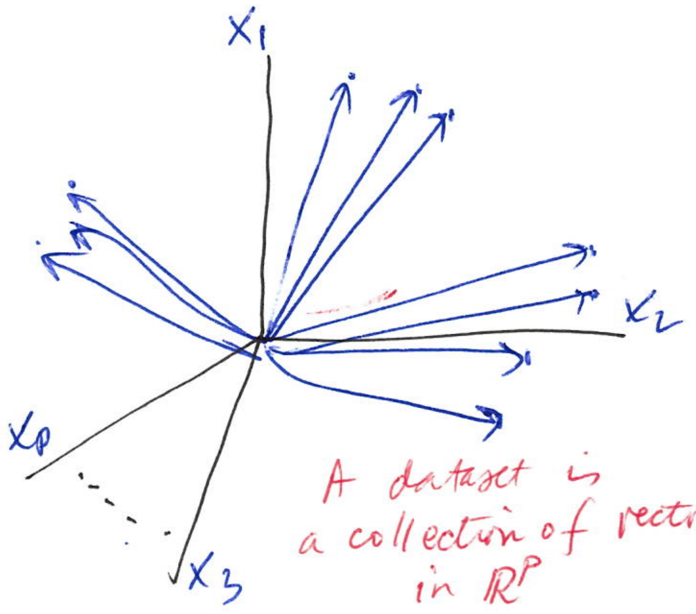
Graph-based learning



A dataset is a collection of vectors in $\mathbb{R}^p$

$$\{\vec{y}_i\}_{i=1}^{N} \quad i = 1, \dots, N$$

$$\vec{y}_i \in \mathbb{R}^p$$

$$\vec{y}_i = \begin{pmatrix} x_1^{(i)} \\ \vdots \\ x_p^{(i)} \end{pmatrix}$$

$$Y^T_{N \times p} = X_{N \times p} = \begin{pmatrix} \vec{y}_1^{\,T} \\ \vdots \\ \vec{y}_N^{\,T} \end{pmatrix}$$

vectors in $\mathbb{R}^p$:

$\rightarrow$ geometry of the dataset in relation to $\left\{ \begin{array}{l} \text{clustering} \\ \& \\ \text{reduced dimensionality} \end{array} \right.$

Key ingredient:
Similarity or dissimilarity (in some cases a distance)

between samples.

$\boxed{\text{An } N \times N \text{ matrix summarising the relationships (pairwise) between samples}}$

Examples :

dissimilarities $D_{N\times N}$ e.g.

$\updownarrow$

similarities $S_{N\times N}$

$$D_{ij} = \|\vec{y}_i - \vec{y}_j\|^2$$ (distances) dissimilarity

$$S_{ij} = \frac{\vec{y}_i \cdot \vec{y}_j}{\|\vec{y}_i\| \, \|\vec{y}_j\|}$$ (cosine similarity)

$$S_{ij} = \rho(\vec{y}_i, \vec{y}_j) = \frac{\mathrm{cov}(\vec{y}_i, \vec{y}_j)}{\sqrt{\mathrm{cov}(\vec{y}_i, \vec{y}_i)}\sqrt{\mathrm{cov}(\vec{y}_j, \vec{y}_j)}}$$ (statistical similarity)

$$D_{ij} = e^{-\frac{\|\vec{y}_i - \vec{y}_j\|^2}{c}}$$ (a kernel matrix)

In most of our unsupervised learning methods: [Pairwise similarities between samples are ~~the~~ the important determinants of the dataset.]

For other problems : ∴ In some cases the problem is directly defined by the pairwise similarities (we don't even know what $\vec{y}_i$ are, just S_{ij})

From $S_{N\times N}$ to Graph representation of the dataset that follows from this information.

Some definitions:

combinatorial object on

Graph: Two sets : nodes and edges.

$G(V,E)$ $\quad V =$ set of vertices (nodes) $i=1,\dots,N$

$E =$ set of edges (links) (i,j) [Pairs of edges]

Aside:

For our case: The set of nodes is the set of samples:

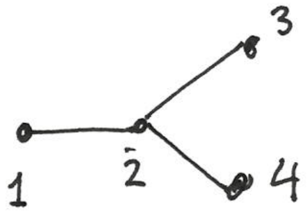
$$\{\vec{y}_i\}_{i=1}^{N} \longmapsto \{i\}_{i=1}^{N}$$

and the edges represent S_{ij}

Example: $\quad G(V,E) \quad V=\{1,2,3,4\}$

$N=4$

$$E=\left\{\begin{matrix}(1,2)\\(2,4)\\(2,3)\end{matrix}\right\}$$



Undirected.

