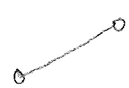
Nonlinear Manifolds, ISOMAP, and LLE

"Def": A (topological) manifold is (topological) space which locally resembles Euclidean space. Each point on an n-dimensional manifold has a neighborhood that can be mapped (continuous, continuous inverse) to $\mathbb{R}^n$.

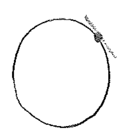
Ex:

One-Dimensional Manifolds

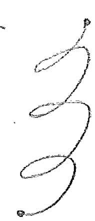
Line - Segment



Circle in $\mathbb{R}^2$

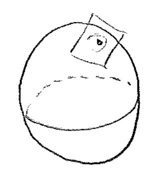


Helix

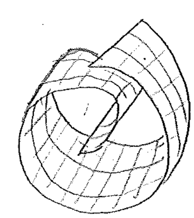


Two-Dimensional Manifolds

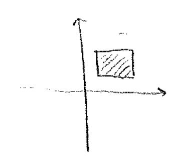
Sphere in $\mathbb{R}^3$



Jelly Roll in $\mathbb{R}^3$



Rectangle in $\mathbb{R}^2$



Disk

k-Dimensional Manifold

Span of $q_1 \ldots q_k \in \mathbb{R}^p$

linearly ind. vectors

Ex. NOT A MANIFOLD, FIGURE 8

INTERSECTION NOT MAPPABLE TO EUCLIDEAN SPACE

"DEF" A <u>SMOOTH</u> (DIFFERENTIABLE) MANIFOLD IS A MANIFOLD WHICH IS LOCALLY SIMILAR ENOUGH TO EUCLIDEAN SPACE TO DO CALCULUS.

(EVERYTHING WE'LL CONSIDER)

DEF: A METRIC d^M, ON A MANIFOLD, M, MEASURES THE DISTANCE BETWEEN POINTS. A RIEMANNIAN MANIFOLD, (M, d^M), IS A SMOOTH MANIFOLD W/ A METRIC

Ex.

<u>LINE</u>

$M = (0,1)$

$d^M(a,b) = |a-b| \quad a,b \in M$

<u>HELIX</u>:

$M =$ HELIX IN $\mathbb{R}^2$

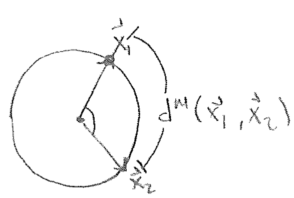
$d^M(a,b) =$ arclength OF SEGMENT BETWEEN a AND b

<u>CIRCLE</u>

$M = \{ \vec{x} \in \mathbb{R}^2 : x_1^2 + x_2^2 = 1 \}$

$d^M(\vec{x}_1, \vec{x}_2) = \cos^{-1}(\vec{x}_1^T \vec{x}_2)$

$=$ ANGLE BETWEEN $\vec{x}_1$ AND $\vec{x}_2$



$*$ d^M IS THE LENGTH OF THE SHORT CURVE (GEODESIC) BETWEEN TWO POINTS ON M.

Embedding Manifolds

Thm: (Nash, 1965) A smooth manifold M can be embedded in a higher dimensional Euclidean space.

Ex: Embedding a line in $\mathbb{R}^1$ in $\mathbb{R}^3$

$$M = (0, 100) \subset \mathbb{R}^1$$

Embedding map

$$\begin{array}{cccc} \psi : & M & \longrightarrow & \mathbb{R}^3 \\ & \rotatebox{90}{$\in$} & & \rotatebox{90}{$\in$} \\ & t & \longrightarrow & \begin{bmatrix} \cos(t) \\ \sin(t) \\ t \end{bmatrix} \end{array}$$

$$\begin{array}{ccc} \text{LINE} & \longrightarrow & \text{HELIX} \\ \text{"1-DIMENSIONAL"} & & \text{"1-DIMENSIONAL"} \end{array}$$

