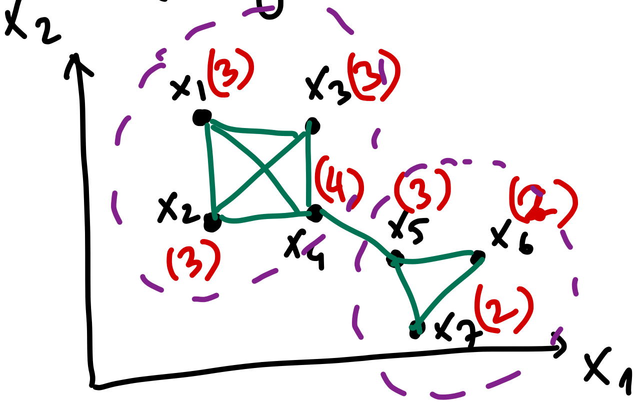
SPECTRAL CLUSTERING

- define local neighborhoods
- if the distance between x_i & x_j is smaller than a threshold they are neighbors.

$$b_{ij} = \begin{cases} 1 & \text{if } \|x_i - x_j\|_2 < \delta \\ 0 & \text{otherwise} \end{cases}$$

$$b_{ij} = \begin{cases} \exp\left[-\dfrac{\|x_i - x_j\|_2^2}{2\sigma^2}\right] & \text{if } \|x_i - x_j\|_2 < \delta \\ 0 & \text{otherwise} \end{cases}$$



$$b_{ii} = 0 \quad \forall i$$

$$d_{ii} = \sum_{j \neq i} b_{ij} \quad \forall i$$

→ # of neighbors of data point i

$$d_{ij} = 0 \quad \forall (i, j \neq i)$$

$$B = \begin{array}{c|ccccccc} & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ \hline x_1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ x_2 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ x_3 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ x_4 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ x_5 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ x_6 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ x_7 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{array}$$

connectivity or adjacency matrix

$$D = \begin{array}{c|ccccccc} & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ \hline x_1 & 3 & & & & & & \\ x_2 & & 3 & & & & 0 & \\ x_3 & & & 3 & & & & \\ x_4 & & & & 4 & & & \\ x_5 & & 0 & & & 3 & & \\ x_6 & & & & & & 2 & \\ x_7 & & & & & & & 2 \end{array}$$

mass, outgoing

$$[3 \;\; -1 \;\; -1 \;\; -1 \;\; 0 \;\; 0 \;\; 0]$$

Laplacian Matrix:

$$L_{N\times N} = D_{N\times N} - B_{N\times N}$$

↳ each row (column) sums up to 0.

$$L_{\text{RANDOM-WALK}} = D^{-1} \cdot L = D^{-1} \cdot (D - B) = \boxed{I - D^{-1} \cdot B}$$

$$L_{\text{SYMMETRIC}} = D^{-1/2} \cdot L \cdot D^{-1/2} = D^{-1/2} \cdot (D - B)\, D^{-1/2} = \boxed{I - D^{-1/2} \cdot B \cdot D^{-1/2}}$$

SPECTRAL CLUSTERING

STEP #1: Find the eigenvectors of normalized $L_{N\times N}$ matrix.

STEP #2: Pick R smallest eigenvectors.

STEP #3: Construct Z matrix as follows:

$$\underline{\underline{Z}} = \left[\underset{\text{1st smallest eigenvector}}{v_1} \quad v_2 \quad \cdots \quad \underset{R^{th}\text{ smallest eigenvector}}{v_R} \right]_{N\times R}$$

STEP #4: Run k-means clustering algorithm on Z matrix to find K clusters.

PARAMETERS:

δ: threshold.

R: # of eigenvectors to be included

K: # of clusters to be found.

HIERARCHICAL CLUSTERING

- finding groups such that instences (data points) in a group are more similar [closer] to each other than instences in different groups.

