Dimensionality Reduction

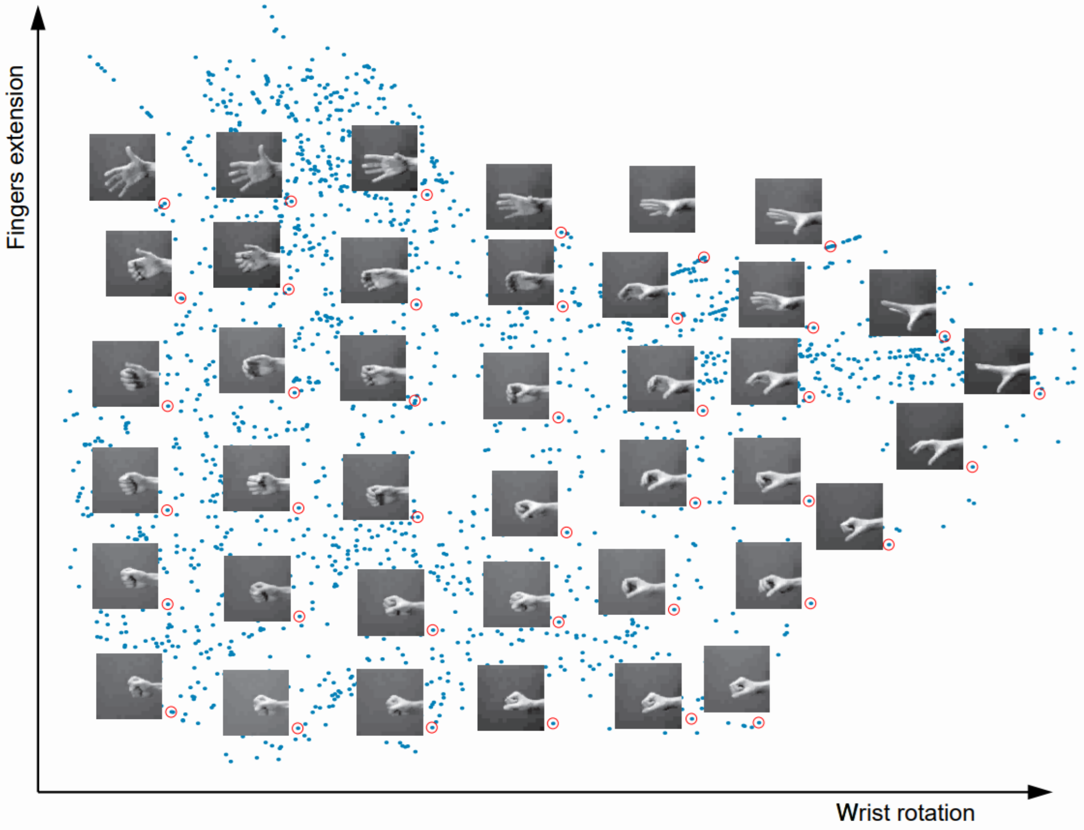
Alessandro Minoli
Flavio Perini
Francesco Torgano

