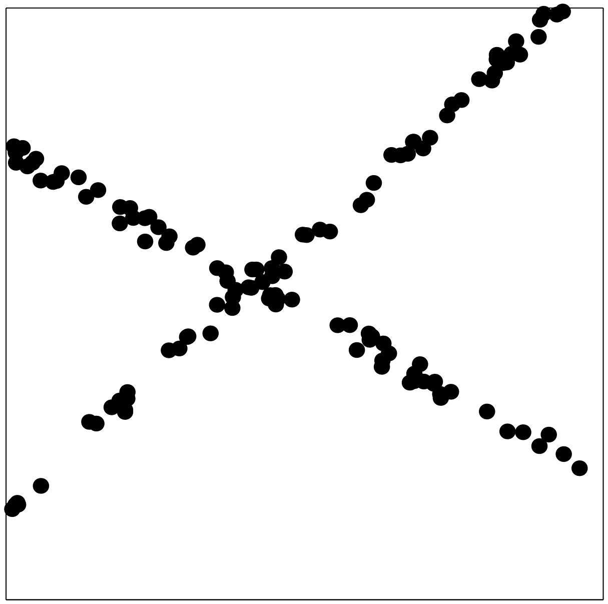
CSI 436/536
Introduction to Machine Learning

Multi-model LLSE and k-means clustering

Professor Siwei Lyu
Computer Science
University at Albany, State University of New York

Fitting multiple linear models

- In many practical problems, we need to deal with multiple models at the same time to separate cooccurring factors attributing to observed data
- fitting two lines to the data set
  - need to find parameters of two lines a_1, a_2, b_1, b_2
  - need to know the assignment of each point to line 1 or line 2
- knowing one solve the other is easy, but solving them at the same time is hard



Problem formulation & solution

- Overall objective function

$$\min_{\alpha_{ij}, w_j} \sum_{i=1}^{n} \sum_{j=1}^{K} \alpha_{ij}(y_i - w_j^T x_i)^2, \text{ s.t. } \alpha_{ij} \in [0,1], \sum_{j=1}^{K} \alpha_{ij} = 1$$

- Solution is a special case of the expectation-maximization (EM) algorithm
  - Starting with initial values of $\alpha_{ij} = \dfrac{1}{K}$,
  - Iterate until convergence
    - Update w_j, for j = 1, …, K (E-step: fitting data to each line)
    - Update α_{ij} (M-step: figuring out the model of each data example belongs)

E-step

- Solving sub-problem with regards to each w_j

  $\min_{w_j} \sum_{i=1}^{n} \alpha_{ij}(y_i - w_j^T x_i)^2$ — this is a weighted LLSE

- Diagonal weight matrix W with $W_{ii} = \alpha_{ij} \geq 0$, and to solve

$$\min_w (y - X^T w)^T W (y - X^T w)$$

- Solution

  $\nabla_w (y - X^T w)^T W (y - X^T w) = 2(XWX^T w - XWy) = 0$

  so $XWX^T w = XWy \Rightarrow w = (XWX^T)^{-1} XWy$

M-step

- Solving sub-problem with regards to each α_{ij}

$$\min_{\alpha_{ij}} \sum_{j=1}^{K} \alpha_{ij}(y_i - w_j^T x_i)^2, \text{ s.t. } \alpha_{ij} \in [0,1], \sum_{j=1}^{K} \alpha_{ij} = 1$$

  - Define $L_{ij} = (y_i - w_j^T x_i)^2$, this reduces to a linear programming (LP) problem for each i, as

$$\min_{\alpha_{ij}} \sum_{j=1}^{K} \alpha_{ij} L_{ij}, \text{ s.t. } \alpha_{ij} \in [0,1], \sum_{j=1}^{K} \alpha_{ij} = 1$$