Component #1: The Distence Function Between Data Points

distence $\Rightarrow$ dissimilarity

distence $\uparrow$ similarity $\downarrow$

distence $\downarrow$ similarity $\uparrow$

$$k(x_i, x_j) = \exp\left[-\frac{\|x_i - x_j\|_2^2}{2\sigma^2}\right]$$

0 $\longleftrightarrow$ 1   0 $+\infty$

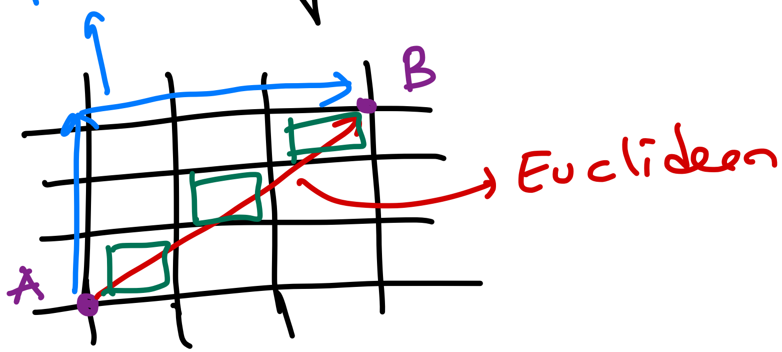
dissimilar $\longleftrightarrow$ similar   similar   dissimilar

Euclideen Distence

$$d(x_i, x_j) = \|x_i - x_j\|_2$$

$$= \sqrt{\left(\sum_{d=1}^{D} (x_{id} - x_{jd})^2\right)}$$

$$= \sqrt{x_i^T x_i - 2x_i^T x_j + x_j^T x_j}$$



Manhatten Distence (city-Block Distence)

$$d(x_i, x_j) = \sum_{d=1}^{D} |x_{id} - x_{jd}|$$

Component #2: The Direction to Proceed

Agglomerative (bottom-to-top)

⇒ combines small clusters into bigger ones

⇒ starts with "N" clusters.

Divisive (top-to-bottom)

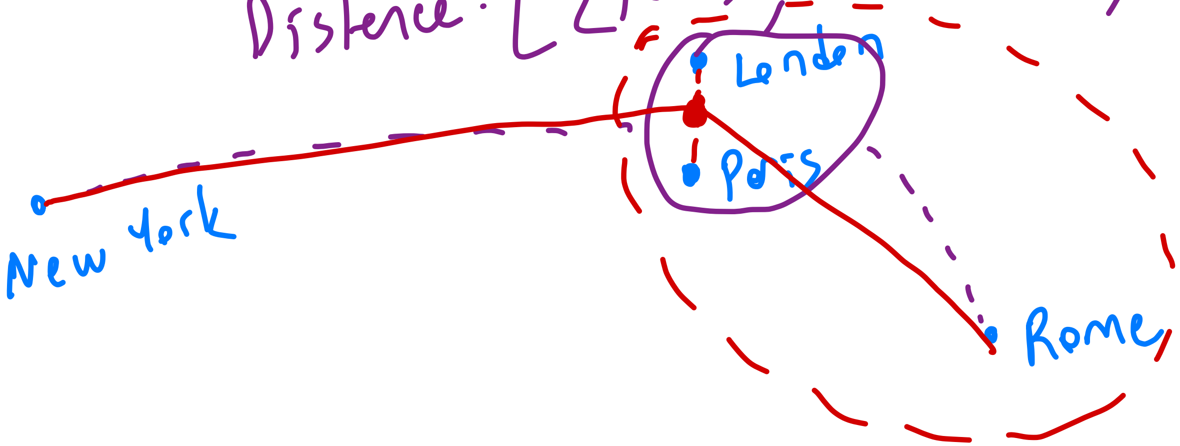
⇒ divides big clusters into smaller ones

⇒ starts with "1" cluster

Component #3: The Distance Function Between Groups of Data Points.

Distance [{Paris, London}, {New York}]

Distance. [{Paris, London}, {Rome}]



Distance [{London, Paris}, {Berlin, Rome}]

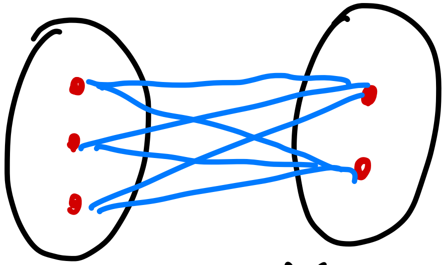
Centroid Clustering:

$$d(C_A, C_B) = \left\| \underbrace{\frac{\sum_{x_i \in C_A} x_i}{|C_A|}}_{\text{centroid of } C_A} - \underbrace{\frac{\sum_{j \in C_B} x_j}{|C_B|}}_{\text{centroid of } C_B} \right\|_2$$

cardinality of C_A (# of members)

