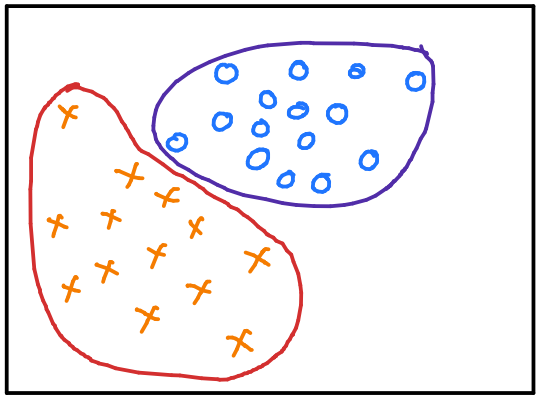
k-means Clustering



- Clustering is an unsupervised ML algorithm
- Idea in clustering
  - Samples within a cluster are similar to each other
  - Samples in different clusters are dissimilar
- We have learned about clustering with GMM using EM algorithm
  - GMM models the cluster probabilistically (soft assignments)

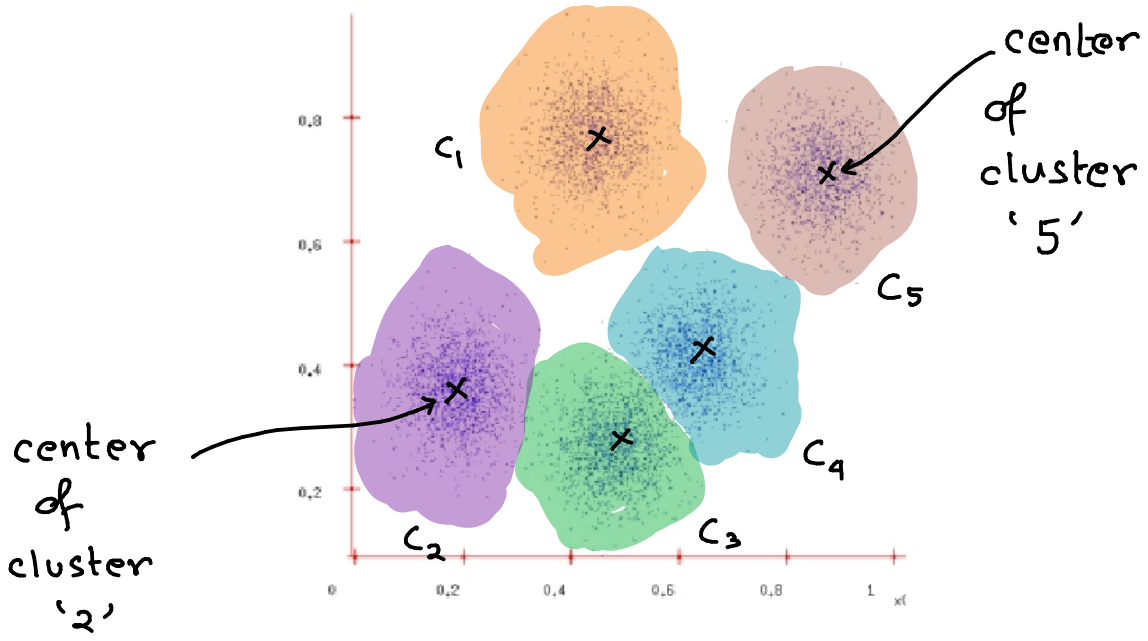
    i.e. $p(\underline{x}_i \mid y = k) = \pi_k \, \mathcal{N}(\underline{x}_i \mid \underline{\mu}_k, \underline{\underline{\Sigma}}_k)$

    probability of data point $\underline{x}_i$ belonging to the 'k' th cluster
- In this lecture, we introduce the k-means clustering algorithm
  - Unlike GMM, in k-means, we do 'hard' cluster assignments and there is no probabilistic model

Intuition of k-means

- k-means assumes that there are 'K' clusters, and each point is close to its cluster center or mean (the average of points in the cluster)
  - If we knew the cluster assignment, we could easily compute the centers
  - If we knew the centers, we could easily compute which points belong to which cluster
  - Chicken and egg problem!

