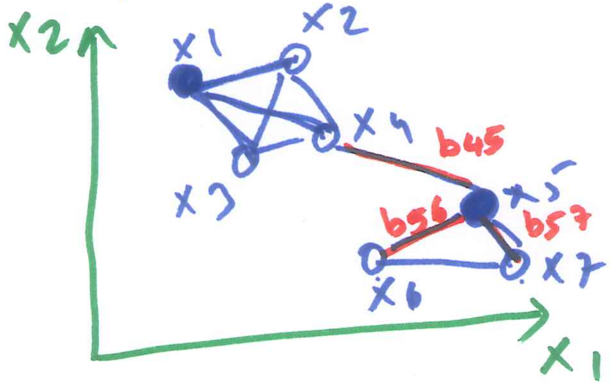
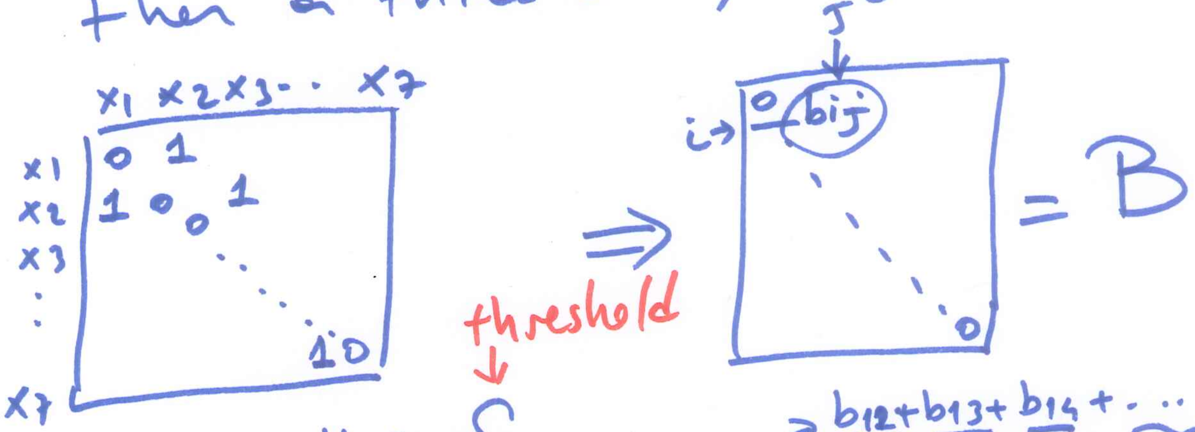
SPECTRAL CLUSTERING

- define local neighborhoods

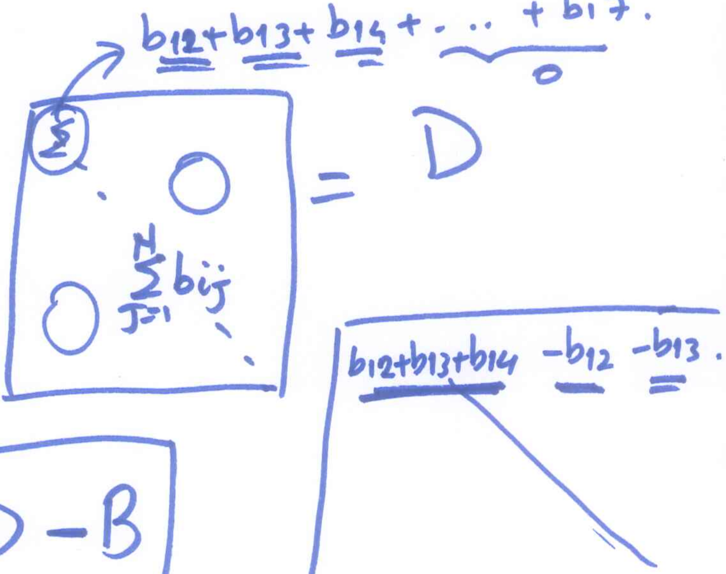


- distance between x_i & x_j is smaller then a threshold, they are neighbors.