Equivalent representation:

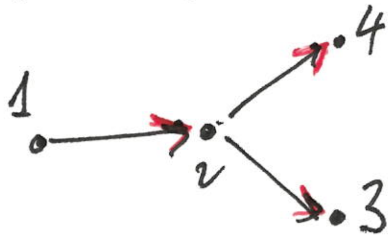
Adjacency matrix: $A_{N\times N}$

$$A=\begin{bmatrix}0&1&0&0\\1&0&1&1\\0&1&0&0\\0&1&0&0\end{bmatrix}$$

$$A_{ij}=\begin{cases}1, & \text{if } i\sim j \text{ (connected)} \\ & (i,j)\in E\\ 0, & \text{otherwise}\end{cases}$$

If undirected, $A = A^T$

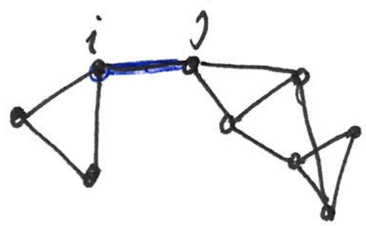
Directed graph.



$$A = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

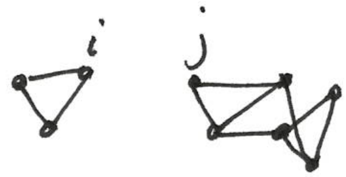
$$A \neq A^T$$

<u>Connected graph:</u> every pair of nodes is connected by a path (weak, for undirected)



connected

1

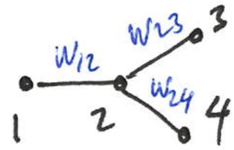


2

Two disconnected components.

See next page for 1 & 2

<u>Weighted graphs:</u>

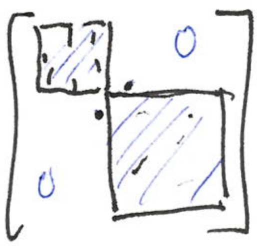
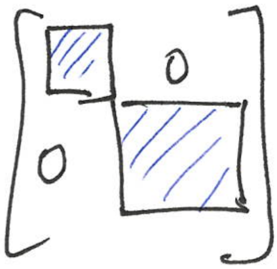


$$W = \begin{bmatrix} 0 & w_{12} & 0 & 0 \\ w_{12} & 0 & w_{23} & w_{24} \\ 0 & w_{23} & 0 & 0 \\ 0 & w_{24} & 0 & 0 \end{bmatrix}$$

Components in the graph:

If A is connected, then
$A_{N \times N}$ has full rank
and cannot be rearranged in
block-diagonal form (irreducible)

If we have more than one
~~disconnected~~ connected component
then A can be written in
block diagonal form:

$\boxed{1}$ $A =$  . After edge is gone $\boxed{2}$ $A =$ 

First connection : <u>Clustering problem</u> is related to
making the adjacency matrix
as block-diagonal as possible
by relabelling of the nodes.

[Remember that nodes are samples]

Second connection : distances between samples

Distances on graphs

- Minimal distance on the graph (geodesic):

geodesic Distance between nodes i and j in an undirected graph:

$$\min_{\{i,j\}} \# \text{ edges in } \{i,j\}$$

where $\{i,j\}$ denotes the set of paths between i and j in the graph.

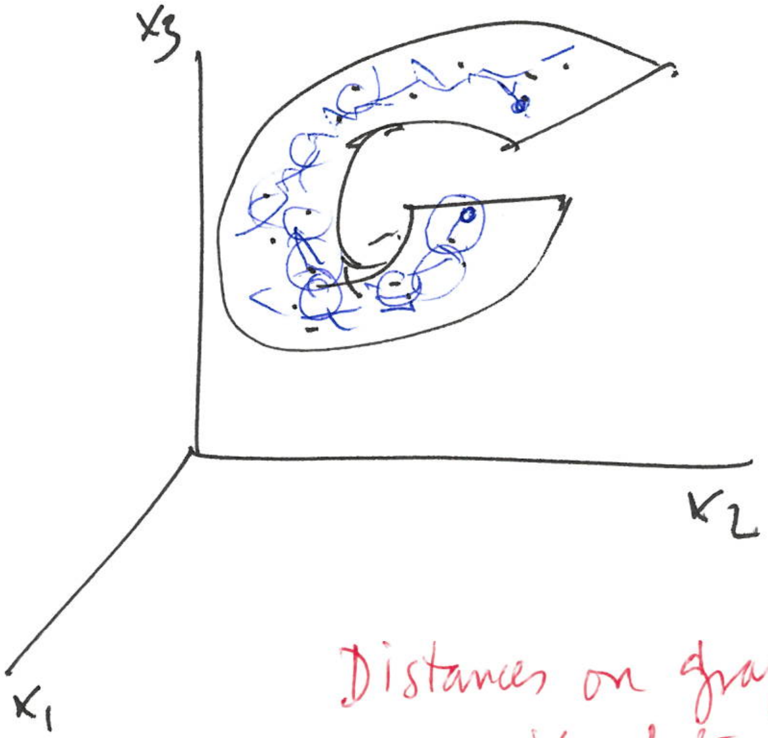
[Breadth first search algorithms (BFS)]

- Average distance between i,j:

$$\langle \# \text{ edges} \rangle_{\{i,j\}}$$

where $\langle \cdot \rangle_{\{i,j\}}$ denotes the average over the set of paths between (i,j)

If graph is weighted, the length of a path is the sum of the weights of the edges on the path



$\vec{y}_i \in \mathbb{R}^3$

Distances on graphs can provide better representations of the intrinsic geometry of the dataset.