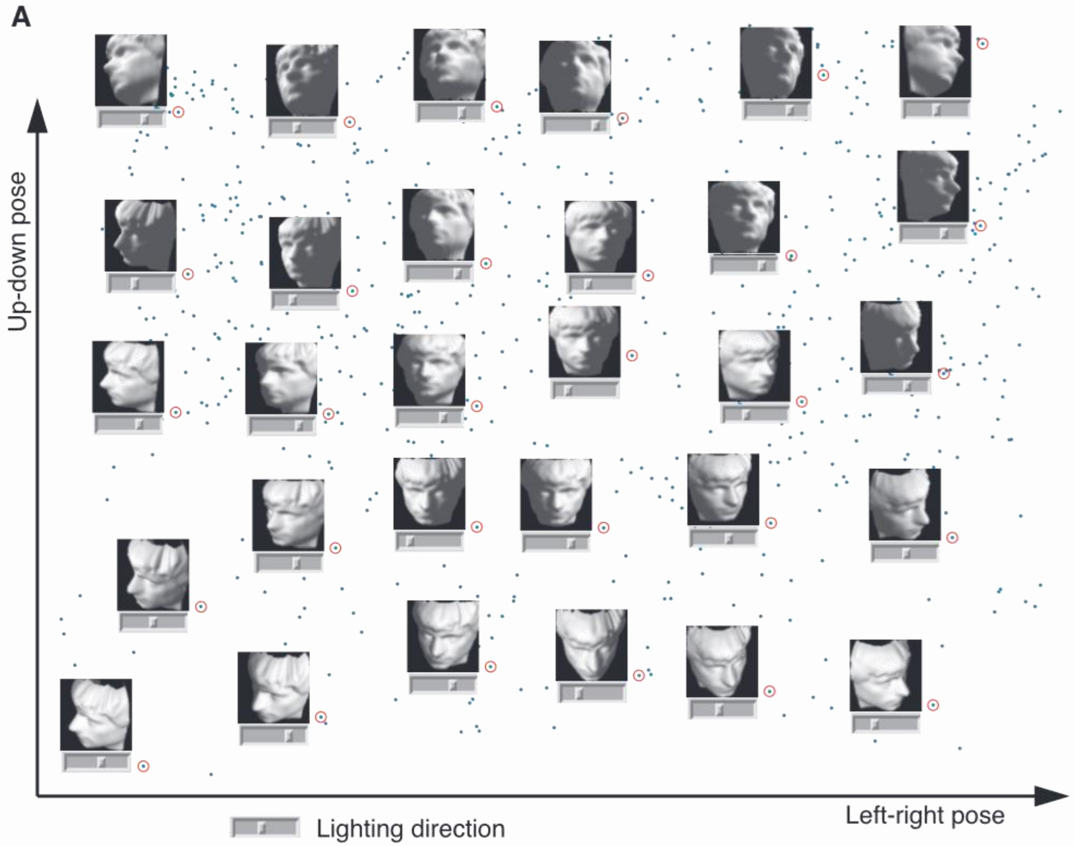
SciViz A.A. 2020-21

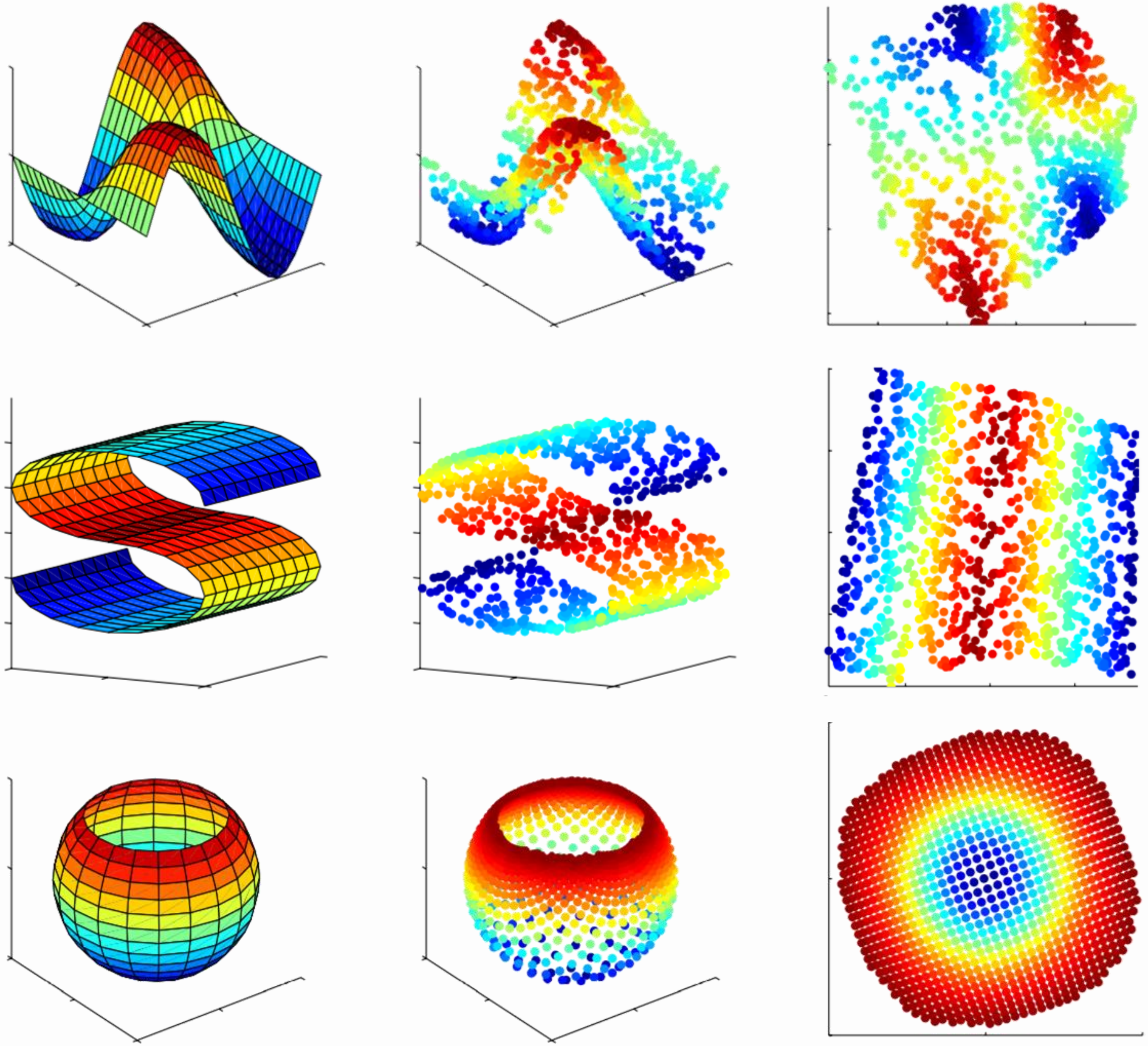
