CLUSTERING

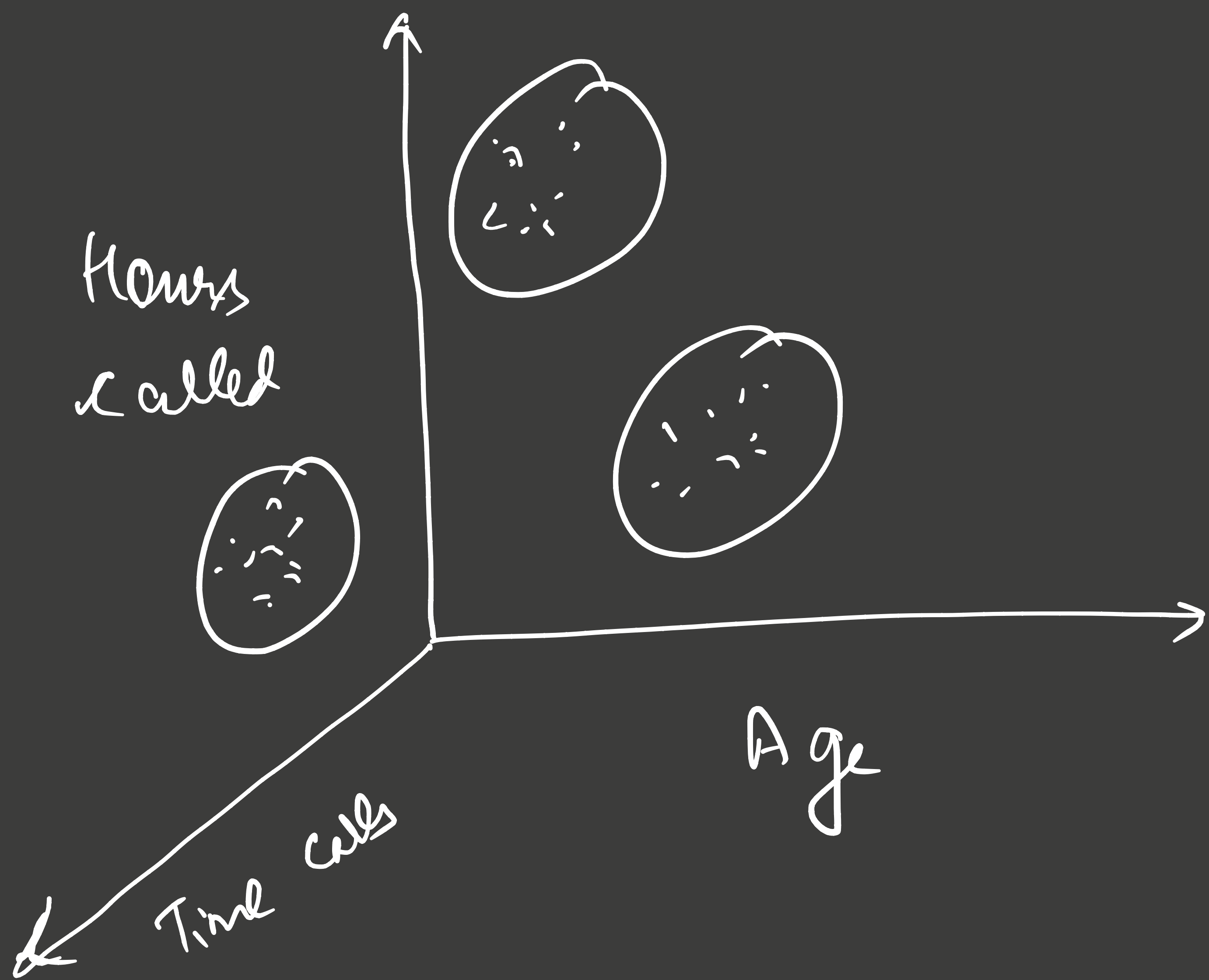
* FIND SUBGROUPS / CLUSTERS IN DATASET

* NEED TO DEFINE SIMILARITY / DISSIMILARITY

* EXAMPLES

  MARKET SEGMENTATION

  — DIFFERENT PLANS FOR DIFFERENT CLASSES / GROUPS



College folks: Age < 30

Hours called: ...