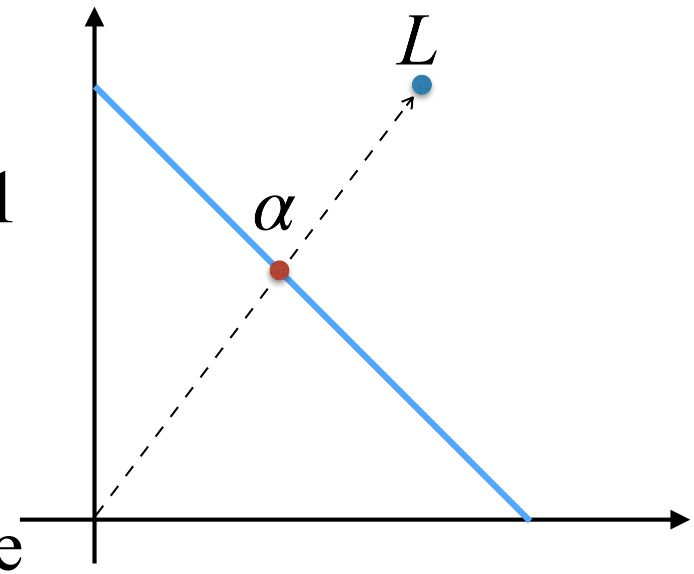
    - We can solve this using LP solver, but this problem has a simple solution

M-step

- Solving the M-step problem

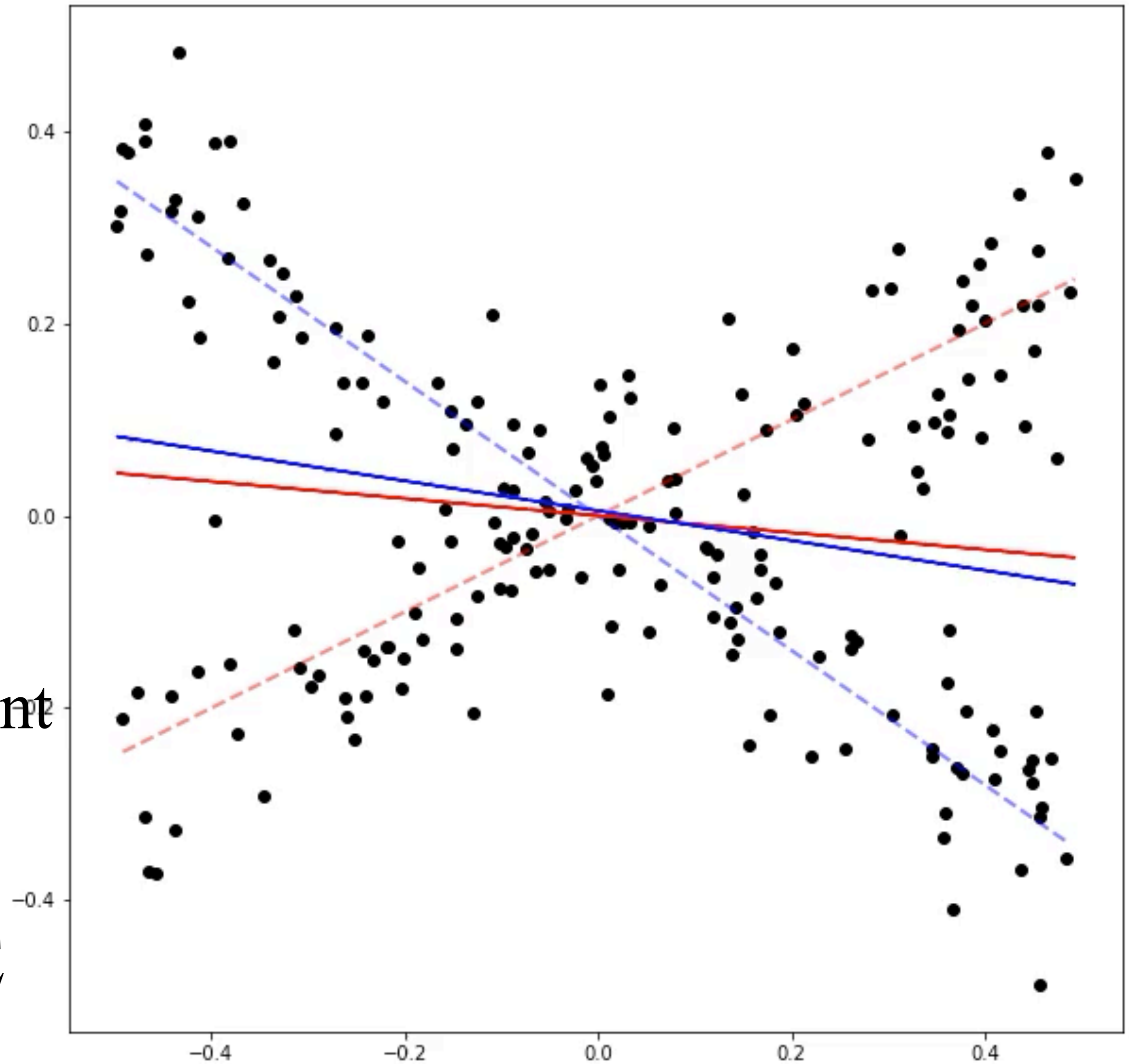
  $$\min_{\alpha_{ij}} \sum_{j=1}^{K} \alpha_{ij} L_{ij}, \text{ s.t. } \alpha_{ij} \in [0,1], \sum_{j=1}^{K} \alpha_{ij} = 1$$