UNIVERSITÀ DEGLI STUDI DI MILANO

ISOMAP

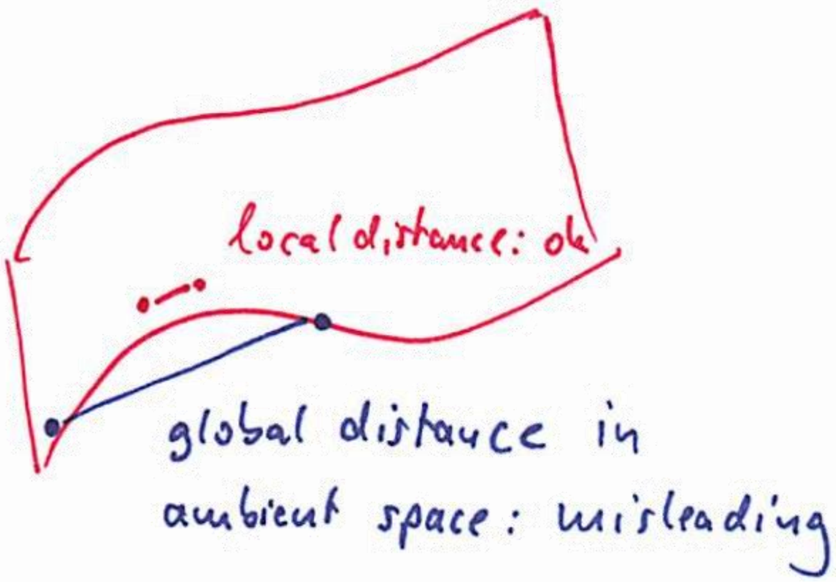
Cosa significa ridurre la dimensionalità?