Single-Link Clustering:

$$d(C_A, C_B) = \min_{\substack{x_i \in C_A \\ x_j \in C_B}} d(x_i, x_j)$$



Complete-Link Clustering:

$$d(C_A, C_B) = \max_{\substack{x_i \in C_A \\ x_j \in C_B}} d(x_i, x_j)$$

Average-Link Clustering:

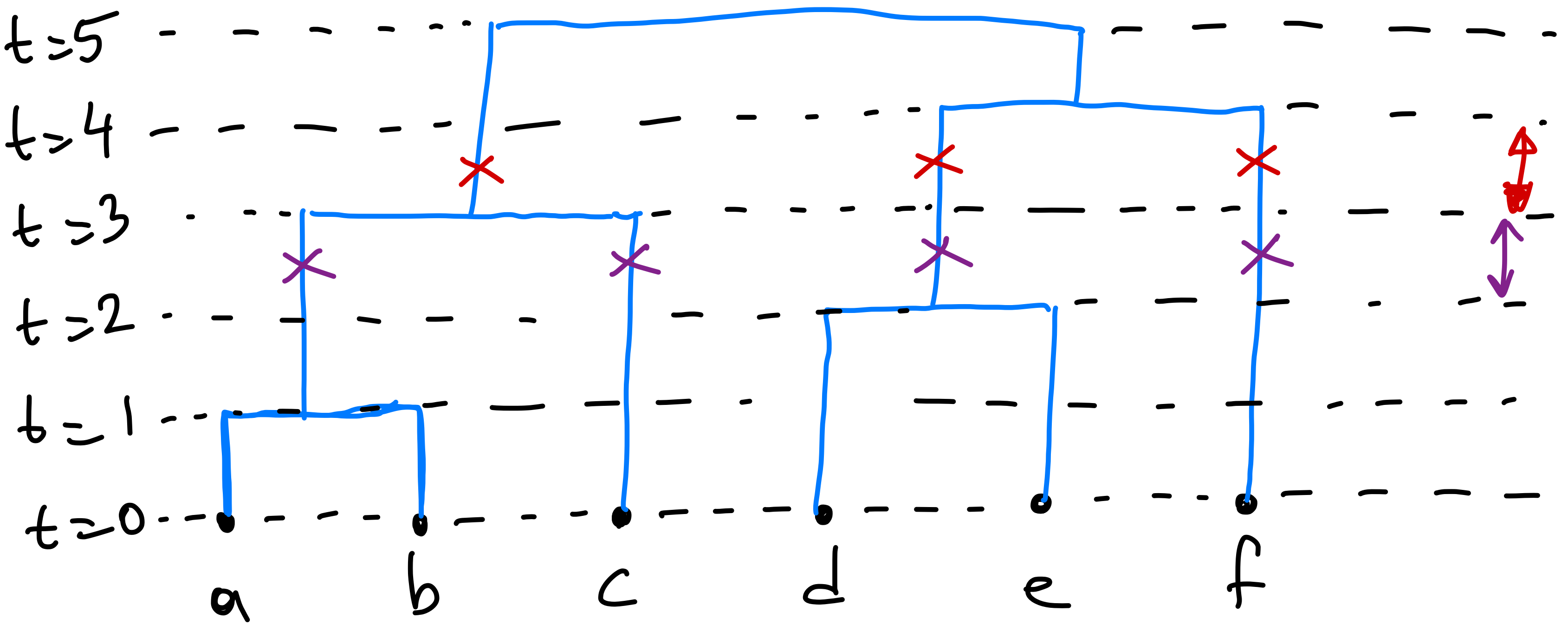
$$d(C_A, C_B) = \frac{\sum_{x_i \in C_A} \sum_{x_j \in C_B} d(x_i, x_j)}{|C_A|\,|C_B|}$$

$\vdots$

$t=0$	6 clusters	a	b	c	d	e	f
$t=1$	5 clusters	ab		c	d	e	f
$t=2$	4 clusters	ab		c	de		f
$t=3$	3 clusters	abc			de		f
$t=4$	2 clusters	abc			def		
$t=5$	1 cluster	abcdef					