  - The constraint set $\alpha_{ij} \in [0,1], \sum_{j=1}^{K} \alpha_{ij} = 1$ is known as the probability simplex
  - L is a vector in the positive orthant, so optimal solution is on the probability simplex
  - Optimal solution can be obtained by Cauchy-Shwartz inequality $\alpha_{ij} = \frac{L_{ij}}{\sum_{k=1}^{K} L_{ik}}$, which satisfies the constraint



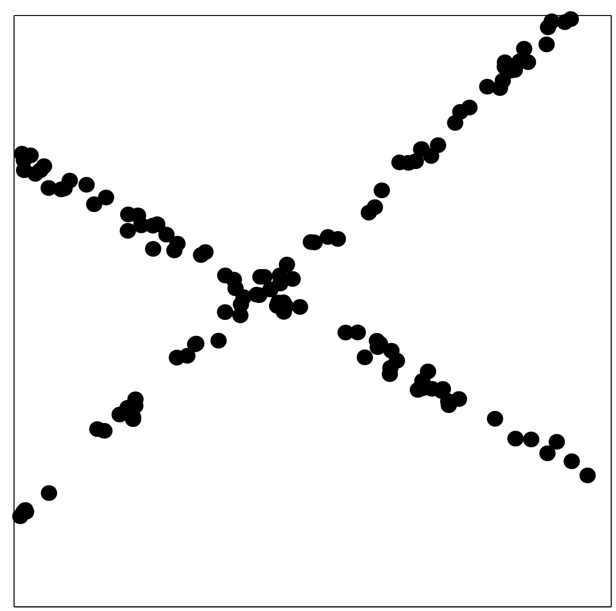
EM-LLSE algorithm

- Stands for expectation-maximization
- Set initial values of the two linear models
  - Iterate until convergence occurs
    - For each model to be considered
      - Compute errors of each data point
      - Update model parameter with Weighted LLSE algorithm



clustering problem

- multi-modal linear LSE is a soft clustering problem
  - Membership to each model/cluster is in the form of a "soft" weight
- We can convert the problem to a hard clustering setting
  - Membership is a "yes/no" binary question
  - One data point can only belong to one cluster