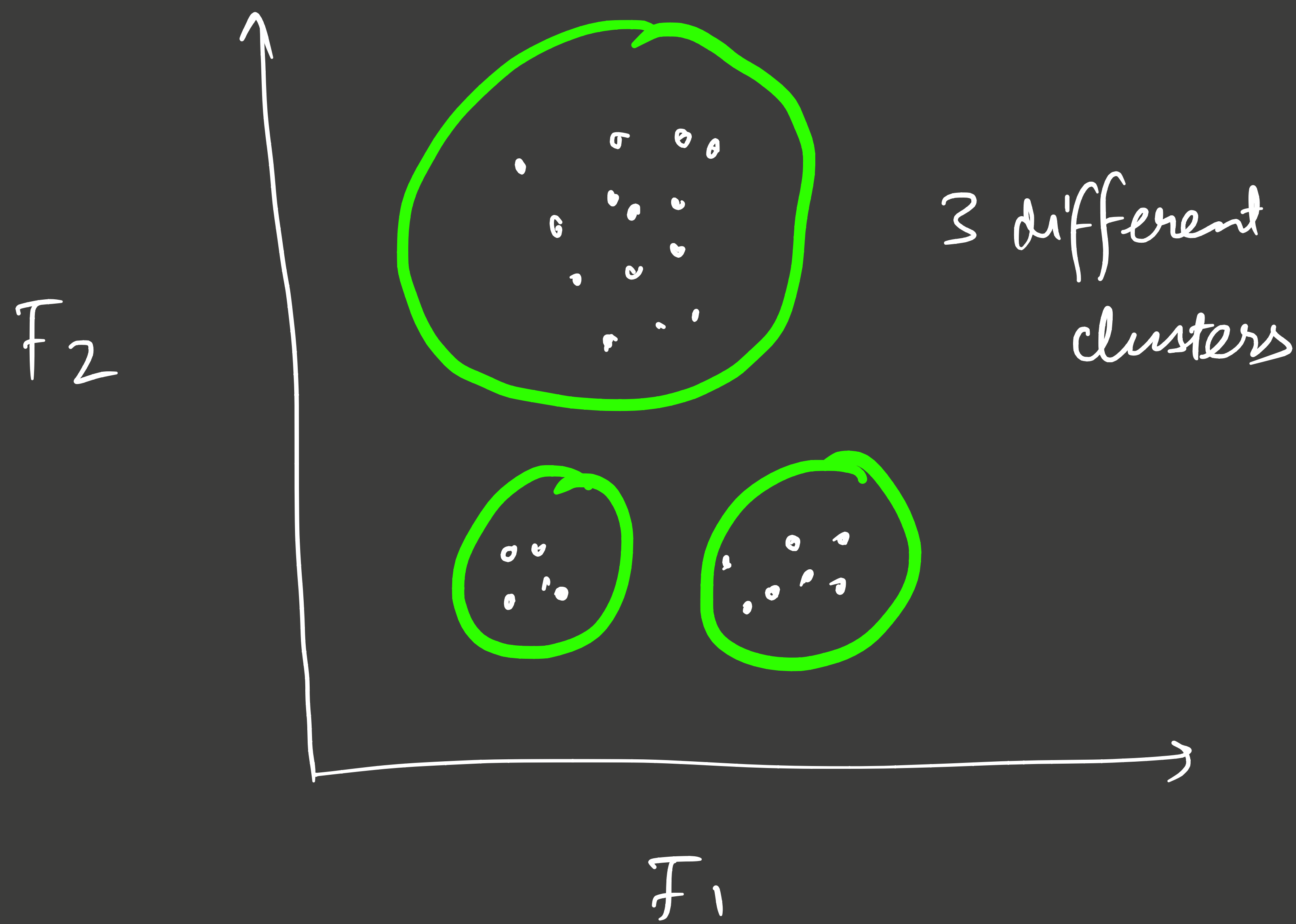
Time: > 9pm

Group 2: Age > 55

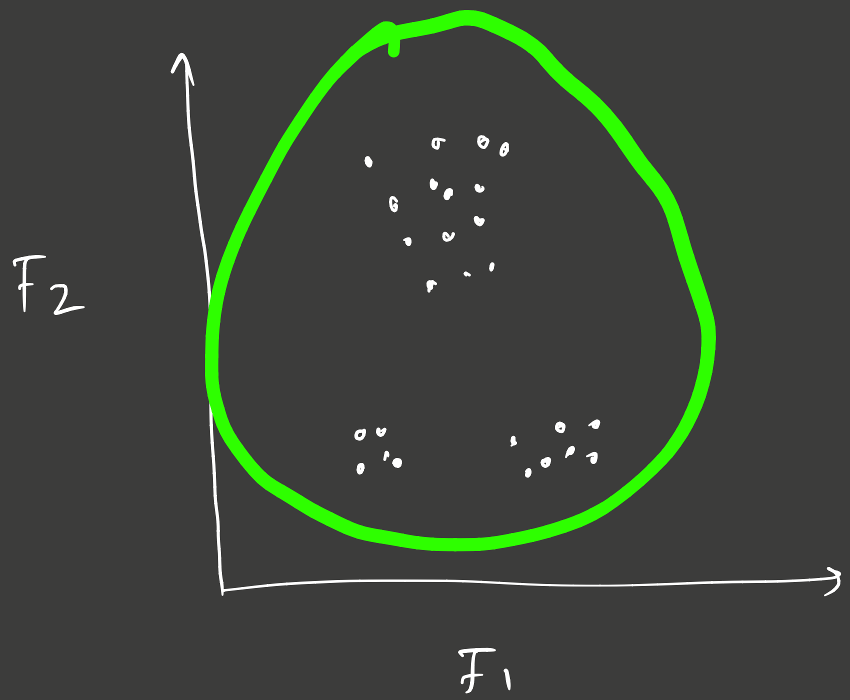
Hours: ...

Time: ...