Ex: Embedding a rectangle in $\mathbb{R}^3$ into a ~~Swiss~~ Jelly roll in $\mathbb{R}^3$

$$M = \left[\frac{3\pi}{2}, \frac{9\pi}{2}\right] \times [0, 15] \in \mathbb{R}^2$$

$$\begin{array}{cccc} \psi : & M & \longrightarrow & \mathbb{R}^3 \\ & (y_1, y_2) & \longrightarrow & \begin{bmatrix} y_1 \cos(y_1) \\ y_1 \sin(y_1) \\ y_2 \end{bmatrix} \end{array}$$

$$\begin{array}{ccc} \text{PLANE} & \longrightarrow & \text{"ROLLED UP" PLANE} \\ \text{2D} & \longrightarrow & \text{2D} \end{array}$$

DATA ON MANIFOLDS

ASSUME

- $\vec{y}_1, \ldots \vec{y}_N$ ARE RANDOMLY SAMPLED FROM A SMOOTH t-DIMENSIONAL MANIFOLD W/ METRIC d^M.

- THESE POINTS ARE (NONLINEARLY) EMBEDDED INTO A HIGH-DIMENSIONAL INPUT SPACE $\mathcal{X} \subset \mathbb{R}^r$ $(t \ll r)$ BY A SMOOTH MAP ψ GIVING DATA

$$\vec{x}_i = \psi(\vec{y}_i)$$

- OBSERVATIONS
- DISTANCE IN $\mathcal{X}$ MEASURED USING EUCLIDEAN DISTANCE

- TRUE (UNKNOWN) LOWER DIMENSIONAL REPRESENTATION

- UNKNOWN EMBEDDING FUNCTION

GOAL:

GIVEN $\vec{x}_1, \ldots \vec{x}_N$ RECOVER $\vec{y}_1, \ldots \vec{y}_N$

($\vec{x}$ RECOVER M AND IMPLICIT REPRESENTATION OF ψ i.e. $\vec{x} = \psi(\vec{y})$, $\vec{y} = \psi^{-1}(\vec{x})$)

NOTE: THE TRUE DIMENSION t IS TYPICALLY UNKNOWN.

WE CAN

1) PICK A DIMENSION FOR VISUALIZATION $(t = 1, 2, 3)$, OR
2) GENERATE FITS AND COMPARE FOR MANY t THEN CHOOSE

* ESTIMATING TRUE t IS AN ACTIVE AREA OF RESEARCH!!

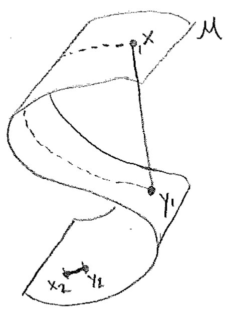
NOTE: HEREAFTER WE'LL ASSUME THE METRIC ON A MANIFOLD IS GIVEN BY THE SMALLEST ARCLENGTH

$$d^{\mathcal{M}}(x,y) = \inf_{C(x,y)} L(C(x,y))$$

$C(x,y)$ = ALL SMOOTH PATHS ON $\mathcal{M}$ STARTING AT x AND ENDING AT y

$$L(C(x,y)) = \int \left\| \frac{dc}{dt} \right\| dt = \underline{\text{arclength}} \text{ OF PATH}$$

Ex.



