L22: K-Means Clustering

w/ Lloyd's Algorithm

Assignment-based Clustering

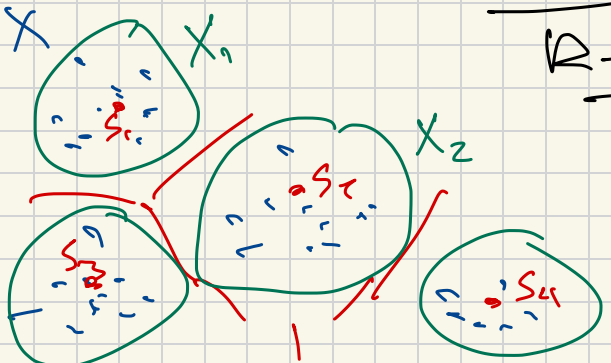
Input $X \subset \mathbb{R}^d$ distance $dl : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_{\geq 0}$

value $k > 1 \equiv$ # clusters $dl(x_1, x_2) = \|x_1 - x_2\|$

Goal: Set of k sites $S = \{s_1, s_2, \ldots s_k\} \subset \mathbb{R}^d$

$X = \bigcup_{j=1}^{k} X_j$ $X_j \subset X$ $X_j = \{x \in X \mid \phi_S(x) = s_j\}$

