Lecture 2. Isomap and LLE

Juno Kim

Department of Mathematics & Statistics
Seoul National University

Manifold Learning, Spring 2022

Table of Contents

1 Isomap

2 Extensions (I)

3 LLE

4 Extensions (II)

Isomap

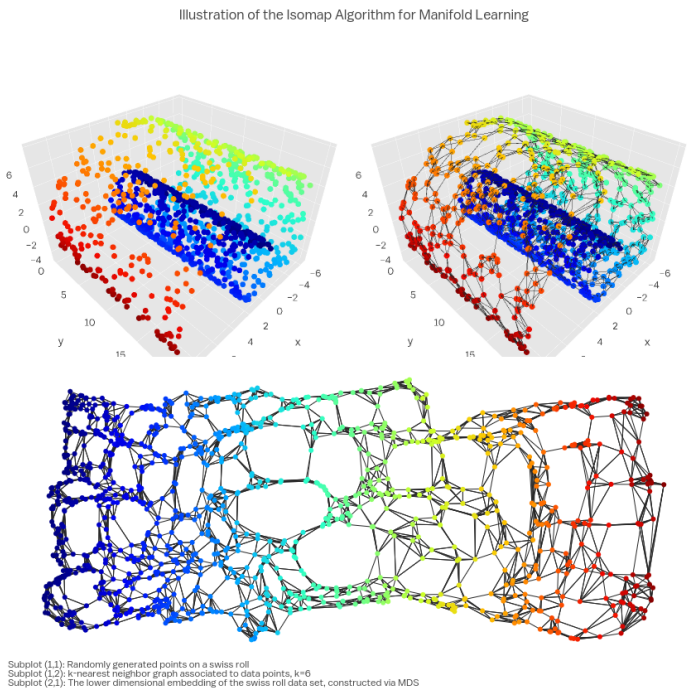

Figure: Applying Isomap to the Swiss roll.

- short for isometric feature mapping algorithm
- a nonlinear generalization of MDS using geodesics
- requires 2 key assumptions:

1. $\mathcal{M}$ is a convex subset of $\mathbb{R}^t$. In particular, $\mathcal{M}$ must be intrinsically flat and contains no holes.
2. The embedding map ψ is an isometry.

Step 1: Nearest-Neighbor Search

- Select an integer K or an $\epsilon > 0$
- Given data $x_1, \cdots, x_n \in \mathcal{X} = \mathbb{R}^r$ with r large, compute the Euclidean distances $d_{ij}^{\mathcal{X}} = \|x_i - x_j\|_{\mathcal{X}}$
- Determine which points are neighbors by (1) connecting each x_i to its K nearest points for all i, or (2) connecting all pairs x_i, x_j with $d_{ij}^{\mathcal{X}} < \epsilon$
- This gives a weighted undirected graph G with weights $d_{ij}^{\mathcal{X}}$
- Time complexity $O(r \log K \cdot n \log n)$ using BallTree

Step 2: Estimating Geodesic Distance

- Estimate the true geodesic distance $d_{ij}^{\mathcal{M}} = d^{\mathcal{M}}(x_i, x_j)$ for every pair of points by computing the *graph distance* d^G
- d_{ij}^G is defined as the length of the shortest path in G between x_i and x_j
- If the data is sampled from a p.d.f. fully supported on $\mathcal{M}$ and $\mathcal{M}$ is indeed flat, $d^G \to d^{\mathcal{M}}$ as $n \to \infty$.
- For dense G, use Floyd-Warshall algorithm: $O(n^3)$
- For sparse G, run Dijkstra's algorithm for each vertex: $O(Kn^2 \log n)$