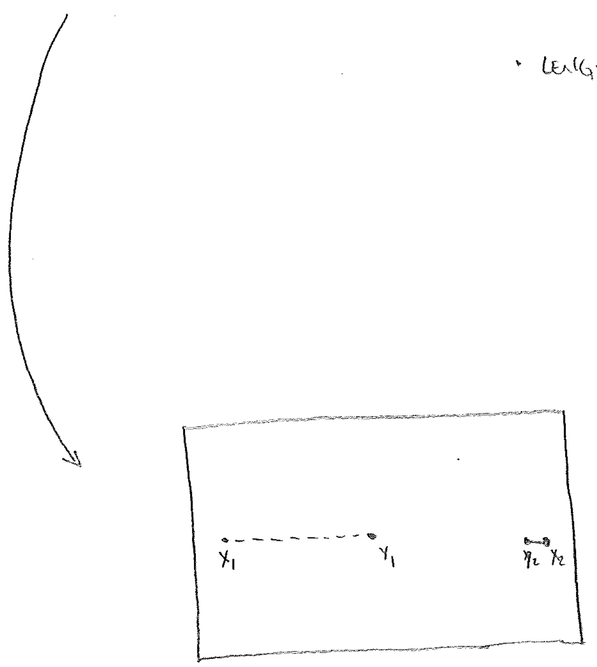
--- IS SHORTEST (BY ARCLENGTH) PATH ON $\mathcal{M}$ CONNECTING x_1 AND y_1

- LENGTH OF --- $>$ $\|x_1 - y_1\|_2$ ← EUCLIDEAN DISTANCE

— IS SHORTEST (BY ARCLENGTH) PATH ON $\mathcal{M}$ CONNECTING x_2 AND y_2

- LENGTH OF — $\|x_2 - y_2\|_2$

ψ^{-1}



ISOMAP (ISOMETRIC FEATURE MAPPING) (TENENBAUM, DE SILVA, LANGFORD 2000)

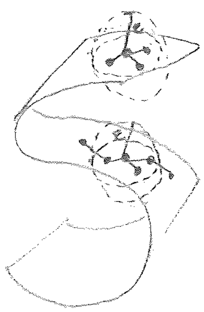
ALGORITHM

1) NEIGHBORHOOD GRAPH FIX INTEGER K OR $\varepsilon > 0$. CALCULATE ALL PAIRED (EUCLIDEAN) DISTANCES

TUNING PARAMETERS

$$d_{ij} = \|\vec{x}_i - \vec{x}_j\|_2$$

EUCLIDEAN

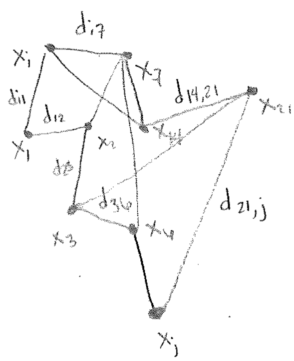


DETERMINE WHICH POINTS ARE "NEIGHBORS" ON THE MANIFOLD $\mathcal{X} = \Psi(\mathcal{U})$ BY CONNECTING EACH POINT TO

i) ITS K-NEAREST NEIGHBORS, OR

ii) ALL POINTS WITHIN DISTANCE ε (EXCLUDING ITSELF)

THIS GIVES A WEIGHTED NEIGHBORHOOD GRAPH $G = G(V, \mathcal{E})$



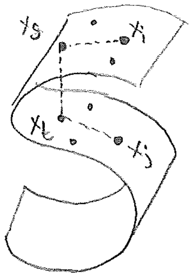
- V = NODES/VERTICES CORRESPONDING TO OBSERVATIONS $\vec{x}_1, \ldots \vec{x}_N$

- $\mathcal{E}$ = WEIGHTED EDGES SO THAT $\varepsilon_{ij} = d_{ij}$ IF NODES $\vec{x}_i$ & $\vec{x}_j$ ARE CONNECTED i.e. IF

  - x_i AND x_j ARE K-NEAREST NEIGHBORS
  - OR $d_{ij} < \varepsilon$

"DELETE ALL EUCLIDEAN DISTANCE THAT ARE TOO LONG, i.e. DON'T MATCH GEODESIC DISTANCE!"