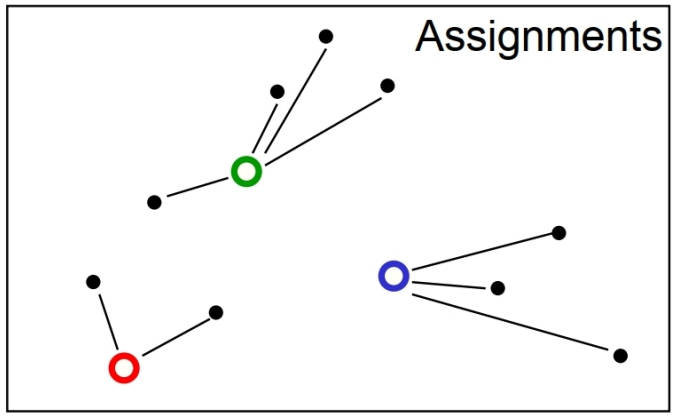
- Heuristically speaking, one could start randomly and alternate between the two!

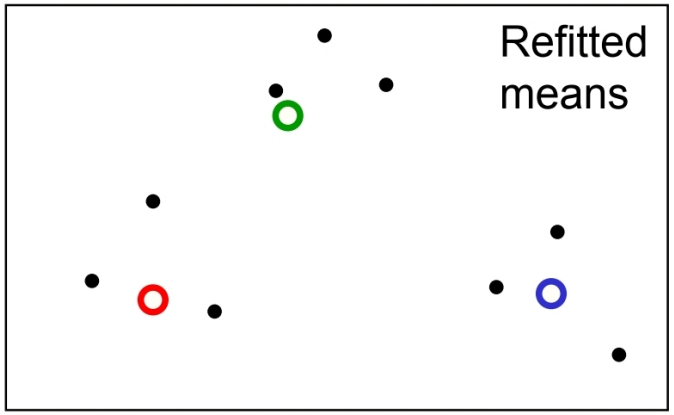


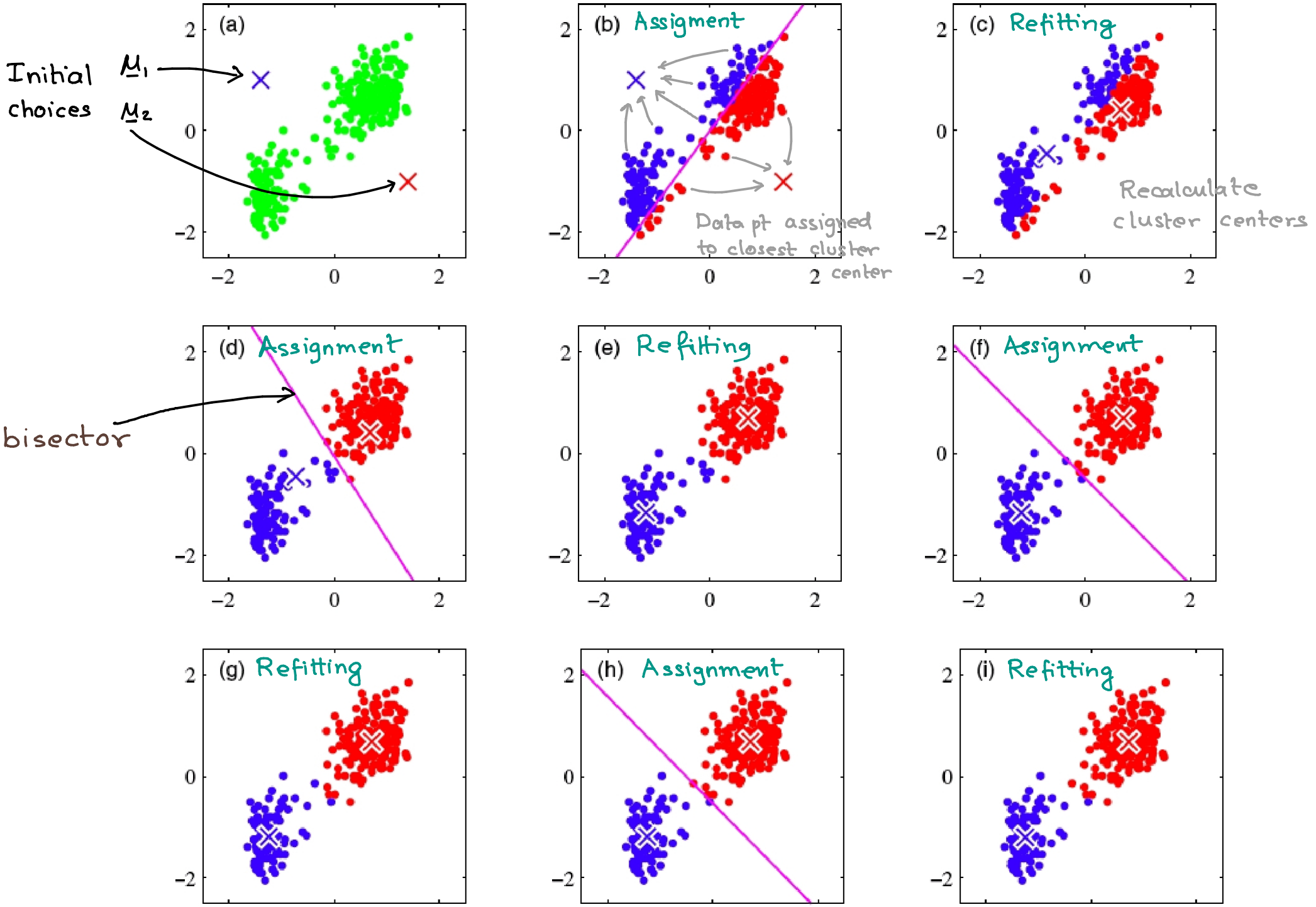
K-means

- Initialization: Randomly initialize cluster centers (or means)

- The algorithm iteratively alternates between two steps:
  - Assignment step: Assign each data point to the closest cluster
  - Refitting step: Move each cluster center to the center of gravity of the data assigned to it






Initial choices
$\underline{\mu}_1$
$\underline{\mu}_2$
(a)
(b) Assigment
Data pt assigned to closest cluster center
