ISOMAP

Machine Learning

IIIy CSE IDP and IDDMP

Refer : Stephen Marsland

Motivation

- **The Curse of Dimensionality**
  - High-dimensional data often lies on a much lower-dimensional manifold
  - Pairwise Euclidean distances become less meaningful as dimensionality grows
- **Nonlinear Structure**
  - Linear methods like PCA cannot “unfold” curved manifolds
  - We need a way to respect the intrinsic geometry of the data
- **Global vs. Local**
  - Methods such as LLE focus on preserving local neighborhoods
  - ISOMAP seeks to preserve **global** manifold structure by using geodesic distances

What Is ISOMAP?

Full Name: Isometric Mapping

Core Idea:

- Build a nearest-neighbor graph on the high-dimensional points
- Approximate **geodesic (manifold) distances** by shortest paths in the graph
- Feed those distances into classical Multi-Dimensional Scaling (MDS)

Key Properties:

- Recovers the low-dimensional manifold isometrically (when assumptions hold)
- Minimizes a global stress function over all pairwise distances
- Retains global topology, not just local patches

Multi-Dimensional Scaling (MDS)

Objective: Find a low-dimensional embedding $\{z_i\}_{i=1}^{N} \subset \mathbb{R}^L$ that preserves pairwise distances

Given: High-dimensional data points $x_1, \ldots, x_N \in \mathbb{R}^M$

Embedding: New coordinates $z_1, \ldots, z_N \in \mathbb{R}^L$ with $L < M$

Relation to PCA:

- If distances are Euclidean, classical MDS is equivalent to PCA
- ISOMAP replaces Euclidean distances with geodesic approximations before applying MDS

MDS Cost Function Choices

Kruskal–Shephard Scaling (Least-Squares)

- Also called classical least-squares MDS
- Treats all pairwise discrepancies equally
- Minimizes $S_{KS}(\mathbf{z}_1, \mathbf{z}_2, \dots \mathbf{z}_N) = \sum_{i \neq i'} (d_{ii'} - \|\mathbf{z}_i - \mathbf{z}_i'\|)^2$
- Good for preserving overall global distances

MDS Cost Function Choices

Sammon Mapping

- Weighted least-squares variant
- Puts **greater emphasis on small (local) distances**
- Minimizes $S_{SM}(\mathbf{z}_1, \mathbf{z}_2, \ldots \mathbf{z}_N) = \sum_{i \neq i'} \frac{(d_{ii'} - \|\mathbf{z}_i - \mathbf{z}_i'\|)^2}{d_{ii'}}$
- Helps keep neighbors correctly spaced on the manifold

Classical MDS (Similarity-Based)

- **Key Idea:**

  Converts distances into **similarities** via the centered inner product

  Builds a Gram (similarity) matrix SSS instead of a distance matrix

- **Workflow Overview:**

  Center the data to compute pairwise inner products $s_{ii'} = (\mathbf{x}_i - \bar{\mathbf{x}}), (\mathbf{x}_i' - \bar{\mathbf{x}})^T$

  The function that needs to be minimised is $\sum_{i \neq i'} \left(s_{ii'} - (\mathbf{z}_i - \bar{\mathbf{z}}), (\mathbf{z}_i' - \bar{\mathbf{z}})^T\right)^2$

- **When It Works Best:**

  Flat (Euclidean) manifolds

  Recovers the same embedding as PCA when input distances are Euclidean

- **Limitations:**

  Cannot directly handle **nonlinear** (curved) manifolds

Multi-Dimensional Scaling (MDS) Algorithm

The Multi-Dimensional Scaling (MDS) Algorithm

- Compute the matrix of squared pairwise similarities $\mathbf{D}$, $\mathbf{D}_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|^2$
- Compute $\mathbf{J} = \mathbf{I}_N - 1/N$ (where $\mathbf{I}_N$ is the $N \times N$ identity function and N is the number of datapoints)
- Compute $\mathbf{B} = -\frac{1}{2}\mathbf{JDJ}^T$
- Find the L largest eigenvalues λ_i of $\mathbf{B}$, together with the corresponding eigenvectors $\mathbf{e}_i$
- Put the eigenvalues into a diagonal matrix $\mathbf{V}$ and set the eigenvectors to be columns of matrix $\mathbf{P}$
- Compute the embedding as $\mathbf{X} = \mathbf{PV}^{1/2}$

From Classical MDS to ISOMAP

Challenge:

- Manifolds with **curvature** (non-flat) cannot be embedded accurately by classical MDS
- True pairwise manifold distances are unknown

ISOMAP's Key Approximation:

I) Local Euclidean Assumption

- For **nearby** points, Euclidean distance ≈ geodesic distance

II) Graph-Based Geodesics

- Connect each point to its k-nearest neighbors (or within radius ϵ)
- Approximate long-range distances by **shortest paths** through this neighbor graph

Outcome:

Embedding that **unfolds curved manifolds**

Preserves **global** manifold geometry rather than just local neighborhoods

Isomap Algorithm

The Isomap Algorithm

- Construct the pairwise distances between all pairs of points
- Identify the neighbours of each point to make a weighted graph G
- Estimate the geodesic distances d_G by finding shortest paths
- Apply classical MDS to the set of d_G

Practical Aspects of ISOMAP

Shortest-Path Computation

- **Floyd–Warshall:** $O(N^3)$time
- **Dijkstra (per source):** $O(N^2)$ time
- Choice depends on dataset size and sparsity of graph

Neighborhood Size Matters

- Too **small** k $\Rightarrow$ graph **fragments** into disconnected components $\Rightarrow$ infinite distances
- Must select k(or radius ϵ) large enough to keep the graph **connected**
- Typically **keep only the largest component**, discarding isolated points

Iris Dataset Example (Figure – 1)

- k=12: only one class remained connected; two classes lost
- Increasing k>50: all three classes preserved, meaningful embedding

Swiss Roll Comparison (Figure – 2)

- ISOMAP and LLE produce **qualitatively similar** “unrolled” embeddings
- Both methods capture the manifold’s intrinsic geometry when applied correctly

Isomap for Iris data

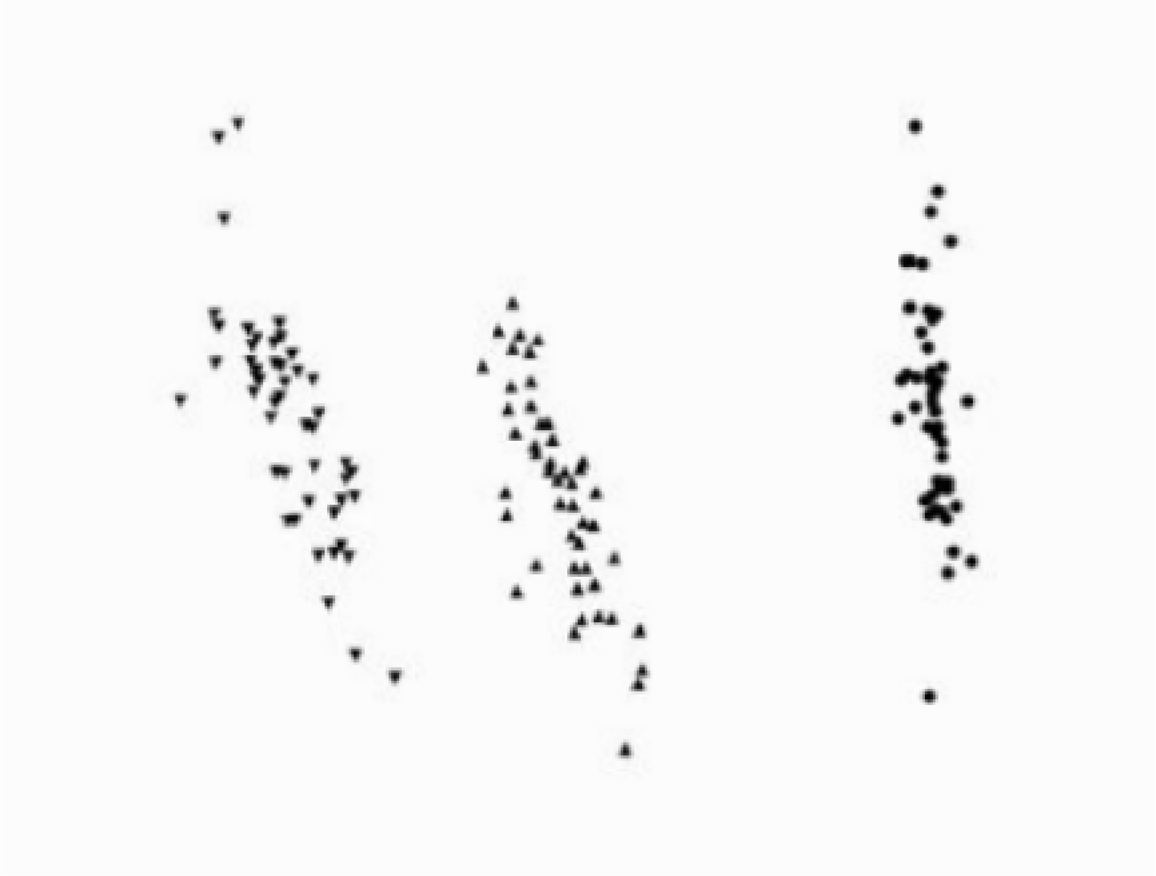

Figure - 1 Isomap transforms the iris data in a similar way to factor analysis, provided that the neighbourhood size is large enough to avoid points becoming disconnected.