2) <u>COMPUTE GRAPH DISTANCES</u> ESTIMATE THE UNKNOWN TRUE <u>GEODESIC DISTANCES</u>, $\{d_{ij}^{\mathcal{X}}\}$, BETWEEN PAIRS OF POINTS IN $\mathcal{X}$, BY GRAPH DISTANCES, $\{d_{ij}^{G}\}$, WITH RESPECT TO GRAPH G



$$d_{ij}^{\mathcal{X}} \approx d_{ij}^{G} = \text{SHORTEST PATH DISTANCE BETWEEN } i\text{th AND } j\text{th NODES IN } G$$

POINTS WHICH ARE NOT NEIGHBORS (CONNECTED DIRECTLY) ARE CONNECTED BY A SERIES OF LINKS

$$x_i \xrightarrow{d_{is}} x_s \xrightarrow{d_{st}} x_t \xrightarrow{d_{tj}} x_j$$

$$d_{ij}^{\mathcal{X}} \approx d_{ij}^{G} = d_{is} + d_{st} + d_{tj}$$

<u>NOTE:</u>

1) SHORTEST PATHS IN A GRAPH CAN BE COMPUTED BY FLOYD'S ALGORITHM ($O(N^3)$) OR DIJKSTRA'S ALGORITHM ($O(kN^2 \log N)$)

2) UNDER SOME ASSUMPTIONS

$$d_{ij}^{G} \xrightarrow{N \to \infty} d_{ij}^{\mathcal{X}}$$

(MORE ON THIS LATER). WE CAN APPROXIMATE GEODESIC DISTANCE (GLOBAL STRUCTURE) USING <u>MANY</u> SMALL EUCLIDEAN DISTANCES (LOCAL STRUCTURE)

3) GIVES RISE TO A DISTANCE MATRIX $D \in \mathbb{R}^{N \times N}$

3) <u>EMBEDDING VIA MDS:</u> TYPICALLY USE CLASSICAL SCALING

ASSESS PERFORMANCE OF ISOMAP BY PLOTTING $1 - R_t^2$, $R_t^2 = \left[\mathrm{corr}\left(D_t^{Y}, D^{G}\right)\right]^2$ AGAINST t FOR $t = 1, 2, \ldots, t^* < N$, D_t^{Y} IS DISTANCES OF ALL PAIRS IN t-DIMENSIONAL CONFIGURATION.

ASSUMPTIONS

1) ISOMETRY: THE MAP $\Psi : \mathcal{M} \longmapsto \mathcal{X}$ IS AN ISOMETRIC EMBEDDING

i.e. Ψ PRESERVES INFINITESIMAL ANGLES AND LENGTHS!

(NOTE: STATEMENT IN IZENMAN EQ. (16.36) IS WRONG/MISLEADING

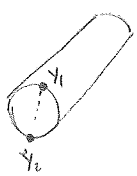
$$d^{\mathcal{M}}(\vec{x}, \vec{y}) = \| \Psi(x) - \Psi(y) \|_2$$ HOLDS INFINITESIMALLY!)

2) CONVEXITY: THE LOWER DIMENSIONAL MANIFOLD IS A CONVEX SUBSET OF $\mathbb{R}^t$ (BUT $\mathcal{X} = \Psi(\mathcal{M})$ NEED NOT BE CONVEX)

$$\vec{y}_1, \vec{y}_2 \in \mathcal{M} \implies s\vec{y}_1 + (1-s)\vec{y}_2 \in \mathcal{M} \quad \forall s \in (0,1) \quad \forall \vec{y}_1, \vec{y}_2 \in \mathcal{M}$$

Ex. NONVEX

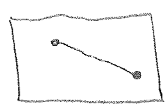
TUBE:



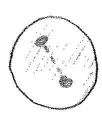
HELIX:

Ex: CONVEX

RECTANGLE:



DISK:



* FOR CONVEX SETS

$$\| \vec{y}_1 - \vec{y}_2 \| = d^{\mathcal{M}}(\vec{y}_1, \vec{y}_2) \,!$$

SHORTEST PATH BETWEEN 2 POINTS IS A STRAIGHTLINE

Ex: ISOMETRY

$$\mathcal{M} = (0, 100) \subset \mathbb{R}$$

$$\Psi : \mathcal{M} \to \mathbb{R}^3$$

$$t \xrightarrow{\Psi} \begin{bmatrix} \frac{1}{\sqrt{2}} \cos(t) \\ \frac{1}{\sqrt{2}} \sin(t) \\ \frac{1}{\sqrt{2}} t \end{bmatrix}$$