$$b_{ij} = \begin{cases} 0 & \text{if the } \|x_i - x_j\| > \overset{\text{threshold}}{\delta} \\ \underbrace{\exp\left(-\frac{\|x_i-x_j\|^2}{2\sigma^2}\right)}_{k(x_i,x_j)} & \text{otherwise.} \end{cases}$$

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


$b_{12}+b_{13}+b_{14}+ \underbrace{\ldots + b_{17}}_{0}$.
Σ
0
0
$\sum_{j=1}^{N} b_{ij}$
= D
$b_{12}+b_{13}+b_{14}$ $-b_{12}$ $-b_{13}$.

Laplacian matrix ⇒ $\boxed{L = D - B}$

↳ each row sums up to 0

$$\underbrace{\mathcal{L}_{\text{RANDOM-WALK}}}_{N\times N} = D^{-1}.L = D^{-1}(D-B) = \underline{I - D^{-1}.B}$$

$$\underbrace{\mathcal{L}_{\text{SYMMETRIC}}}_{N\times N} = \underline{D^{-1/2}.L\,D^{-1/2}} = I - D^{-1/2}.B.D^{-1/2}$$

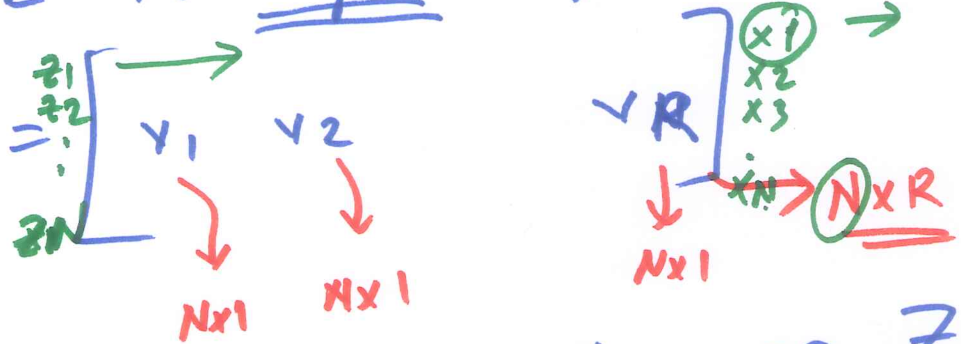
} normalized $\mathcal{L}$ matrices.

SPECTRAL CLUSTERING

Step 1: Find the eigenvectors of normalized $\mathcal{L}$ matrix

Step 2: Pick R largest eigenvectors.

Step 3: Z



Step 4: Run k-means algorithm on Z to find K clusters.

PARAMETERS

δ: distance threshold

R: # of eigenvectors to be included

K: # of clusters.

$k(x_i, x_j)$ $d(x_i, x_j)$

HIERARCHICAL CLUSTERING

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

First Component: Distance function between data points.

distance ⇒ dissimilarity.

distance ↑ similarity ↓

distance ↓ similarity ↑

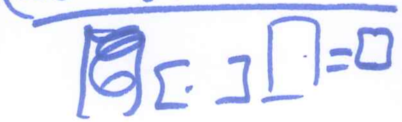
$$\downarrow\ \exp\left[-\frac{\overbrace{\{\|x_i-x_j\|_2^{2}\}}^{\text{distance}\uparrow}}{2\sigma^2}\right] = \exp\left[-\frac{d(x_i,x_j)^2}{2\sigma^2}\right]$$