Hard multi-modal LLSE

- Overall objective function
  $$\min_{\alpha_{ij}, w_j} \sum_{i=1}^{n} \sum_{j=1}^{K} \alpha_{ij}(y_i - w_j^T x_i)^2, \text{ s.t. } \alpha_{ij} \in \{0,1\}, \sum_{j=1}^{K} \alpha_{ij} = 1$$
  - Note the difference is that the constraint changes from a continue interval [0,1] to a discrete binary set {0,1}
  - The summation constraint determines that each data point is assigned to exactly one model
- solving this problem directly is NP-hard because It involves enumerating all possible clusters
- We can solve it with the EM algorithm for an approximate solution

EM algorithm for hard multi-modal LLSE

- Solving sub-problem with regards to each w_j

$$\min_{w_j} \sum_{i=1}^{n} \alpha_{ij}(y_i - w_j^T x_i)^2 = \min_{w_j} \sum_{i=1}^{n} 1_{(x_i,y_i)\sim model_j}(y_i - w_j^T x_i)^2$$

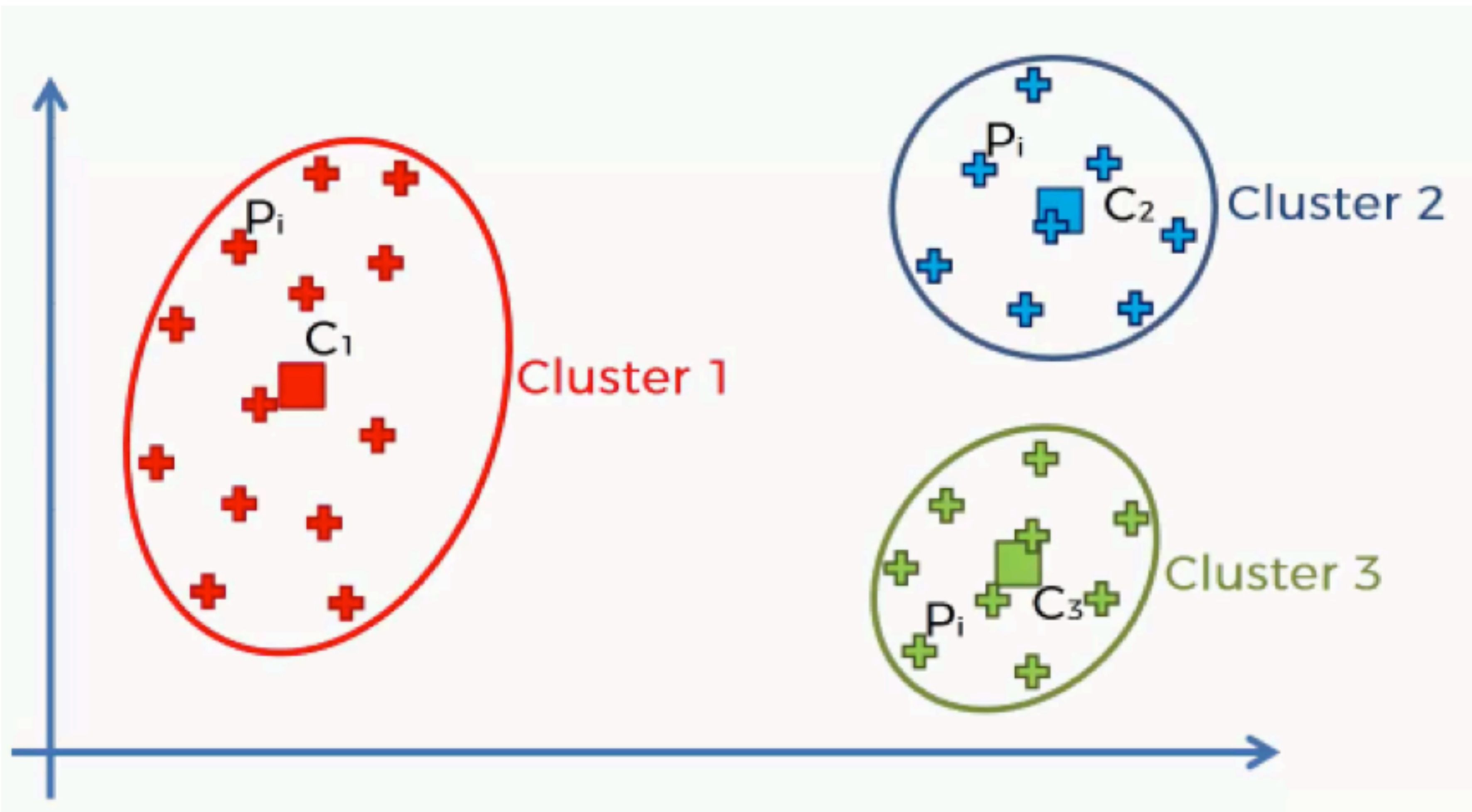
- Solving the M-step problem

$$\min_{\alpha_{ij}} \sum_{j=1}^{K} \alpha_{ij} L_{ij}, \text{ s.t. } \alpha_{ij} \in \{0,1\}, \sum_{j=1}^{K} \alpha_{ij} = 1$$

  - Solution is given by $\alpha_{ij} = 1_{j=\arg\min_k L_{ik}}$
  - Intuitively, data point i belongs to the model that gives the minimum error

Clustering algorithm

- We can extend the multi-modal LLSE algorithm to clustering algorithms



k-means algorithm

- One of the most popular and simplest clustering algorithms, also known as Lloyd algorithm, or vector quantization

- n d-dim data in matrix X
- $C \geq 2$ clusters