Step 3: Spectral Embedding

- Apply MDS to the proximity data d_{ij}^G to obtain reconstructed points y_i in t-dimensional feature space $\mathcal{Y} = \mathbb{R}^t$.
- For $\mathbf{A}^G = (-\frac{1}{2}(d_{ij}^G)^2)$, compute $\mathbf{B}^G = \mathbf{H}\mathbf{A}^G\mathbf{H}$ and retrieve the eigenvectors u_j corresponding to its t largest eigenvalues λ_j.
- G is quasi-isometrically embedded into $\mathcal{Y}$ via:

$$(y_1, \cdots, y_n) = (\sqrt{\lambda_1}u_1, \cdots, \sqrt{\lambda_t}u_t)^T$$

- To choose t, plot the squared correlation coeffient $1 - R_t^2$ of the n^2 distances $d_{ij}^{\mathcal{Y},t} := \|y_i - y_j\|_{\mathcal{Y}}$ and d_{ij}^G, and identify the bend.
- Eigenvalue decomposition complexity is equal to matrix multiplication: in practice, $O(n^3) \sim O(n^{2.8})$ with Strassen

Remark

- Choice of K & ϵ dictates the success of Isomap. Large values can introduce false connections (short-circuits), a few of which may severely alter the spectral embedding.
- This issue also arises from noisy data, where outliers lie far away from the embedded manifold.
- Conversely, small values may create a graph too sparse to effectively approximate geodesics.
- Robustness may be achieved by preprocessing outliers, identifying problematic paths, incorporating global information, etc.

Table of Contents

1 Isomap

2 Extensions (I)

3 LLE

4 Extensions (II)

Landmark Isomap

- Isomap works best when $n \leq 10^3$. Efficiency is significantly compromised for much larger datasets.
- Shortest path calculations for every pair of points have large redundancies. L-Isomap selects m landmark points and only computes d^G from every point to each landmark.
- Landmarks may be randomly chosen, or selected to better represent the global geometry. In practice, $m \sim 50$.
- d^G estimation time is improved to $O(Kmn \log n)$

Landmark Isomap

We then apply 2-step 'Landmark-MDS.' Complexity: $O(m^2 n)$

Part 1. Apply classical MDS to the m landmarks $\ell_1, \cdots, \ell_m$ and their graph distances to obtain estimates $\hat{\ell}_i \in \mathbb{R}^t$, where $\mathbf{L} = (\hat{\ell}_1, \cdots, \hat{\ell}_m) = (\sqrt{\lambda_1} u_1, \cdots, \sqrt{\lambda_t} u_t)^T$.

Part 2. For each point x, use the graph distances $d_{i,x}^G$ between x and ℓ_i to embed $\hat{x}$ in $\mathbb{R}^t$, minimizing least-squares loss.

Exercise. Derive LMDS. Hint:

- Compute the ideal values α_i of $\ell_i^T x$ from $d_{i,x}^G$'s.
- Solve the optimization problem $\hat{x} = \text{argmin} \left\| \mathbf{L}^T x - \alpha \right\|$ using the Moore-Penrose inverse $\mathbf{L}^+$. Here, $\mathbf{L}^+ = (u_1/\sqrt{\lambda_1}, \cdots)$

TCIE

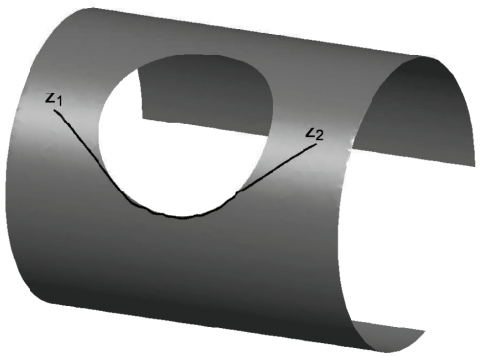

Figure: A geodesic curving around boundary.

- *Topologically constrained isometric embedding* (Rosman, 2008) relaxes the 1st condition: $\mathcal{M} \subset \mathbb{R}^t$ need not be convex.
- Holes in the manifold distort geodesics by forcing them to wrap around the boundary $\partial\hat{\mathcal{M}}$. TCIE discards such paths in the optimization process.

TCIE

We detect boundary points by comparing x to the set of its neighbors $N(x)$.