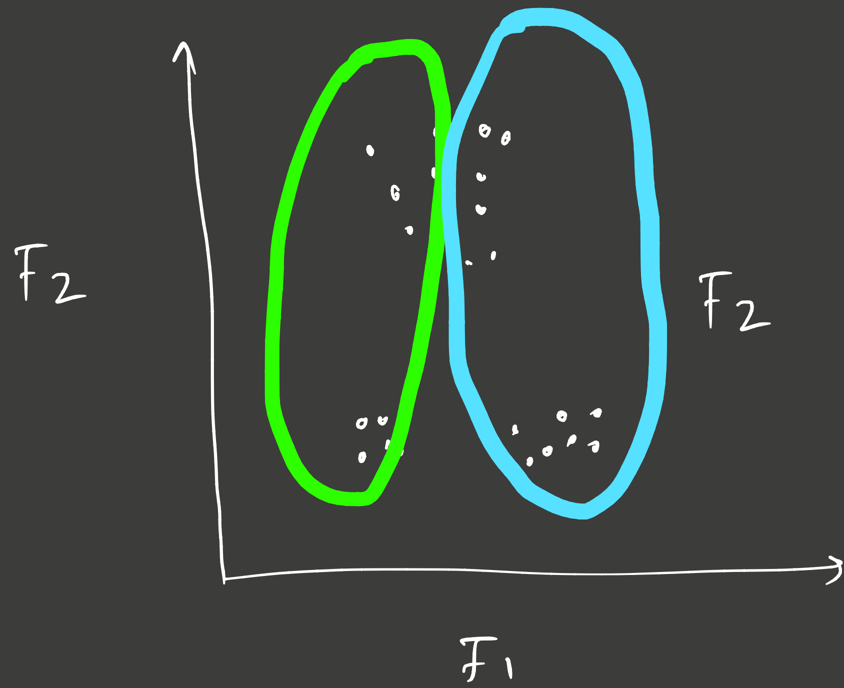
K-MEANS CLUSTERING



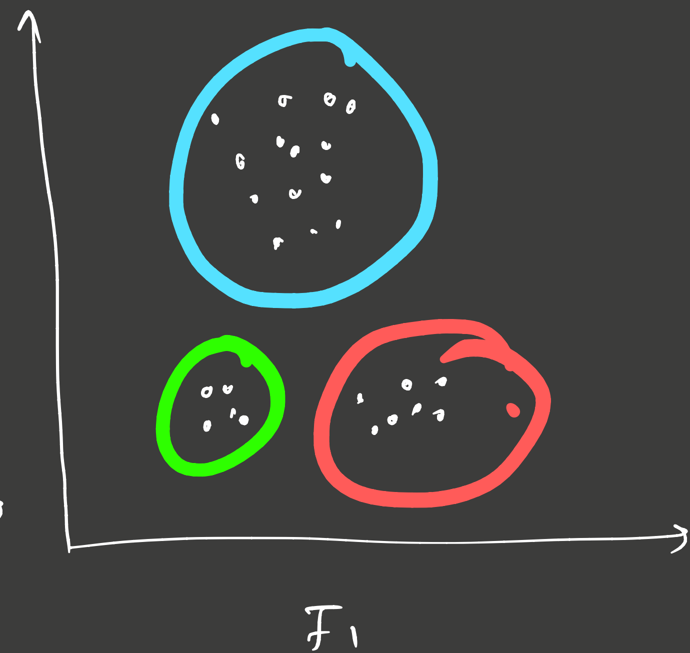
K=1 cluster



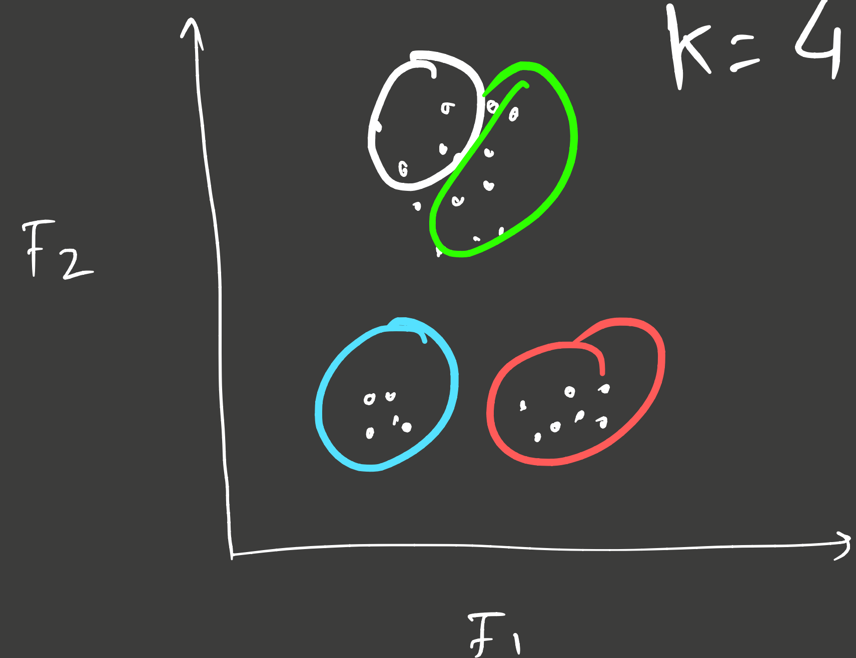
K=2 clusters



K=3 clusters

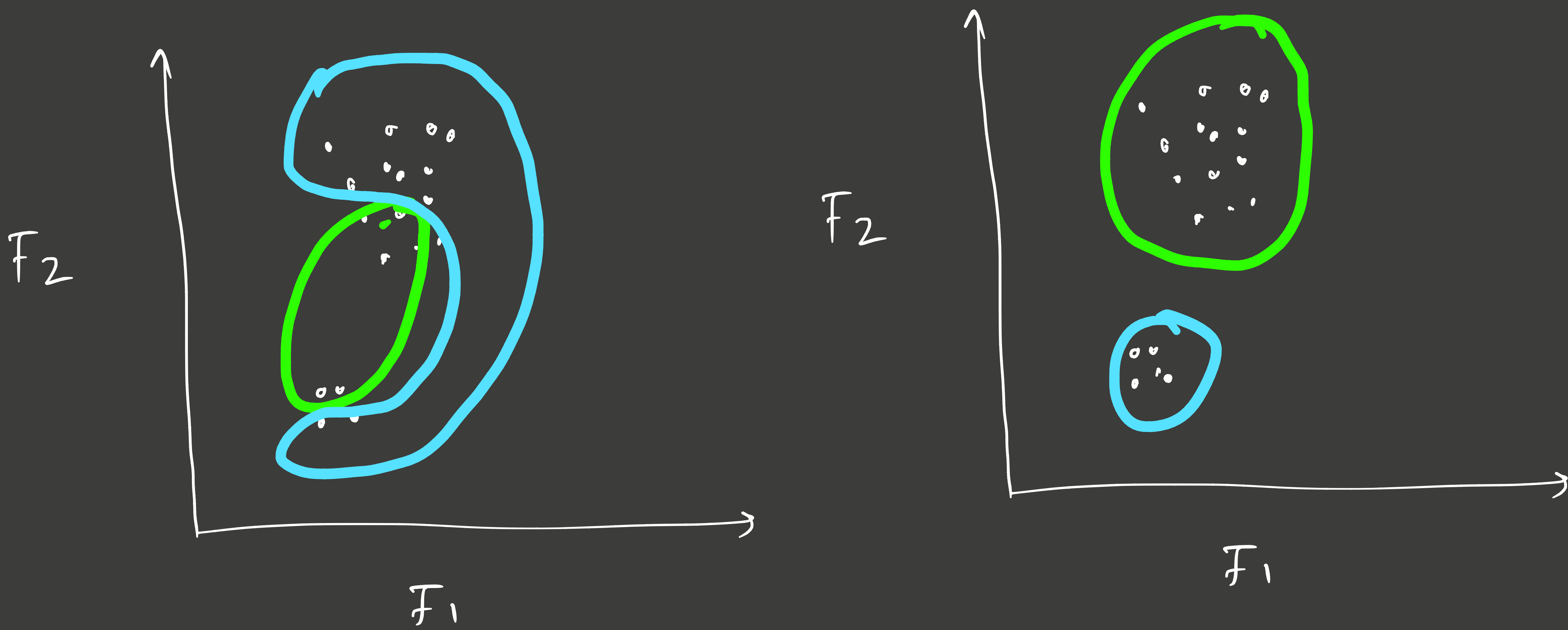


K=4 clusters



K-Means setting

* N points in R^d space
* C_i : set of points in i^{th} clusters
* $C_1 \cup C_2 \cup \dots C_K = \{1, \dots n\}$
* $C_k \cap C_{k'} = \{\phi\}$ for $k \neq k'$



WHICH CLUSTERING IS
BETTER & WHY?

K-Means Intuition

* GOOD CLUSTERING: WITHIN CLUSTER VARIATION IS SMALL (WCV)

* Objective: $\min\limits_{C_1, C_2 .. C_K} \left(\sum_{i=1}^{K} WCV(C_i) \right)$

$\downarrow$

Total WCV is as small as possible

K-Means Intuition

* Objective: $\min_{C_1, C_2 .. C_K} \left(\sum_{i=1}^{K} WCV(C_i) \right)$

$$WCV(C_i) = \frac{1}{|C_i|} \underbrace{\sum_{a \in C_i} \sum_{b \in C_i} \|x_a - x_b\|_2^2}_{\text{squared Euclidean distance b/w all pairs of points}}$$

$|C_i|$: # points in C_i

K-MEANS ALGORITHM

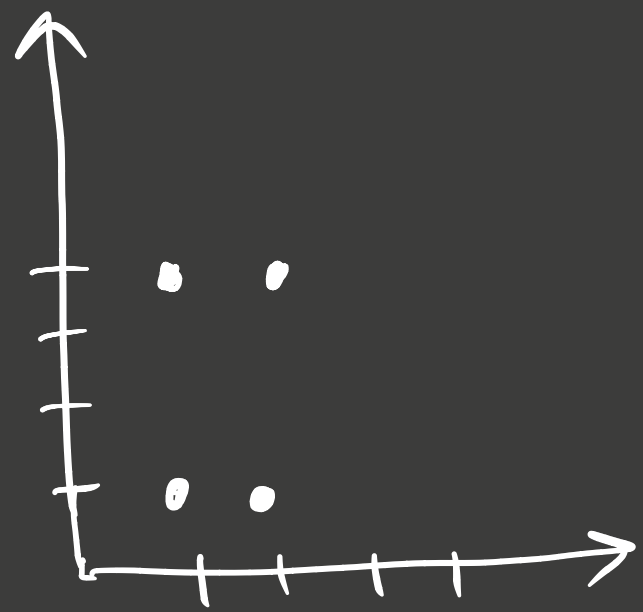
* Randomly assign cluster # (1, .. K) to every point.

* Iterate till convergence:

  * For each cluster C_i compute centroid (mean over points in C_i over 'd' dimensions)

  * Assign each observation to cluster whose centroid is closest

EXAMPLE RUN



WHY K-MEANS WORKS?

$$WCV(C_i) = \frac{1}{|C_i|} \sum_{a \in C_i} \sum_{b \in C_i} \|x_a - x_b\|_2^2 \quad \ldots (1)$$

$x_i \in \mathbb{R}^d$ = Centroid for i^{th} cluster

$$= \frac{1}{|C_i|} \sum_{a \in C_i} x_a$$

$\therefore$ Rewrite (1) as:

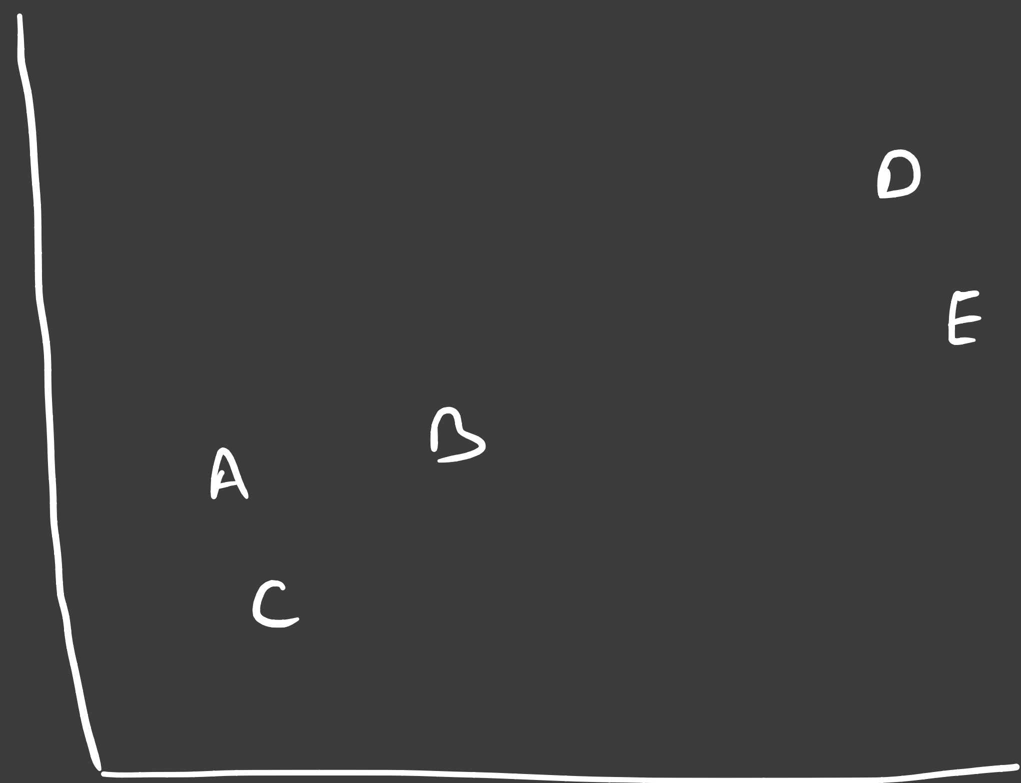
$$WCV(C_i) = 2 \sum_{a \in C_i} \|x_a - x_i\|_2^2$$

K-Means gives local optima!

HIERARCHICAL CLUSTERING

* Clustering for "all" # of clusters.
    * NO need to 'pre-specify' $\underline{\underline{K}}$ like K-means.

