

---

---

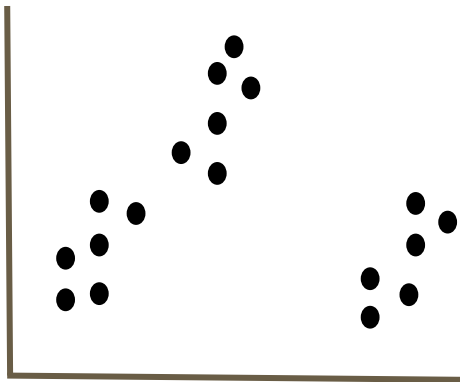
# Clustering - Kmeans

— Boston University CS 506 - Lance Galletti —

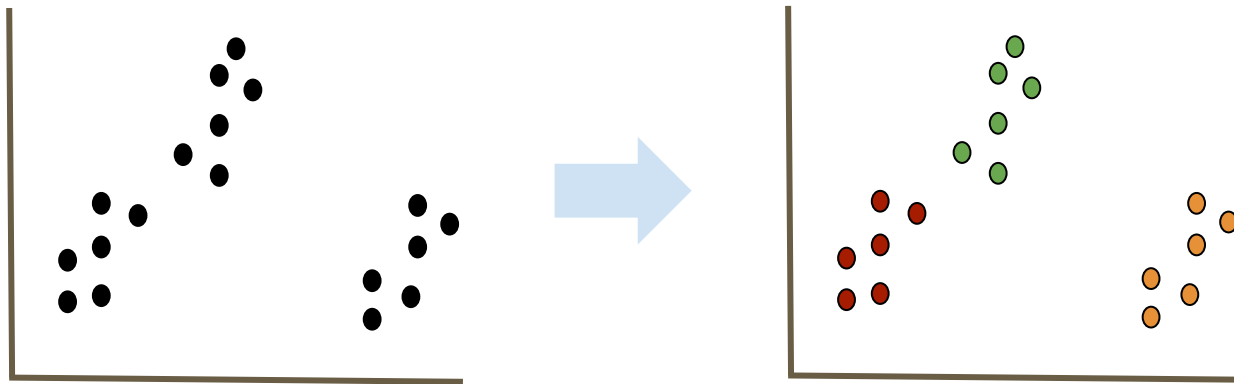
---

---

# What is a Clustering



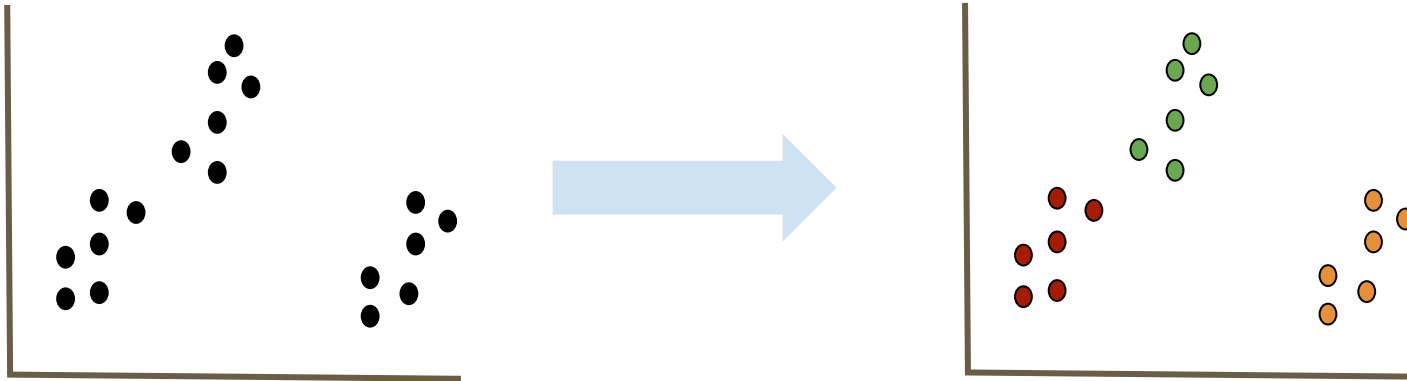
# What is a Clustering



# What is a Clustering

A clustering is a grouping / assignment of objects (data points) such that objects in the same group / cluster are:

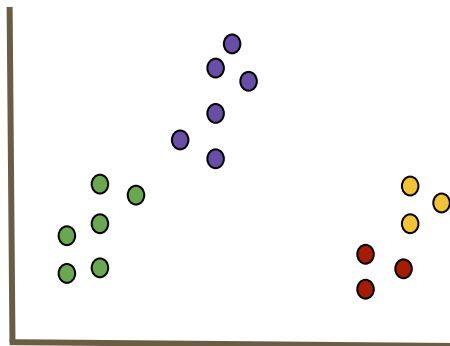
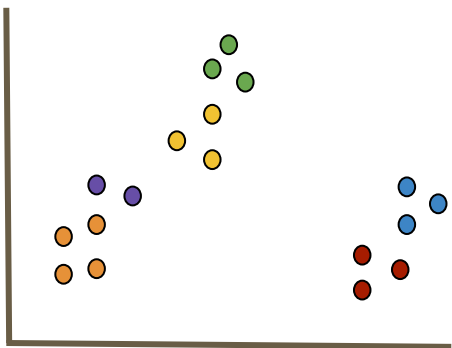
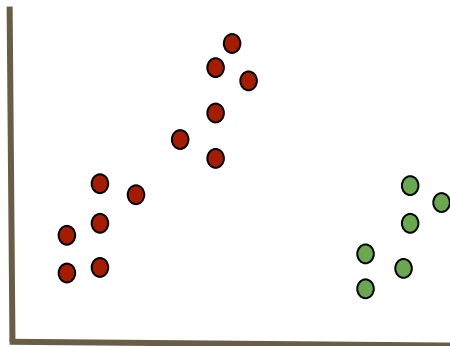
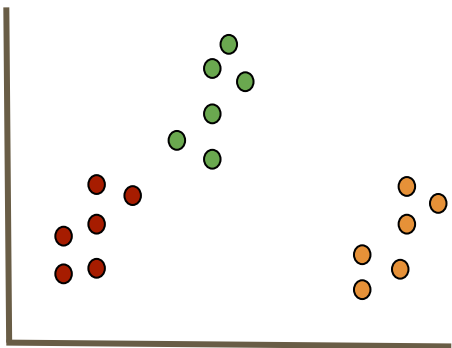
- similar to one another
- dissimilar to objects in other groups



# Applications

- Outlier detection / anomaly detection
  - Data Cleaning / Processing
  - Credit card fraud, spam filter etc.
- Feature Extraction
- Filling Gaps in your data
  - Using the same marketing strategy for similar people
  - Infer probable values for gaps in the data (similar users could have similar hobbies, likes / dislikes etc.)

# Clusters can be Ambiguous



# Types of Clusterings

## **Partitional**

Each object belongs to exactly one cluster

## **Hierarchical**

A set of nested clusters organized in a tree

## **Density-Based**

Defined based on the local density of points

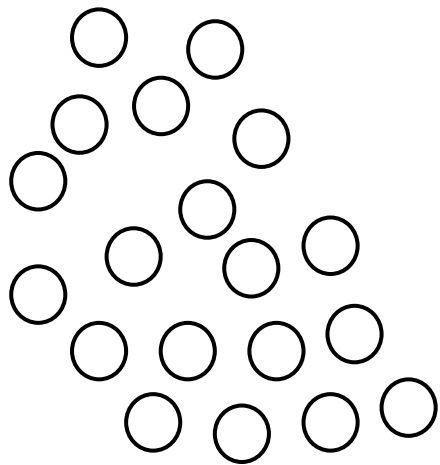
## **Soft Clustering**

Each point is assigned to every cluster with a certain probability

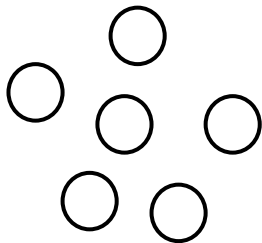
# Partitional Clustering

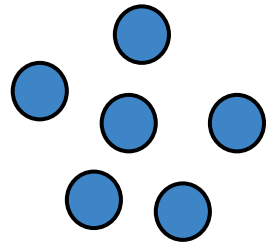
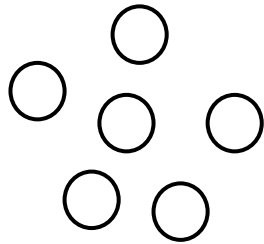
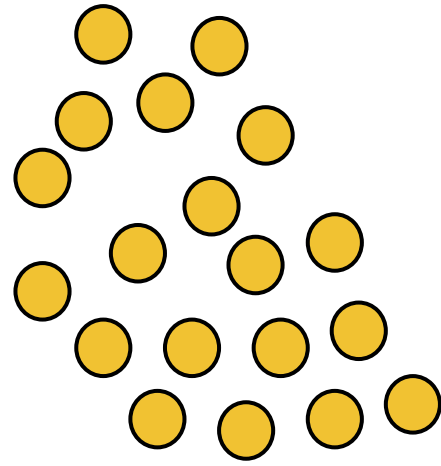
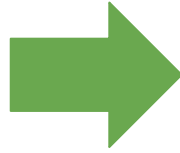
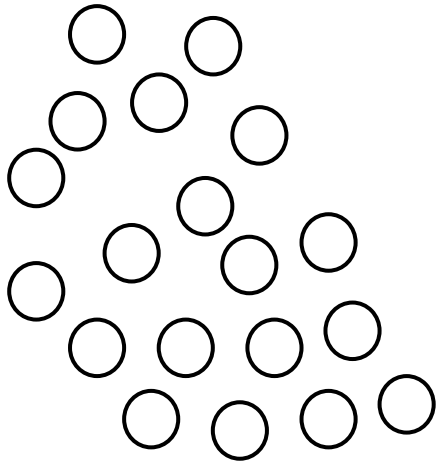


