - Eigenvalue is 0.9010

Generalized Heat Kernels

- Find Gaussian kernel $K_\varepsilon(x,y) = \exp\left(-\frac{\|x-y\|^2}{\varepsilon^2}\right)$
- Normalize kernel

$$K^{(\alpha)}(x,y) = \frac{K_\varepsilon(x,y)}{p^\alpha(x)p^\alpha(y)} \quad where \quad p(x) = \int K_\varepsilon(x,y)dm(y)$$

- Renormalized kernel

$$A_\varepsilon(x,y) = \frac{K^{(\alpha)}(x,y)}{\sqrt{d^{(\alpha)}(x)}\sqrt{d^{(\alpha)}(y)}} \quad where \quad d^{(\alpha)}(x) = \int K^{(\alpha)}(x,y)dm(y)$$

  - α=1, Laplacian-Beltrami operator, separate geometry from density
  - α=0, classical normalized graph Laplacian
  - α=1/2, backward Fokkar-Planck operator

General Diffusion Map

- P.S.D. Radial basis kernel $K_\varepsilon(x,y) = h\left(\frac{\|x-y\|^2}{\varepsilon^2}\right)$
- Normalize kernel

$$K^{(\alpha)}(x,y) = \frac{K_\varepsilon(x,y)}{p^\alpha(x)p^\alpha(y)} \quad where \quad p(x) = \int K_\varepsilon(x,y)dm(y)$$

- Markov kernel

$$a_\varepsilon^{(\alpha)}(x,y) = \frac{K^{(\alpha)}(x,y)}{d^{(\alpha)}(x)} \quad where \quad d^{(\alpha)}(x) = \int K^{(\alpha)}(x,y)dm(y)$$

- Diffusion Operator:

$$A_\varepsilon^{(\alpha)} f(x) = \int a_\varepsilon^{(\alpha)}(x,y) f(y) p(y) dy, \qquad p(x) = \frac{\exp(-U(x))}{Z}$$

$$\Delta_\varepsilon^{(\alpha)} = \frac{I - A_\varepsilon^{(\alpha)}}{\varepsilon}$$

Convergence of Diffusion Map [Coifman-Lafon 2005]

- Uniform sampling: Laplacian eigenmap converges to Laplacian-Beltrami operators [Belkin-Niyogi]
- Nonuniform sampling with $p(x)$
  - α=1: $\Delta_e^{(1)} = \frac{I - A_e^{(1)}}{e} = \Delta_0 + O(e^{1/2})$ where Δ_0 is Laplacian-Beltrami operator on Riemannian manifolds
  - α=1/2: backward Fokkar-Planck operator
  - α=0: classical normalized graph laplacian

Convergence Theorem [Coifman-Lafon 2006]

Theorem 7.1. Let $\mathcal{M} \in \mathbb{R}^p$ be a compact smooth submanifold, $q(x)$ be a probability density on $\mathcal{M}$, and $\Delta_{\mathcal{M}}$ be the Laplacian-Beltrami operator on $\mathcal{M}$.

$$\lim_{t\to 0} L_{t,\alpha} = \frac{\Delta_{\mathcal{M}}(f q^{1-\alpha})}{q^{1-\alpha}} - \frac{\Delta_{\mathcal{M}}(q^{1-\alpha}))}{q^{1-\alpha}}. \tag{67}$$

This suggests that

- for $\alpha = 1$, it converges to the Laplacian-Beltrami operator $\lim_{t\to 0} L_{t,1} = \Delta_{\mathcal{M}}$;
- for $\alpha = 1/2$, it converges to a Schrödinger operator whose conjugation leads to a forward Fokker-Planck equation;
- for $\alpha = 0$, it is the normalized graph Laplacian.

Hessian LLE

- Laplacian LLE

$$f^T L f = \sum_{i \geq j} w_{ij}(f_i - f_j)^2 \geq 0 \sim \int \|\nabla_M f\|^2 = \int (\mathrm{trace}(f^T \mathcal{H} f))^2$$

where $\mathcal{H} = [\partial^2 / \partial_i \partial_j] \in \mathbb{R}^{d \times d}$ is the Hessian matrix.

