**San José State University**

**Math 253: Mathematical Methods for Data Visualization**

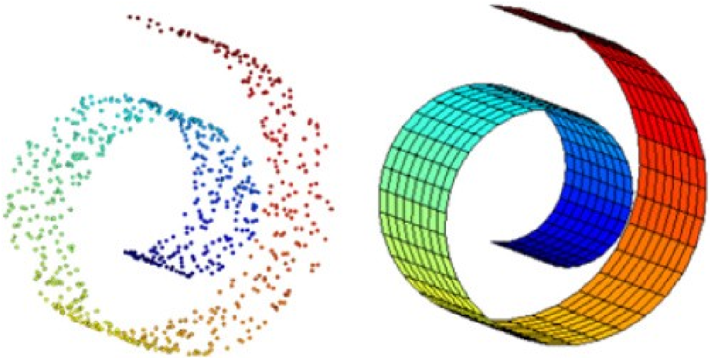
# Isometric Feature Mapping (ISOmap)

Dr. Guangliang Chen

# ISOmap

Briefly, ISOmap is MDS combined with a special metric, called **geodesic distance**, for reducing the dimensionality of data sampled from a smooth manifold:

- **Paper**: *A Global Geometric Framework for Nonlinear Dimensionality Reduction, J. B. Tenenbaum, V. de Silva and J. C. Langford, Science 290 (5500): 2319–2323, December 2000*
- **Website**: `https://web.mit.edu/cocosci/isomap/isomap.html`



(a) (b)

## Motivation

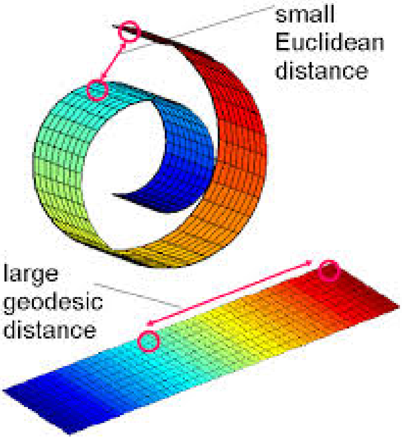
Consider applying PCA to the Swissroll data. There are the following drawbacks:

- The PCA dimension needs to be higher, and sometimes much higher, than the manifold dimension (otherwise PCA may project faraway points along the manifold to nearby locations);
- PCA cannot capture the curved dimensions (its principal directions are generally not meaningful).

## The ISOmap approach to dimension reduction

Instead of preserving the Euclidean distance, we will apply MDS to preserve the **geodesic distance**, which

- captures the true, nonlinear geometry corresponding to the curved dimension
- allows to see the transitioning along the manifold (and thus the global structure).

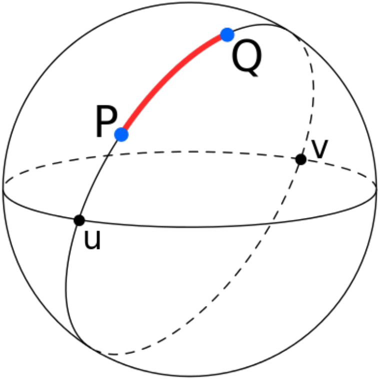


## How to find geodesic distances

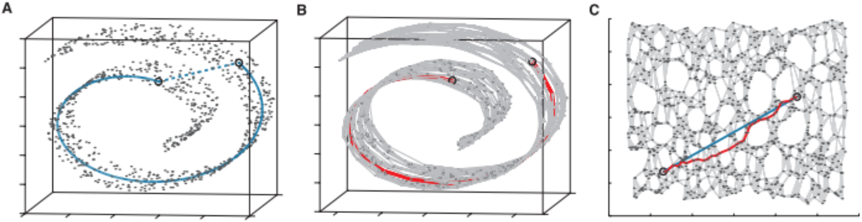
The geodesic distance of two data points that live in a manifold is the shortest distance along the manifold.

On a sphere, it is just the great-circle distance.

The exact geodesic distances are often impossible to find (unless we know the true manifold).



In practical settings where we are only given a data set $X$ sampled from an unknown manifold $\mathcal{M}$, we can approximate the true geodesic distances $d_{\mathcal{M}}(i,j)$ by the shortest-path distances $d_G(i,j)$ on a nearest-neighbor graph $G$ built on the data set.



