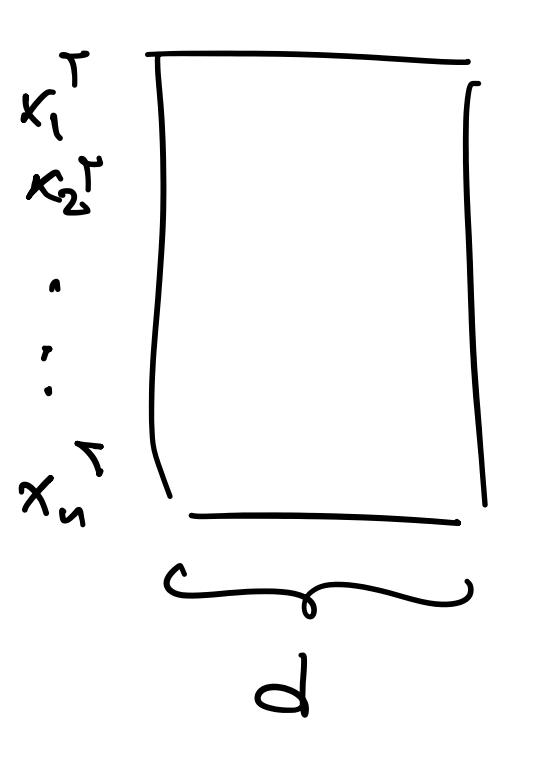
Lecture 9

Monday, September 30, 2024 9:57 AM

Clustering

Q: given D, find groups.



Partition of $D = \{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n\}$

$\vec{x}_i \in \mathbb{R}^d$

n points in d-dim space

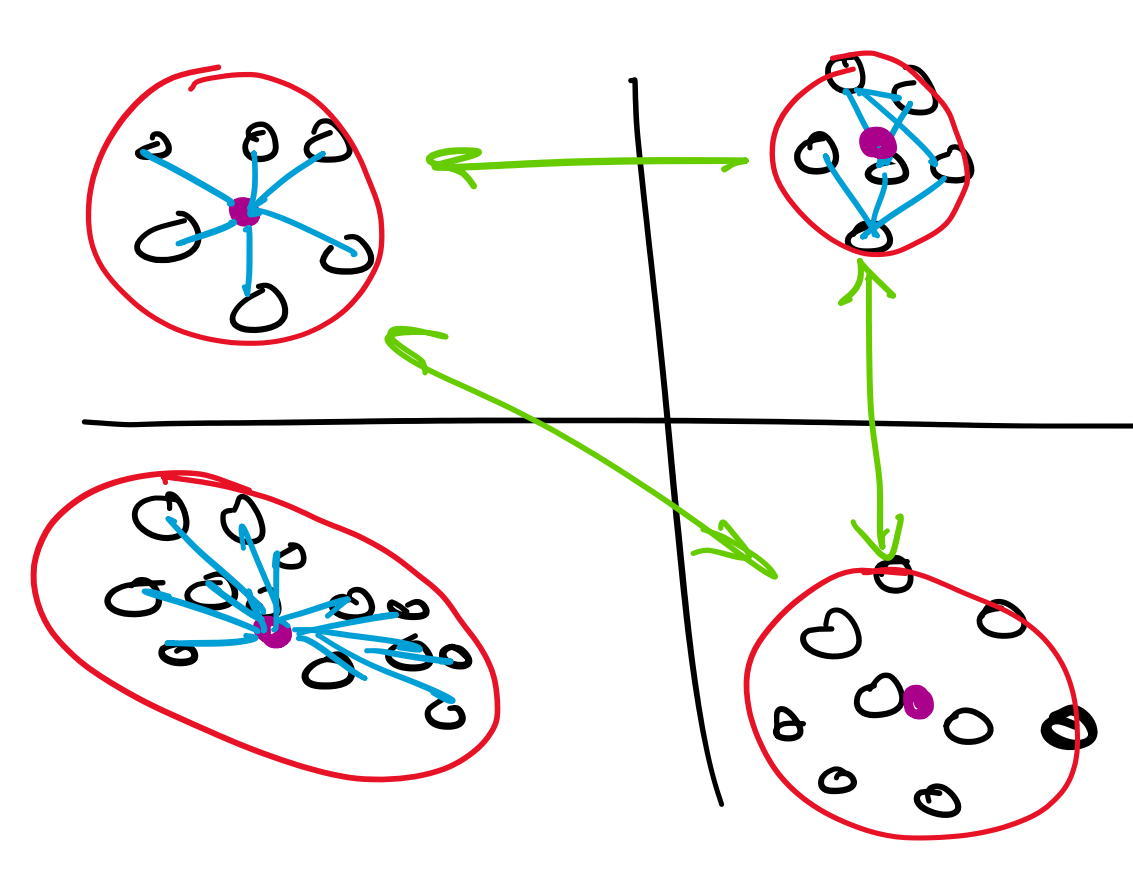
find k - parts
↖ user-specified # of clusters