- Hessian LLE

$$\min \int \|\mathcal{H} f\|^2, \quad \|f\| = 1$$

- Laplacian kernel: const + linear + bilinear
- Hessian kernel: const + linear functions

Note that: $\Delta(f) = trace\big(H(f)\big)$

Hessian LLE Algorithm (I)

Algorithm 3: Hessian LLE Algorithm

Input: A weighted undirected graph $G = (V, E, d)$ such that

1 $V = \{x_i \in \mathbb{R}^p : i = 1, \ldots, n\}$

2 $E = \{(i, j) :$ if j is a neighbor of i, i.e. $j \in \mathcal{N}_i\}$, e.g. k-nearest neighbors

Output: Euclidean k-dimensional coordinates $Y = [y_i] \in \mathbb{R}^{k \times n}$ of data.

3 ***Step 1***: Compute local PCA on neighborhood of x_i, for,

$$\tilde{X}^{(i)} = [x_{i_1} - \mu_i, ..., x_{i_k} - \mu_i]^{p \times k} = \tilde{U}^{(i)} \tilde{\Sigma} (\tilde{V}^{(i)})^T, \quad x_{i_j} \in \mathcal{N}(x_i),$$

where $\mu_i = \sum_{j=1}^{k} x_{i_j} = \frac{1}{k} X_i \mathbf{1}$, $\tilde{U}^{(i)} = [\tilde{U}_1^{(i)}, ..., \tilde{U}_k^{(i)}]$ is an approximate tangent space at x_i;

Continued...

Hessian LLE Algorithm (II)

4 ***Step 2***: Hessian estimation, assumed d-dimension: define

$$M = [1, \tilde{V}_1, ..., \tilde{V}_k, \tilde{V}_1\tilde{V}_2, ..., \tilde{V}_{d-1}\tilde{V}_d] \in \mathbb{R}^{k\times(1+d+\binom{d}{2})}$$

where $\tilde{V}_i\tilde{V}_j = [\tilde{V}_{ik}\tilde{V}_{jk}]^T \in \mathbb{R}^k$ denotes the elementwise product (Hadamard product) between vector $\tilde{V}_i$ and $\tilde{V}_j$. Now we perform a Gram-Schmidt Orthogonalization procedure on M, get

$$\tilde{M} = [1, \hat{v}_1, ..., \hat{v}_k, \hat{w}_1, \hat{w}_2, ..., \hat{w}_{\binom{d}{2}-1}] \in \mathbb{R}^{k\times(1+d+\binom{d}{2})}$$

Define Hessian by

$$[H^{(i)}]^T = [last \quad \binom{d}{2} \quad columns \quad of \quad \tilde{M}]_{k\times\binom{d}{2}}$$

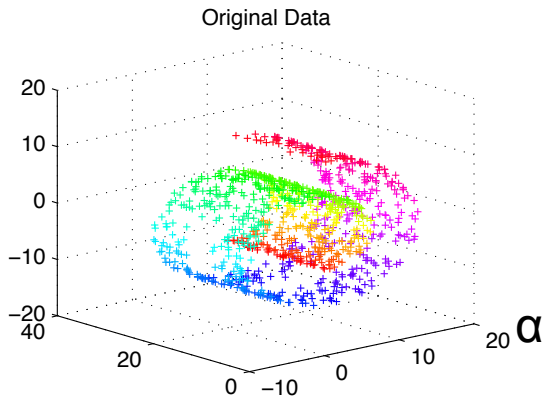
as the first $d+1$ columns of $\tilde{M}$ consists an orthonormal basis for the kernel of Hessian.

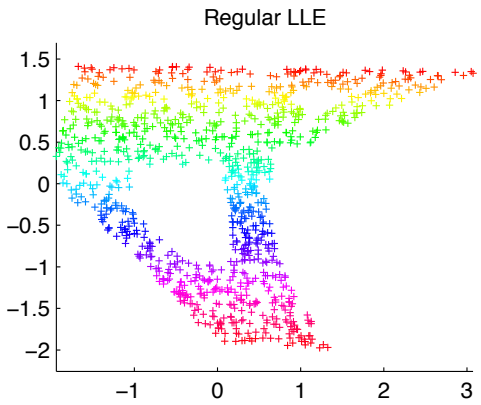
5 ***Step 3***: Define

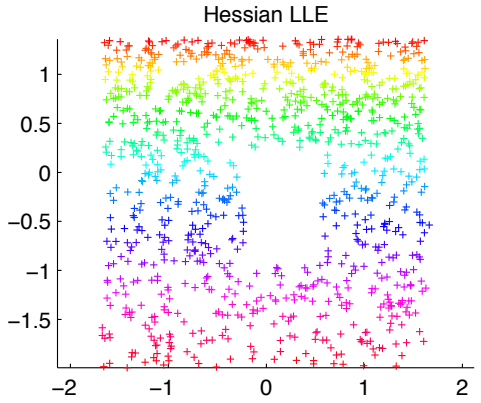
$$K = \sum_{i=1}^{n} S^{(i)} H^{(i)T} H^{(i)} S^{(i)T} \in \mathbb{R}^{n\times n}, \quad [x_1, .., x_n]S^{(i)} = [x_{i_1}, ..., x_{i_k}],$$