hard clustering $X_j \cap X_{j'} = \emptyset$ $j \neq j'$



k-means

$S^* = \underset{|S|=k}{\operatorname{argmin}} \sum_{i=1}^{n} \|x_i - \phi_S(x_i)\|^2 = \text{cost}(X, S)$

$\phi_S(x) = \underset{s_j \in S}{\operatorname{argmin}} \|x - s_j\|$

Lloyd's Algorithm

0: Initialize — Choosing k pts $S \subset X$

$S = \{s_1, s_2, \ldots s_k\}$

1: repeat

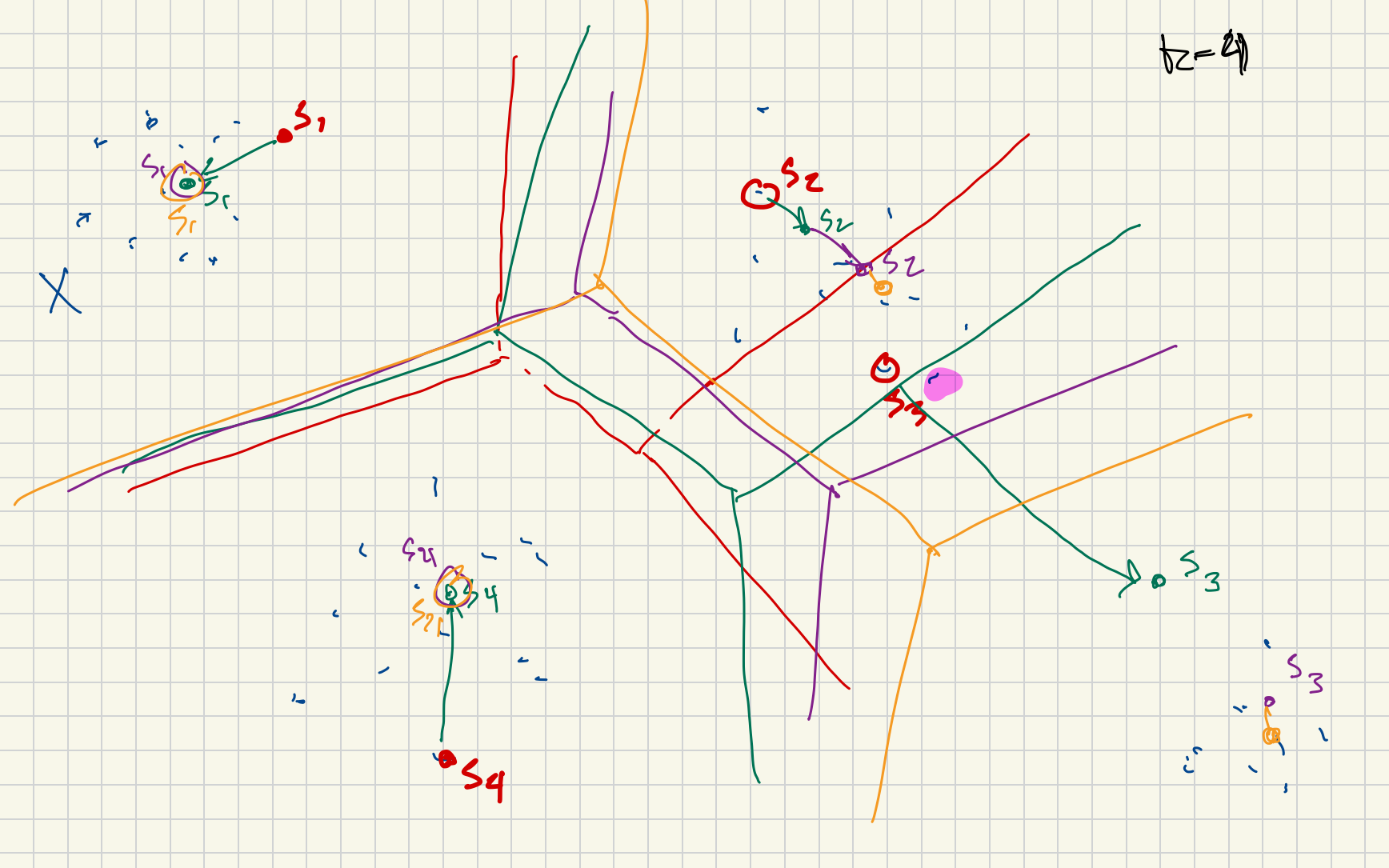
(a) $\forall x_i \in X$: assign x_i to X_j (ith subset of X "cluster" associated w/ s_j) so $\phi_S(x_i) = s_j$