(c) Refitting
Recalculate cluster centers
bisector
(d) Assignment
(e) Refitting
(f) Assignment
(g) Refitting
(h) Assignment
(i) Refitting

K-means Objective

What is actually being optimized?

k-means clustering amounts to selecting the 'k' clusters such that the distances of the points to the cluster centers, summed over all data points, is minimized:

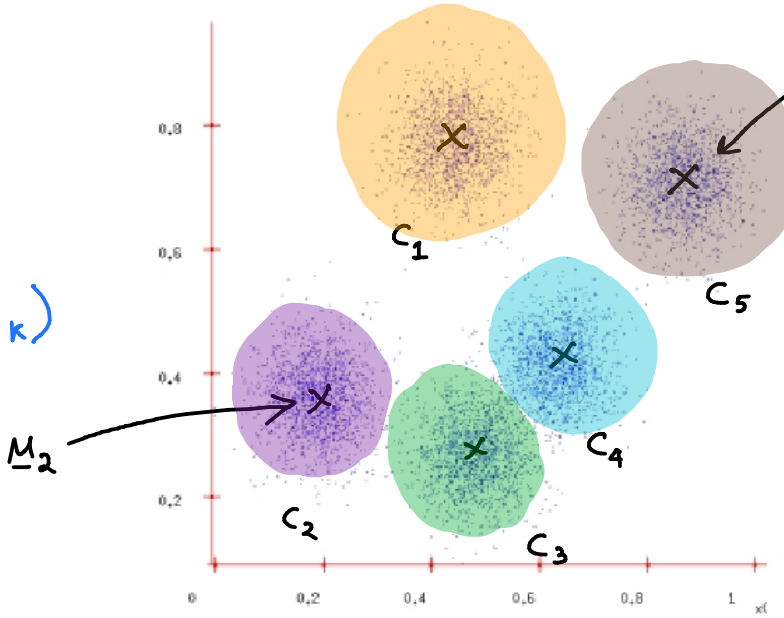
$$\{\hat{r}_{ik}, \hat{\underline{\mu}}_k\} = \underset{\{r_{ik}, \underline{\mu}_k\}}{\arg\min} \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \, \| \underline{x}_i - \underline{\mu}_k \|_2^2$$