$C_1, C_2, \ldots, C_k$

C_i is a set of points

① $C_i \subseteq D$
② $C_i \cap C_j = \phi$ ⟵ disjoint clusters
③ $\bigcup_{i=1}^{k} C_i = D$ ⟵ every point belongs to some cluster

] Partition



Similarity vs distance

1) minimize the intra cluster distance (max the intra cluster similarities)

2) maximize the inter-cluster distance (minimize the inter-cluster similarities)

SSE ≡ sum of squared errors

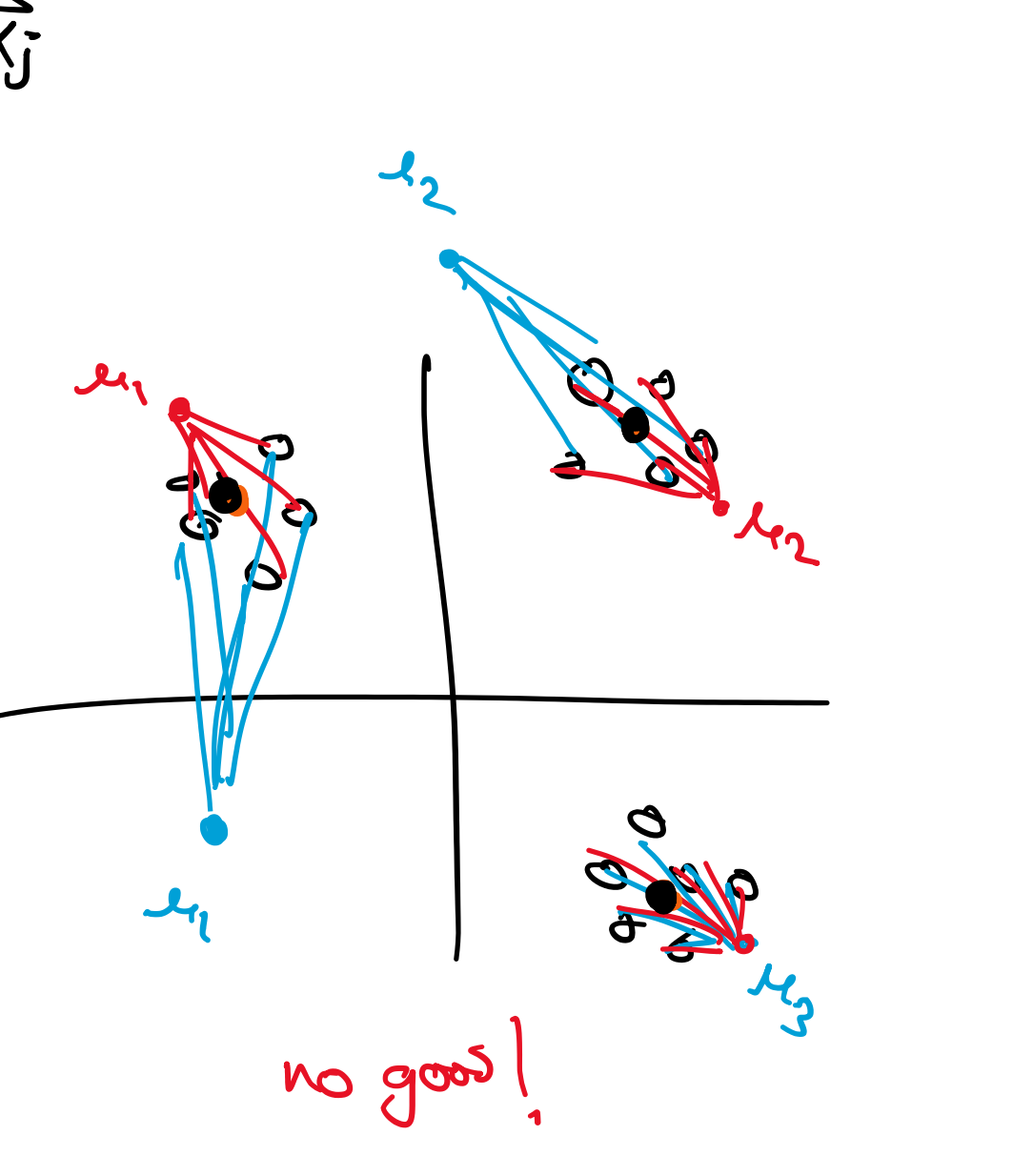
C_i = set of points in that cluster

↳ 'representative member'

Mean $\vec{\mu}_i = \frac{1}{|C_i|} \sum_{\vec{x}_j \in C_i} \vec{x}_j$

$$\min_{\vec{\mu}_1, \vec{\mu}_2, \vec{\mu}_k} SSE = \sum_{i=1}^{k} \left(\sum_{j=1}^{|C_i|} \| \vec{x}_j - \vec{\mu}_i \|^2 \right)$$

$\vec{\mu}_1, \vec{\mu}_2, \vec{\mu}_k$: unknown; $\sum_{i=1}^{k}$: # of clusters; $\vec{x}_j$: some point in cluster C_i; $\vec{\mu}_i$: mean for cluster C_i