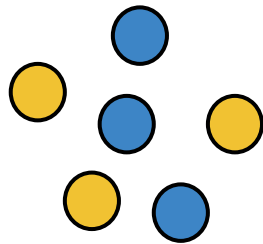
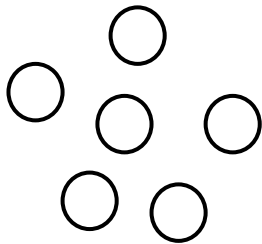
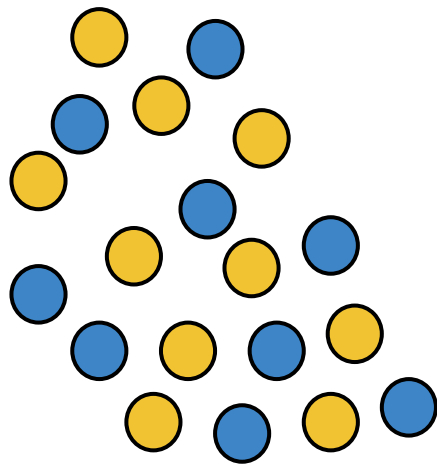
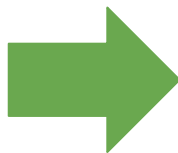
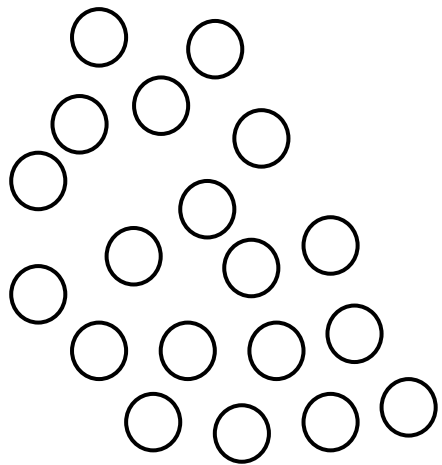
# Partitional Clustering

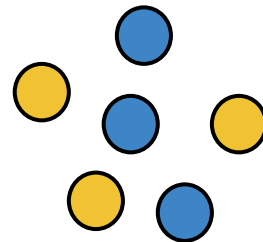
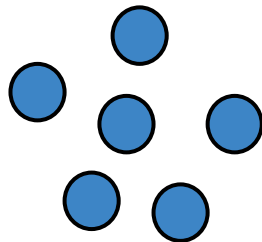
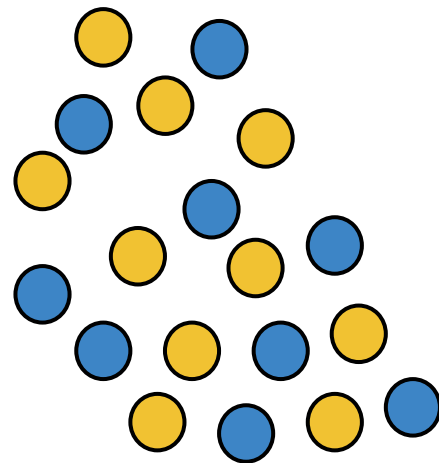
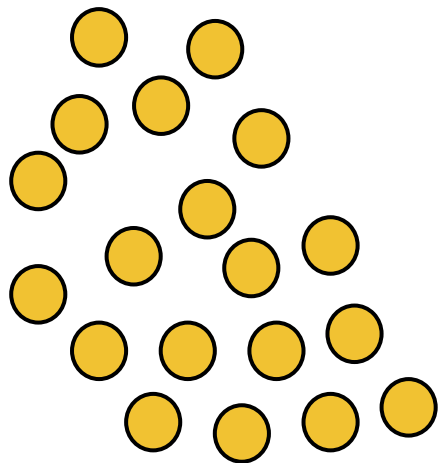


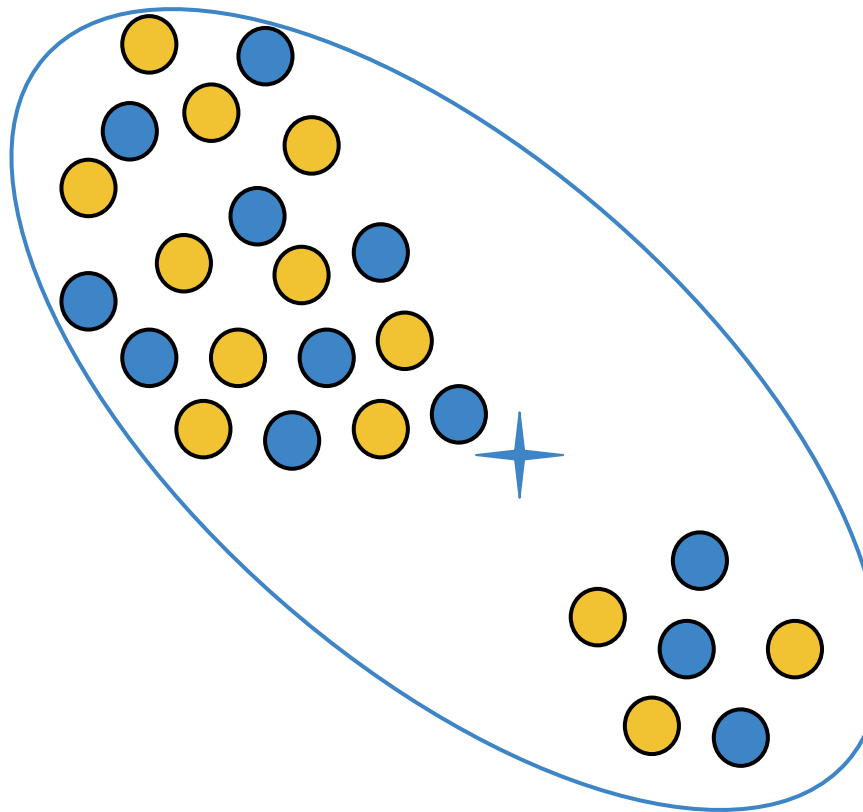
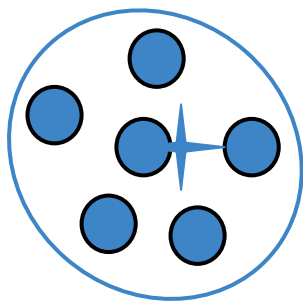
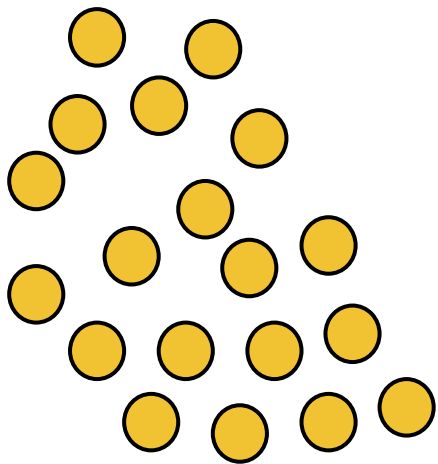
**Goal:** partition dataset into  $k$  partitions

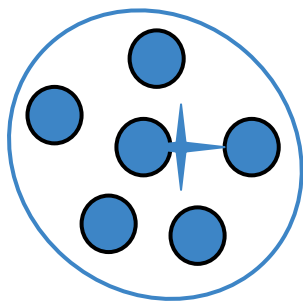
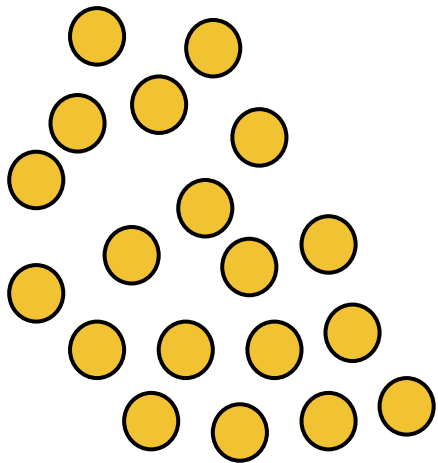




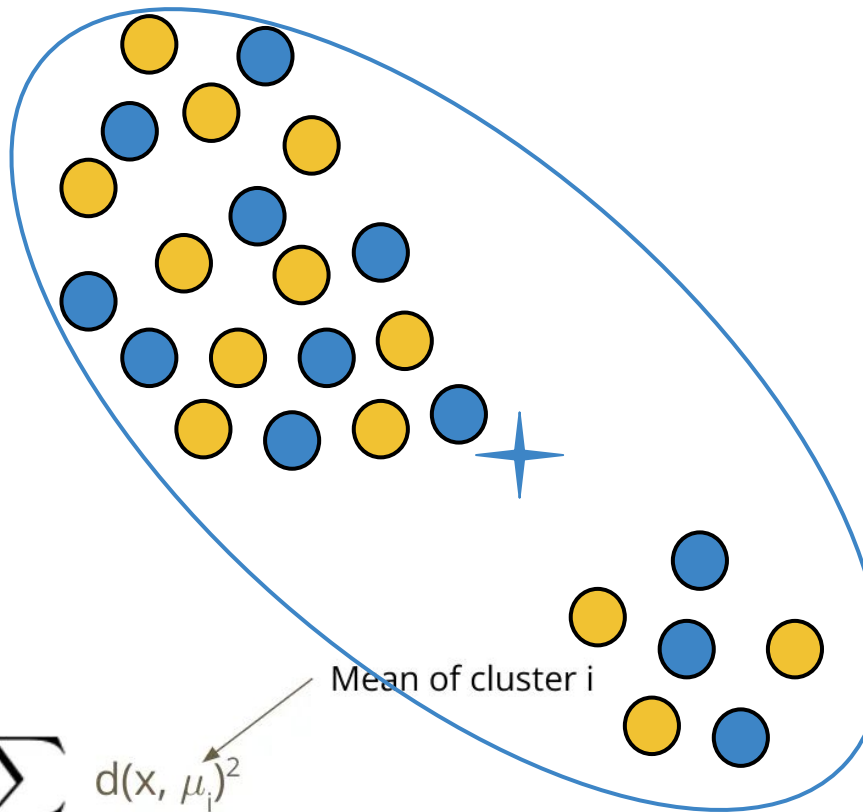








$$\frac{1}{|C_i|} \sum_{x \in C_i}$$



Mean of cluster i

Cluster i

# Cost Function

$$\sum_i^k \sum_{x \in C_i} d(x, \mu_i)^2$$

- Way to evaluate and compare solutions
- Hope: can find some algorithm that find solutions that make the cost small

# K-means

Given  $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$  our dataset,  $\mathbf{d}$  the euclidean distance, and  $\mathbf{k}$

Find  $\mathbf{k}$  centers  $\{\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_k\}$  that minimize the **cost function**:

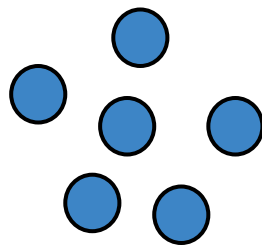
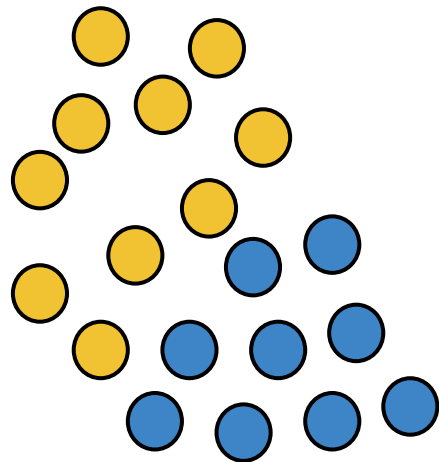
$$\sum_i^k \sum_{x \in C_i} d(x, \mu_i)^2$$