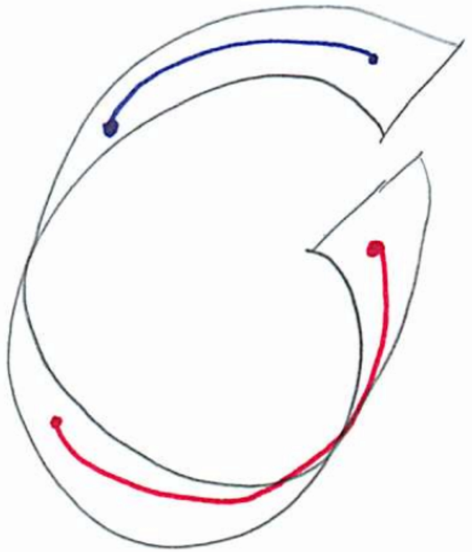


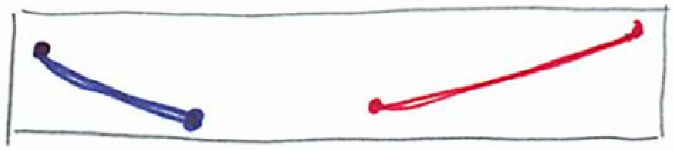




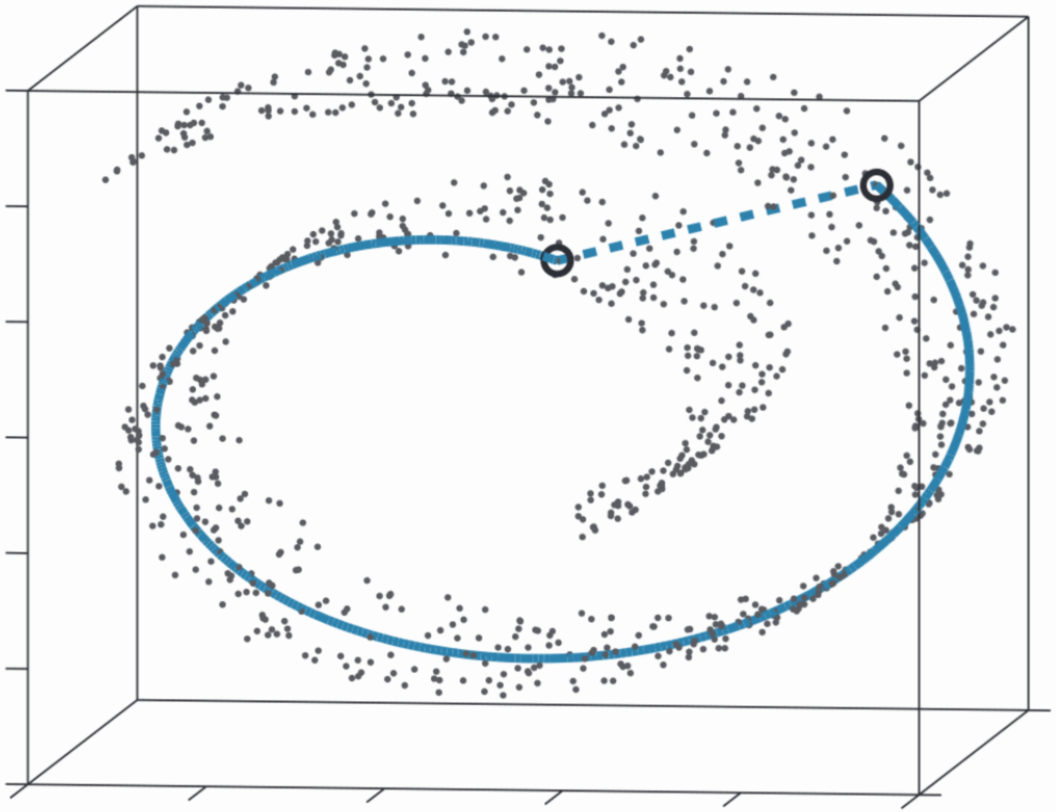
L'intuizione







Algoritmo di ISOMAP



Input:

- $\boldsymbol{X}$ (ad alta dimensionalità)