$\underline{\mu}_k$: center of cluster k

mean of all data pts $\underline{x}_i \in C_k$

$r_{ik} \in \{0, 1\} \quad \forall \, i, k$

$r_{ik} = 1 \Rightarrow \underline{x}_i$ belongs to cluster k (with center $\underline{\mu}_k$)



$$\underline{\mu}_5 = \frac{1}{|C_5|} \sum_{i=1}^{N} r_{i5} \, \underline{x}_i$$

$|C_5|$: # of data pts that belong to cluster '5'

How to optimize?

Optimization problem:

$$\{\hat{r}_{ik}, \hat{\underline{\mu}}_k\} = \arg\min_{\{r_{ik}, \underline{\mu}_k\}} \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \| \underline{x}_i - \underline{\mu}_k \|_2^2$$

- This is a combinatorial optimization which is NP-hard to solve
- An alternating minimization strategy is used to solve the optimization:
  - If we fix the center $\{\underline{\mu}_k\}$, then we can easily find the optimal assignments r_{ik} for each sample $\underline{x}_i$

$$\{\hat{r}_{ik}\} = \arg\min_{\{r_{ik}\}} \sum_{k=1}^{K} r_{ik} \| \underline{x}_i - \underline{\mu}_k \|_2^2$$

That is, assign each point to the cluster with the nearest center

e.g. if $\underline{x}_i$ is assigned to cluster k

$$r_{i1} = 0, \quad r_{i2} = 0, \ldots, \quad r_{ik} = 1, \ldots, \quad r_{iK} = 0$$

How to optimize?

Optimization problem:

$$\min \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \, \| \underline{x}_i - \underline{\mu}_k \|_2^2$$

- An alternating minimization strategy is used to solve the optimization:
  - Similarly, if we fix the assignments r_{ik}, then we can easily find optimal centers $\underline{\mu}_k$

$$\frac{\partial}{\partial \underline{\mu}_l} \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \, \| \underline{x}_i - \underline{\mu}_k \|_2^2 = 0$$

$$\Rightarrow \quad 2 \sum_{i=1}^{N} r_{il} \, (\underline{x}_i - \underline{\mu}_l) = 0$$

$$\Rightarrow \quad \boxed{\hat{\underline{\mu}}_l = \frac{\sum_{i=1}^{N} r_{il} \, \underline{x}_i}{\sum_{i=1}^{N} r_{il}}}$$

K-means algorithm (also called Lloyd's algorithm)

Data: $\{\underline{x}_i\}_{i=1}^N$, number of cluster K

Procedure:

- Initialization: Set K cluster means $\underline{\mu}_1, \ldots, \underline{\mu}_K$ to random values
- Repeat until convergence (until assignments do not change)
  - Assignment: Each data point $\underline{x}_i$ is assigned to nearest center