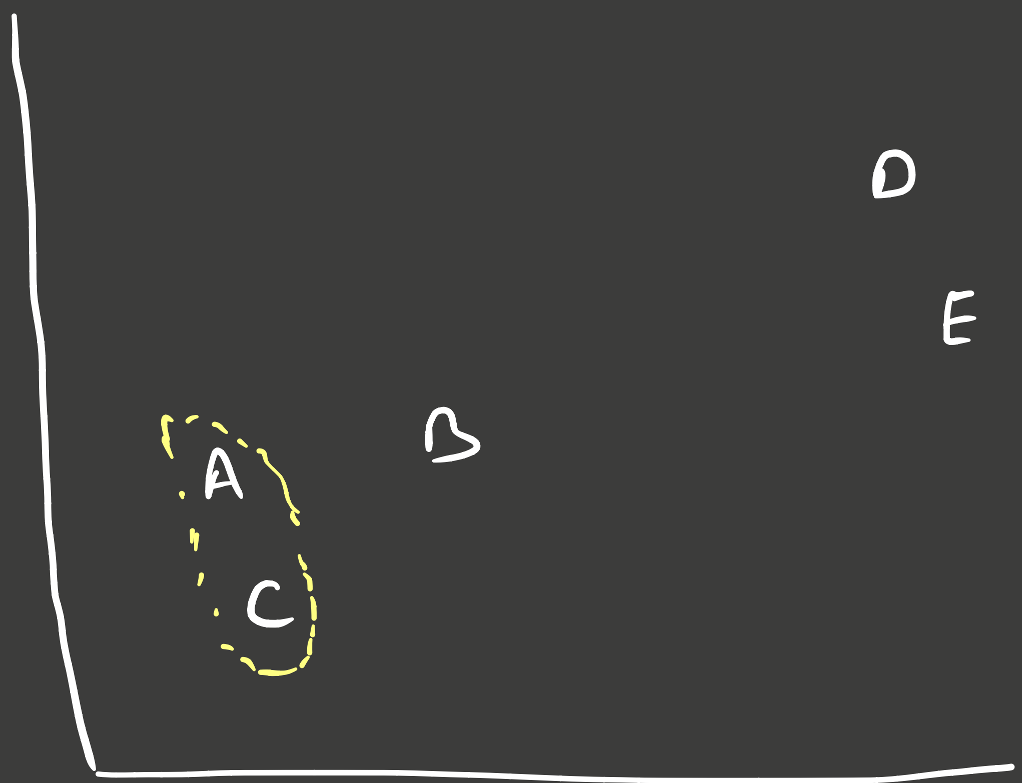
HIERARCHICAL CLUSTERING



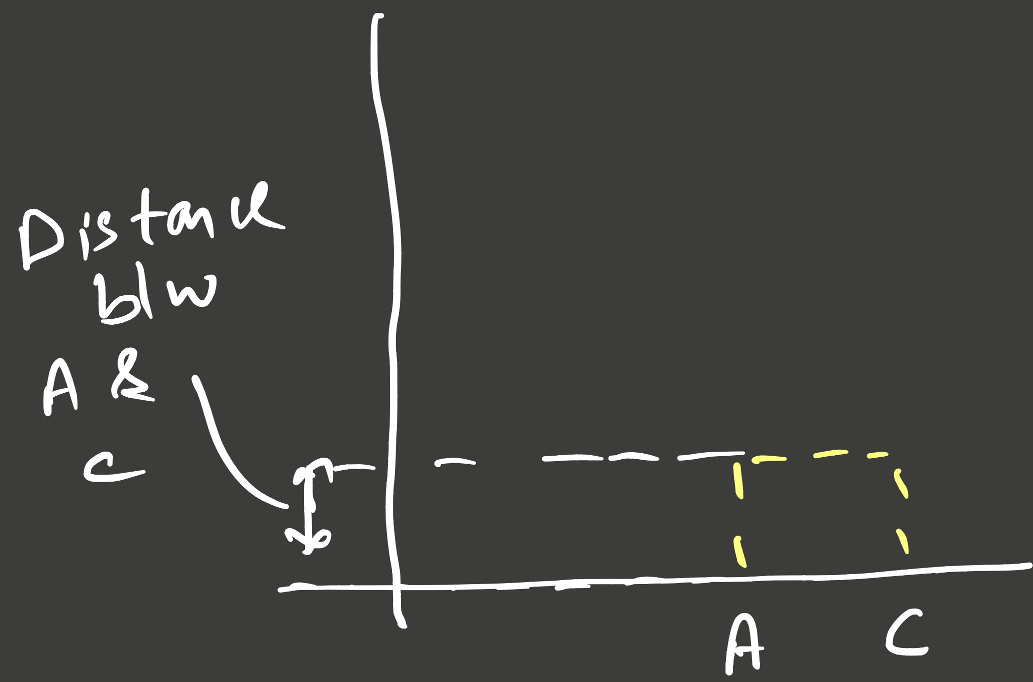
Dendogram

HIERARCHICAL CLUSTERING

* Start with each point in own cluster
* Identify closest 2 clusters → Merge

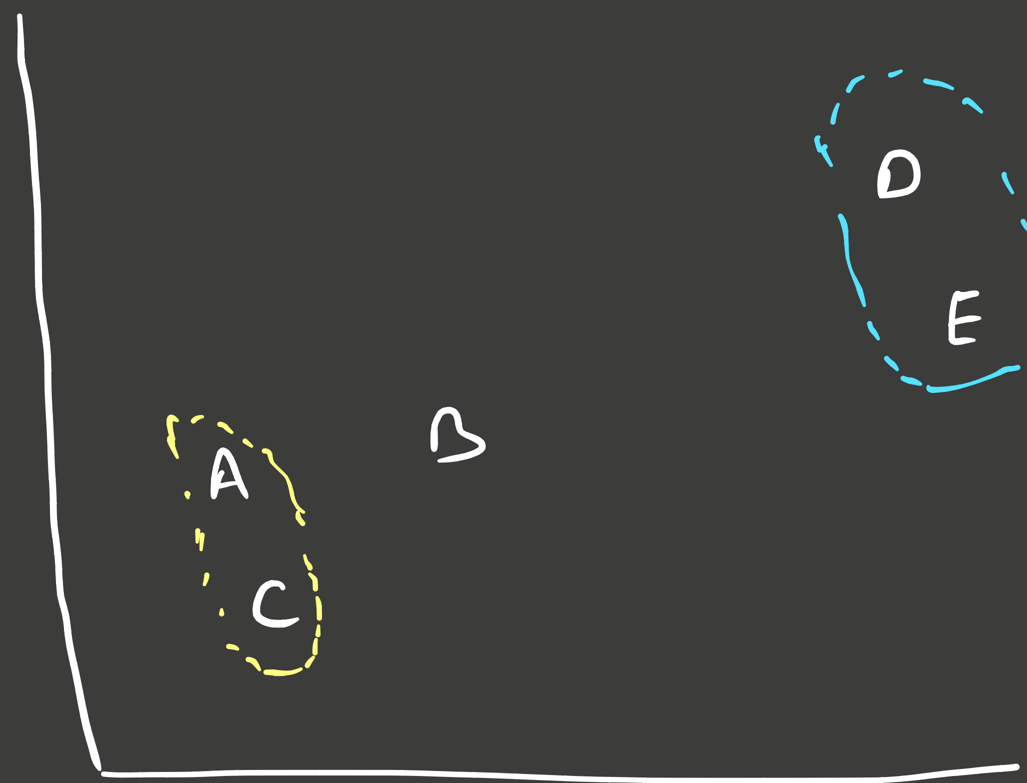


Dendogram

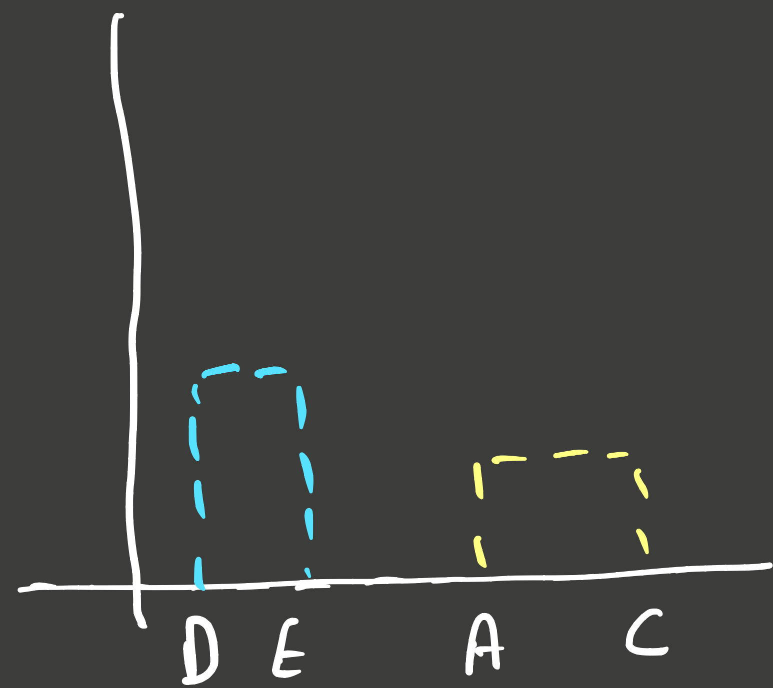


HIERARCHICAL CLUSTERING

* Start with each point in own cluster
  * Identify closest 2 clusters → Merge
  * Repeat