- una funzione di **distanza** $d(x_i, x_j)$, scegliamo la euclidea

1) Costruzione del grafo pesato

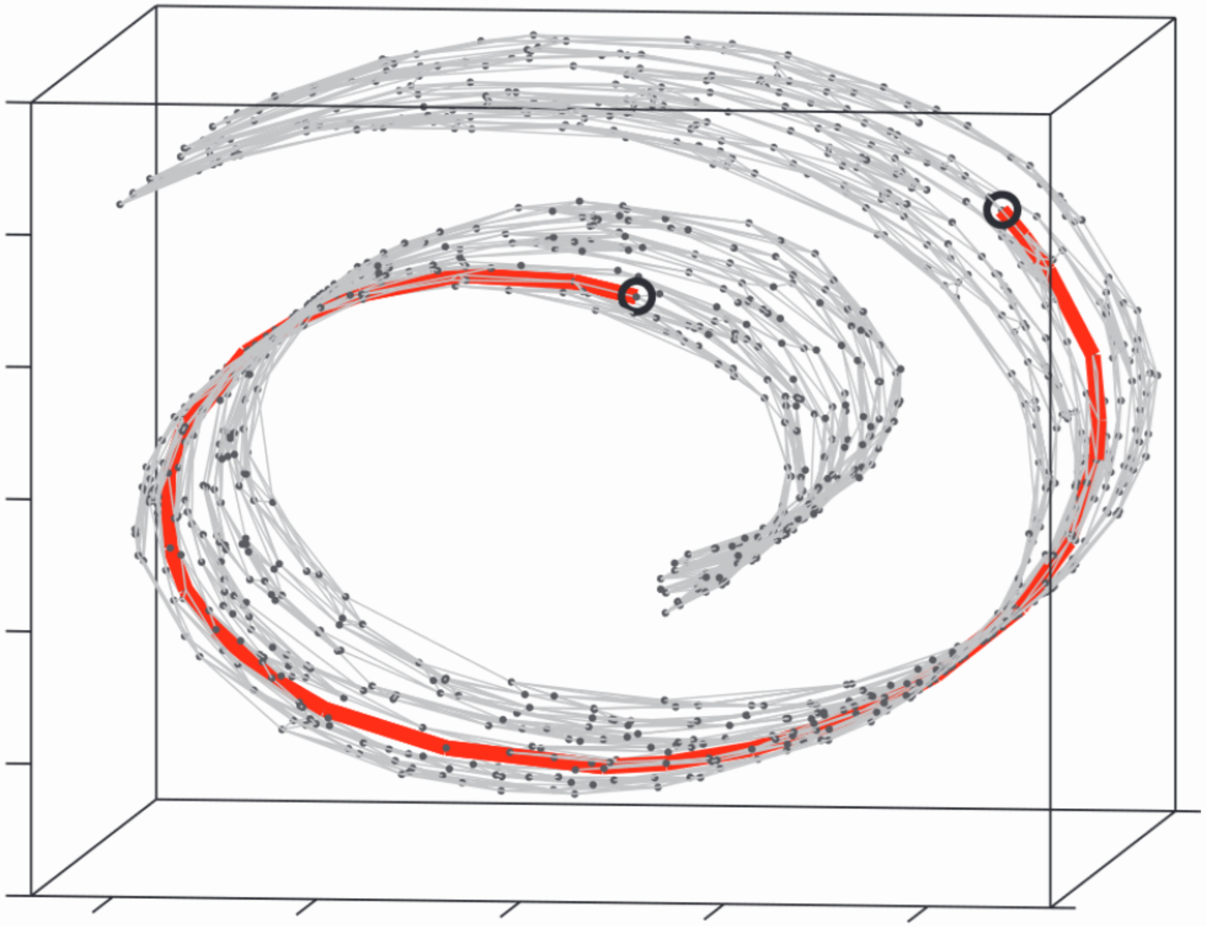
Troviamo i neighbors $\mathcal{N}_i$ di ogni punto x_i