(b) $\forall s_j \in S$: update $s_j = \frac{1}{|X_j|} \sum_{x \in X_j} x = \text{average}(X_j)$

until ("converged" or R steps)

R: 10 or 20

like "Expectation-Maximization" (EM)


s_1
s_2
s_3
s_4
s_3

Does Lloyd's Also Always Converge? Yes

$$\text{Cost}(X, S) = \sum_{x_i \in X} \|x_i - q_S(x_i)\|^2 \quad \text{step (a)}$$

$$= \sum_{s_j \in S} \left(\sum_{x \in X_j} \|s_j - x\|^2 \right) \quad \text{step (b)}$$

optimizing

* not increase

Claim each step (a) or (b) decreases Cost(X, S)

step (a): $\forall x \in X$ | assign to X_j so $q_S(x) = s_j$ closest site.

step (b): $\forall s_j \in S$ | $s_j = \frac{1}{|X_j|} \sum_{x \in X_j} x$ ← set to average

$s_j^* = \underset{s}{\text{argmin}} \sum_{x \in X_j} \|x - s\|^2$

Must be $S_t \to S_{t+1} \to S_{t+2}$

claim $S_t \neq S_{t+1}$ $\quad cost(X, S_t) > cost(X, S_{t+1})$

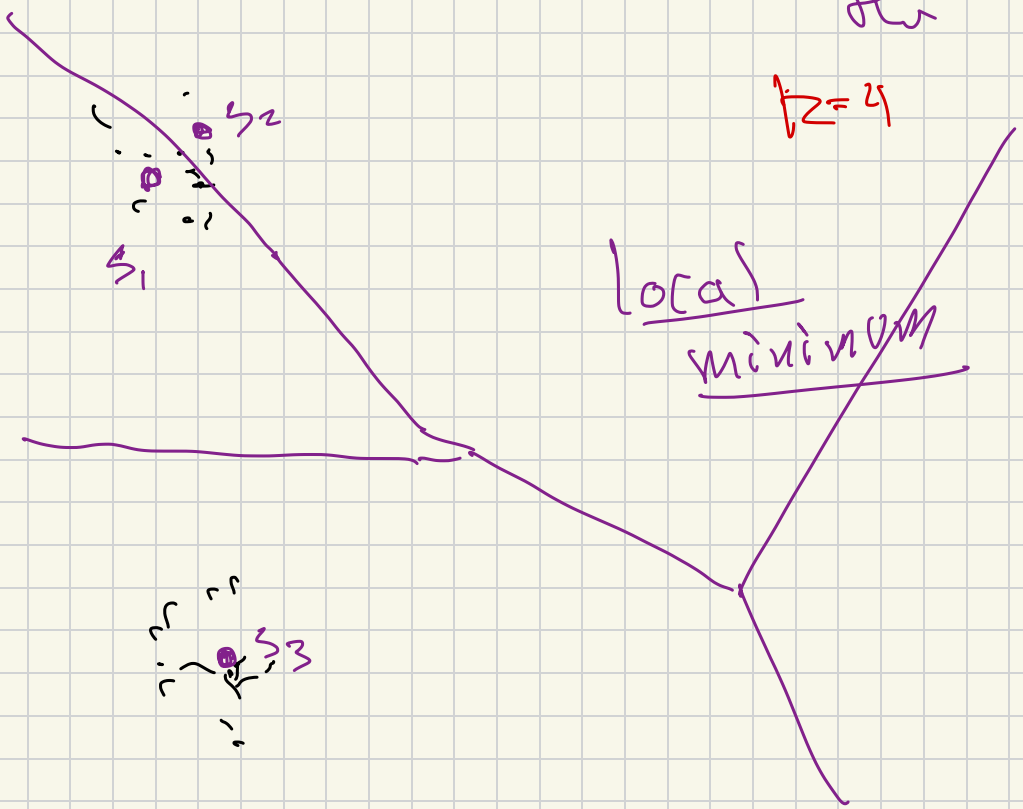
$cost(X, S_{t+1}) > cost(X, S_{t+2})$

$cost(X, S_t) > cost(X, S_{t+2})$

$\Rightarrow \quad S_t \neq S_{t+2}$

Lloyd's Algo

May not converge to the optimal solution!



$k=4$

local minimum

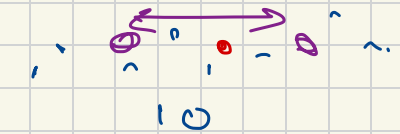
How to Address non-optimal convergence?

① Try more initializations of random

step 0: select $S_c X$ at random.

⇒ Initialize each S_j is empty. $X_j \subset X$ at random.

② Force sites not too close



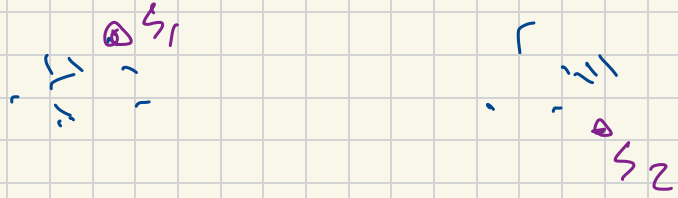
if stuck and points close → randomly reassign to the point.

sometimes $X_j = \emptyset$

(3) Better way to initialize

→ incrementally increase k.

always far from existing sites



→ Gonzalez Also greedy

→ k-means++ (randomized)