Method 1. (distance-based) $x \in \partial\hat{\mathcal{M}}$ if, for some separation δ,

$$\mu(x) := \left\| x - \frac{1}{|N(x)|} \sum\nolimits_{y \in N(x)} y \right\| > \delta$$

Method 2. (direction-based) $x \in \partial\hat{\mathcal{M}}$ if

$$\frac{1}{|N(x)|} \#\{z \in N(x) : \langle x - y, x - z \rangle > 0\}$$

exceeds a certain ratio θ for enough $y \in N(x)$

TCIE

- Numerically solve the following weighted optimization problem using gradient descent:

$$\text{argmin} \sum_{i,j} w_{ij}(d_{ij}^{G} - d_{ij}^{\mathcal{Y},t})^2$$

  where $w_{ij} = 0$ if the shortest path between x_i, x_j meets $\partial\hat{\mathcal{M}}$, and 1 otherwise.
- The problem is nonconvex and can converge to local minima such as folds – utilize *graduated optimization* ($w_{ij}^{(0)} \equiv 1 \ldots$)
- *Vector extrapolation* (Smith, 1987) or *multiresolution* (Platt, 2004) methods accelerate convergence.

C-Isomap

- *Conformal Isomap* (Silva, 2002) relaxes the 2nd condition: ψ only needs to be conformal.
- ψ is locally isometric up to a scale factor $s(x)$. Assume the hidden data is sampled uniformly from $(\mathcal{M}, d)$
- Observed local data density approximates $1/s(x)^t$
- $\sqrt{\mu(x_i)\mu(x_j)}$ is an asymtotically accurate estimator of s near neighbors x_i, x_j – and independent of t
- In Step 1 of Isomap, replace the graph weights by $\frac{\|x_i - x_j\|}{\sqrt{\mu(x_i)\mu(x_j)}}$

A conformal map between Riemannian manifolds $(\mathcal{M}, g)$, $(\mathcal{N}, h)$ is a diffeomorphism $f : \mathcal{M} \to \mathcal{N}$ which induces a conformal equivalence of metric tensors, i.e. $g = s \cdot f^* h$ for some $s > 0$.

C-Isomap

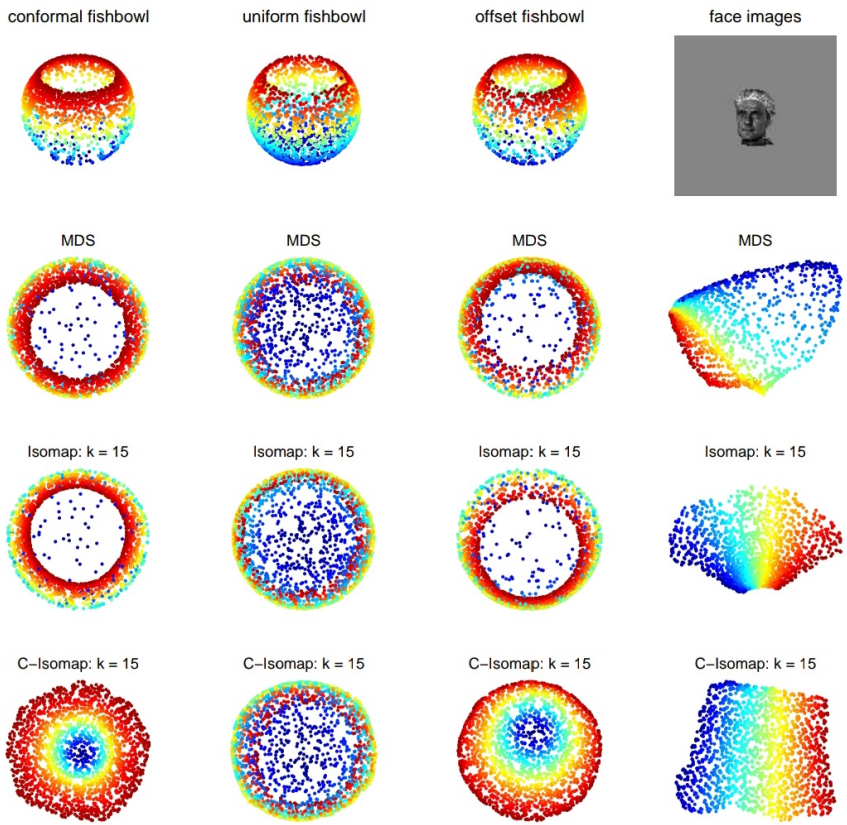

Figure: Fishbowl data (stereographic/uniform/offset) and face images (2 parameters: distance/direction). Rows: MDS/Isomap/C-Isomap.