K-nearest neighbors $\Rightarrow$ $|\mathcal{N}_i| = k$

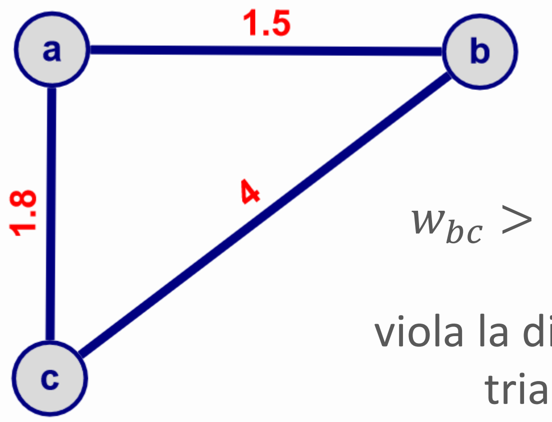
Fixed radius $\Rightarrow \mathcal{N}_i = \{ j \mid \|x_j - x_i\| \leq r \}$

Uniamo i neighbors $\mathcal{N}_i$ a x_i con archi di peso

$w_{ij} = \|x_j - x_i\|, \forall j \in \mathcal{N}_i$ (distanze **locali**)



- grafo **connesso**
- vale la **disuguaglianza triangolare**



$$w_{bc} > w_{ba} + w_{ac}$$
$$\Downarrow$$

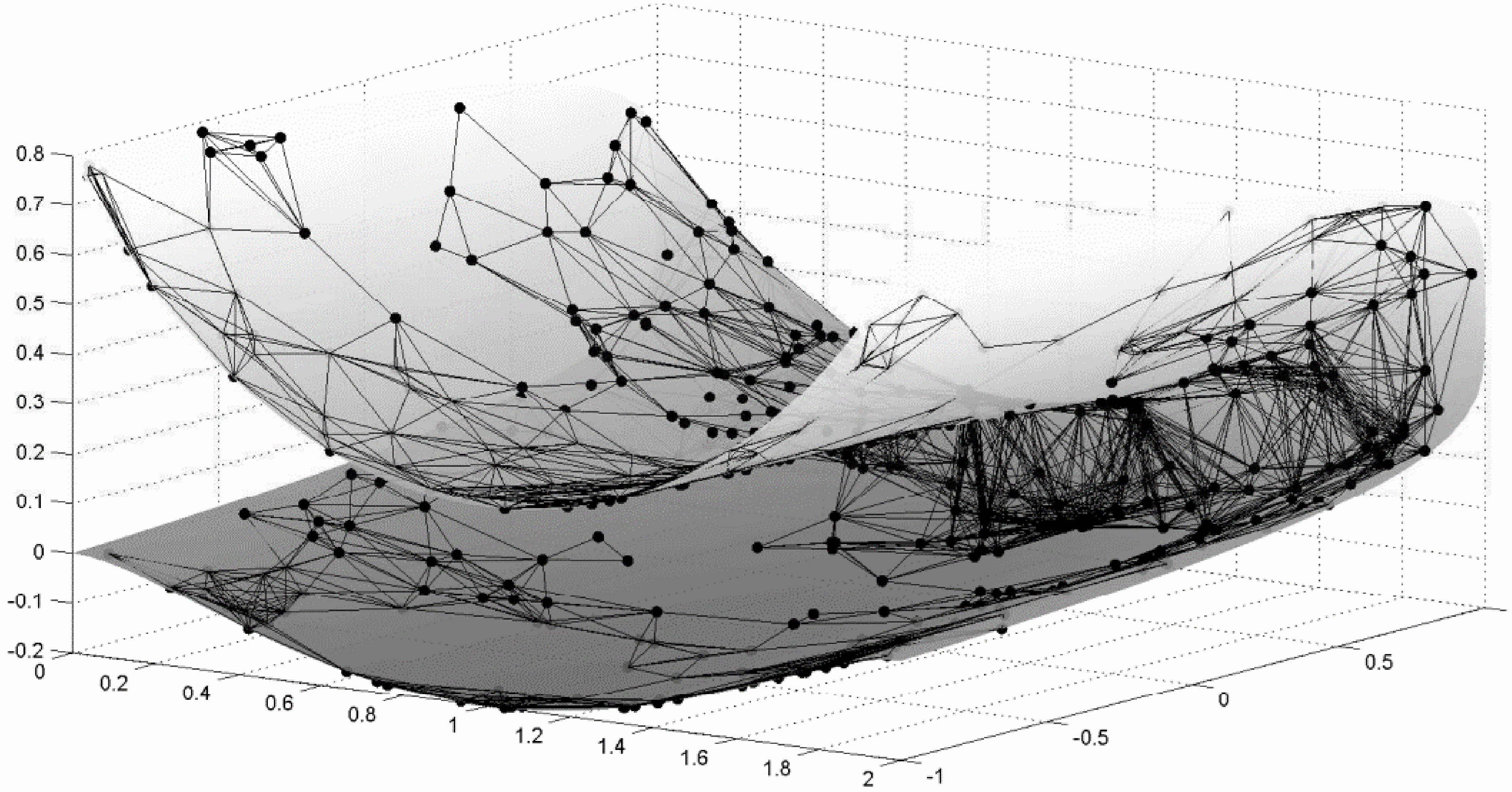
viola la disuguaglianza triangolare

2) Ricava la matrice di distanze D

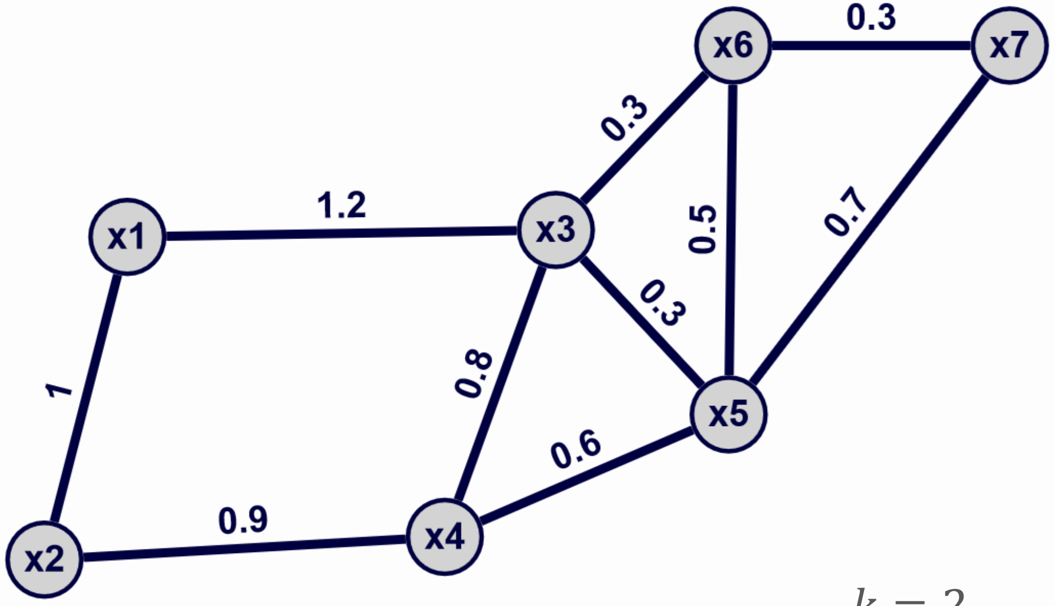
calcolando le distanze d_{sp} dei cammini minimi fra tutte le coppie di punti usando l'algoritmo di Dijkstra o di Floyd-Warshall