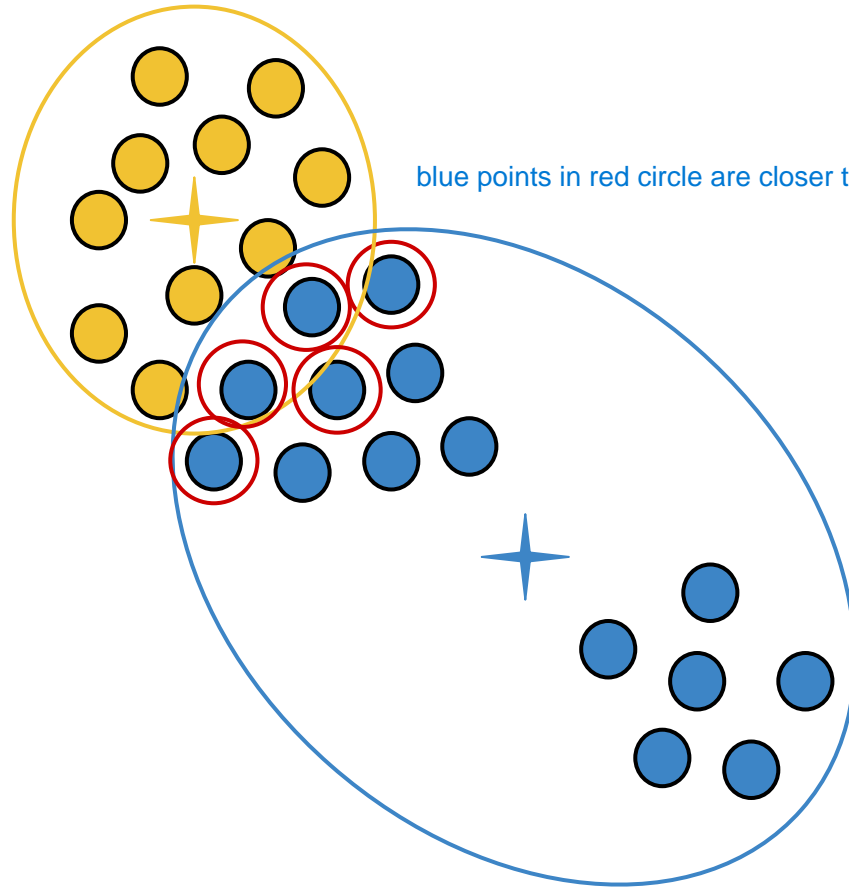
When  $\mathbf{k}=1$  and  $\mathbf{k}=n$  this is easy. Why?

1 cluster including all points; each point has a cluster

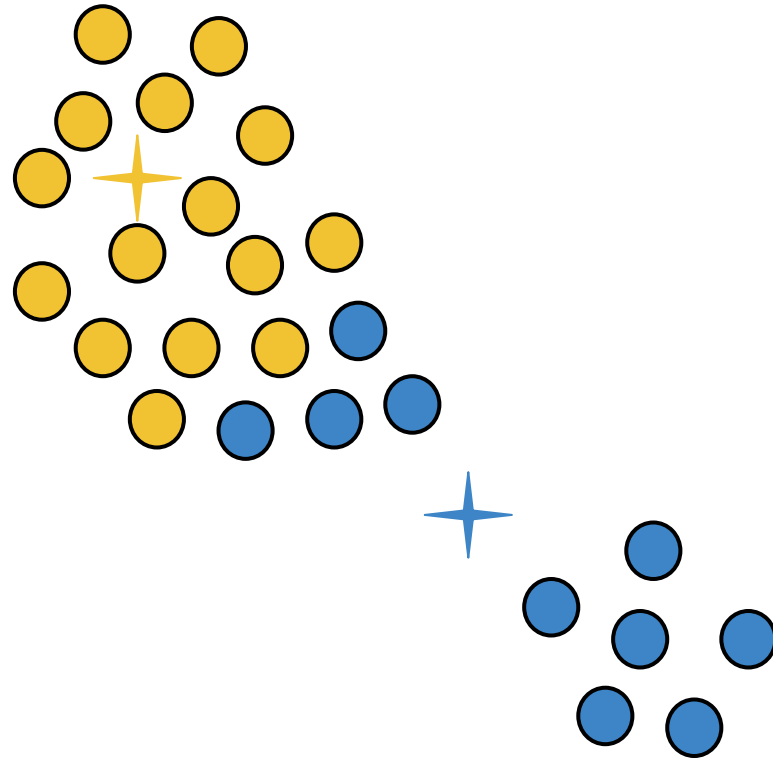
When  $\mathbf{x}_i$  lives in more than 2 dimensions, this is a very difficult (**NP-hard**) problem

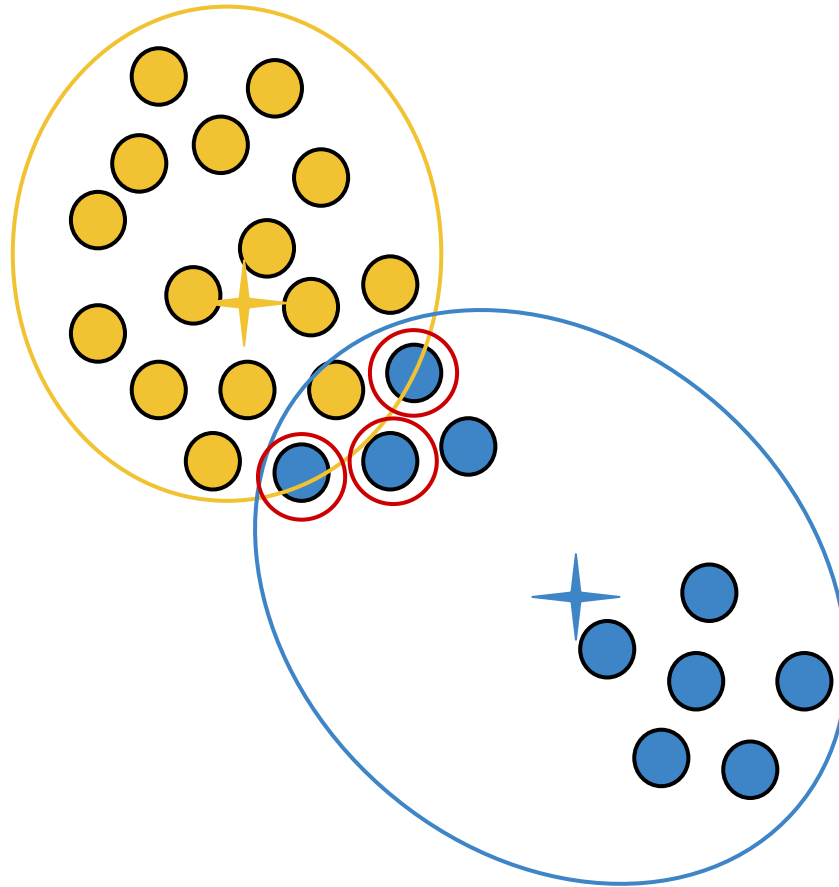


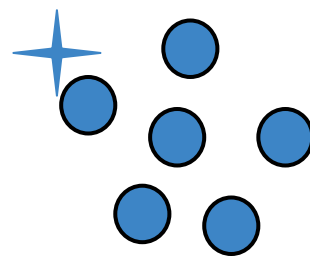
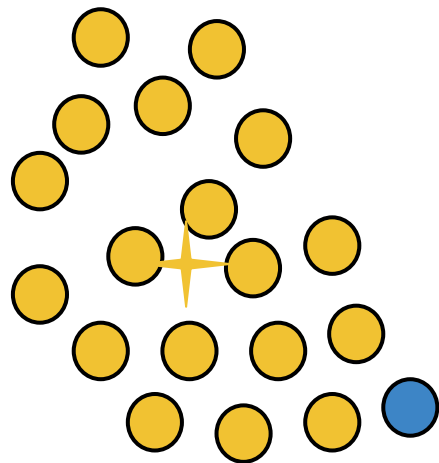


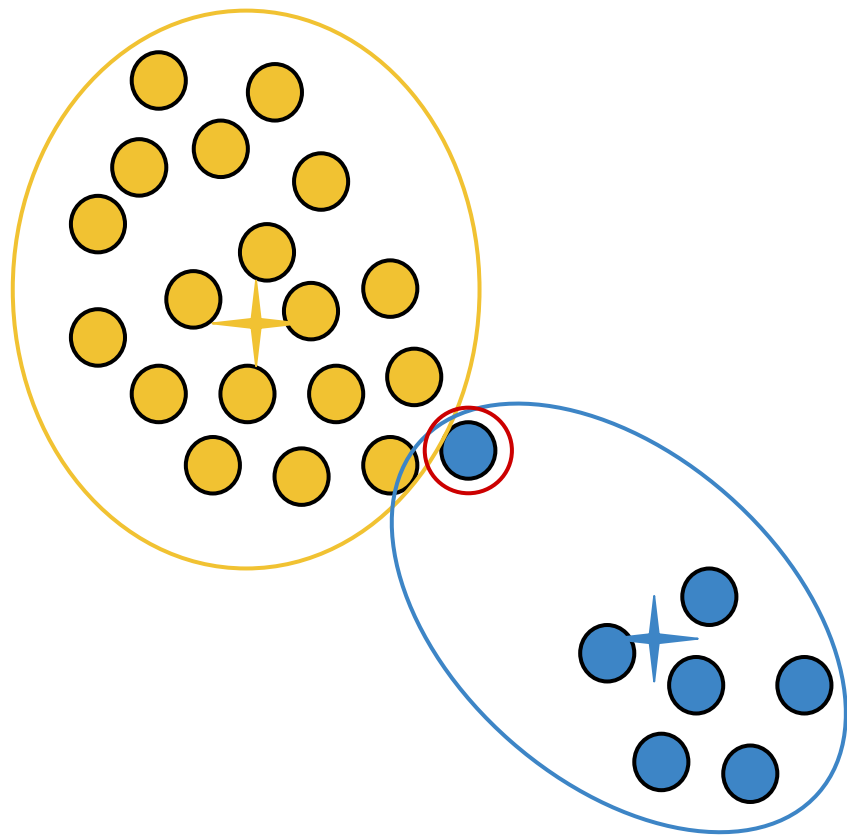


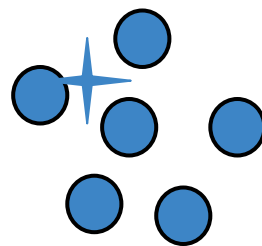
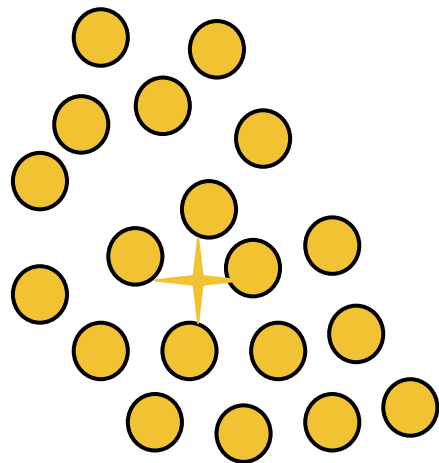
blue points in red circle are closer to the mean of yellow cluster