- $a, b \in \mathcal{M} \implies d^{\mathcal{M}}(a, a + \Delta a) = |\Delta a|$

- $d^{\mathcal{X}}(\Psi(a), \Psi(a + \Delta a))$

$$= \left| \int_a^{a+\Delta a} \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2} \, dt \right|$$

$$= \left| \int_a^{a+\Delta a} \sqrt{\frac{1}{2} + \frac{1}{2}} \, dt \right| = |\Delta a|$$

ANGLE BETWEEN NEARBY VECTORS IS PRESERVED

(ALL ANGLES IN $\mathcal{M}$ ARE 0!)

Strengths

- When assumptions are met : sampling on $\mathcal{X}$ is sufficiently dense so that approx.

$$d_{ij}^{G} \approx d_{ij}^{\mathcal{X}}$$

is good, then ISOMAP performs well

- Question: What kinds of embeddings satisfy isometry assumption?

  :) Distance preserving $\longrightarrow$ rotations, folds/twists, rolls

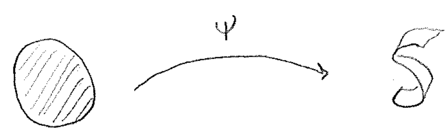
  :-| Uniform stretching $\longrightarrow$ $\vec{y}_i \longrightarrow c\vec{y}_i \longrightarrow \psi(c\vec{y}_i) = \vec{x}_i$

  ISOMAP

- Question: What restrictions does convexity assumption add?

  ISOMAP works best for intrinsically flat M

ψ



Sheet   "Rolled up sheet"

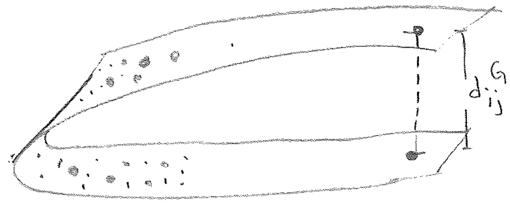
"Geodesic distance"   "Geodesic distance"

Weaknesses

- Computational: Floyd's algorithm : Classical MDS $\rightarrow$ $\mathcal{O}(N^3)$ computations

  Dijkstra $\mathcal{O}(kN^2 \log N)$ Landmark ISOMAP

Highly Curved $\mathcal{X}$ / Insufficient Samples / ε, K too big



Dense Sampling   Sparse Sampling

* Proofs regarding require δ covering of $\mathcal{X}$, i.e.

  $\forall \vec{x} \in \mathcal{X}$ there is a sample $\vec{x}_i$ s.t. $\|\vec{x} - \vec{x}_i\|_2 < \delta$,

  which may require a lot of samples

* Setting K / ε too large can cause the same issue

Parting Thoughts on ISOMAP

* ISOMAP seeks to preserve geodesic distances between all pairs of points

  1) x, y far apart (via geodesic distance) on $\mathcal{X}$

     $\rightarrow$ preserving these relationship(s) on $\mathcal{M}$ is global

  2) Convexity requirement enforces

     "Geodesic" "=" "Euclidean"