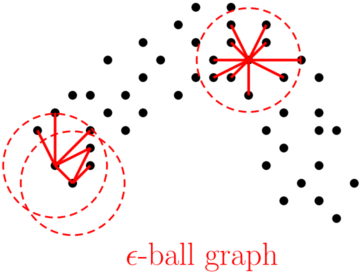
Detailed steps:

1. Build a neighborhood graph $G$ from the given data by connecting only "nearby points" with edges weighted by their Euclidean distances, i.e.,

$$d_X(i,j) = \|\mathbf{x}_i - \mathbf{x}_j\| \quad \text{if } \mathbf{x}_i, \mathbf{x}_j \text{ are "close" (and 0 otherwise)}$$

   where "closeness" is defined in one of the following ways:

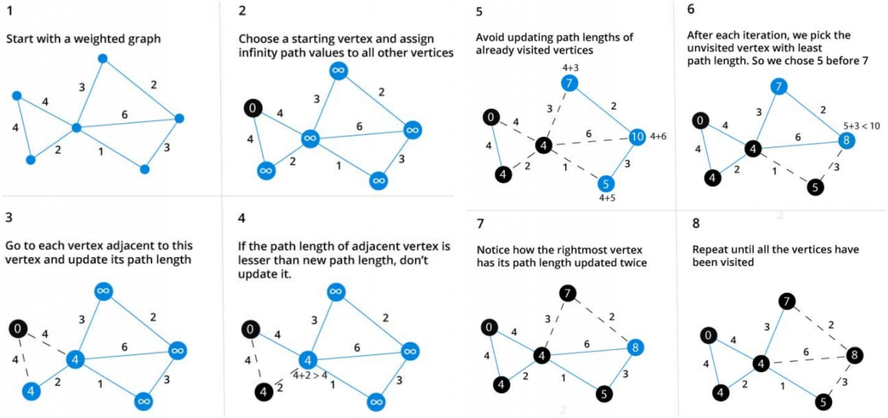
   - $\epsilon$**-ball approach**: For each $\mathbf{x}_i$, another point $\mathbf{x}_j$ is close if and only if $\|\mathbf{x}_i - \mathbf{x}_j\| \le \epsilon$, or
   - $k$**NN approach**: For each point $\mathbf{x}_i$, $\mathbf{x}_j$ is close if it is among the the $k$ nearest neighbors of $\mathbf{x}_i$.




2. Apply Dijkstra's algorithm[1] with the nearest neighbor graph $G$ (constructed by either method) to find the shortest-path distances for all pairs of data points ($d_G(i, j)$).

[1] https://upload.wikimedia.org/wikipedia/commons/5/57/Dijkstra_Animation.gif

# ISOmap




(Source: `https://www.programiz.com/dsa/dijkstra-algorithm`)

## The ISOmap algorithm

**Input**: Pairwise distances $d_X(i,j)$ of data points in the input space, embedding dimension $k \geq 1$, neighborhood graph method ($\epsilon$-ball or $k$NN)

**Output**: A $k$-dimensional representation of the data $\mathbf{Y} \in \mathbb{R}^{n\times k}$.

1. Construct a neighborhood graph $G$ from the given distances $d_X(i,j)$ using the specified method
2. Compute the shortest-path distances $d_G(i,j)$ between all vertices of $G$ by using Dijkstra's algorithm.
3. Apply MDS with $d_G(i,j)$ as input distances to find a $k$-dimensional representation $\mathbf{Y}$ of the original data

## Implementations

MATLAB:

- Code by the ISOmap authors:
  https://web.mit.edu/cocosci/isomap/isomap.html
- Matlab Toolbox for Dimensionality Reduction by van der Maaten:
  https://lvdmaaten.github.io/drtoolbox/
- MANI: Manifold Learning Toolkit (by T. Wittman):
  http://macs.citadel.edu/wittman/Research/Mani/mani.m

Python: https://scikit-learn.org/stable/modules/generated/sklearn.manifold.Isomap.html

**Demonstration**

See

`https://www.cs.cmu.edu/~bapoczos/Classes/ML10715_2015Fall/slides/ManifoldLearning.pdf`

## Summary

We presented the MDS method for dimensionality reduction which aims to preserve certain kind of distances of the data:

Special cases:

- Euclidean distance: PCA
- Geodesic distance: ISOmap