↑ ↓

$d_E(P_1,P_2)=\underline{\underline{2\sqrt{2}}}$

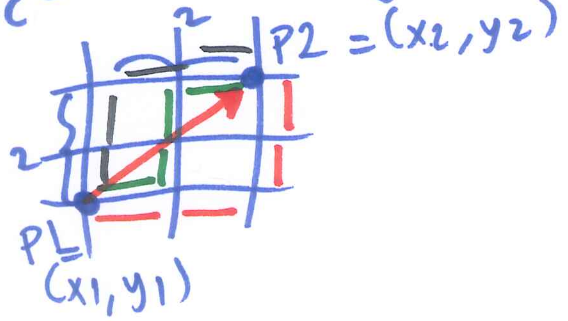
$d_C(P_1,P_2)=4$

Euclidean distance $= \|x_i-x_j\|_2 = \sqrt{\sum_{d=1}^{D}(x_{id}-x_{jd})^2} = \sqrt{(x_i-x_j)^T(x_i-x_j)}$



City-block distance $= \sum_{d=1}^{D}|x_{id}-x_{jd}|$

⋮



Second Component: Direction to proceed.

Agglomerative.

→ start with N clusters

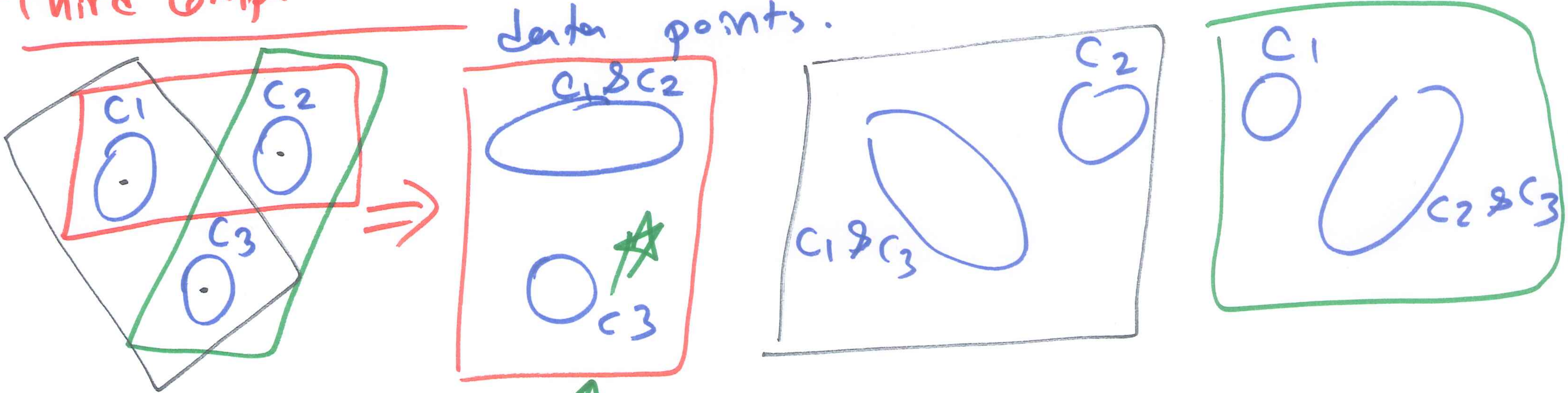
→ combine small clusters into bigger ones.

Divisive

→ Start with one cluster

→ Divide big cluster into smaller ones.

Third Component: Distance function between groups of data points.



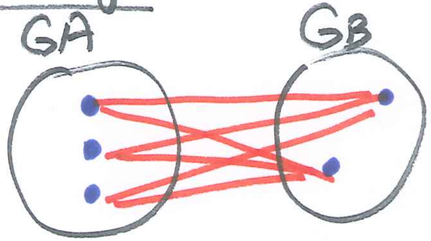
$d(C_1, C_2)$

$d(C_1, C_3)$

$d(C_2, C_3)$

} pick the smallest.