- C_k, $k = 1,\ldots,C$ cluster index set
  - $i \in C_k$, then x_i is in the k^{th} cluster
- cluster membership indicator

$$c_{ik} = \begin{cases} 1 & i \in C_k \\ 0 & i \notin C_k \end{cases}$$

- no dual cluster membership
- cluster representatives μ_k

K-means algorithm

- Iterates between two steps
  - Cluster assignment: update cluster membership indicator
  - Cluster representation: refine representative of each cluster
  - optimality: minimize representation error

$$\sum_{i=1}^{n}\sum_{k=1}^{C} c_{ik} d(\vec{x}_i, \vec{\mu}_k)$$

  - d is a distance metric, usually, assume euclidean norm

$$\sum_{i=1}^{n}\sum_{k=1}^{C} c_{ik} \|\vec{x}_i - \vec{\mu}_k\|^2$$

Optimizing k-means

- solving this problem directly is NP-hard because It involves enumerating all possible clusters

$$\min_{c_{ik},\vec{\mu}_k} \sum_{i=1}^{n} \sum_{k=1}^{C} c_{ik} \|\vec{x}_i - \vec{\mu}_k\|^2$$

- solving by coordinate descent
    - starting from initial guesses of μk,
    - repeat until convergence
        - fixing μk, find optimal cik - regrouping
        - fixing cik, update μk - reassigning
- This guarantees to converge [why?]

derivation

regrouping

M step

$$\min_{c_{ik}} \sum_{k=1}^{C} \sum_{i=1}^{m} c_{ik} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|_2^2$$

$$c_{ik} = \begin{cases} 1 & k = \text{argmin}_j \|\mathbf{x}_i - \boldsymbol{\mu}_j\|_2^2 \\ 0 & k \neq \text{argmin}_j \|\mathbf{x}_i - \boldsymbol{\mu}_j\|_2^2 \end{cases}$$