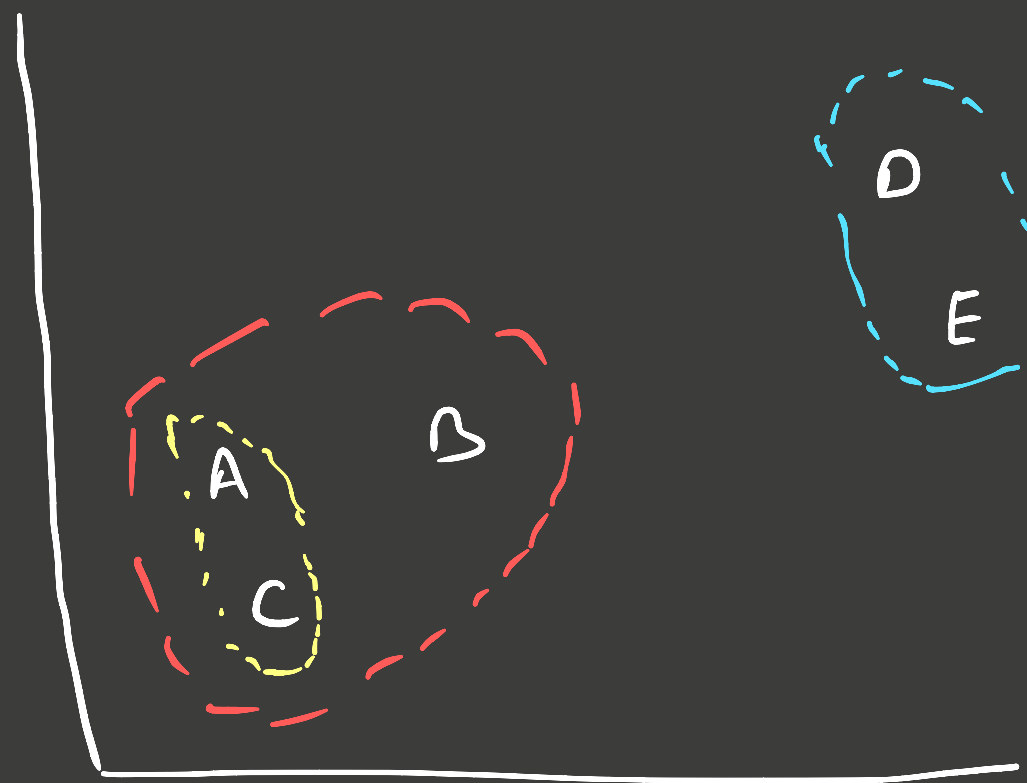


Dendrogram

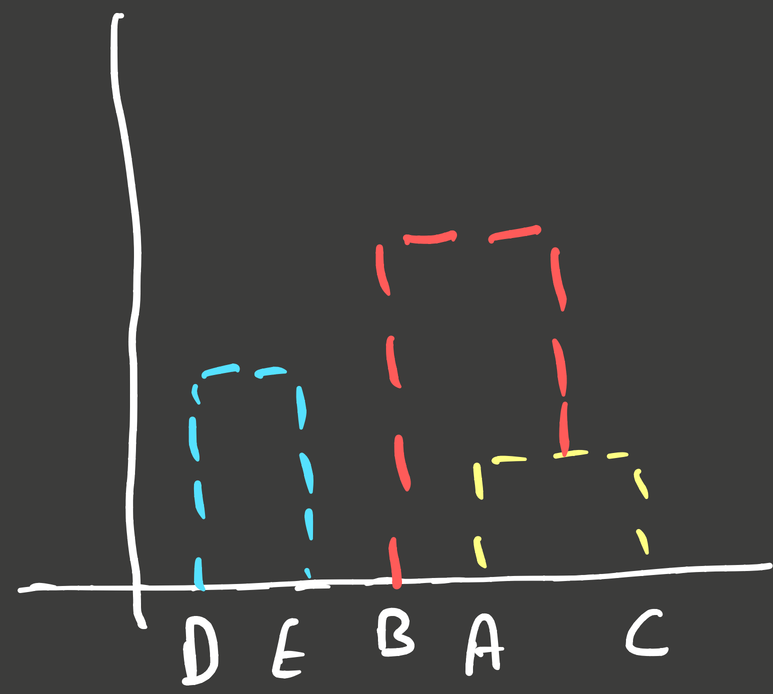


HIERARCHICAL CLUSTERING

* Start with each point in own cluster
    * Identify closest 2 clusters → Merge
    * Repeat