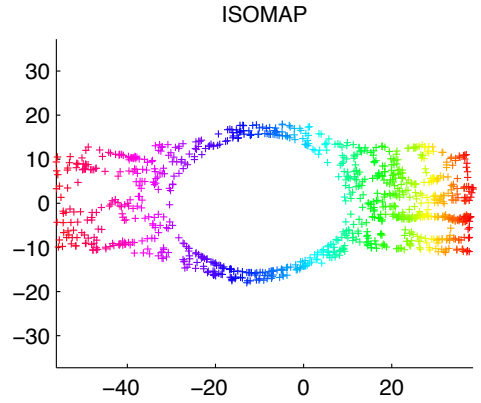
find smallest $d+1$ eigenvectors of K and drop the smallest eigenvector, the remaining d eigenvectors will give rise to a d-embedding.

Comparisons on Swiss Roll with holes









Two Assumptions on ISOMAP

(ISO1) *Isometry.* The mapping ψ preserves geodesic distances. That is, define a distance between two points m and m' on the manifold according to the distance travelled by a bug walking along the manifold M according to the shortest path between m and m'. Then the isometry assumption says that

$$G(m, m') = |\theta - \theta'|, \qquad \forall m \leftrightarrow \theta, m' \leftrightarrow \theta',$$

where $|\cdot|$ denotes Euclidean distance in $\mathbb{R}^d$.

(ISO2) *Convexity.* The parameter space Θ is a convex subset of $\mathbb{R}^d$. That is, if θ, θ' is a pair of points in Θ, then the entire line segment $\{(1-t)\theta + t\theta' : t \in (0,1)\}$ lies in Θ.

Convexity is hard to meet: consider two balls in an image which never intersect, whose center coordinate space (x_1,y_1,x_2,y_2) must have a hole.

Relaxations (Donoho-Grimes'2003)

(LocISO1) *Local Isometry.* In a small enough neighborhood of each point m, geodesic distances to nearby points m' in M are identical to Euclidean distances between the corresponding parameter points θ and θ'.

(LocISO2) *Connectedness.* The parameter space Θ is a open connected subset of $\mathbb{R}^d$.

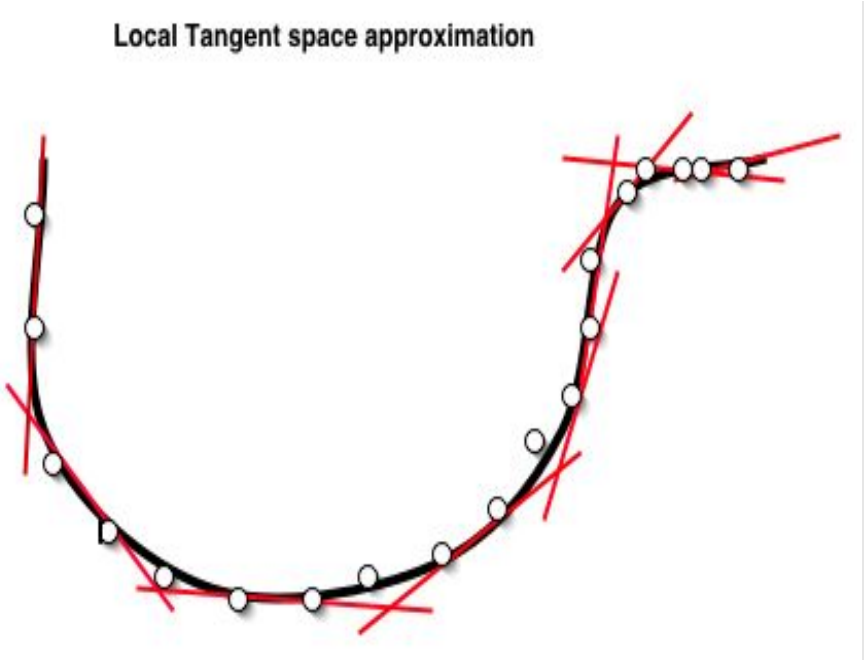
Convergence of Hessian LLE (Donoho-Grimes)

Theorem 1 *Suppose $M = \psi(\Theta)$ where Θ is an open connected subset of $\mathbb{R}^d$, and ψ is a locally isometric embedding of Θ into $\mathbb{R}^n$. Then $\mathcal{H}(f)$ has a $d+1$ dimensional nullspace, consisting of the constant function and a d-dimensional space of functions spanned by the original isometric coordinates.*

We give the proof in Appendix A.