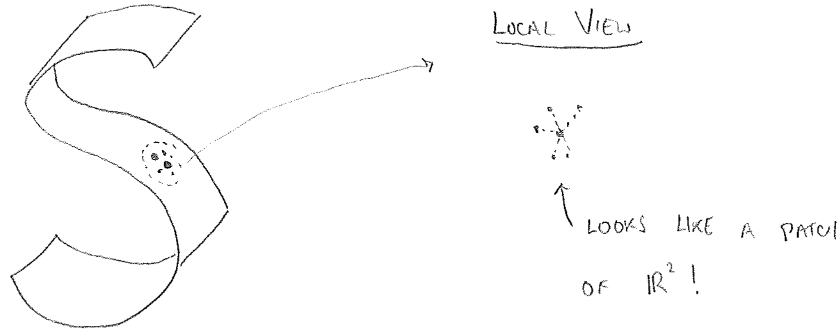
     is low dimensional coordinates

Ex: Swiss roll, helix, folded and rolled washers

See Canvas > Files > Numerical Examples > ISOMAP_LLE

LLE (Local Linear Embedding) (Roweis, Saul 2001)

Motivation



* Build a low dimensional representation that preserves the local geometry between points

i.e. cut a patch → learn geometry
" " "

"stitch patches together in lower dimensions"

Algorithm

1) Neighborhood Graph (same as Isomap)

- Find all pairwise Euclidean distance. For each point save only the K nearest points, $N_i^K \subset \{1, .., i-1, i+1, .. N\}$
- Want K small enough so that patch containing $\vec{x}_j$, $j \in N_i^k$ has little curvature, i.e. looks like a (hyper) plane

2) <u>CONSTRAINED LEAST - SQUARES FIT:</u>

RECONSTRUCT $\vec{x}_i$ BY A LINEAR COMBINATION OF ITS K-NEAREST NEIGHBORS

$$\vec{x}_i \approx \hat{x}_i = \sum_{j=1}^{N} w_{ij} \vec{x}_j \qquad \underline{w_{ij} = 0 \quad \text{IF} \quad j \notin N_i^k}$$

WITH THE CONSTRAINT $\sum_j w_{ij} = 1$.

LETTING $W \in \mathbb{R}^{N \times N}$ BE THE MATRIX OF WEIGHTS, WE WANT TO MINIMIZE

$$E(W) = \sum_{i=1}^{N} \left\| \vec{x}_i - \sum_{j=1}^{N} w_{ij} \vec{x}_j \right\|_2^2$$

SUBJECT TO THE CONSTRAINTS

$$W 1_N = 1_N$$
$$W_{ij} = 0 \quad j \notin N_i^k \quad i = 1, \dots, N$$

<u>NOTE:</u>

1) <u>ROTATIONS/RESCALING:</u> THE MINIMIZER OF $E(W)$ IS UNCHANGED IF $\vec{x}_1, \dots, \vec{x}_N$ ARE ROTATED OR RESCALED

- $a \in \mathbb{R}$, $a \neq 0$

$$\sum_{i=1}^{N} \left\| a\vec{x}_i - \sum_{j=1}^{N} w_{ij} a \vec{x}_j \right\|_2^2 = a^2 \sum_{i=1}^{N} \left\| \vec{x}_i - \sum_j w_{ij} x_j \right\|_2^2$$

- U IS ORTHONORMAL

$$\sum_{i=1}^{N} \left\| U\vec{x}_i - \sum_{j=1}^{N} w_{ij} U \vec{x}_j \right\|_2^2 = \sum_{i=1}^{N} \left\| U \left(\vec{x}_i - \sum_{j=1}^{N} w_{ij} x_j \right) \right\|_2^2$$
$$= \sum_{i=1}^{N} \left\| \vec{x}_i - \sum_{j=1}^{N} w_{ij} \vec{x}_j \right\|_2^2$$

2) SUM TO 1 CONSTRAINT ENFORCES INVARIANCES TO TRANSLATIONS

$$\sum_{i=1}^{N} \left\| (x_i + \vec{a}) - \sum_{j=1}^{N} w_{ij} (\vec{x}_j + \vec{a}) \right\| = \sum_{i=1}^{N} \left\| \vec{x}_i - \sum_{j=1}^{N} w_{ij} \vec{x}_j \right\|_2^2$$