$$k^{(i)} = \arg\min_j \| \underline{x}_i - \underline{\mu}_j \|$$

and the responsibilities

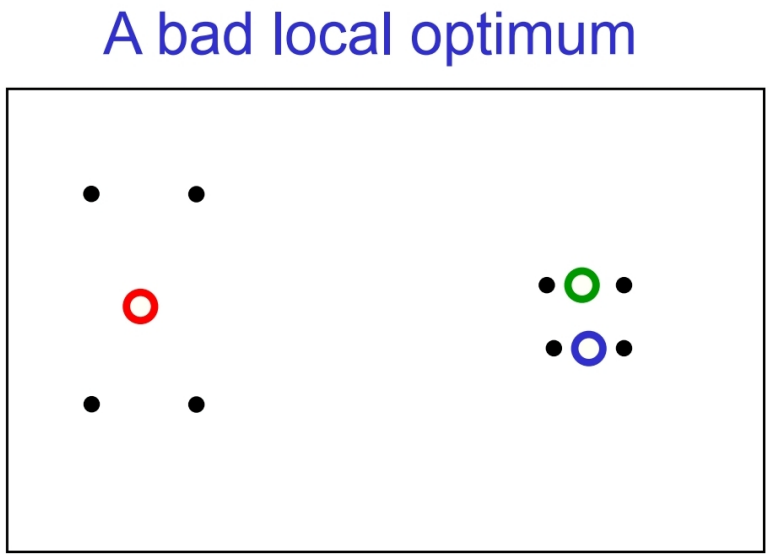
$$r_{ik} = \mathbb{I}\left[k^{(i)} = k\right] \quad \text{for } k = 1, \ldots, K$$

  - Refitting: Each center is set to mean of data assigned to it

$$\underline{\mu}_k = \frac{\sum_i r_{ik}\, \underline{x}_i}{\sum_i r_{ik}}$$

Convergence of k-means algorithm

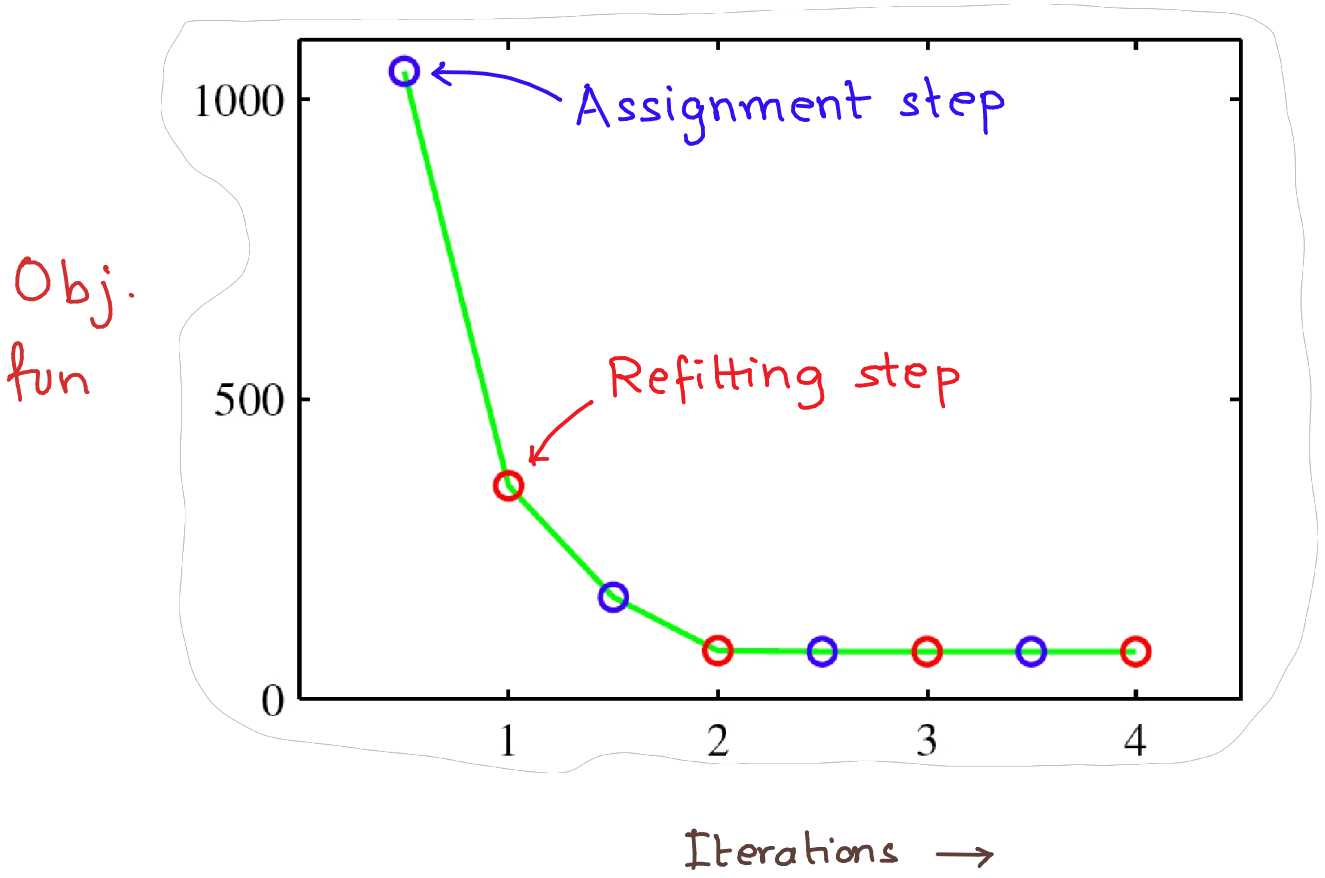
- Similar to the EM algorithm, Lloyd's algorithm converges to a stationary point of the objective function, but is not guaranteed to find the global optimum