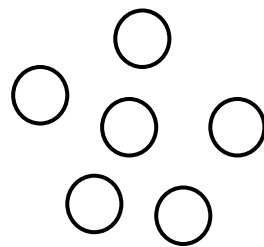
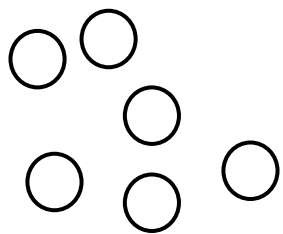
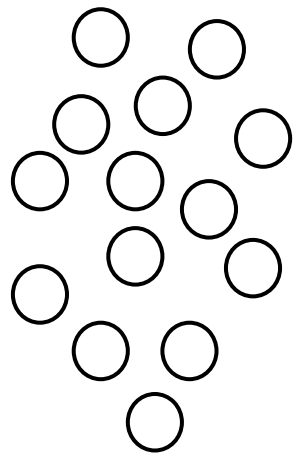


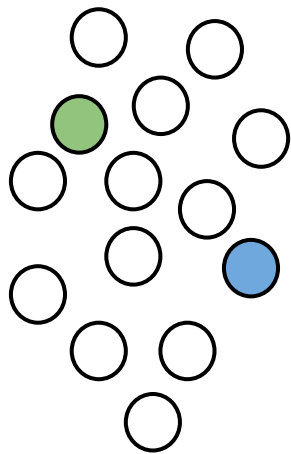


# K-means - Lloyd's Algorithm

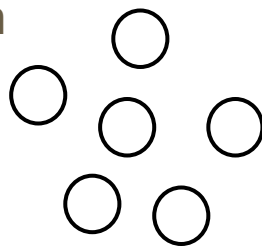
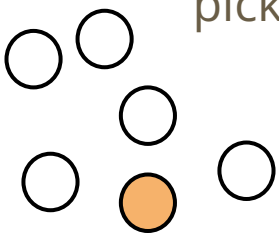
1. Randomly pick  $k$  centers  $\{\mu_1, \dots, \mu_k\}$
2. Assign each point in the dataset to its closest center
3. Compute the new centers as the means of each cluster
4. Repeat 2 & 3 until convergence

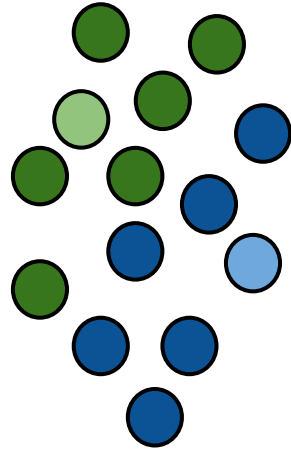




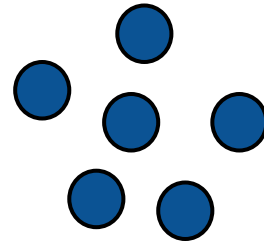
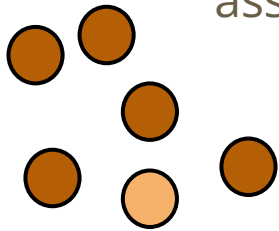


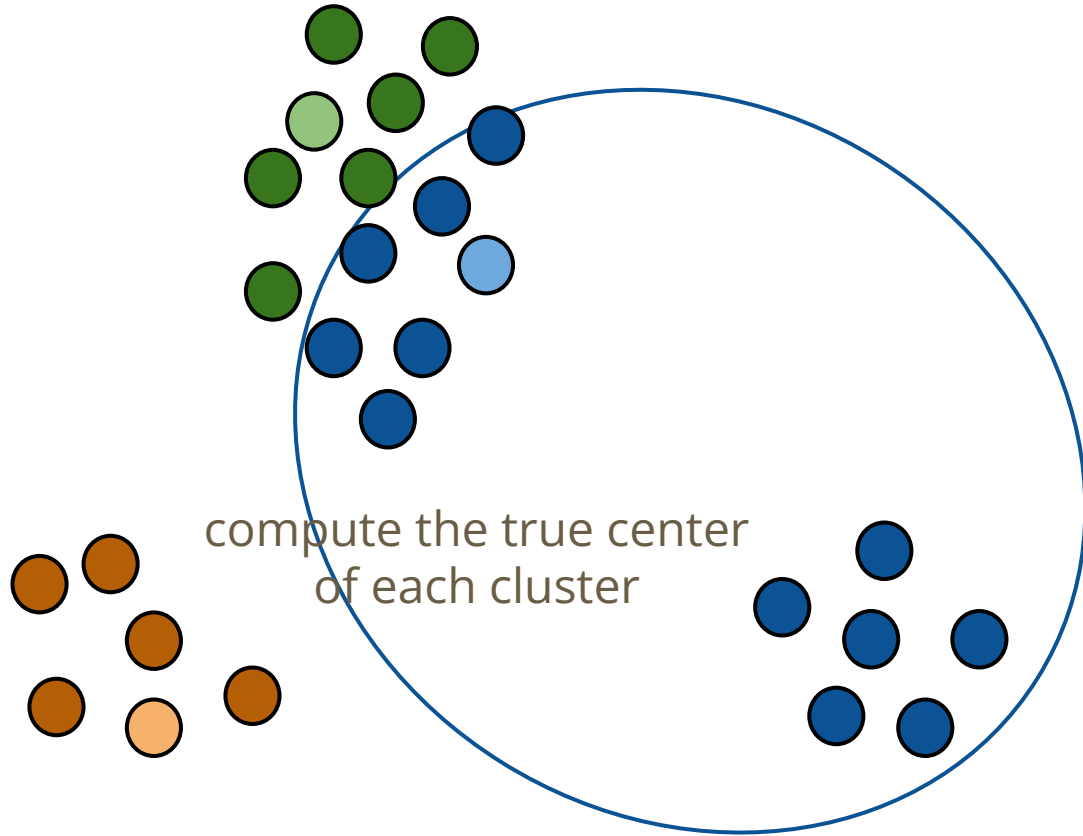
pick  $k$  centers at random

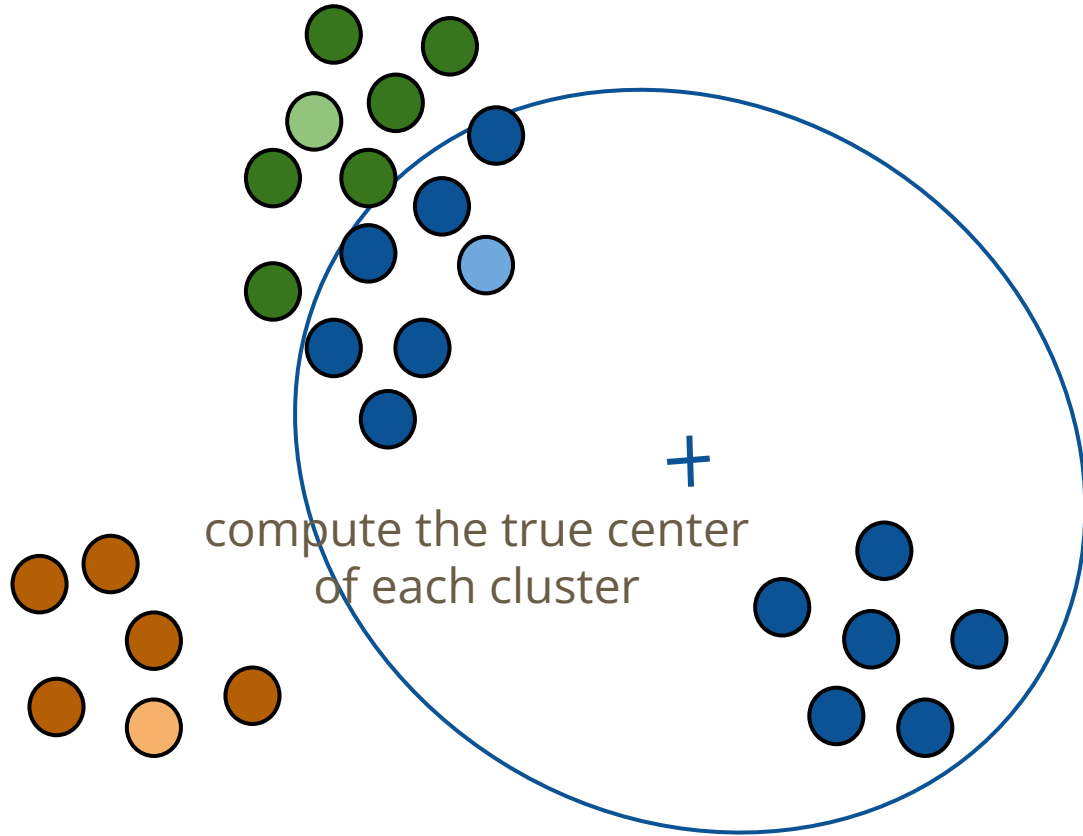


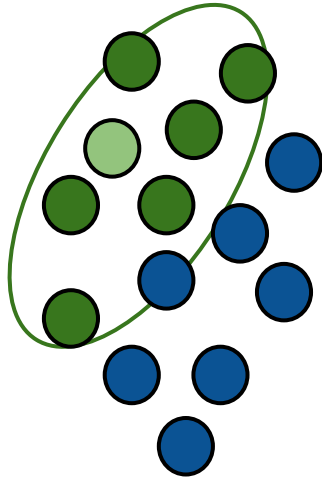


assign points to closest  
center



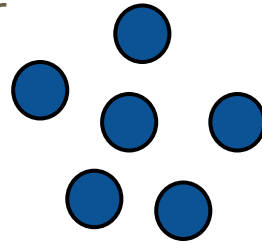
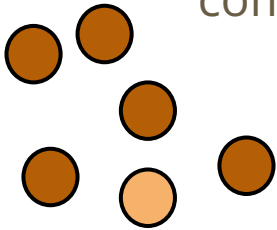