Corollary 2 *Under the same assumptions as Theorem 1, the original isometric coordinates θ can be recovered, up to a rigid motion, by identifying a suitable basis for the null space of $\mathcal{H}(f)$.*

Local Tangent Space Alignment



$$\min_{Y} \sum_{i \sim j} \|y_i - U_i U_j^T y_j\|^2$$

where U_i is a local PCA basis for tangent space at point $x_i \in \mathbb{R}^p$.

LTSA Algorithm (Zha-Zhang'02)

Algorithm 4: LTSA Algorithm

Input: A weighted undirected graph $G = (V, E, d)$ such that

1 $V = \{x_i \in \mathbb{R}^p : i = 1, \ldots, n\}$

2 $E = \{(i, j) :$ if j is a neighbor of i, i.e. $j \in \mathcal{N}_i\}$, e.g. k-nearest neighbors

Output: Euclidean k-dimensional coordinates $Y = [y_i] \in \mathbb{R}^{k \times n}$ of data.

3 ***Step 1***: Compute local PCA on neighborhood of x_i, $x_{i_j} \in \mathcal{N}(x_i)$,

$$\tilde{X}^{(i)} = [x_{i_1} - \mu_i, ..., x_{i_k} - \mu_i]^{p \times k} = \tilde{U}^{(i)} \tilde{\Sigma} (\tilde{V}^{(i)})^T,$$

where $\mu_i = \sum_{j=1}^{k} x_{i_j} = \frac{1}{k} X_i \mathbf{1}$, $\tilde{U}^{(i)} = [\tilde{U}_1^{(i)}, ..., \tilde{U}_k^{(i)}]$ is an approximate tangent space at x_i. Define

$$G_i = [1/\sqrt{k}, \tilde{V_1}^{(i)}, ..., \tilde{V_d}^{(i)}]^{k \times (d+1)};$$

4 ***Step 2***: Alignment (kernel) matrix

$$K^{n \times n} = \sum_{i=1}^{n} S_i W_i W_i^T S_i^T, \quad W_i^{k \times k} = I - G_i G_i^T,$$

where selection matrix $S_i^{n \times k} : [x_{i_1}, ..., x_{i_k}] = [x_1, ..., x_n] S_i^{n \times k}$;

5 ***Step 3***: Find smallest $d + 1$ eigenvectors of K and drop the smallest eigenvector, the remaining d eigenvectors will give rise to a d-embedding.

From LTSA to Connection Laplacian

LTSA (Zhang-Zha'02):

$$\min_Y \sum_{i\sim j} \|y_i - U_i U_j^T y_j\|^2$$

where U_i is a local PCA basis for tangent space at point $x_i \in \mathbb{R}^p$.

Vector Connection Laplacian (Singer-Wu'12):

$$\min_Y \sum_{i\sim j} \|y_i - O_{ij} y_j\|^2, \quad O_{ij} = \arg\min_O \|U_i - O_{ij} U_j\|^2$$

where U_i is a local PCA basis for tangent space at point $x_i \in \mathbb{R}^p$.