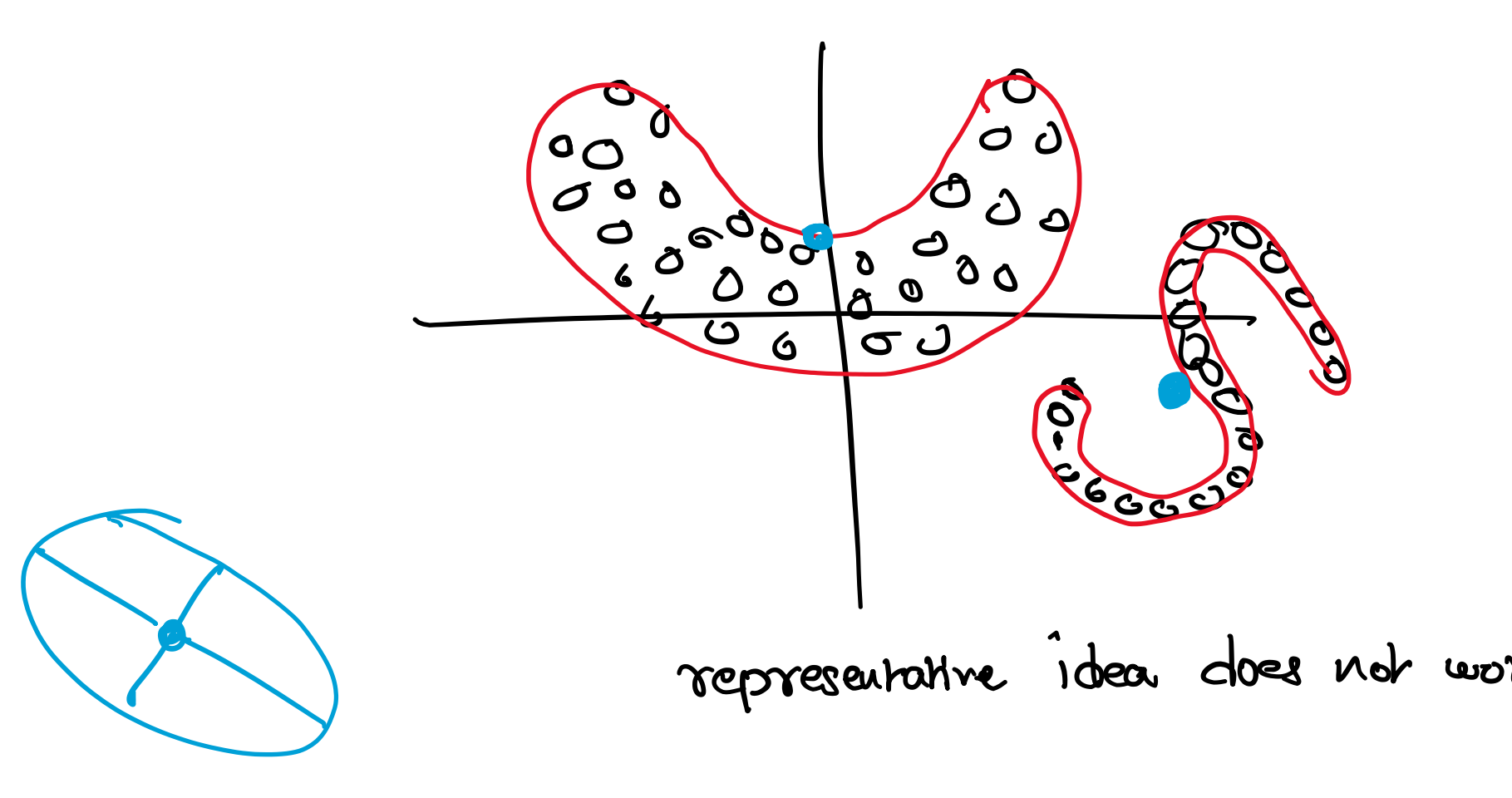


to find optimal solution, i.e. $\vec{\mu}_1, \vec{\mu}_2, \ldots, \vec{\mu}_k$ is NP-hard!