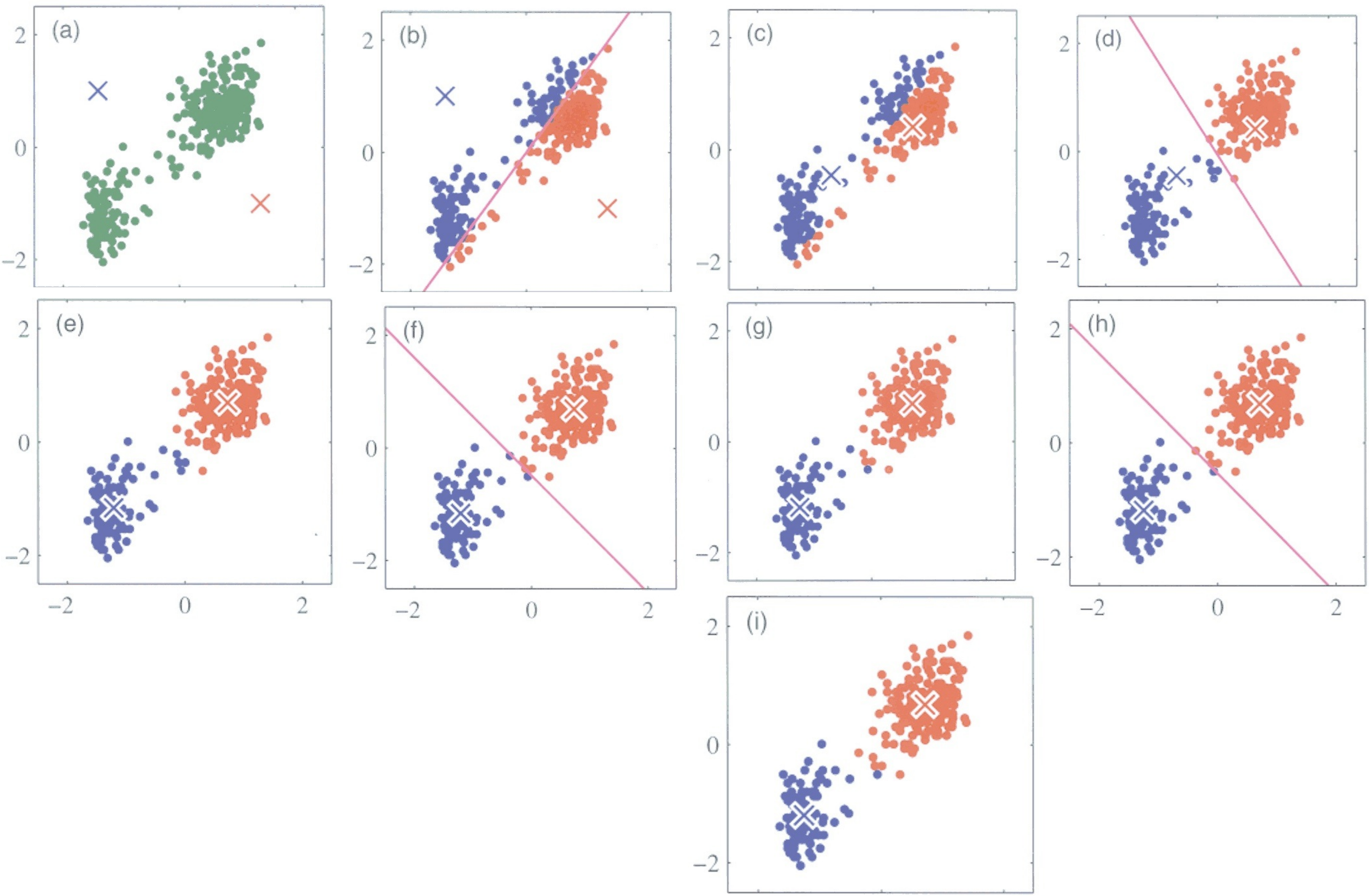
There is also a soft version of the k-means clustering