  - In practice, run it multiple times, each time with a different initialization and pick the result of the run with smallest objective function value

Convergence of k-means algorithm

- Test of convergence: If the assignments do not change in the assignment step, then converged (to at least a local minimum)



k-means should not confused with k-NN

- k-means and k-NN are different, though they have certain similarities
- Both k-means and k-NN use Euclidean distances to define similarities in input space
- Both are sensitive to the normalization of the input values
- However, KNN is a supervised learning method, while k-means is an unsupervised learning method
- The 'k' in the two methods have different meaning

Choosing the number of clusters

- The number of clusters K has to be chosen apriori for both GMM and k-means algorithm for clustering

- Increasing K will reduce training loss (or reduce the objective function)
  - If $K = N$, then each data point will have its own cluster

- Cross-validation techniques are needed to guide selection of K
  - But they need to be adapted to unsupervised setting
  (There is no new data error E_{new} for clustering)

- For GMM, one can use the likelihood of the validation data to find K

Training set $\{\underline{x}_i\}_{i=1}^{N}$ Validation set $\{\tilde{\underline{x}}_i\}_{i=1}^{N_v}$

$K = 1 \rightarrow M^{(1)}, \hat{\underline{\theta}}^{(1)}$ $\quad P\left(\{\tilde{\underline{x}}_i\}_{i=1}^{N_v} \mid \hat{\underline{\theta}}^{(1)}, M^{(1)}\right) = 0.2$

$K = 2 \rightarrow M^{(2)}, \hat{\underline{\theta}}^{(2)}$ $\quad P\left(\{\tilde{\underline{x}}_i\}_{i=1}^{N_v} \mid \hat{\underline{\theta}}^{(2)}, M^{(2)}\right) = 0.45$ ✓ $\rightarrow M = 2$ optimal

$K = 3 \rightarrow M^{(3)}, \hat{\underline{\theta}}^{(3)}$ $\quad P\left(\{\tilde{\underline{x}}_i\}_{i=1}^{N_v} \mid \hat{\underline{\theta}}^{(3)}, M^{(3)}\right) = 0.1$

Choosing the number of clusters

- The validation methods should be handled with care
- In supervised learning, our goal is to obtain good predictions, so minimizing new data error makes sense
- In clustering, the goal is not necessarily to minimize "clustering loss" but to gain insights by finding a small number of clusters
  - So we may prefer a smaller number of clusters even if it gives not-so good validation loss
- The ELBOW method is often used for selecting K
  - plot of loss (either training, validation, or both)