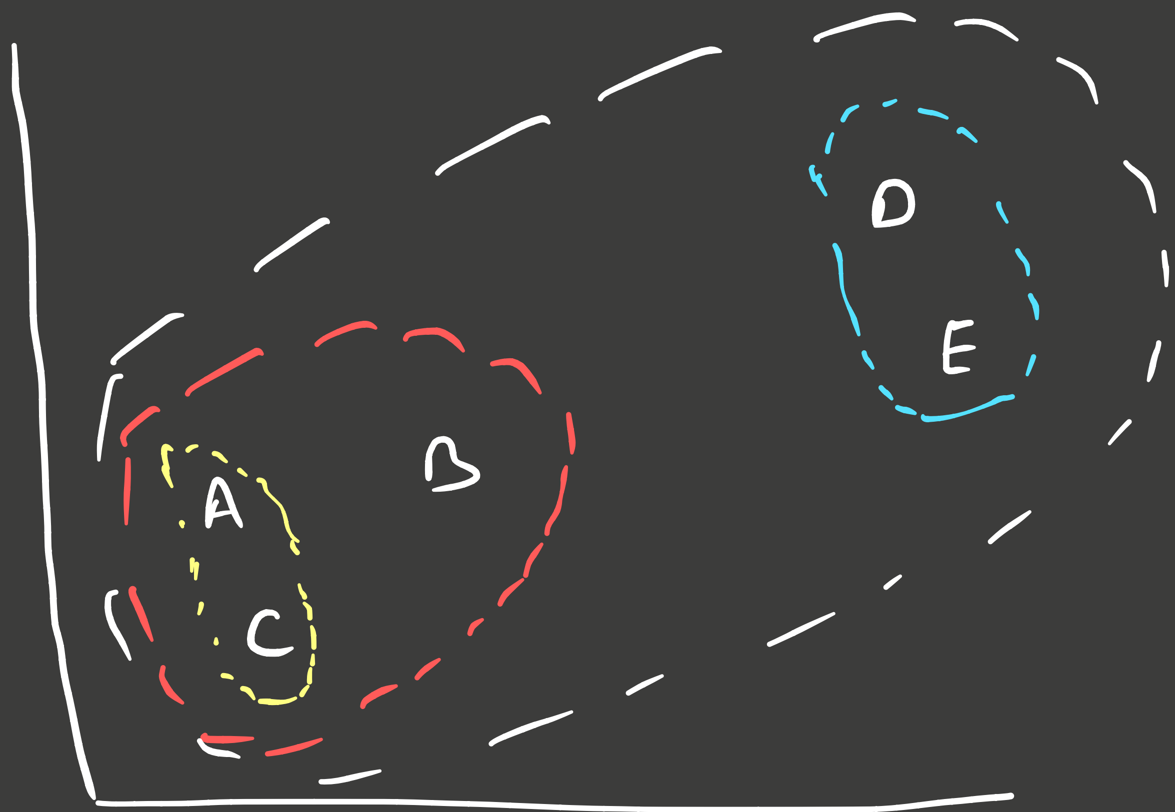


Dendogram

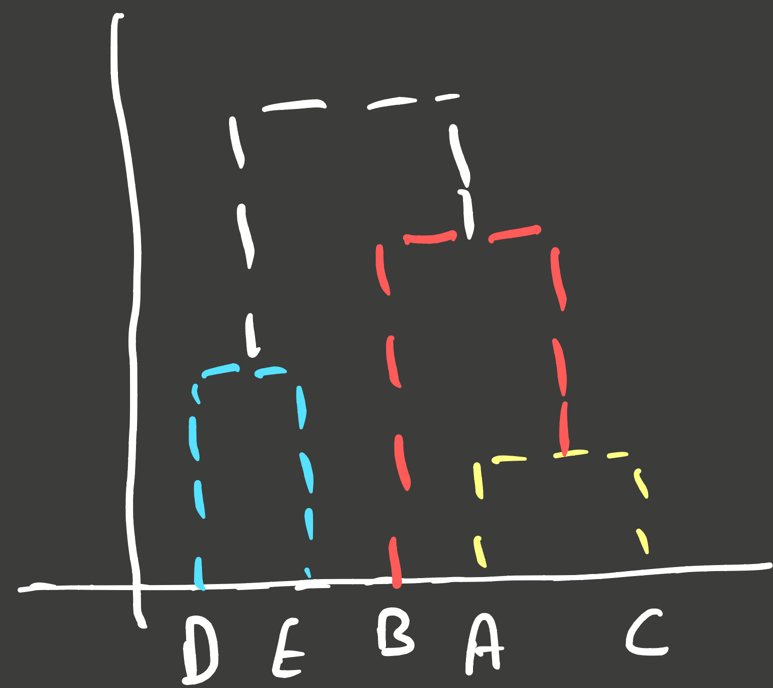


HIERARCHICAL CLUSTERING

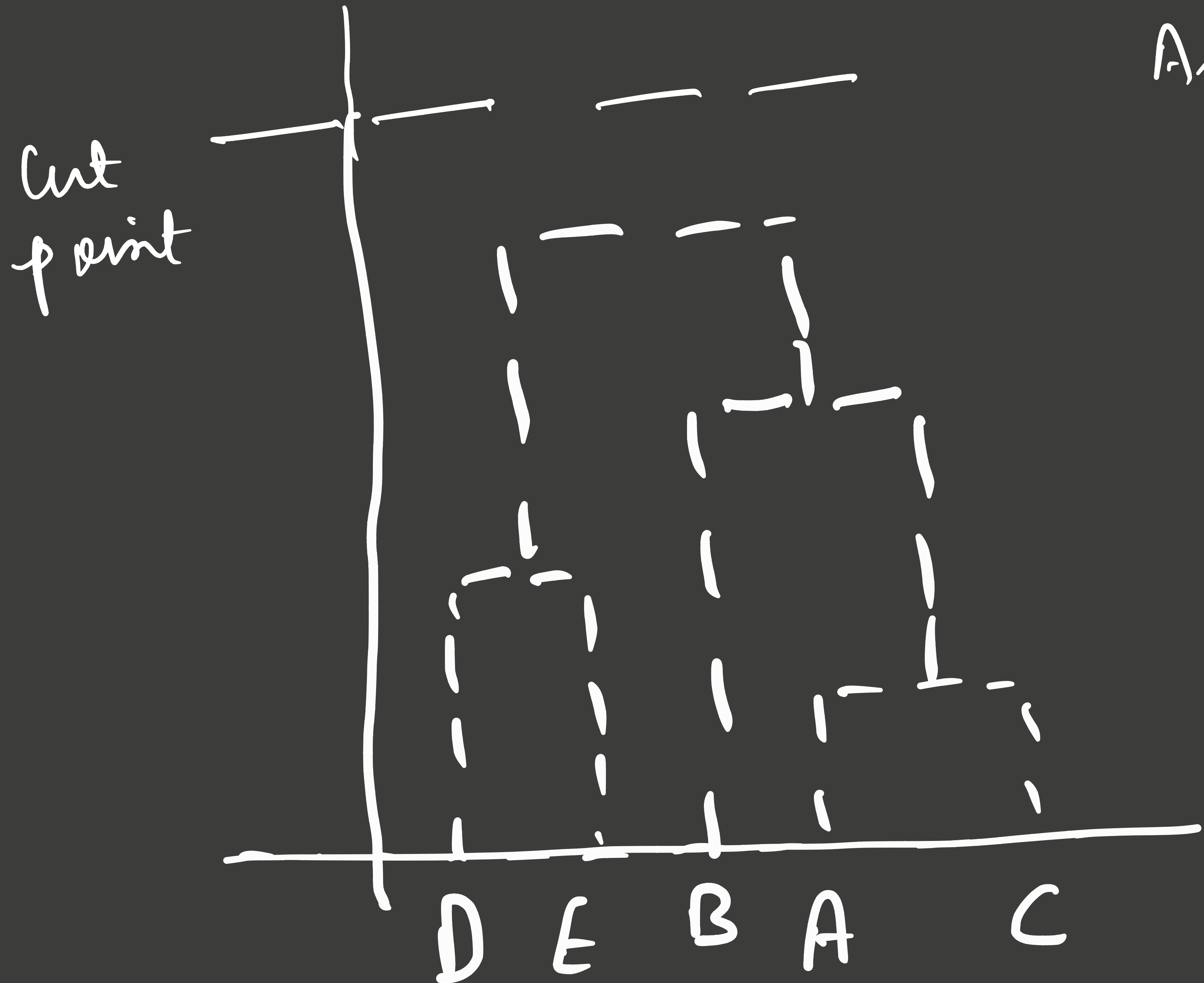
* Start with each point in own cluster
    * Identify closest 2 clusters → Merge
    * Repeat
* End when all points in single cluster