reassigning

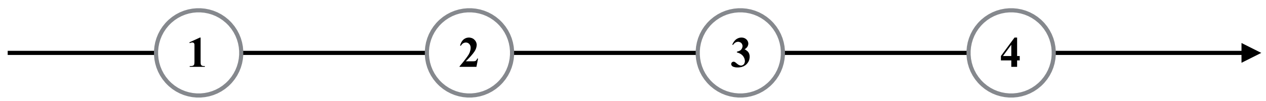
E step

$$\begin{aligned} & \frac{\partial}{\partial \boldsymbol{\mu}_k} \sum_{i=1}^{m} \sum_{k=1}^{C} c_{ik} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|_2^2 \\ = \quad & \frac{\partial}{\partial \boldsymbol{\mu}_k} \sum_{\mathbf{x}_i \in \mathcal{C}_k} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|_2^2 \\ = \quad & 2 \sum_{\mathbf{x}_i \in \mathcal{C}_k} (\mathbf{x}_i - \boldsymbol{\mu}_k) = 2 \sum_{\mathbf{x}_i \in \mathcal{C}_k} \mathbf{x}_i - 2|\mathcal{C}_k| \boldsymbol{\mu}_k = 0 \\ \Rightarrow \quad & \boldsymbol{\mu}_k = \frac{1}{|\mathcal{C}_k|} \sum_{\mathbf{x}_i \in \mathcal{C}_k} \mathbf{x}_i \end{aligned}$$

Example



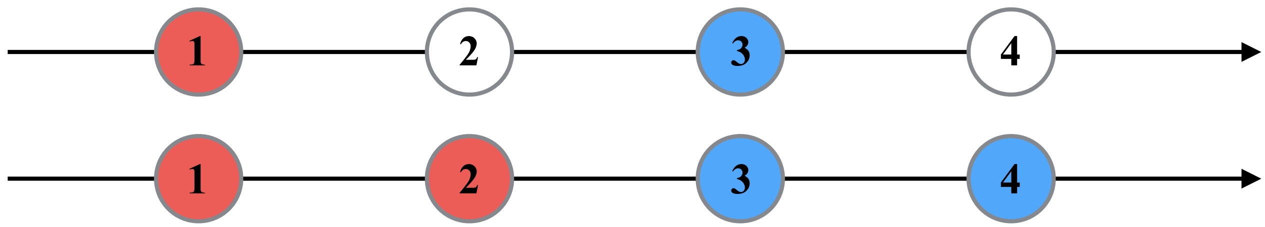
Problem



- Consider the situation above where we need to cluster 4 data points into two clusters: to break ties, we take the left closest center in membership reassignment
    - Initial cluster centers as {1,3}
    - Initial cluster centers as {2,4}

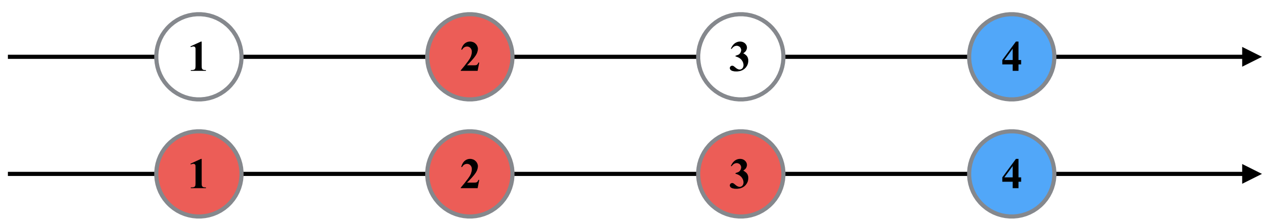
No guarantee for local minimum

- Initial cluster centers as {1,3}



  - Objective = $(1/2)^2 + (1/2)^2 + (1/2)^2 + (1/2)^2 = 1$

- Initial cluster centers as {2,4}



  - Objective = $1 + 0 + 1 + 0 = 2$

- conclusion: k-means algorithm may not converge to a local minimum of the objective function