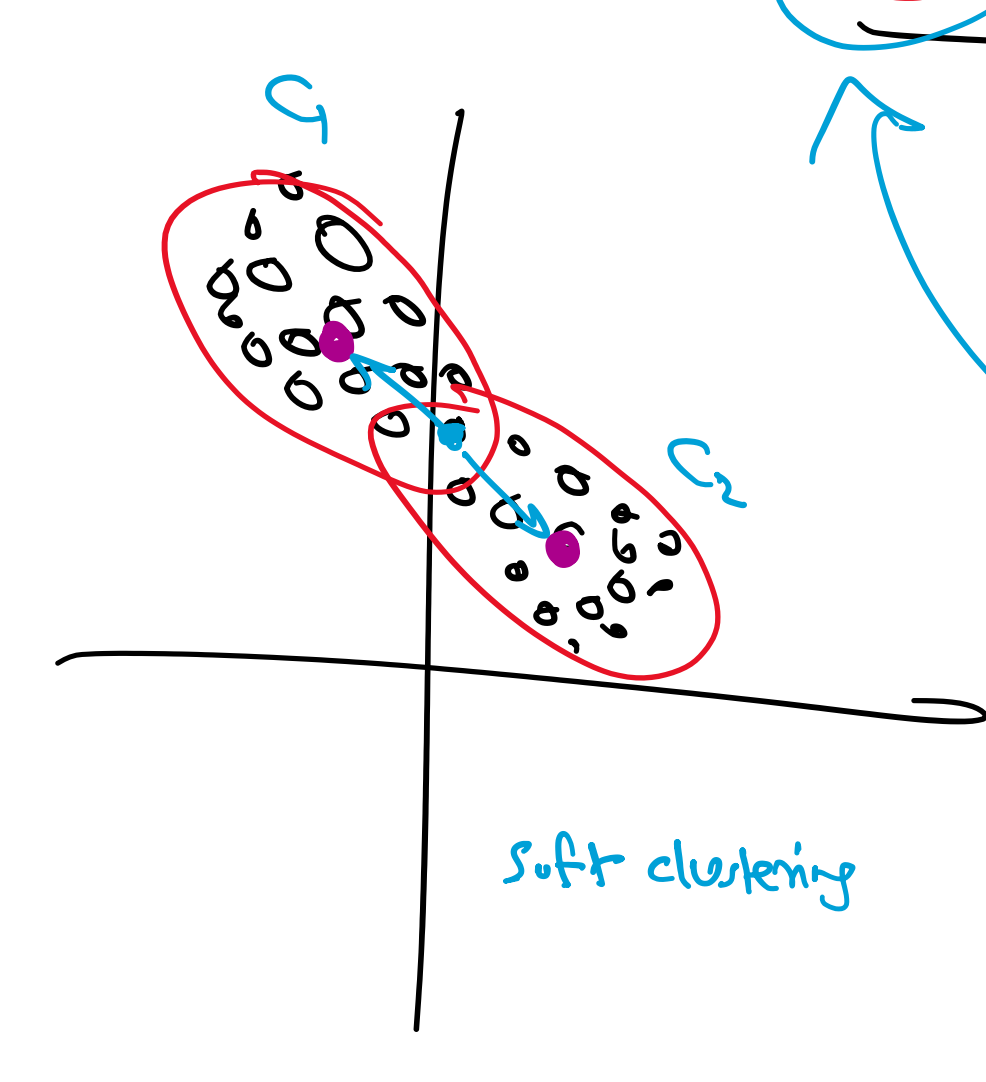
Naive Algo:

1) Enumerate all partitions of D into k groups ← exponential # of these
2) evaluate the SSE objective
3) return the one with least SSE value