(distanze **geodesiche**)

Come scegliere k



Matrice di distanze $\boldsymbol{D}$



$k = 2$

	x1	x2	x3	x4	x5	x6	x7
x1	0	1	1.2	1.9	1.5	1.5	1.8
x2	1	0	1.7	0.9	1.5	2.0	2.2
x3	1.2	1.7	0	0.8	0.3	0.3	0.6
x4	1.9	0.9	0.8	0	0.6	1.1	1.3
x5	1.5	1.5	0.3	0.6	0	0.5	0.7
x6	1.5	2.0	0.3	1.1	0.5	0	0.3
x7	1.8	2.2	0.6	1.3	0.7	0.3	0

3) Applica metric MDS con D come input per ottenere X (a bassa dimensionalità), cioè un embedding del dataset che preserva le distanze geodesiche

Data $D \in \mathbb{R}^{n \times n}$, $d_{ij} = \|x_i - x_j\|$, ricava i punti $(x_i)_{i=1\dots n} \in \mathbb{R}^d$

Questo problema di embedding è chiamato **(metric) multi-dimensional scaling**

Più generalmente il problema è:

Dati gli oggetti $x_1, \dots, x_n \in X$ (ad alta dimensionalità) ,
trova un embedding $\Phi : X \rightarrow \mathbb{R}^d$ t.c. $\|\Phi(x_i) - \Phi(x_j)\| = d_{ij}$

Per una generale D, **non possiamo** ottenere un tale embedding **senza distorsione** dei dati

Classic MDS

Data una D di **distanze euclidee** possiamo esprimere in termini di entrate di D
la **matrice di Gram S** con entrate $s_{ij} = \left(\langle x_i, x_j \rangle\right)_{ij=1\dots n}$ in questo modo:

$$\boldsymbol{s_{ij}} = \langle x_i, x_j \rangle = \cdots = \frac{\mathbf{1}}{\mathbf{2}}\left(\boldsymbol{d_{1i}}^2 + \boldsymbol{d_{1j}}^2 - \boldsymbol{d_{ij}}^2\right)$$

S è una matrice definita positiva, perciò possiamo decomporla nella forma $S = XX^t,\ X \in \mathbb{R}^{n \times d}$

L' $\boldsymbol{i}$ -esima riga di $\boldsymbol{X}$ sarà l'embedding del punto $\boldsymbol{x_i}$ in $\mathbb{R}^{\boldsymbol{d}}$

Per trovare X calcoliamo la decomposizione spettrale di $S = V\Lambda V^t$
Tipicamente, scegliamo una dimensione $d \leq n$ e poniamo:

- V_d uguale alle prime d colonne di V
- Λ_d uguale alla matrice diagonale $d \times d$ con i primi d autovalori sulla diagonale
- $X = V_d\sqrt{\Lambda_d}$

Metric MDS

Se D non è di sole distanze euclidee non riusciremo a ricavare un embedding perfetto.

Perciò definiamo una funzione di **stress**, ad esempio:

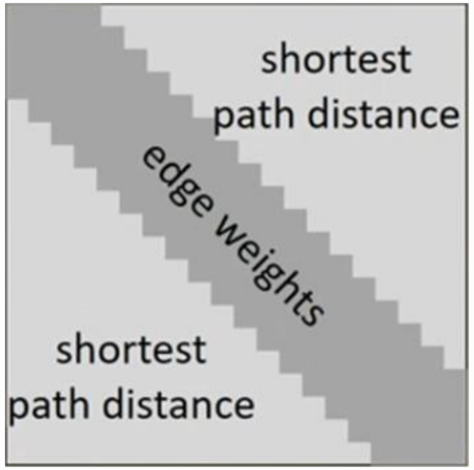
$$stress(embedding) = \sqrt{\sum_{i=1}^{n} \sum_{j=1|i\neq j}^{n} \left(\left\|x_i - x_j\right\| - d_{ij}\right)^2}$$

Cercheremo di trovare un embedding $x_1, \ldots, x_n$ che minimizzi lo stress tramite un algoritmo standard di ottimizzazione non convessa (es. discesa del gradiente).