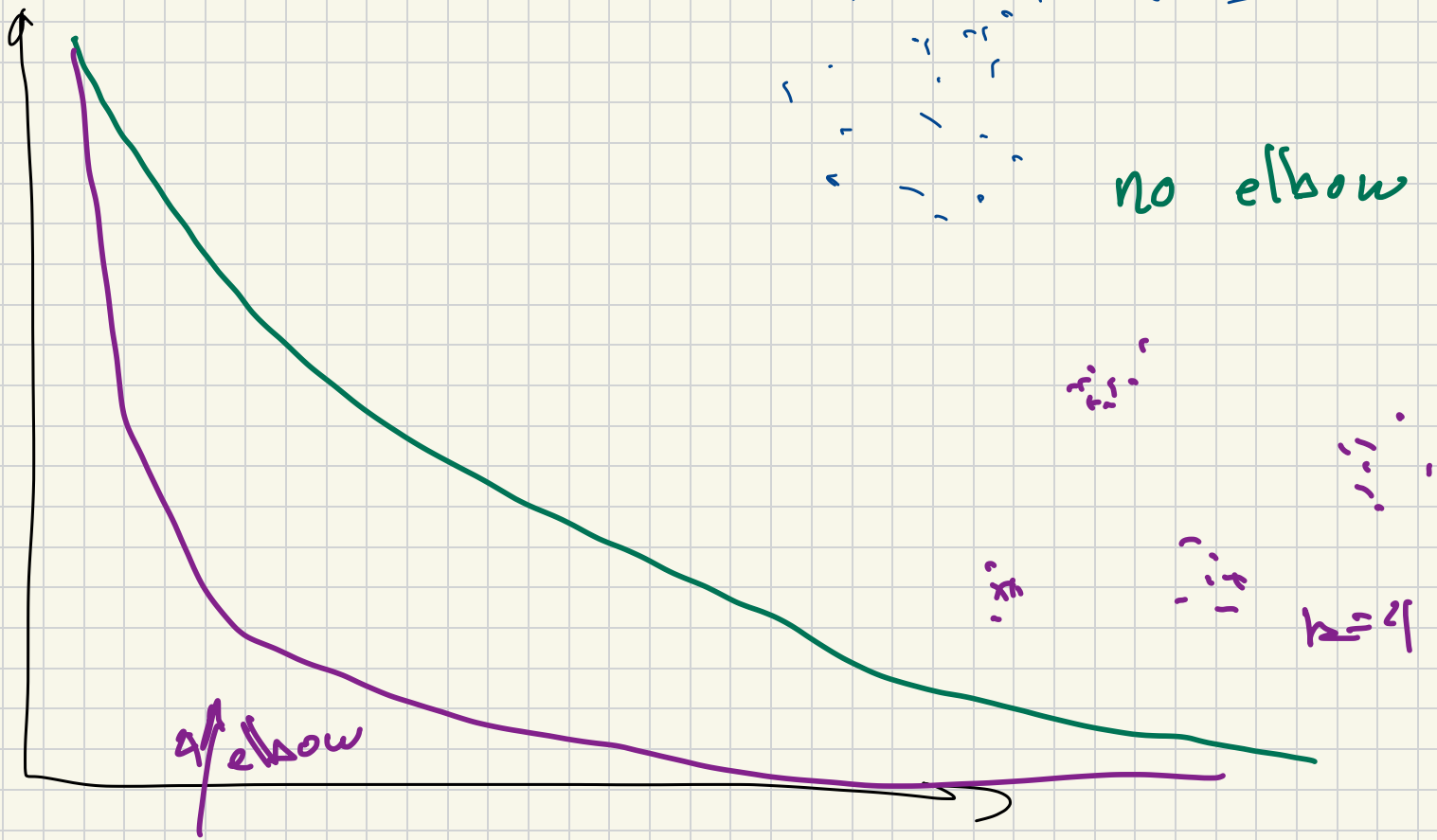
(4) Run w/ much larger $k' > k$

↳ then combine

How to decide k?

Goal $\text{cost}(X, S) = \sum_{x_i \in X} \|x_i - \phi_S(x_i)\|^2$

$\text{cost}(X, S^*, k)$ always decreases as k increases


blob
no elbow
k=4
4/elbow

How many clusters?