Agglomerative

Dendrogram



K=3 clusters

$C_1=\{a,b,c\}$

$C_2=\{d,e\}$

$C_3=\{f\}$

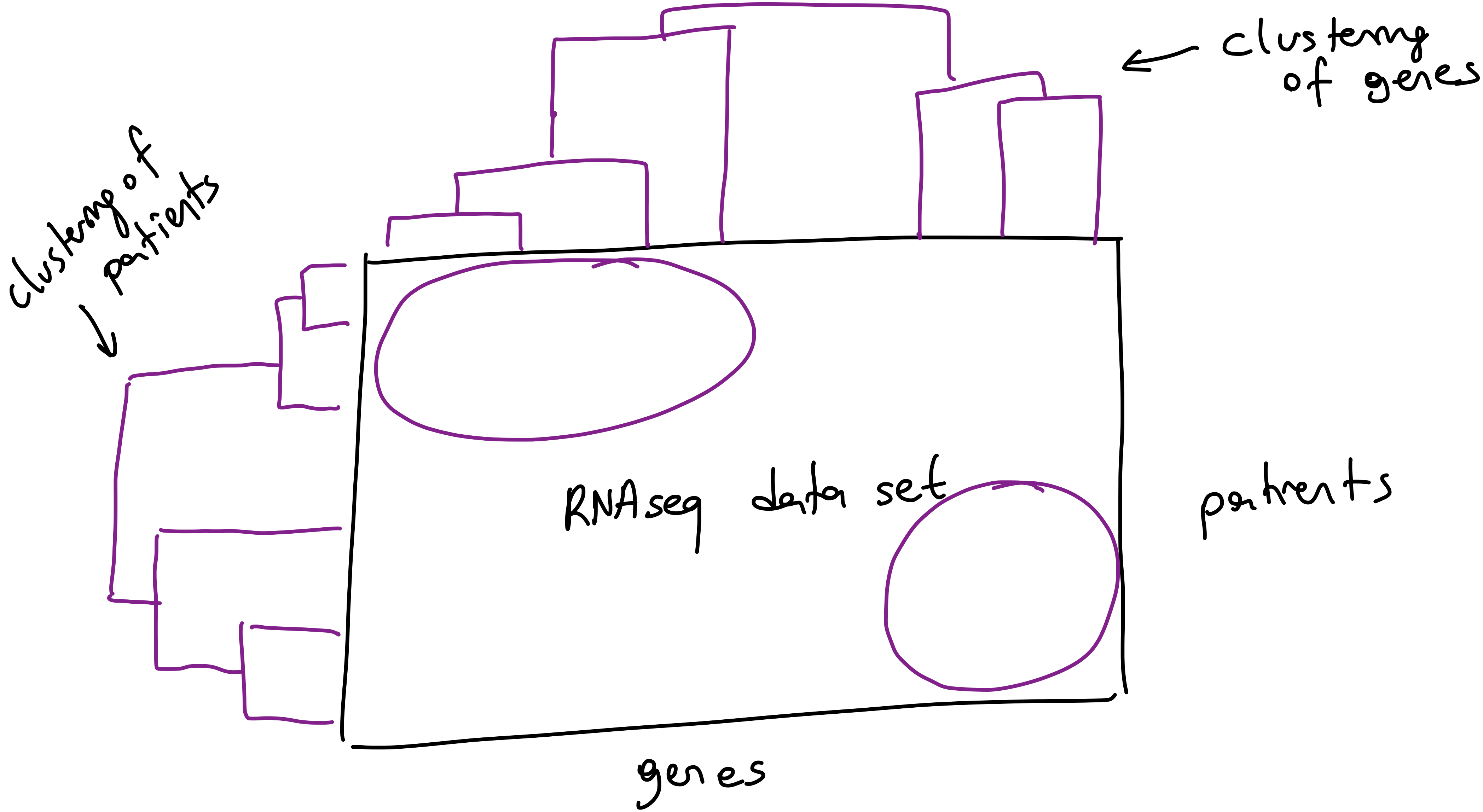
K=4 clusters

$C_1=\{a,b\}$

$C_2=\{c\}$

$C_3=\{d,e\}$

$C_4=\{f\}$



"bi-clustering"