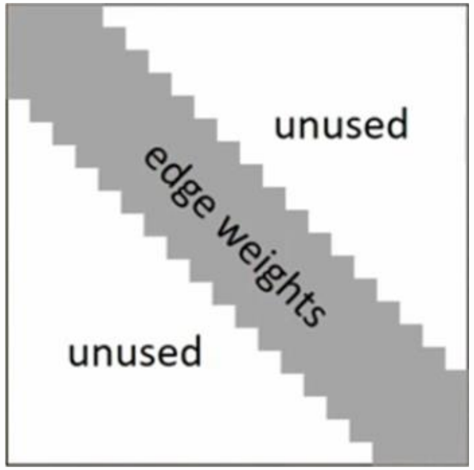
- Garanzia teorica che, con $x_1, \dots, x_n$ campioni presi uniformemente da un manifold “buono” , per $n \to \infty$ e $k \approx log\, n$ i cammini minimi sul grafo pesato convergono alle distanze geodesiche fra i campioni

- $O[Dlog(k)Nlog(N)] + O[N^2(k + \log(N))] + O[dN^2]$

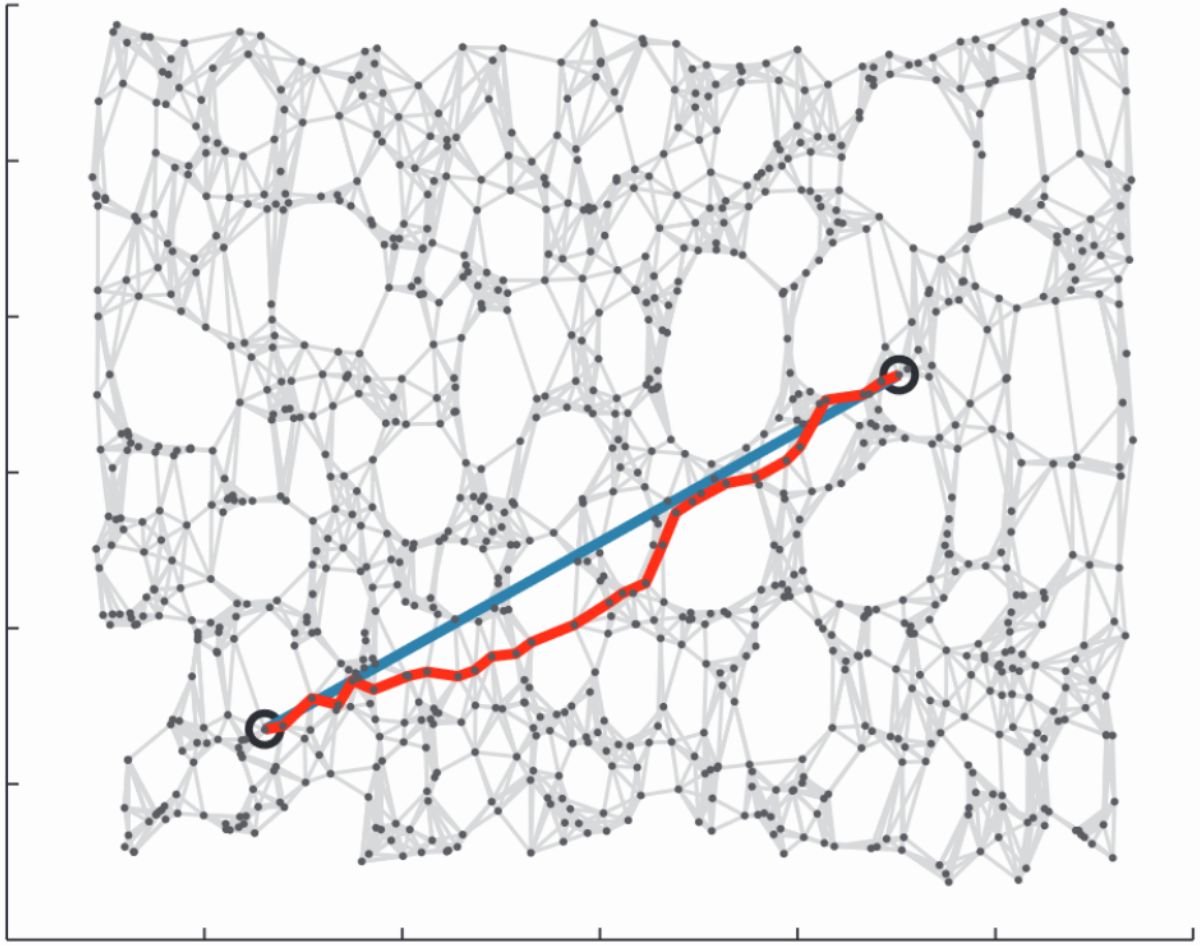
- Problema dei buchi nel manifold



ISOMAP (global)



LLE (local)



IsoMap projection of MNIST digits dataset (size: 1797, time: 2.96s)

IsoMap projection of Fashion-MNIST dataset (size: 5000, time: 21.29s)