Table of Contents

1 Isomap

2 Extensions (I)

3 LLE

4 Extensions (II)

LLE

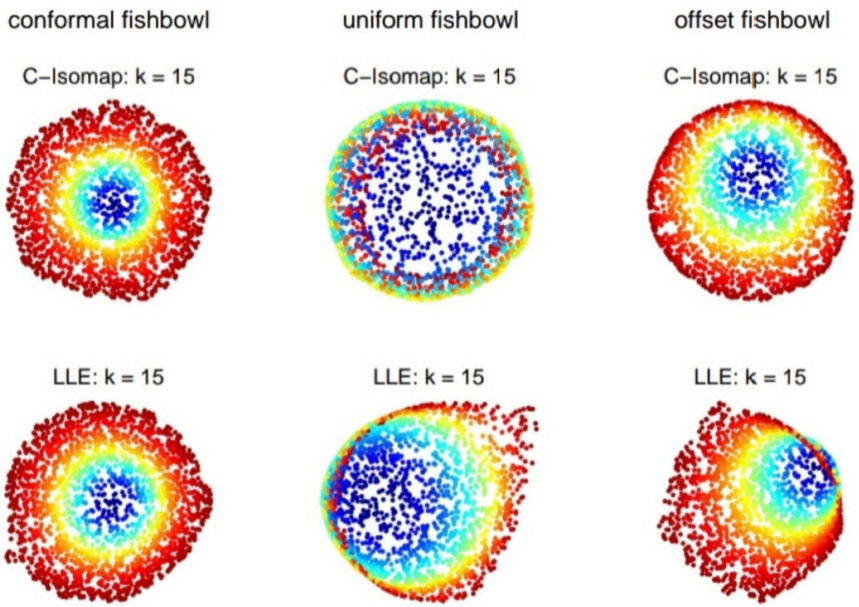

Figure: Fishbowl data, C-Isomap vs LLE.

Local linear embedding focuses on preserving local structure. It is more efficient and less susceptible to false connections than Isomap, but may distort the global geometry.

Step 2: Constrained Fit

- In Step 1 (KNN), fix $K \ll n$. Write $j \in N_i$ for $x_j \in N(x_i)$
- We reconstruct x_i by a linear combination of its neighbors $\sum_{j \in N_i} w_{ij} x_j$, where $\sum_{j \in N_i} w_{ij} = 1$ and $w_{i\ell} = 0$ if $\ell \notin N_i$
- Solve for optimal weights $\mathbf{W} = (w_{ij})$,

$$\hat{\mathbf{W}} = \text{argmin} \sum_{i=1}^{n} \|x_i - \sum_{j \in N_i} w_{ij} x_j\|^2$$

- The nonzero entries of the ith column of $\hat{\mathbf{W}}$ are given by the minimizer $\hat{w}_i$ of $w_i^T \mathbf{G}_i w_i$, where $1_K^T w_i = 1_K$ and

$$(\mathbf{G}_i)_{jk} = (x_i - x_j)^T (x_i - x_k), \quad j, k \in N_i$$

Exercise. Show that $\hat{w}_i = (1_K^T \mathbf{G}_i^{-1} 1_K)^{-1} \mathbf{G}_i^{-1} 1_K$ using Lagrange multipliers.

Step 3: Spectral Embedding

- Fix $\hat{\mathbf{W}}$ and find the $(t \times n)$-matrix $\mathbf{Y} = (y_1, \cdots, y_n)$ solving

$$\operatorname{argmin} \sum_{i=1}^{n} \|y_i - \sum_{j \in N_i} \hat{w}_{ij} y_j\|^2 = \operatorname{argmin} \operatorname{tr}(\mathbf{Y}\mathbf{M}\mathbf{Y}^T)$$

  subject to rigid motion invariance constraints $\bar{y} = \frac{1}{n}\mathbf{Y}1_n = 0$ and $\hat{\Sigma}_y = \frac{1}{n}\mathbf{Y}\mathbf{Y}^T = \mathbf{I}_t$