Dendogram

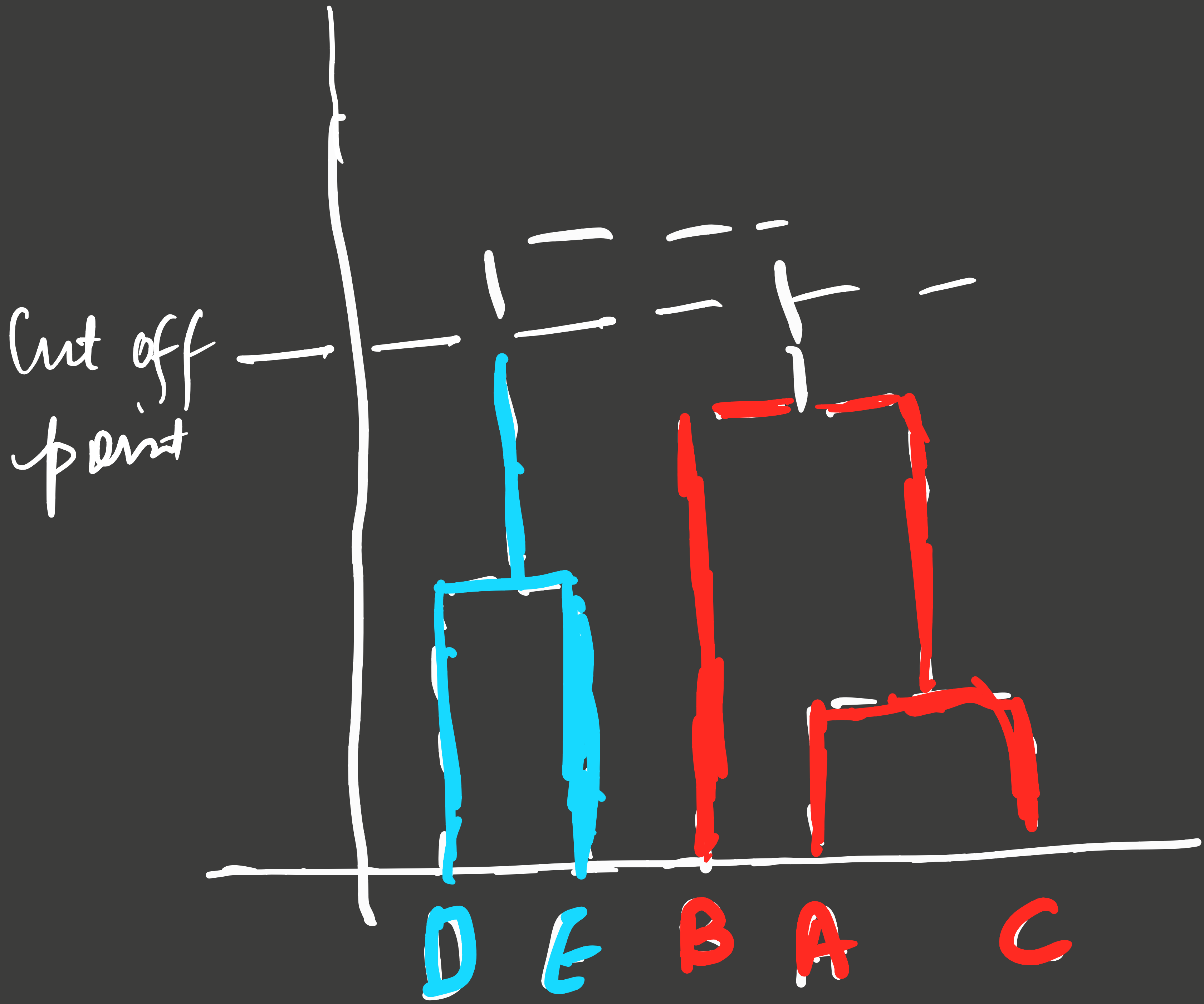


HIERARCHICAL CLUSTERING



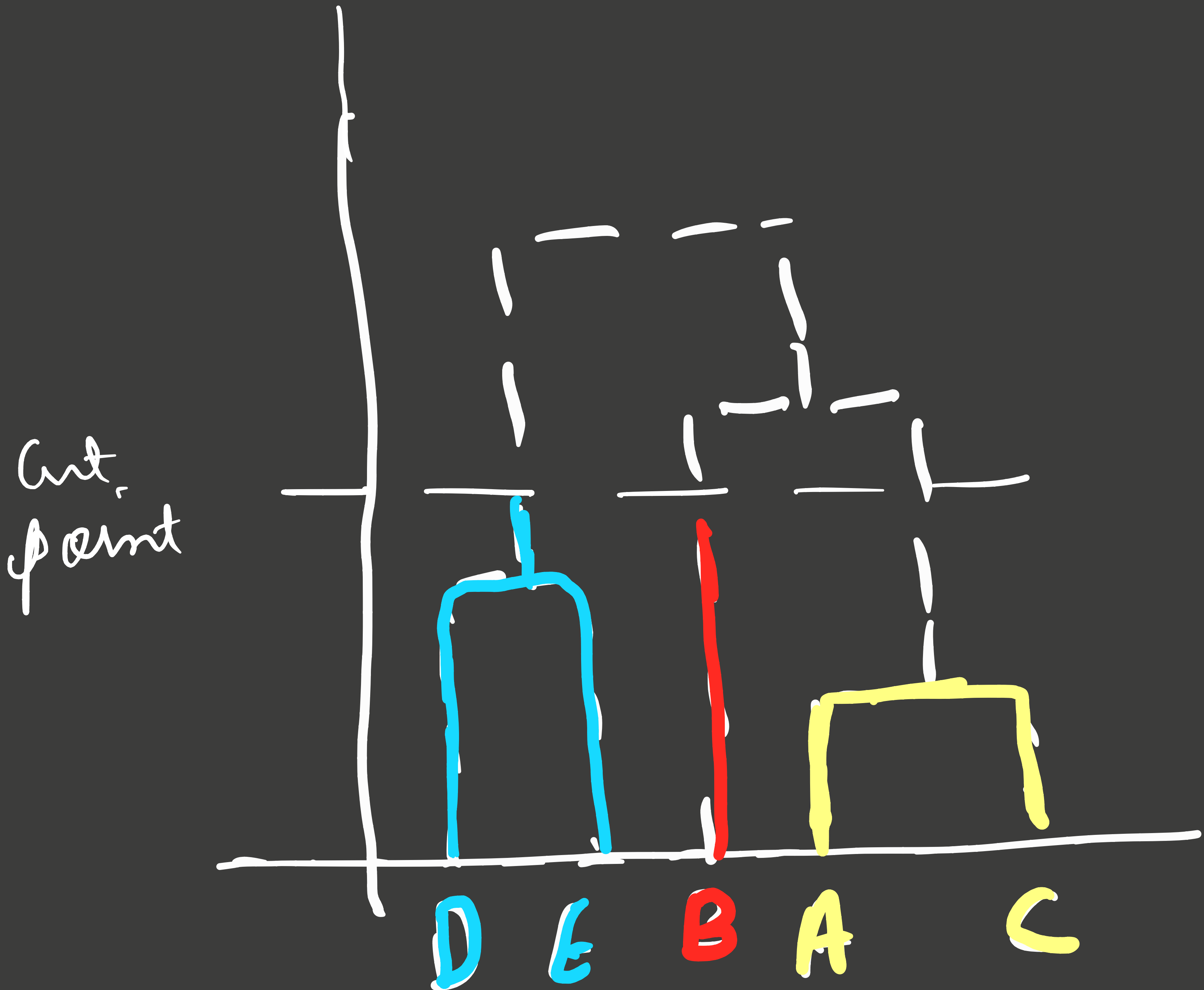
All points belong to same cluster

HIERARCHICAL CLUSTERING



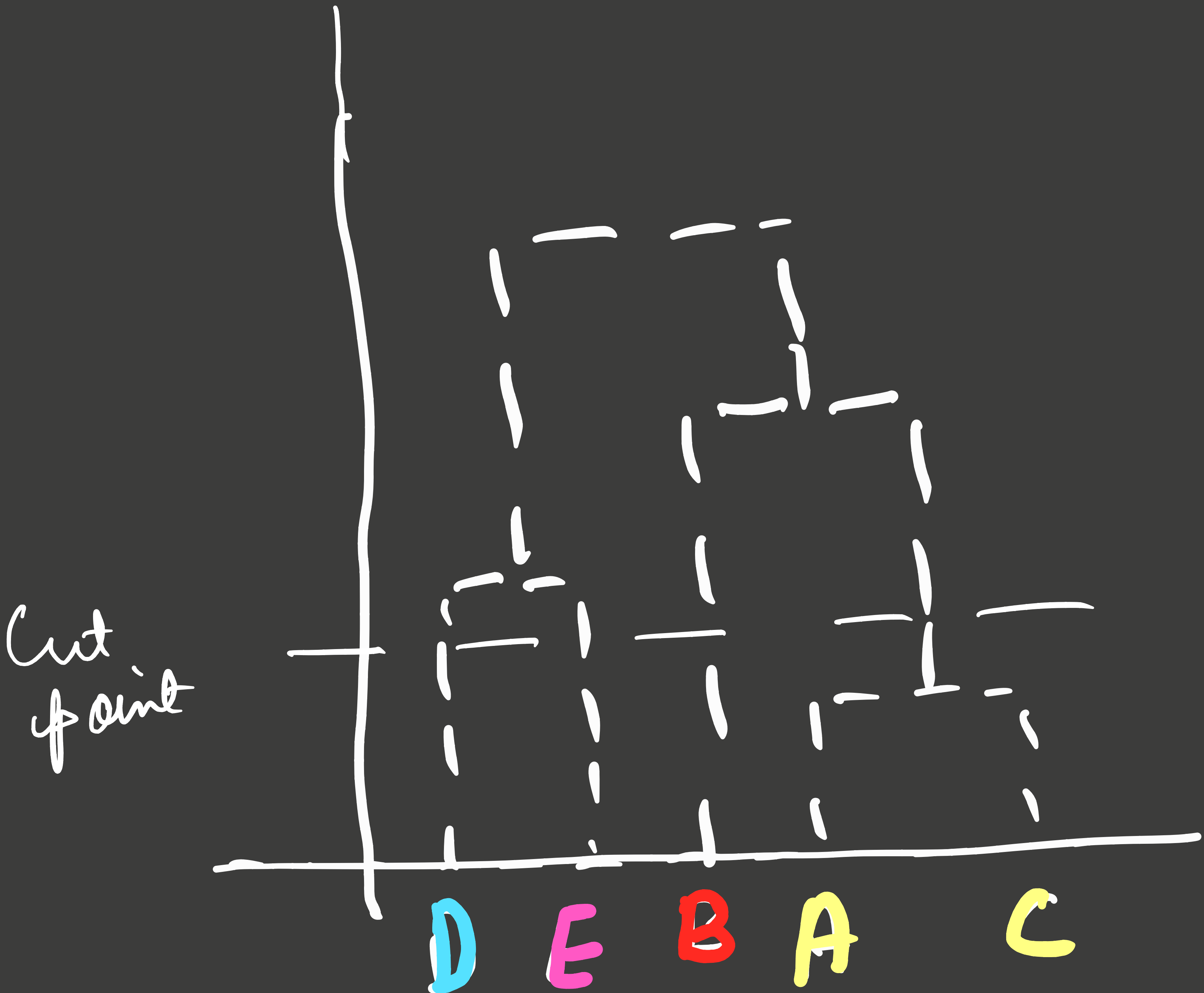
2 Clusters

HIERARCHICAL CLUSTERING



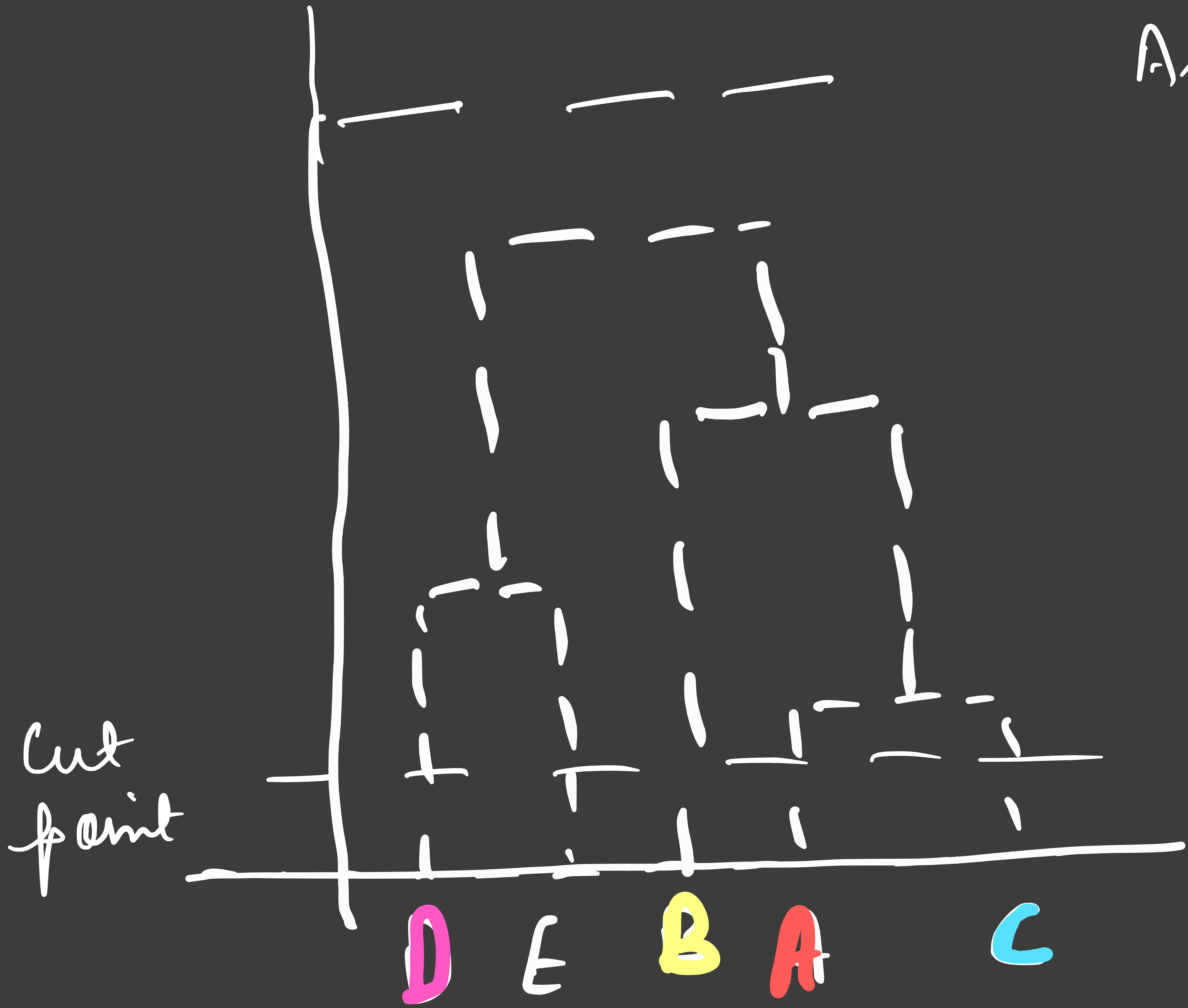
3 clusters