Isomap for Swiss Roll Dataset

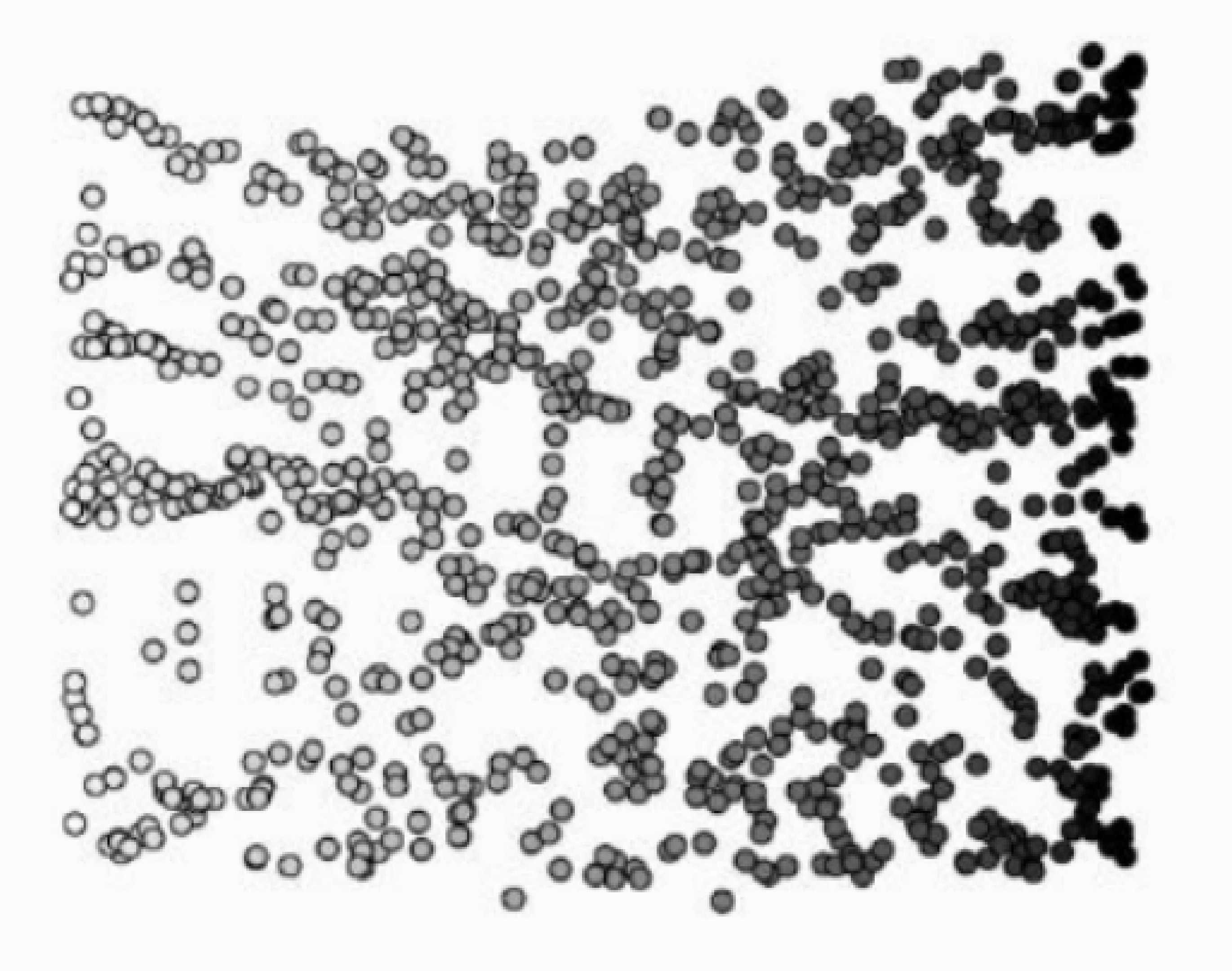

Figure - 2 Isomap also produces a good remapping of the swissroll dataset.

ISOMAP vs. LLE

- **Global vs. Local Focus**
  - **ISOMAP:** Preserves **global** geodesic distances across the entire manifold
  - **LLE:** Preserves only **local** linear reconstructions within each neighborhood
- **Computational Cost**
  - **ISOMAP:** Needs all-pairs shortest paths → $O(N^2)–O(N^3)$
  - **LLE:** Solves sparse local linear systems → typically **much faster**
- **Strengths & Weaknesses**

Aspect	ISOMAP	LLE
Distance Accuracy	Good for both near and far points	Accurate only for nearby points
Global Topology	Captures overall manifold shape	May “fold” distant regions together
Scalability	Heavier for large N	More scalable to large datasets

Summary of ISOMAP

Motivation:

- Overcome the curse of dimensionality and capture nonlinear manifold structure
- Preserve **global** geodesic distances rather than just local neighborhoods

Key Features:

- Produces an **isometric** low-dimensional embedding when manifold assumptions hold
- Reveals curved structures like the Swiss Roll and preserves class separations

When to Use ISOMAP:

- You need to preserve **global relationships** on a manifold
- Your dataset size permits graph-based shortest-path computations
- You seek a **straightforward pipeline** leveraging classical MDS