- where $\mathbf{M}$ is the symmetric matrix $(\mathbf{I}_n - \hat{\mathbf{W}})^T(\mathbf{I}_n - \hat{\mathbf{W}})$
- Note $\lambda_n(\mathbf{M}) = 0$ with corresponding eigenvector $\frac{1}{\sqrt{n}}1_n$. The remaining t *smallest* eigenvalues $\lambda_{n-t} > \cdots > \lambda_{n-1}$ give the optimal solution: $\hat{\mathbf{Y}} = (u_{n-t}, \cdots, u_{n-1})^T$
- The sparseness of $\mathbf{W}, \mathbf{M}$ allows for efficient computation. LLE can also embed some non-flat, nonconvex manifolds.

Table of Contents

1 Isomap

2 Extensions (I)

3 LLE

4 Extensions (II)

Gram Matrix

Here, we recall some properties of the Gram matrix $\mathbf{G} = \mathbf{X}^T\mathbf{X}$, $X = (x_1, \cdots, x_n)$, where $x_i \in \mathbb{R}^r$. Note that $\mathbf{G}_i$ is a 'translated' Gram matrix.

- $\mathbf{G}$ is positive semi-definite: $\alpha^T\mathbf{G}\alpha = \|\sum \alpha_i x_i\|^2 \geq 0$
- Any positive semi-definite matrix can be realized as a Gram matrix via Cholesky decomposition. This representation is unique up to $\mathbf{O}(k)$ for fixed k.
- $\operatorname{rank}\mathbf{G} = \dim\operatorname{col}\mathbf{X}$ and $\det\mathbf{G} = \|x_1 \wedge \cdots \wedge x_n\|^2$

Thus, since $N(x_i)$ lies close to an affine subspace of dimension t (i.e. the tangent space at x_i), $\mathbf{G}_i$ may become singular if $K > t$.

Exercise. Prove the above properties.

Modified LLE

- Two problems arise when $\mathbf{G}_i$ is (almost) singular: (1) $\hat{w}_i$ is numerically unstable, and (2) multiple approximately optimal weight vectors exist, some yielding wrong embeddings.
- For (1), MLLE (Zhang, 2006) solves a regularized system:

$$(\mathbf{G}_i + \gamma \|\sqrt{\mathbf{G}_i}\|_F^2 \mathbf{I}_K) v_i = 1_K, \quad \hat{w}_i = \frac{v_i}{1_K^T v_i} \quad (\gamma > 0)$$

- If $\mathbf{G}_i$ has s_i near-zero singular values, then s_i independent approximately optimal weights $\hat{w}_i^{(\ell)}$, $\ell \leq s_i$ exist. For (2), MLLE minimizes the following total cost:

$$MSSE(\mathbf{Y}) = \sum_{i=1}^{n} \sum_{\ell=1}^{s_i} \|y_i - \sum_{j \in N_i} \hat{w}_i^{(\ell)} y_j\|^2$$

LTSA

- *Local tangent space alignment* (Zhang, 2004) attempts to preserve local structure as exemplified by tangent hyperplanes, without calculating weights directly.
- LTSA first performs PCA on each neighborhood $N(x_i)$ to obtain tangent coordinates $\theta_j^{(i)}$ for all $j \in N_i$
- Then, *global* coordinates $\tau_i \in \mathbb{R}^t$ are found which best respect these coordinates up to an affine transformation:

$$\tau_j = \mu_i + \mathbf{L}_i \theta_j^{(i)} + \epsilon_j^{(i)} \qquad \forall j \in N_i \quad \forall i$$

- Collecting into matrix notation,

$$\mathbf{T}_i = \mu_i 1_K^T + \mathbf{L}_i \Theta_i + \mathbf{E}_i \quad \forall i$$

- For fixed $\mathbf{T}_i$, the error $\|\mathbf{E}_i\|_F^2$ is minimized by:

$$\mu_i = \frac{1}{K}\mathbf{T}_i 1, \quad \mathbf{L}_i = \mathbf{T}_i \left(I - \frac{1}{K} 1 1^T \right) \Theta_i^+$$