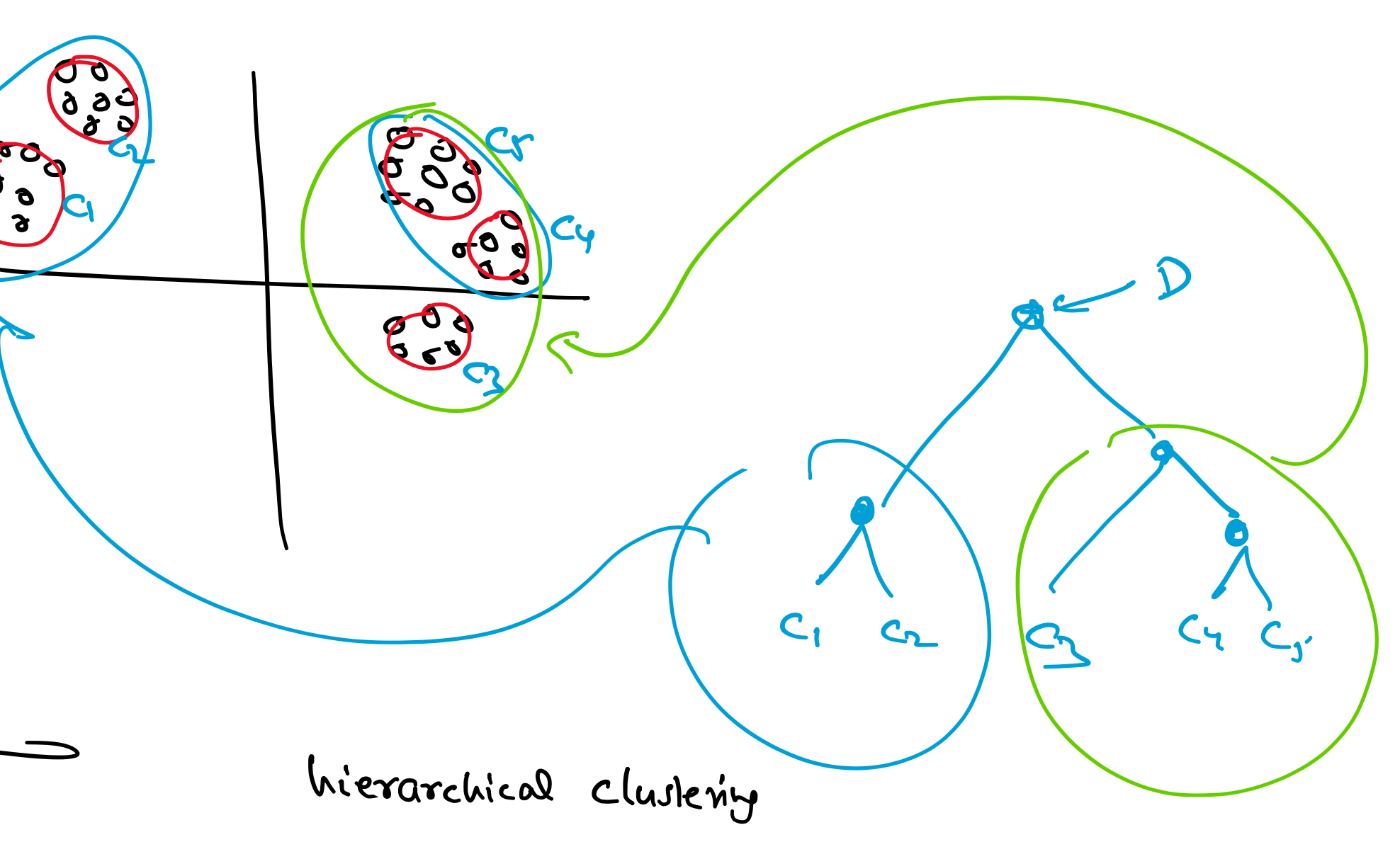
| Branch-and-bound



representative idea does not work



soft clustering



hierarchical clustering

K-Means : greedy algorithm (iterative)

$$SSE = \sum_{i=1}^{k} \sum_{j=1}^{|C_i|} \| \vec{x}_j - \vec{\mu}_i \|^2$$

Unknown parameters: $\vec{\mu}_1, \vec{\mu}_2, \ldots, \vec{\mu}_k$
k-means, one per cluster

---

0) random initialization

Pick any k points $\in D$ and designate as the k means

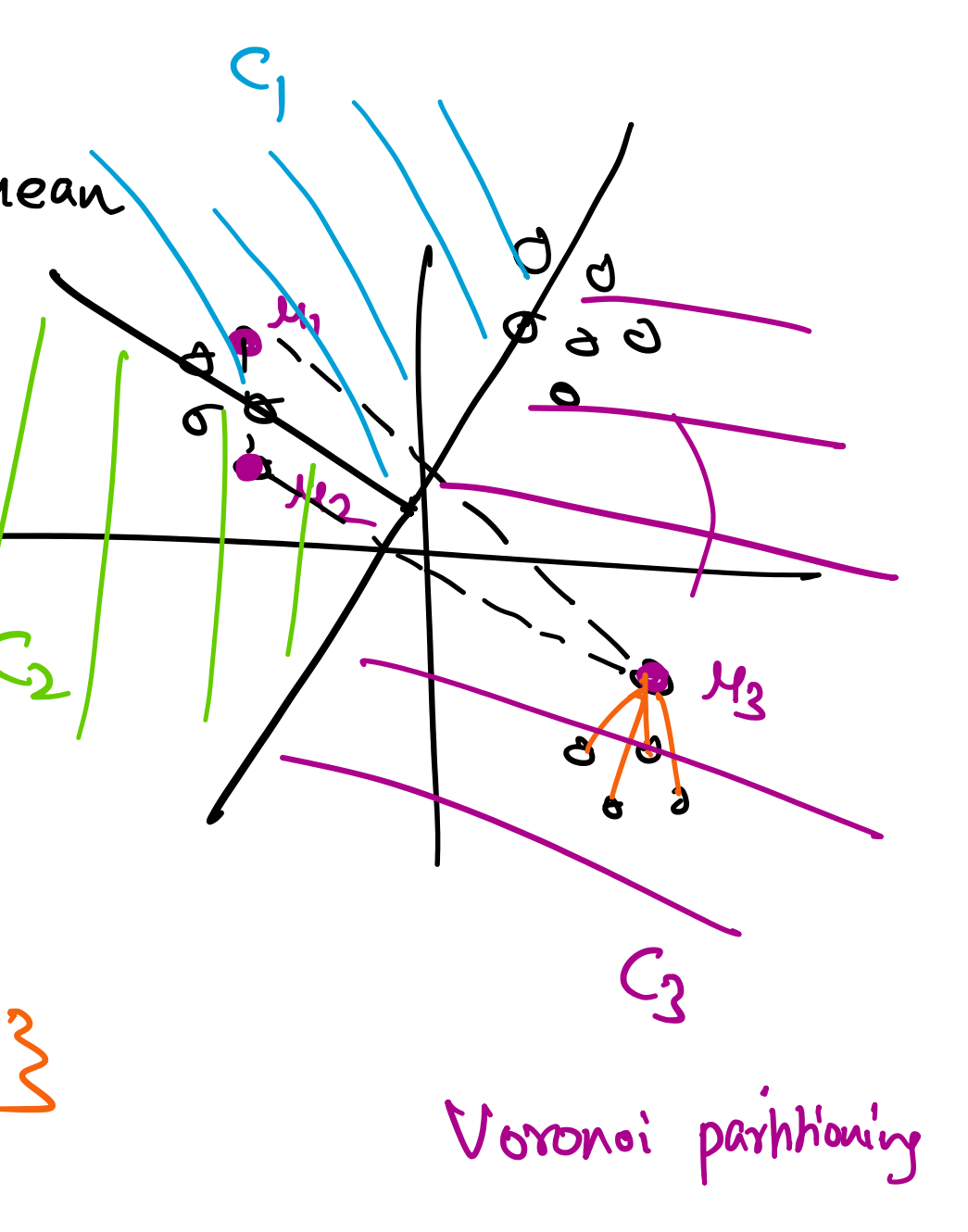
e.g. $\vec{\mu}_1 = \vec{x}_{10}$ $\quad n = 100$
$\vec{\mu}_2 = \vec{x}_{99}$ $\quad k = 3$
$\vec{\mu}_3 = \vec{x}_5$