FINDING WEIGHTS

WANT TO FIND

$$\hat{W} = \underset{W}{\operatorname{argmin}} \sum_{i=1}^{N} \left\| \vec{x}_i - \sum_{j=1}^{N} w_{ij} \vec{x}_j \right\| = \underset{W}{\operatorname{argmin}} \left\| \sum_{j=1}^{N} w_{ij} (\vec{x}_i - \vec{x}_j) \right\|^2$$

SUBJECT TO CONSTRAINTS $W \mathbb{1}_N = \mathbb{1}_N$, $W_{ij} = 0 \quad j \notin N_i^k$

- WE MINIMIZE ONE ROW AT A TIME

$$\left\| \sum_{j} w_{ij} (x_i - x_j) \right\|^2 = \vec{w}_i^T G_i \vec{w}_i$$

$$w_i = (w_{i1}, \dots w_{iN})^T \qquad (i^{th} \text{ ROW OF } W)$$

$$G \in \mathbb{R}^{N \times N}, \quad G_{jk} = \begin{cases} (x_i - x_j)^T (x_i - x_k) & j, k \in N_i^k \\ 0 & \text{ELSE} \end{cases}$$

VERY SPARSE

LAGRANGE MULTIPLIERS

$$f(\vec{w}_i) = w_i^T G \vec{w}_i - \lambda (\mathbb{1}_N^T \vec{w}_i - 1)$$

$$\nabla_{w_i} f = 2G \vec{w}_i - \lambda \mathbb{1}_N = 0$$

$$\Rightarrow G \vec{w}_i = \frac{\lambda}{2} \mathbb{1}_N$$

NOT INVERTIBLE BUT SUBMATRIX G_{jk} TYPICALLY IS!

$$"\ \vec{w}_i = \frac{G_i^{-1} \mathbb{1}_N}{\mathbb{1}_N^T G_i^{-1} \mathbb{1}_N}\ "$$

$$\left(\begin{array}{l} \text{IF} \quad \bar{G}_i \in \mathbb{R}^{k \times k} \quad \text{IS} \quad \bar{G}_{jk} = (x_i - x_j)^T (x_i - x_k) \\ \bar{w}_i \in \mathbb{R}^k \quad \text{IS} \quad \vec{w}_i = (w_{ij_1}, w_{ij_2}, \ldots w_{ij_k})^T \qquad j_1, \ldots j_k \in N_i^k \\ \qquad \bar{w}_i = \dfrac{\bar{G}^{-1} \cdot 1_k}{1_k^T \bar{G}^{-1} 1_k} \quad \text{RESCALE SO} = 1 \end{array} \right)$$

REPEAT FOR $i = 1, \ldots N$ TO OBTAIN $\hat{W}$

3) <u>EIGENPROBLEM:</u>

IF WEIGHTS W_{ij} $j = 1, \ldots N$ REFLECT THE LOCAL, INTRINSIC GEOMETRY NEAR $\vec{x}_i$. THUS, IF $\vec{y}_1, \ldots y_N$ ARE THE LOWER DIMENSIONAL REPRESENTATION OF THE MANIFOLD,

$\vec{y}_1, \ldots \vec{y}_N$ SHOULD <u>MINIMIZE</u> $\displaystyle\sum_{i=1}^{N} \left\| \vec{y}_i - \sum_{j=1}^{N} \hat{w}_{ij} y_j \right\|^2$

* NOW $\hat{W}$ IS FIXED, OPTIMIZE OVER $\vec{y}_1, \ldots \vec{y}_N$

WANT TO FIND $\vec{y}_1, \ldots y_N \in \mathbb{R}^d$ <u>MINIMIZING</u>

$$\sum_{i=1}^{N} \left\| \vec{y}_i - \sum_{j=1}^{N} \hat{w}_{ij} \vec{y}_j \right\|^2 = \left\| Y - WY \right\|^2 \qquad Y = \begin{bmatrix} \vec{y}_1^{\,T} \\ \vdots \\ y_N^T \end{bmatrix}$$

$$= \left\| (I - W) Y \right\|^2 = \operatorname{tr}\left(Y^T (I - \hat{W})^T (I - W) Y \right)$$