- Let $\mathbf{S}_i$ be the 0-1 selection matrix such that $\mathbf{T}\mathbf{S}_i = \mathbf{T}_i$ for all global coordinates $\mathbf{T} = (\tau_1, \cdots, \tau_n)$
- Now find $\mathbf{T}$ subject to $\mathbf{T}\mathbf{T}^T = I$ minimizing the total error

$$\begin{aligned} SSE(\mathbf{T}) &= \sum \|\mathbf{E}_i\|_F^2 = \sum \left\| \mathbf{T}_i \left(I - \frac{1}{K} 1 1^T \right) (I - \Theta_i^+ \Theta_i) \right\|^2 \\ &\equiv \sum \|\mathbf{T}\mathbf{S}_i \mathbf{W}_i\|^2 = \|\mathbf{T}\mathbf{S}\mathbf{W}\|_F^2 \end{aligned}$$

  where $\mathbf{S} = (\mathbf{S}_1, \cdots)$ and $\mathbf{W} = \mathrm{diag}(\mathbf{W}_1, \cdots)$
- Optimal $\hat{\mathbf{T}}$ is given by the eigenvectors of $\mathbf{B} = \mathbf{S}\mathbf{W}\mathbf{W}^T\mathbf{S}^T$ with the $t+1$ smallest eigenvalues, excluding $\mathbf{B}1_n = 0$.

The Frobenius norm is defined as $\|\mathbf{A}\|_F^2 = \sum_{i,j} |A_{ij}|^2 = \mathrm{tr}(\mathbf{A}\mathbf{A}^T) = \sum \sigma_k^2(\mathbf{A})$.

Robust LLE

- Classical estimators obtained by e.g. least-squares methods are sensitive to violations of model assumptions.
- Robust statistics aims to develop procedures which reduce the influence of distributional deviations (outliers).
- Example: mean vs median
- RLLE (Chang, 2005) first performs local *robust* PCA on each $N(x_i)$ to measure how likely x_i comes from $\mathcal{M}$, and reduces its influence accordingly.

- Recall that in PCA, we minimize the total error

$$SSE(\mathbf{X}) = \sum \|\epsilon_j\|^2 = \sum \|x_j - \nu - \mathbf{A}z_j\|^2$$

- In robust PCA, we instead minimize $SSE(\alpha, \mathbf{X}) = \sum \alpha_j \|\epsilon_j\|^2$ where the weights $\alpha_j > 0$, $\sum \alpha_j = 1$ measure how close x_j is from the affine subspace to be fitted.
- The true values of α are unknown. Ideally, a large ϵ_j should induce large α (following some convex law ρ).
- However, this creates a cyclic dependency:

$$\alpha \xrightarrow{\min SSE(\alpha,\cdot)} \nu, \mathbf{A} \xrightarrow{\epsilon_j = x_j - \nu - \mathbf{A}z_j} \epsilon \xrightarrow{\alpha_j = \rho(\|\epsilon_j\|)} \alpha$$

- Exploit this dependency to form an iterative procedure, starting from ordinary PCA ($\alpha_j^{(0)} \equiv 1/K$) and running until the weights stabilize:

$$\alpha^{(m)} \xrightarrow{SSE} \nu^{(m)}, \mathbf{A}^{(m)} \to \epsilon^{(m)} \xrightarrow{\rho} \alpha^{(m+1)}$$

- Now each $x_j \in N(x_i)$ has a weight $\alpha_j(x_i)$. Fixing j, repeat for all $i \in N_j$ and sum to obtain a total *reliability score* β_j
- Discard outliers with very low $\beta_i < \tau$ which lie far from $\mathcal{M}$
- Finally, modify Step 3 of LLE to:

$$\hat{\mathbf{Y}}_R = \text{argmin} \sum_{\beta_i \geq \tau} \beta_i \left\| y_i - \sum_{j \in N_i} \hat{w}_{ij} y_j \right\|^2$$

Replacing $\|\cdot\|^2$ with Huber loss $L_c(x) = \begin{cases} |x|^2 & \text{if } |x| \leq c \\ 2c|x| - c^2 & \text{if } |x| > c \end{cases}$ and using $\rho(x) = \min(1, c/|x|)$ are popular methods.