Iterative improvement

1) given $\vec{\mu}_1, \vec{\mu}_2, \ldots, \vec{\mu}_k$

assign each point to the closest mean
partitioning step

$\forall \vec{x}_j \in D$
$\forall \vec{\mu}_i$
[find $\delta_{ij} = \| \vec{x}_j - \vec{\mu}_i \|^2$
↓
assign $\vec{x}_j$ to C_{i^*}
where $i^* = \arg\min_{i=1}^{k} \{ \delta_{ij} \}$



result in $C_1, C_2, \ldots, C_k$ (partitioning of D)

2) Given $C_1, C_2, \ldots, C_k$

update the means

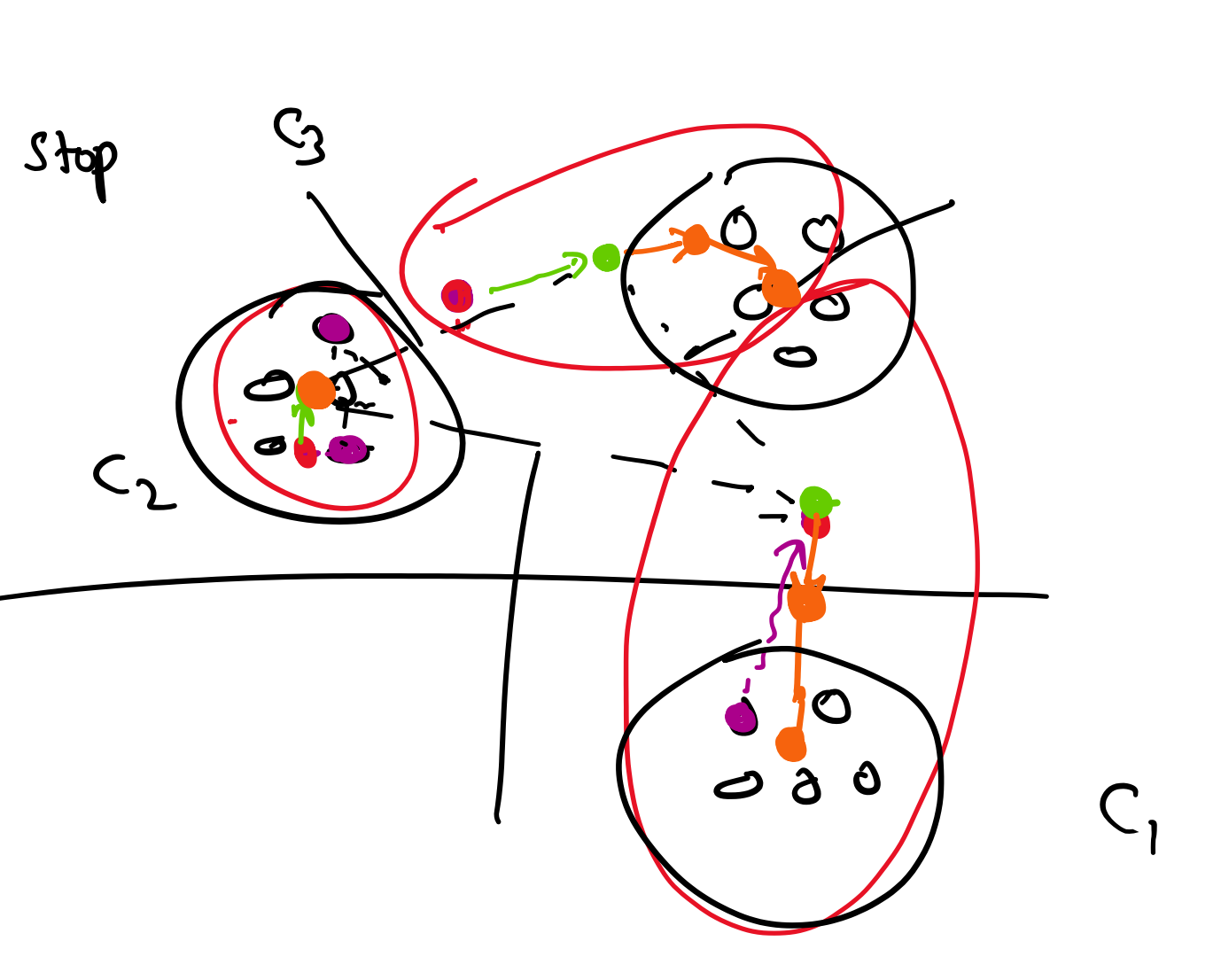
$\vec{\mu}_i = \frac{1}{|C_i|} \sum_{\vec{x}_j \in C_i} \vec{x}_j$