HIERARCHICAL CLUSTERING



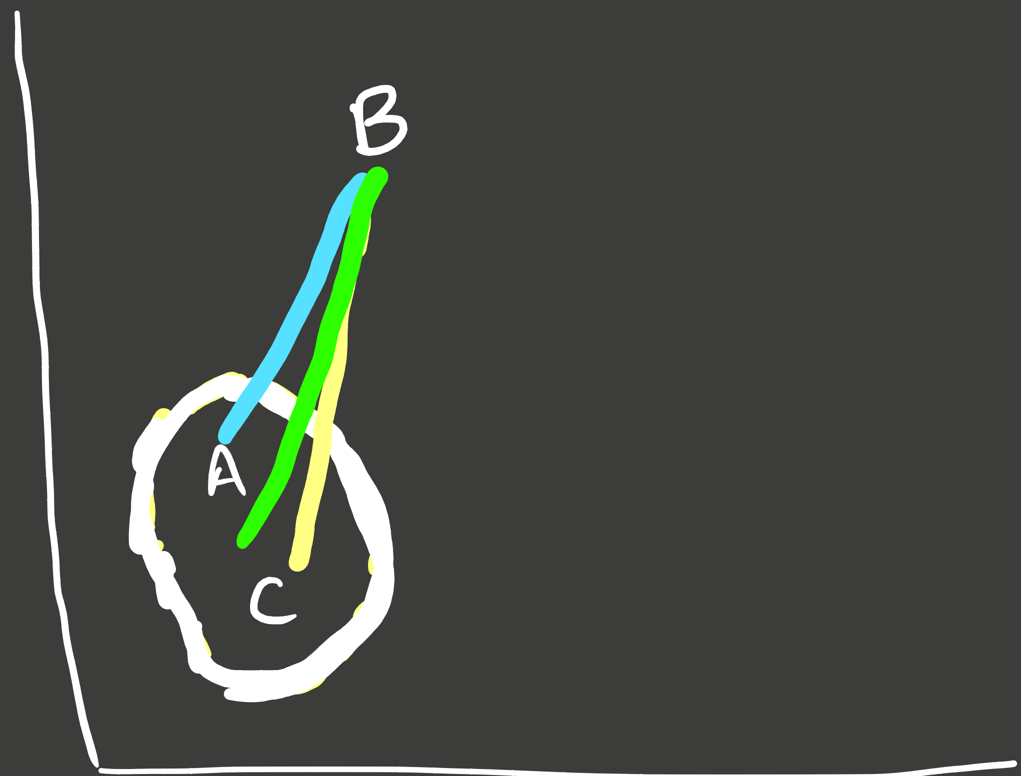
4 clusters

HIERARCHICAL CLUSTERING



All points belong to different clusters

JOINING CLUSTERS / LINKAGES



COMPLETE

MAX INTER CLUSTER DISSIMILARITY

SINGLE

MIN. INTER CLUSTER DISSIMILARITY

CENTROID

DISSIMILARITY B/W CLUSTER CENTROIDS