W/ CONSTRAINTS $\underbrace{1_N^T Y = \vec{0}}_{\substack{\text{MEAN } 0 \\ \text{FIXES TRANSLATION}}}$, $\underbrace{\frac{1}{N} \sum_{i=1}^{N} \vec{y}_i y_i^T = I_d}_{\text{FIXES ROTATION/SCALE}}$

$M = (I-W)^T(I-W)$ IS SYMMETRIC $\Rightarrow$ N ORTHOGONAL EIGENVECTORS!

<u>NOTE</u>: 1) $M = (I-W)^T(I-W)\mathbb{1}_N = \vec{0}$

- $\vec{V}_1$ IS EIGENVECTOR W/ MEAN ZERO!
- ALL OTHER EIGENVECTORS $\perp$ TO $\mathbb{1}_N$
- NOT GEOMETRICALLY MEANINGFUL $\longrightarrow$ USE REMAINING d EIGENVECTORS $\vec{V}_2, \ldots \vec{V}_{d+1}$

* SORTING EIGENVALUES (AND CORRESPONDING EIGENVECTORS) IN INCREASING ORDER

$$0 = \lambda_1 < \lambda_2 \leq \lambda_3 \ldots \leq \lambda_N$$

2) LET $\vec{V}_i$ BE NORMALIZED SO THAT

$$\|v_i\|^2 = N \quad \text{THEN}$$

$$Y = \begin{bmatrix} \vec{V}_2, \ldots \vec{V}_{d+1} \end{bmatrix} \quad \text{SATISFIES} \quad \mathbb{1}_N Y = \vec{0}^T$$

$$\frac{1}{N} Y^T Y = I_d .$$

LOWER DIMENSIONAL REPRESENTATION!

<u>EX</u>: SWISS ROLL, HELIX, FOLDED AND ROLLED WASHERS

SEE CANVAS > FILES > NUMERICAL EXAMPLES > ISOMAP_LLE

ASSESSING THE PERFORMANCE OF LLE & CHOOSING DIMENSION OF LOW-DIMENSIONAL REP

- RECALL, LLE SEEKS TO PRESERVE LOCAL BEHAVIOR THROUGH NEAREST NEIGHBOR RECONSTRUCTION.

- ONE METHOD OF MEASURING ERROR IS THE PRESERVATION OF NEAREST NEIGHBORS

  - $\vec{x}_1 \ldots \vec{x}_N$ ORIGINAL DATA IN HIGH-DIMENSIONS

    η_i^k = INDICES OF K NEAREST NEIGHBORS OF $\vec{x}_i$

  - $\vec{y}_1 \ldots y_N$ LOW-DIMENSIONAL COORDINATES

    ν_i^k = INDICES OF K NEAREST NEIGHBORS OF $\vec{y}_i$

  - RANK BASED SCORES (LEE, RENARD, ET AL. 2013)

RANDOM PROJECTION

PERFECT

$$Q_{NX}(K) = \sum_{i=1}^{N} \frac{|\nu_i^K \cap \eta_i^K|}{KN} \in [0, 1]$$

$$R_{NX}(K) = \frac{(N-1)\, Q_{NX}(K) - K}{N-1-K} \in$$

FOR FIXED K-NEAREST NEIGHBORS, WE CAN COMPUTE AT DIFFERENT DIMENSIONS OF $\vec{y}_1, \ldots \vec{y}_N$

- GENERATE DIMENSION V. ERROR PLOTS

  $\uparrow Q_{NX}, R_{NX}$

- CAN ALSO USE DISTANCE BASED METHODS (CONNECTION AS IN ISOMAP) BUT EUCLIDEAN (NOT GEODESIC) DISTANCE IS TYPICALLY USED IN MOST IMPLEMENTATIONS.