Comparisons of Manifold Learning Techniques

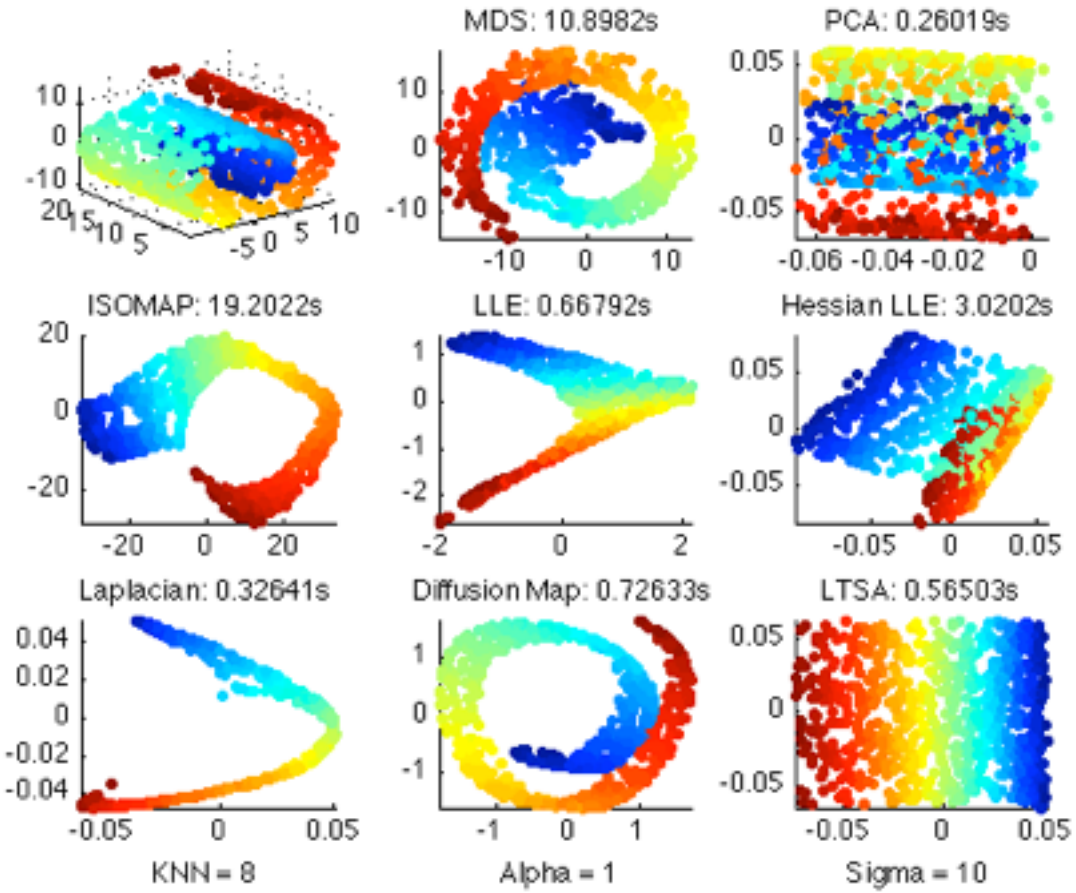
- MDS
- PCA
- ISOMAP
- LLE
- Hessian LLE
- Laplacian LLE
- Diffusion Map
- Local Tangent Space Alignment
- Matlab codes: mani.m

Courtesy of Todd Wittman

How To Compare

- Speed
- Manifold Geometry
- Non-convexity
- Curvature
- Corners
- Noise
- Non-uniform Sampling
- Sparse Data
- Clustering
- High-Dimensional Data: *Can the method process image manifolds?*
- Sensitivity to Parameters
  - K Nearest Neighbors: *Isomap, LLE, Hessian, Laplacian, KNN Diffusion*
  - Sigma: *Diffusion Map, KNN Diffusion*

Speed on Swiss Roll



Now go to Todd Wittman's slides, page 13th…

Reference

- Tenenbaum, de Silva, and Langford, A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science* 290:2319-2323, 22 Dec. 2000.
- Roweis and Saul, Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* 290:2323-2326, 22 Dec. 2000.
- M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation, 15:1373–1396, 2003.
- D. L. Donoho and C. Grimes. Hessian eigenmaps: New locally linear embedding techniques for high-dimensional data. PNAS 100 (10): 5591–5596 2003.
- R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. PNAS 102 (21):7426-7431, 2005.
- M. Hein, J.-Y. Audibert, and U. von Luxburg. From graphs to manifolds - weak and strong pointwise consistency of graph Laplacians. In P. Auer and R. Meir, editors, Proc. of the 18th Conf. on Learning Theory (COLT), pages 486–500, Berlin, 2005. Springer.
- Ulrike von Luxburg, Mikhail Belkin, and Olivier Bousquet. Consistency of spectral clustering. Ann. Statist. Volume 36, Number 2 (2008), 555-586.

Reference

- M. Belkin and P. Niyogi. Convergence of Laplacian Eigenmaps. 2006. Short version NIPS 2008.
- Singer, Amit. From graph to manifold Laplacian: The convergence rate. *Applied and Computational Harmonic Analysis*. 2006.
- Zhenyue Zhang and Hongyuan Zha, Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment, SIAM Journal of Scientific Computing, 2002
- Singer, Amit and Hau-Tieng Wu, Vector Diffusion Map and the Connection Laplacian. Communications on Pure and Applied Mathematics, 65 (8): 1067-1144, 2012. Matlab VDM codes downloaded at
  - https://sites.google.com/site/hautiengwu/home/download

Acknowledgement

- Slides stolen from M. Belkin, R. Coifman, G. Lerman,et al.