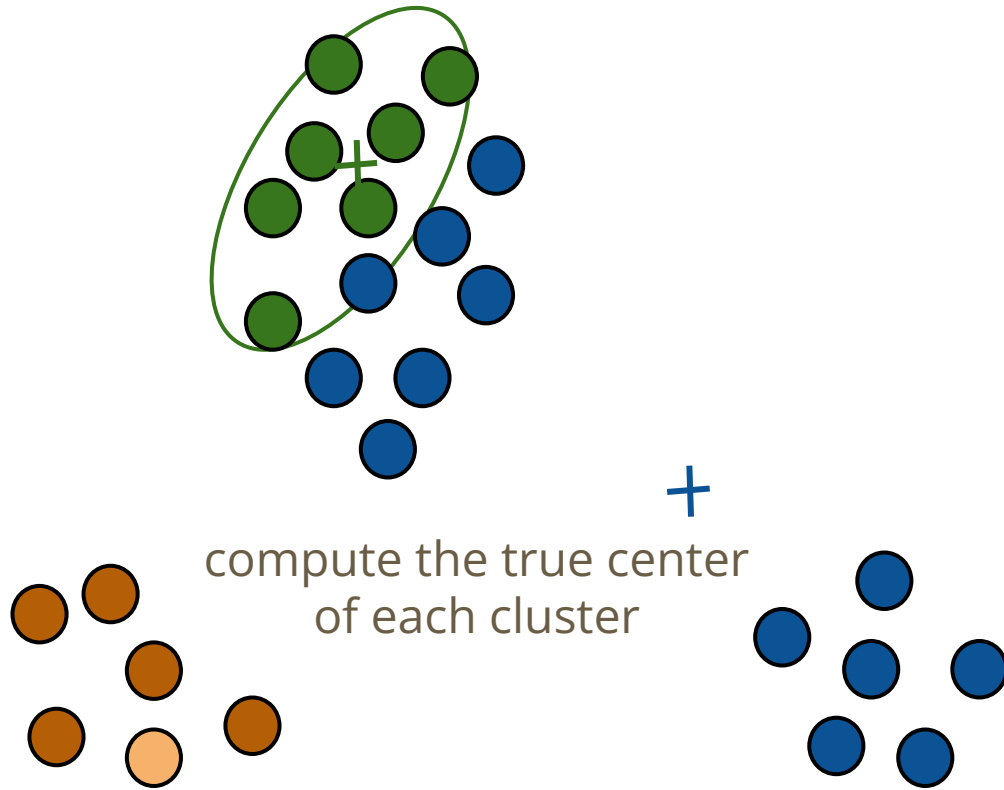


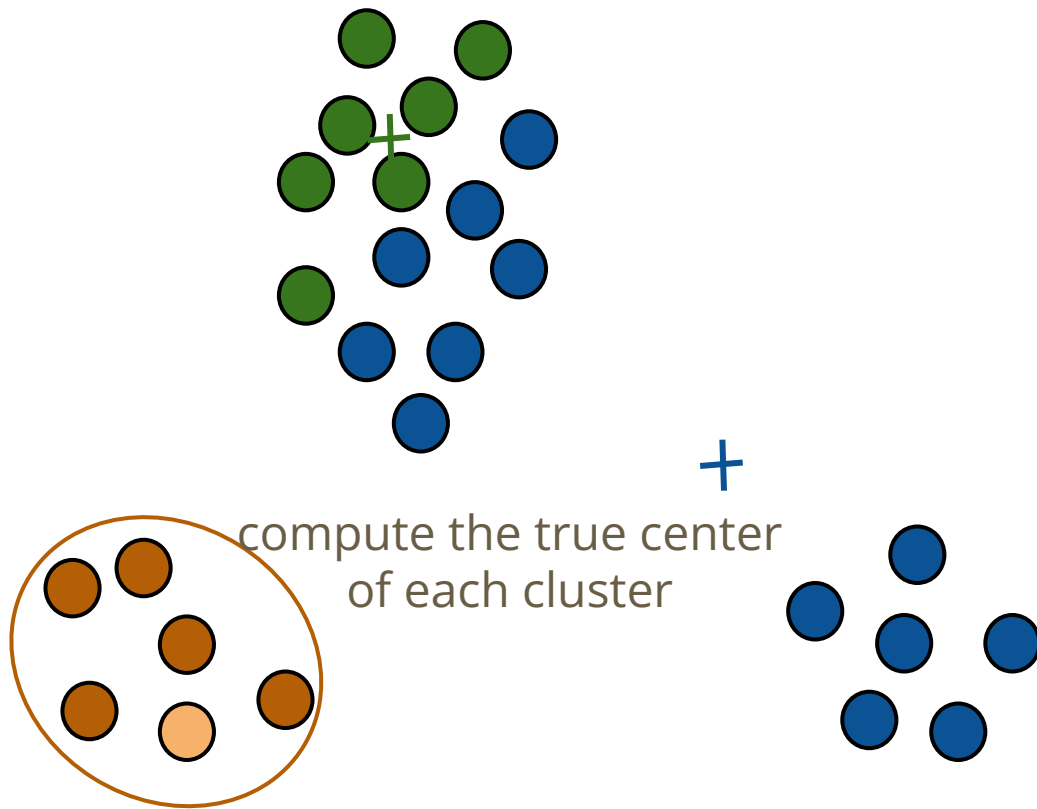


+

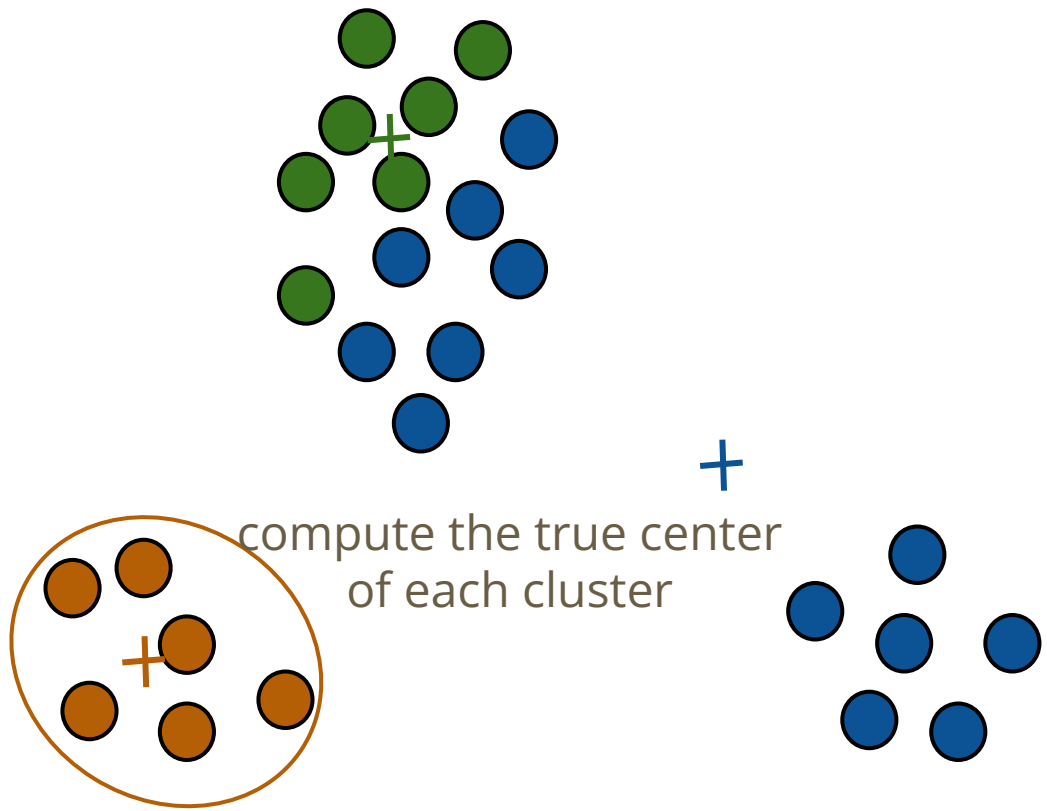
compute the true center  
of each cluster