result in new estimate for the k means

3) repeat 1) & 2) until convergence

$\varepsilon = \sum_{i=1}^{k} \| \vec{\mu}_i^{t} - \vec{\mu}_i^{t-1} \|^2$ $\quad$ t is step

if $\varepsilon < 10^{-5}$ stop



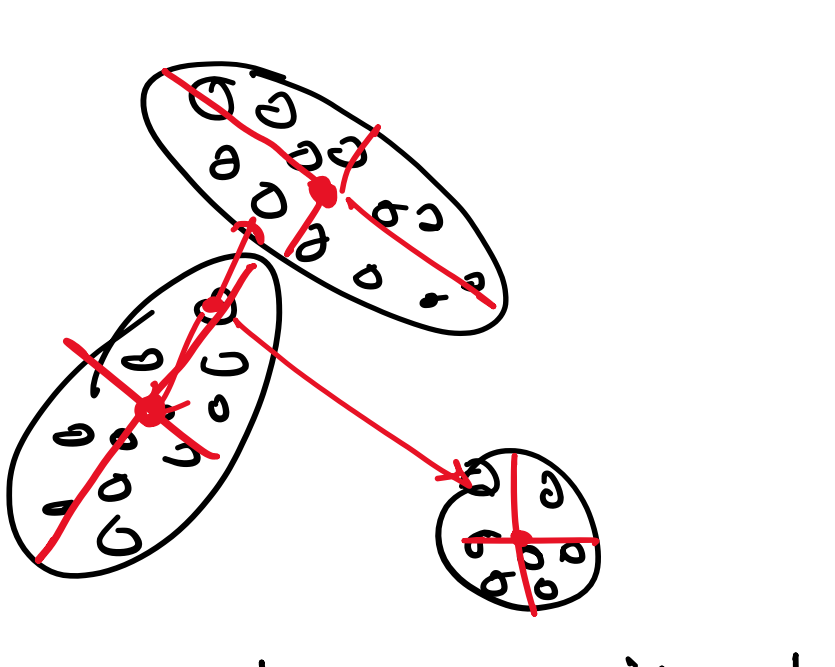
Probabilistic / soft assignment

$\{C_1, C_2, \ldots, C_k\}$
no longer a partition

[$P(C_i | \vec{x}_j) \equiv w_{ij}$
probability that point j ($\vec{x}_j$) belongs to cluster i (C_i)]

$\forall C_i, i = 1 \ldots k$

$\sum_{i=1}^{k} w_{ij} = 1 \quad \forall \vec{x}_j$



clustering: mixture of Gaussians

C_i → mean $\vec{\mu}_i$ d
→ cov matrix Σ_i d×d
→ size $P(C_i)$ {prior probability}

$$\Theta = \begin{Bmatrix} \text{means} & \text{cov} & \text{priors} \\ \vec{\mu}_1, & \Sigma_1, & P(C_1) \\ \vec{\mu}_2, & \Sigma_2, & P(C_2) \\ \vdots & & \\ \vec{\mu}_k, & \Sigma_k, & P(C_k) \end{Bmatrix}$$ Unknown

---

Expectation - Maximization Algorithm (EM)

0) Random initialization

a) Pick any k (random) points as $\vec{\mu}_1, \vec{\mu}_2, \ldots, \vec{\mu}_k$
b) assign each point $\vec{x}_j \in D$ to closest mean to obtain $\{C_1, C_2, \ldots, C_k\}$
c) for each C_i, compute Σ_i

$P(C_i) = \frac{|C_i|}{n}$

---

1) Expectation step: (soft assignment)

Given $\vec{\mu}_i, \Sigma_i, P(C_i)$ $\forall i = 1 \ldots k$
($\vec{\mu}_i$: d, Σ_i: d×d, $P(C_i)$: 1)

Compute $w_{ij} = P(C_i | \vec{x}_j)$?
$\forall \vec{x}_j \in D, \forall i = 1 \ldots k$

Computing for each point $\vec{x}_j$, the prob that it belongs to each of the k clusters

2) Maximization step (update params)

Given $w_{ij} = P(C_i | \vec{x}_j)$ $\forall x_j \in D, \forall C_i, i = 1 \ldots k$

Compute update to Θ

$\vec{\mu}_i = \frac{\sum_{j=1}^{n} w_{ij} \vec{x}_j}{\sum_{j=1}^{n} w_{ij}}$ $\quad$ weighted sum/mean
add up each point proportional to its weight/prob of belonging to C_i

e.g. uniform
$w_{ij} = \frac{1}{n}$
$\Rightarrow \frac{\sum_{j=1}^{n} \frac{1}{n} \vec{x}_j}{\sum_{j=1}^{n} \frac{1}{n}}$, $\sum_{j=1}^{n} \frac{1}{n} = 1$
$= \frac{1}{n} \sum \vec{x}_j$
mean!