Single-link clustering: $d(G_A, G_B) = \underline{\min}_{x_i \in G_A, x_j \in G_B} \; d(x_i, x_j)$

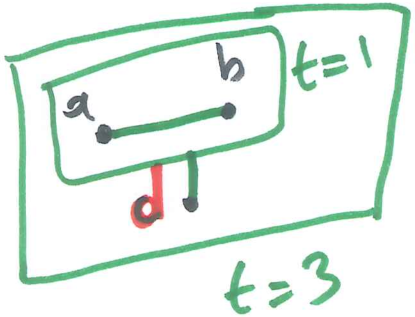


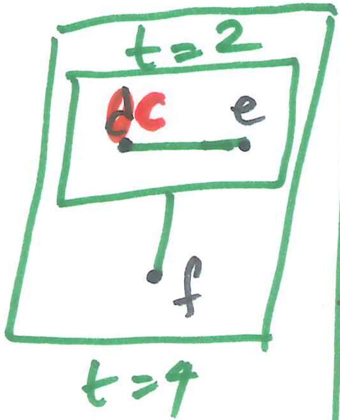
Complete-link clustering: $d(G_A, G_B) = \max_{x_i \in G_A, x_j \in G_B} \; d(x_i, x_j)$

Average-link clustering: $d(G_A, G_B) = \dfrac{\sum_{x_i \in G_A} \sum_{x_j \in G_B} d(x_i, x_j)}{\underbrace{|G_A|}_{\text{\# of data points in } G_A} \cdot \underbrace{|G_B|}_{\text{\# of data points in } G_B \text{ (cardinality)}}}$

Centroid clustering: $d(G_A, G_B) = \left\| \dfrac{\sum x_i}{|G_A|} - \dfrac{\sum x_j}{|G_B|} \right\|_2$

⋮





t=0 6 clusters {a}, {b}, {c}, {d}, {e}, {f}

t=1 5 clusters {a,b}, {c}, {d}, {e}, {f}

t=2 4 clusters {a,b}, {c}, {d,e}, {f}

t=3 3 clusters {a,b,c}, {d,e}, {f}

t=4 2 clusters {a,b,c}, {d,e,f}

t=5 1 cluster {a,b,c,d,e,f}

([illegible]) = [illegible]! / [illegible]!

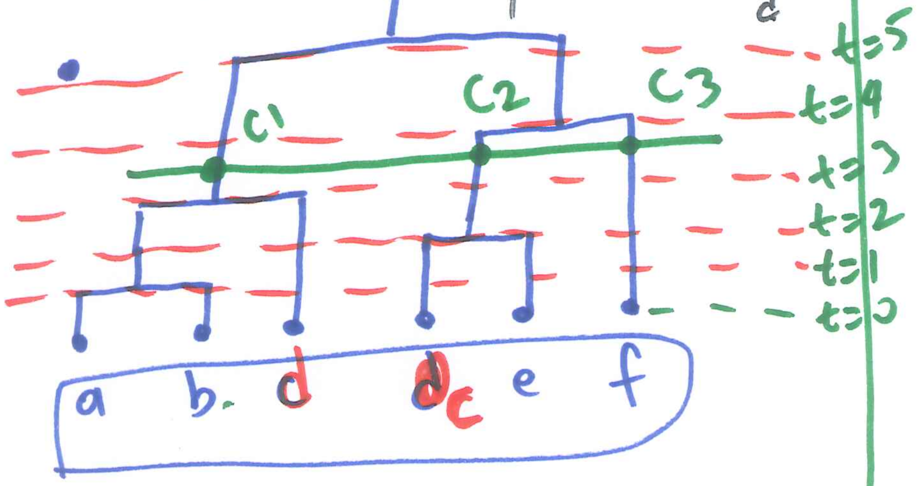
Dendrogram

abcd	ef	abcf	~~de~~
abce	df	abdf	ce
abde	cf	acdf	be
acde	bf	bcdf	ae
bcde	af		d
			d
			d

$\frac{n!}{k!(n-k)!}$

$$\begin{array}{l}
\cancel{6} \rightarrow 5 \text{ vs } 1 \\
15 \rightarrow \boxed{4 \text{ vs } 2} \\
\underline{20} \rightarrow \underline{\underline{3}} \text{ vs } \underline{\underline{3}} \\
41
\end{array}$$

2 vs 4



$\Longrightarrow$

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

$C_2 = \{d, e\}$

$C_3 = \{f\}$