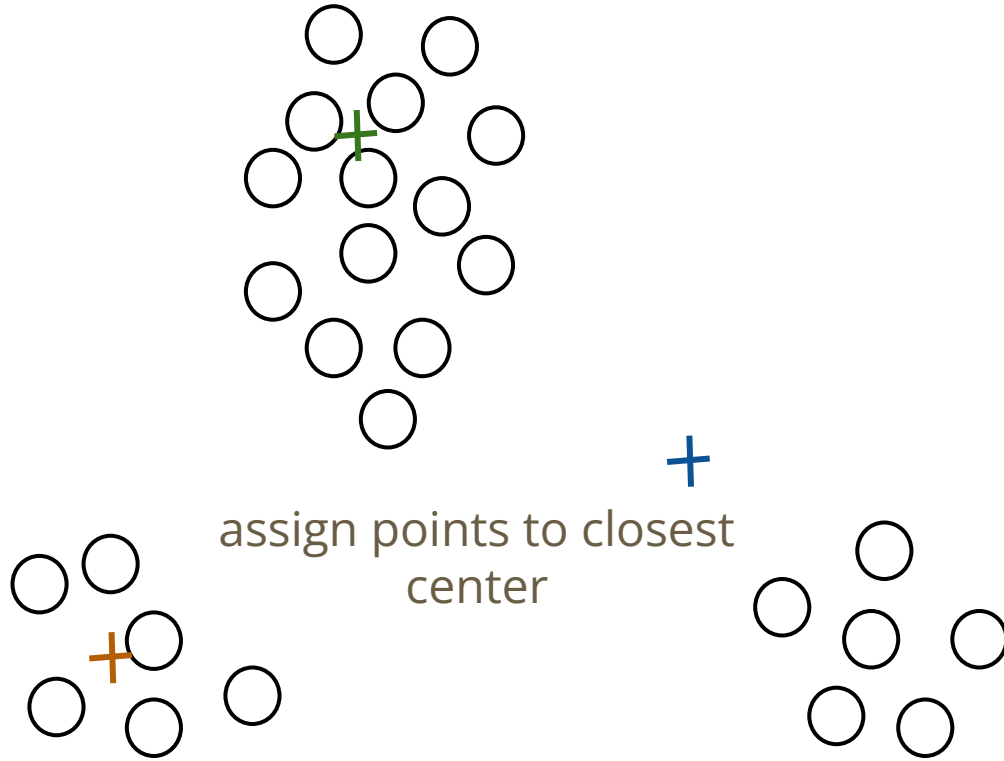


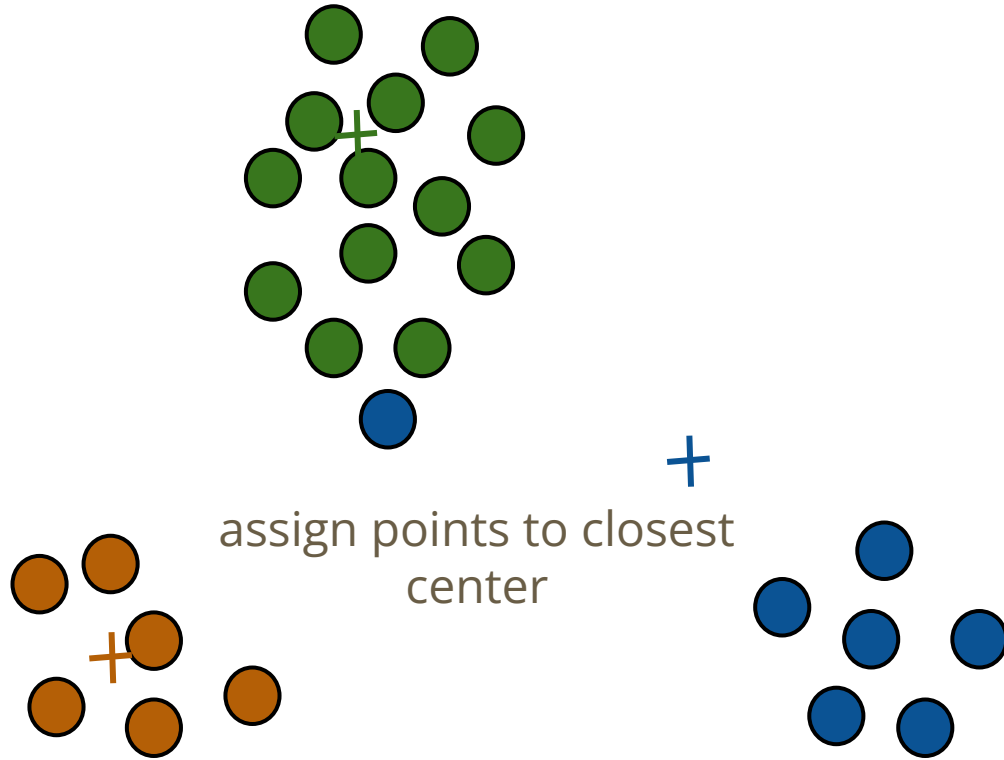


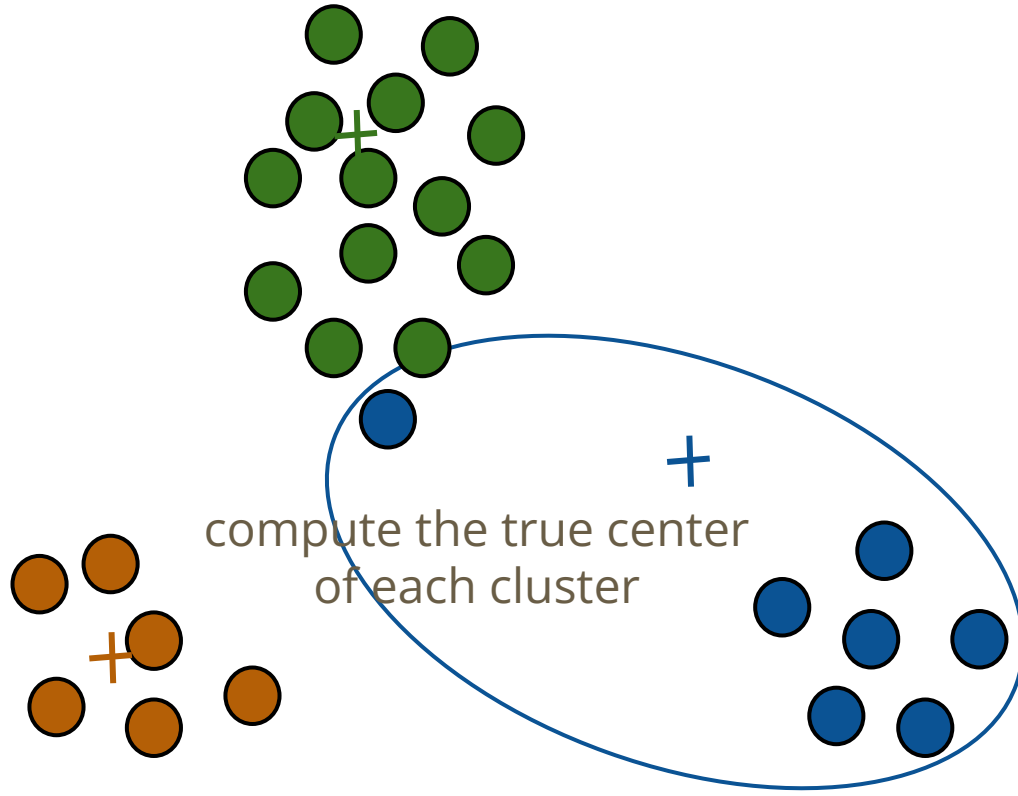


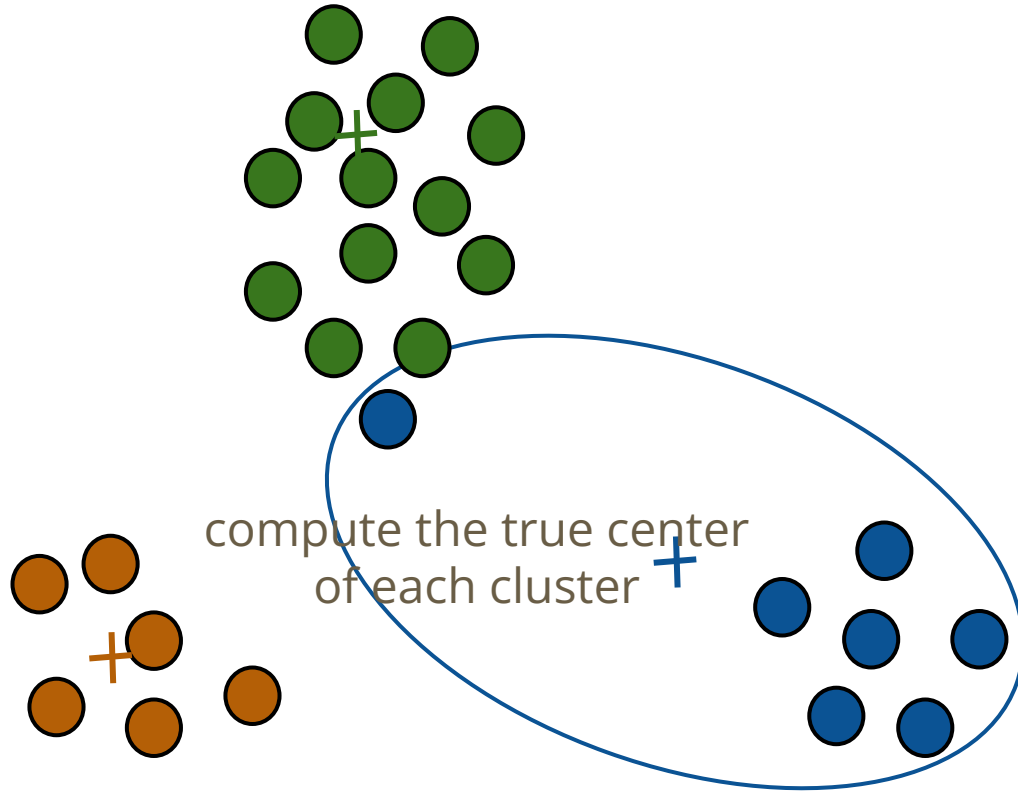


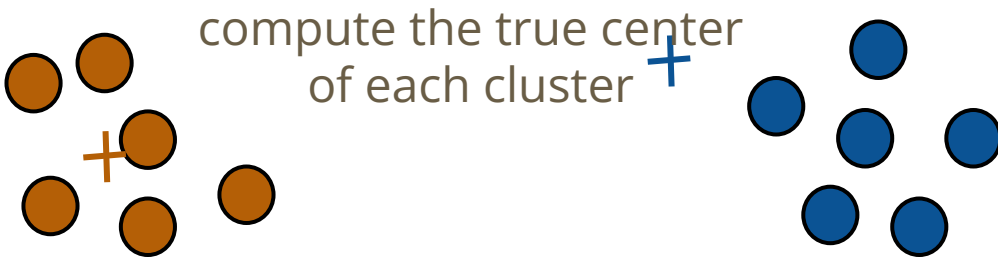
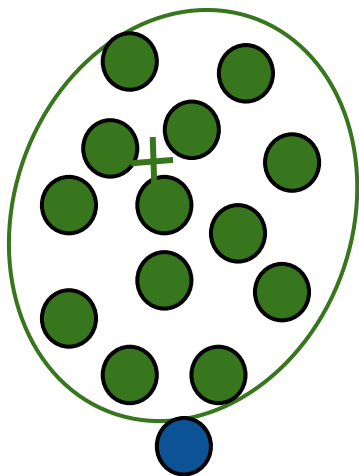


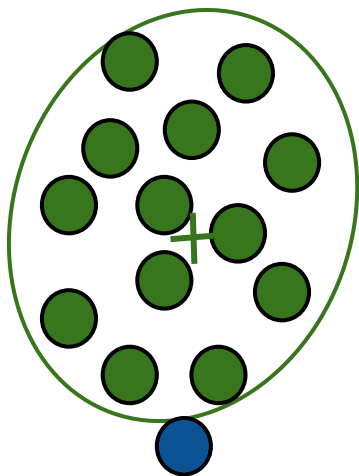




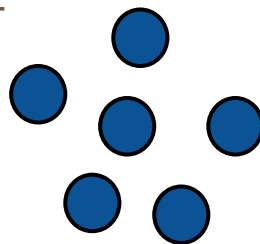
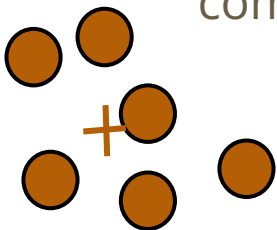


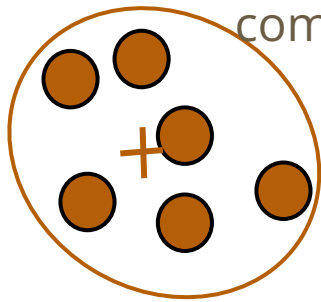
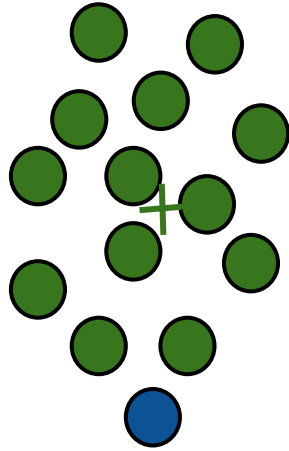