$\Sigma_i = \frac{\sum_{j=1}^{n} w_{ij} (\vec{x}_j - \vec{\mu}_i)(x_j - \mu_i)^T}{\sum_{j=1}^{n} w_{ij}}$ ← weighted outer products for cov

$\Sigma = \frac{\sum (x_i - \mu)(x_i - \mu)^T}{n}$

$P(C_i) = \frac{\sum_{j=1}^{n} w_{ij}}{n}$ ⟵ total 'weight' for C_i over all the points

3) stop when $\varepsilon < 10^{-5}$

$\varepsilon = \sum_{i=1}^{k} \| \vec{\mu}_i^{t} - \vec{\mu}_i^{t-1} \|^2$

---

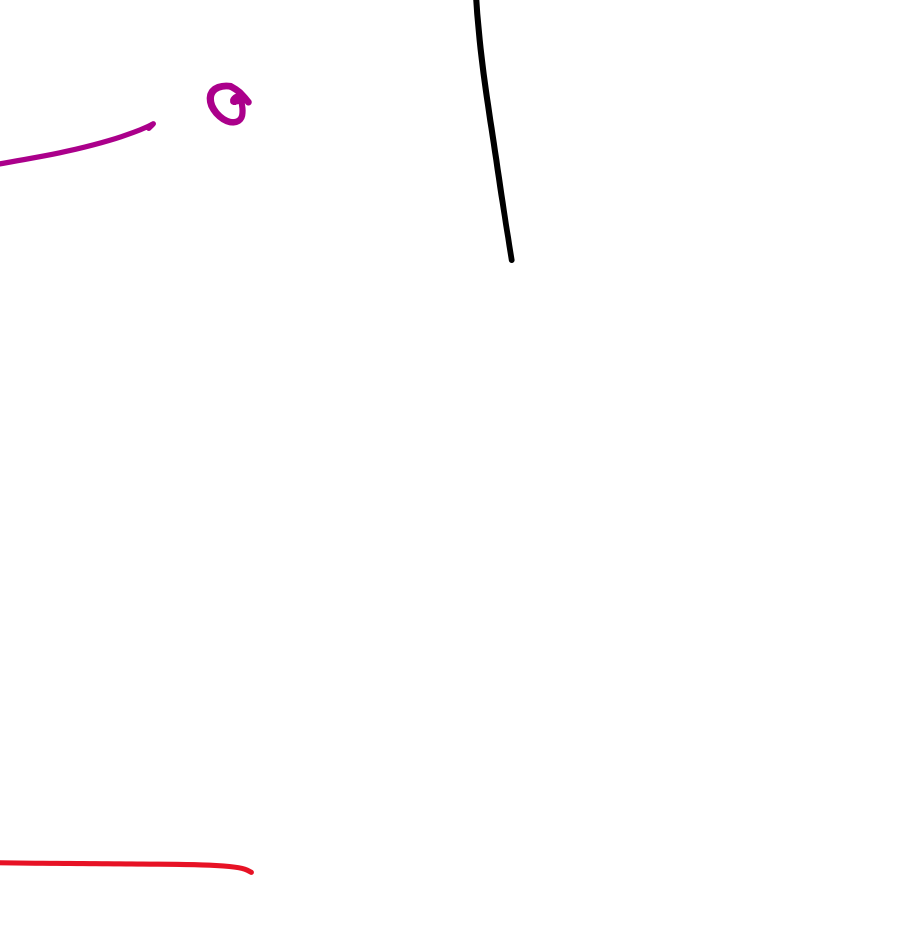
how to compute

$w_{ij} = P(C_i | \vec{x}_j)$ $\quad$ Prob that $\vec{x}_j$ belongs to C_i

$P(C_i | x_j) = \frac{P(\vec{x}_j | C_i) \cdot P(C_i)}{P(\vec{x}_j)}$ (likelihood, prior; $P(\vec{x}_j)$ ← evidence)

Posterior probability

$= \frac{P(\vec{x}_j | C_i) \cdot P(C_i)}{\sum_{i=1}^{k} P(x_j | C_i) \cdot P(C_i)}$ ($P(x_j)$)



$P(C_i | x_j) = \frac{N(\vec{x}_j | \vec{\mu}_i, \Sigma_i) \cdot P(C_i)}{P(x_j)}$

---

w_{ij} ⟶ hard clustering

assign each point to the max prob cluster