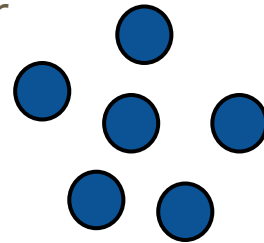


compute the true center  
of each cluster +

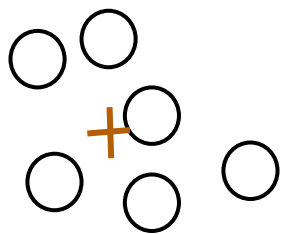
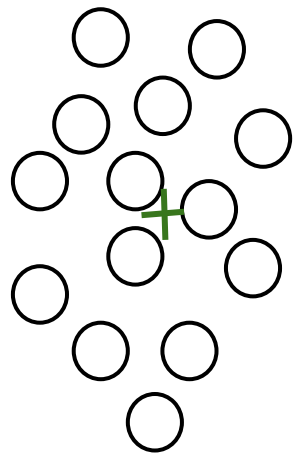




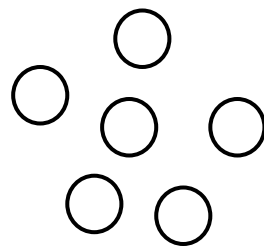
compute the true center  
of each cluster +

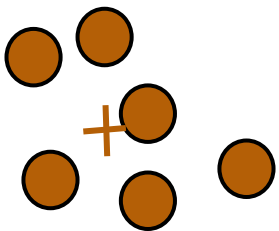
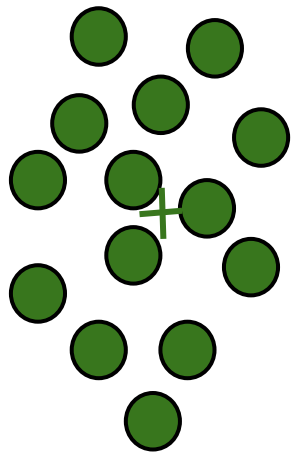




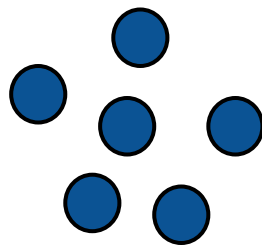


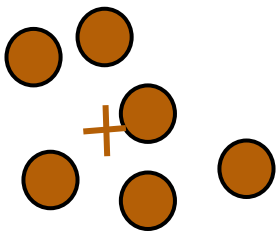
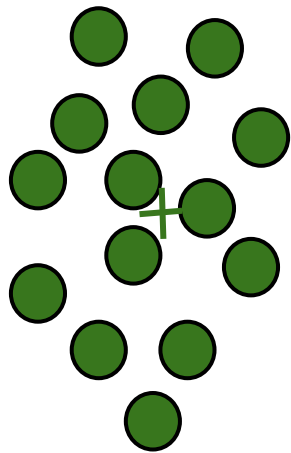
+



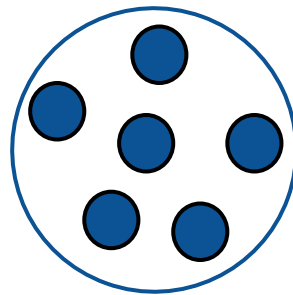


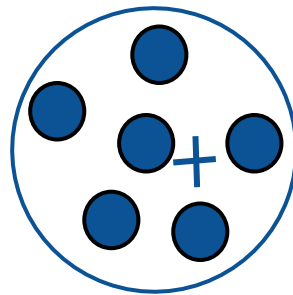
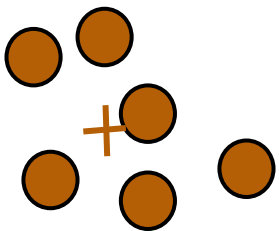
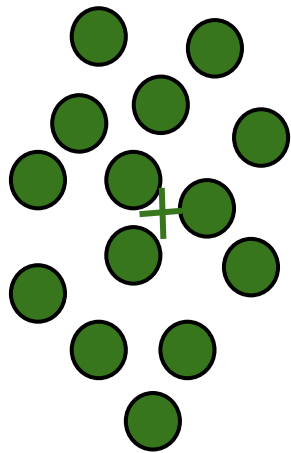
+

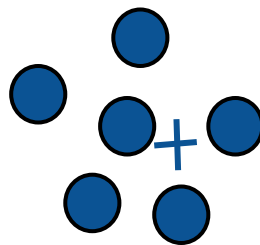
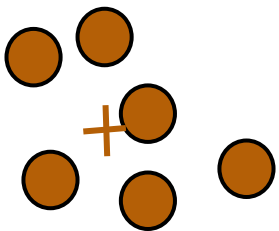
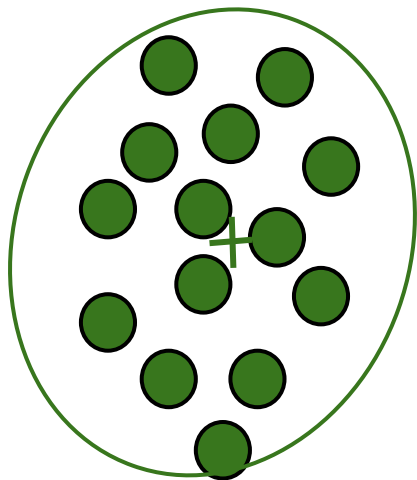


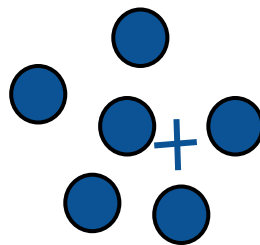
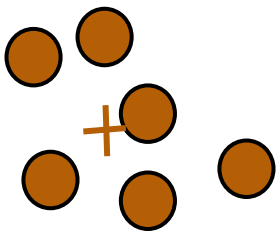
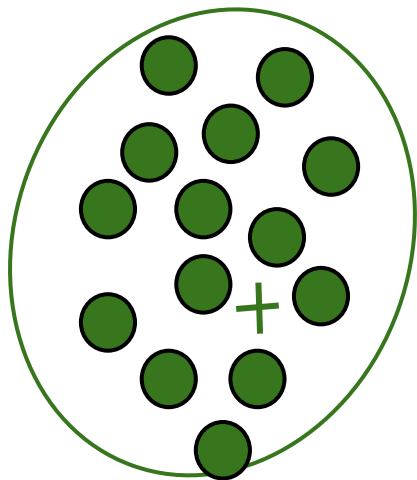


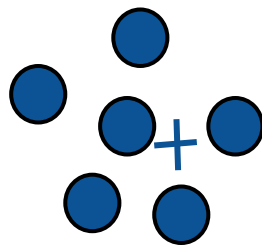
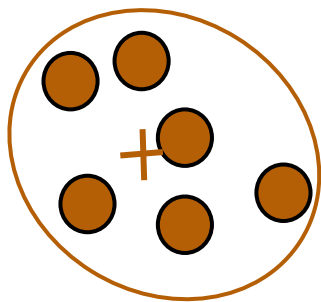
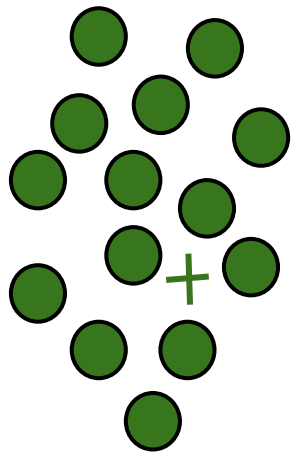
+

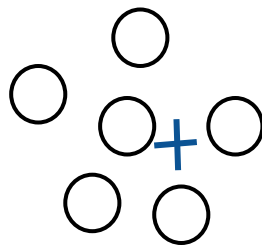
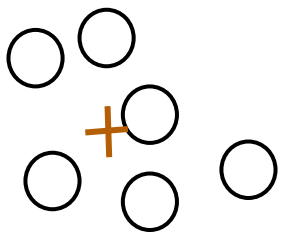
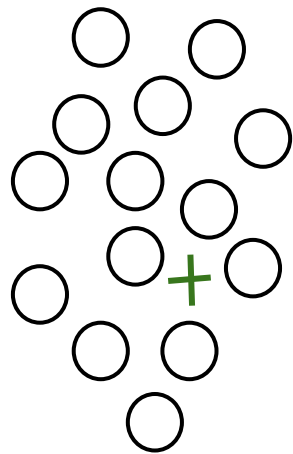




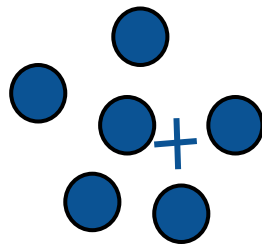
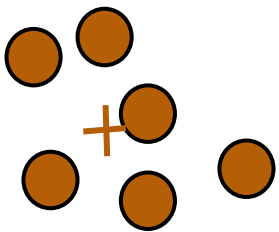
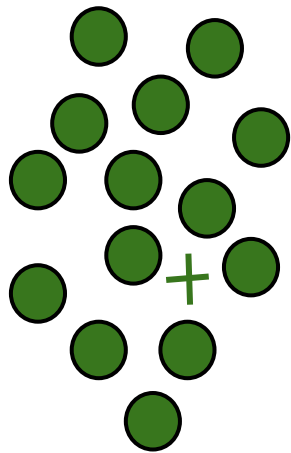


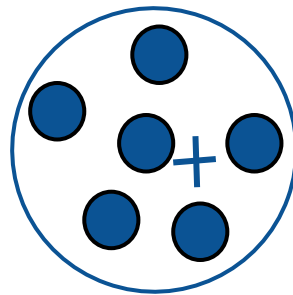
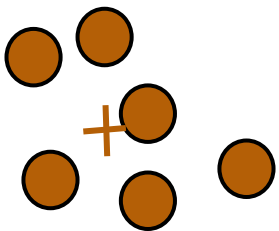
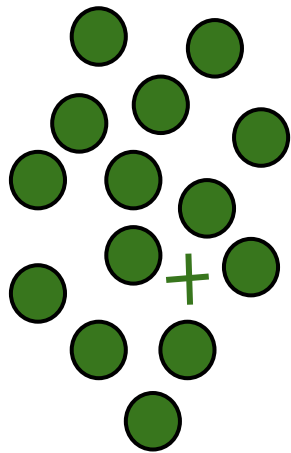


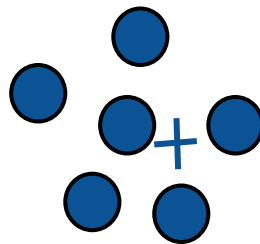
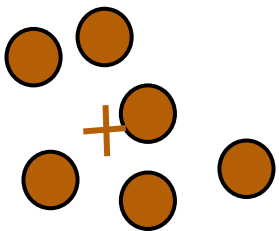
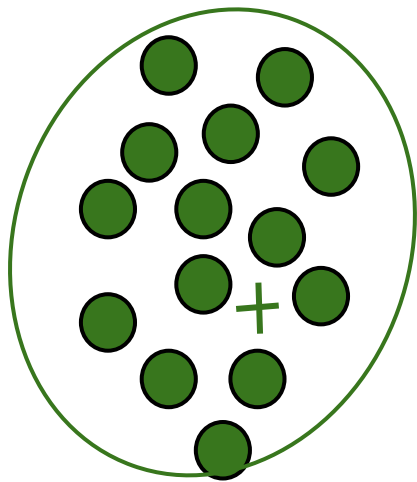


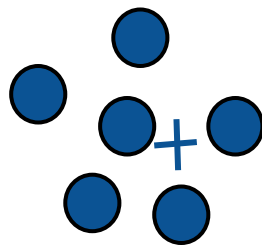
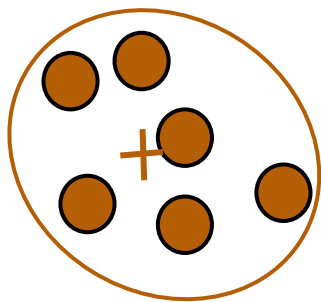
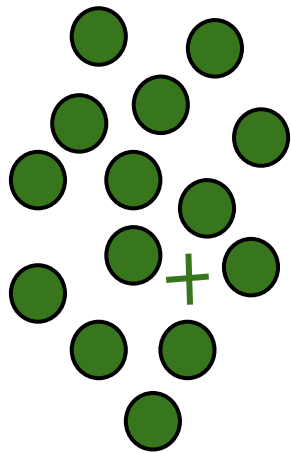


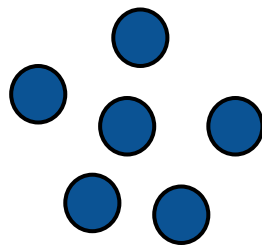
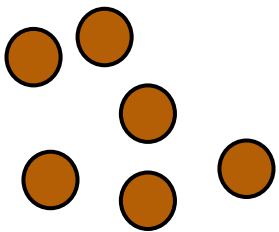
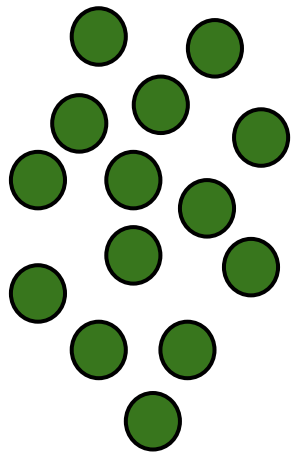








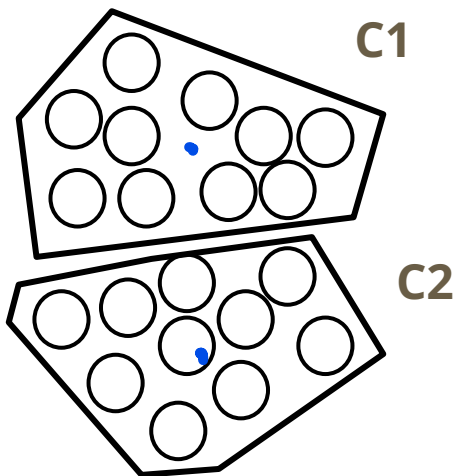




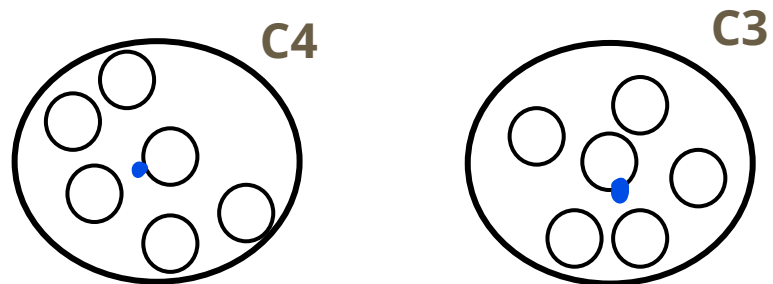
# Questions

do they converge?

1

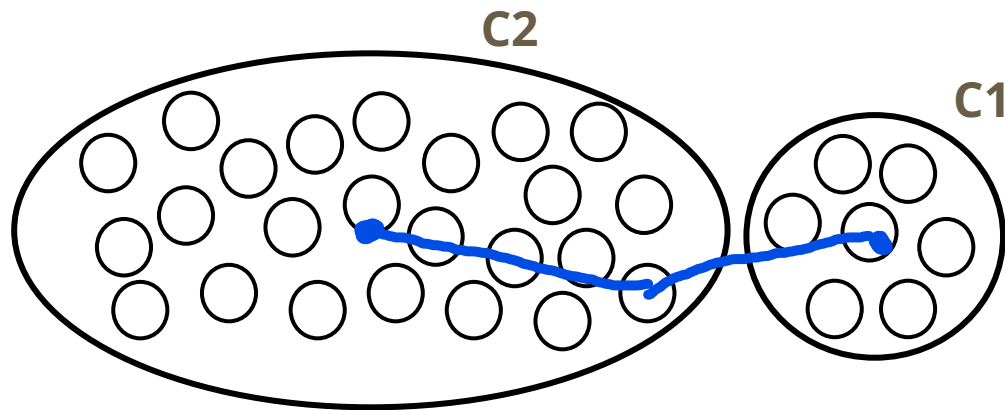


yes



2

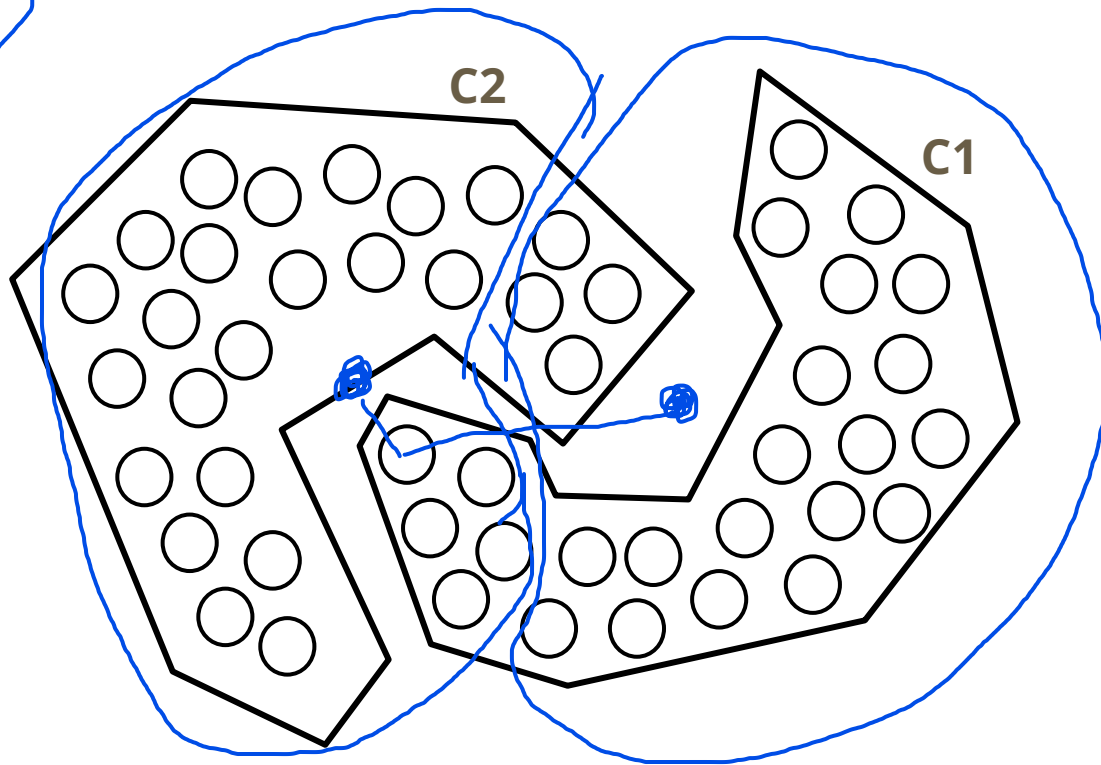
no



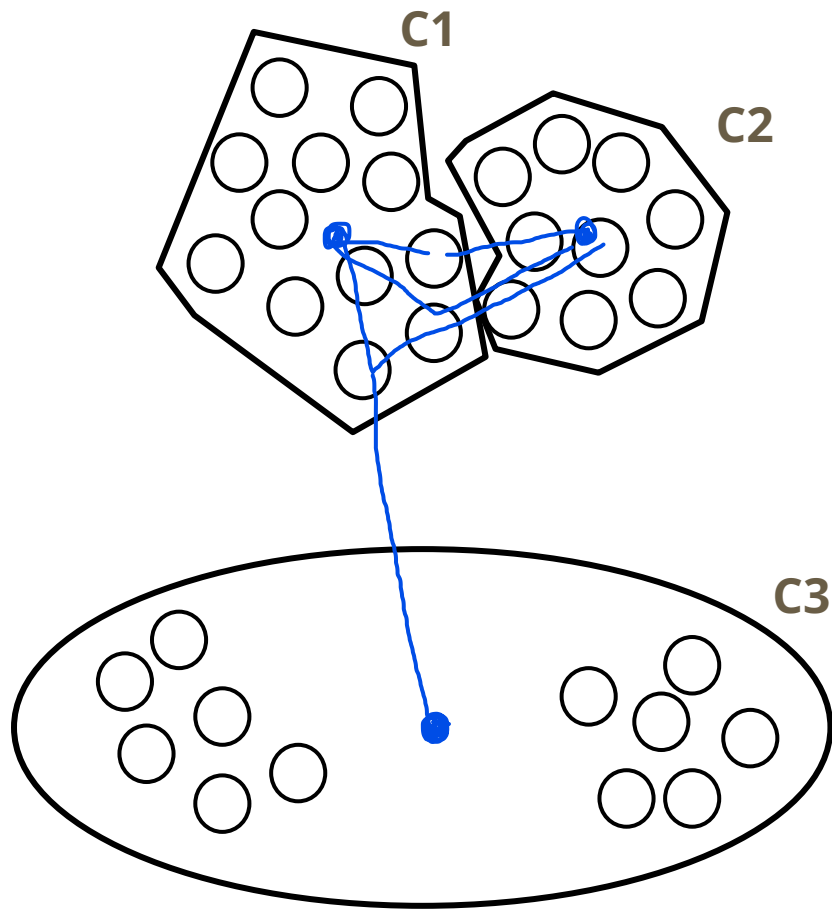


3

no



4



5

C2

yes

C1

