Unsupervised Learning: Clustering

Rishabh Iyer

University of Texas at Dallas

Based on the slides of Nick Rouzzi and Vibhav Gogate

Clustering

Clustering systems:

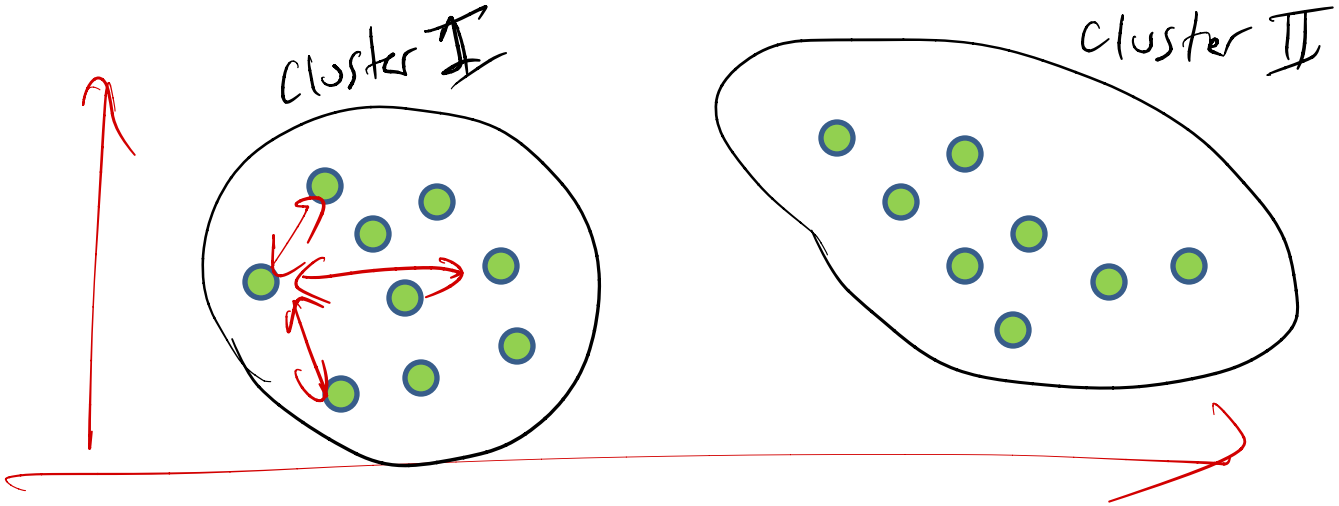
- Unsupervised learning
- Requires data, but no labels
- Detect patterns, e.g., in
  - Group emails or search results
  - Customer shopping patterns
- Useful when don't know what you're looking for...
  - But often get gibberish

Photos

Features / Attributes need to be useful

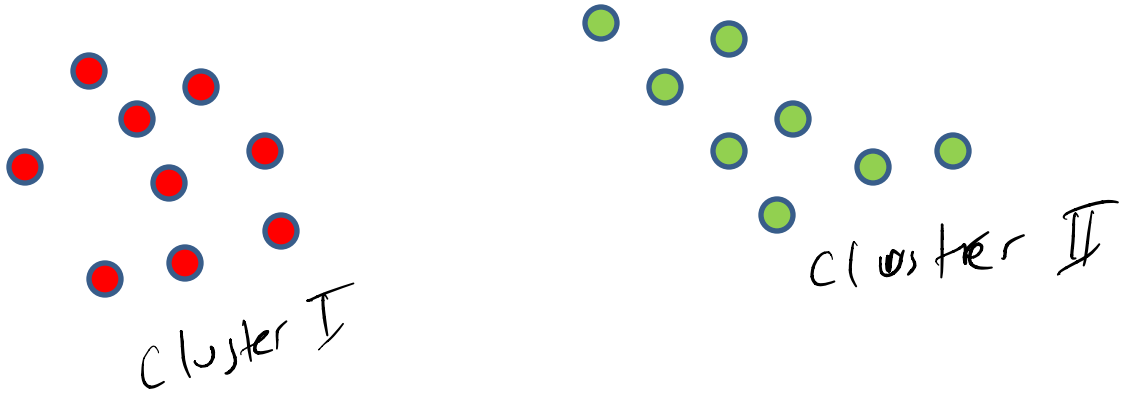
Clustering

- Want to group together parts of a dataset that are close together in some metric
  - Useful for finding the important parameters/features of a dataset



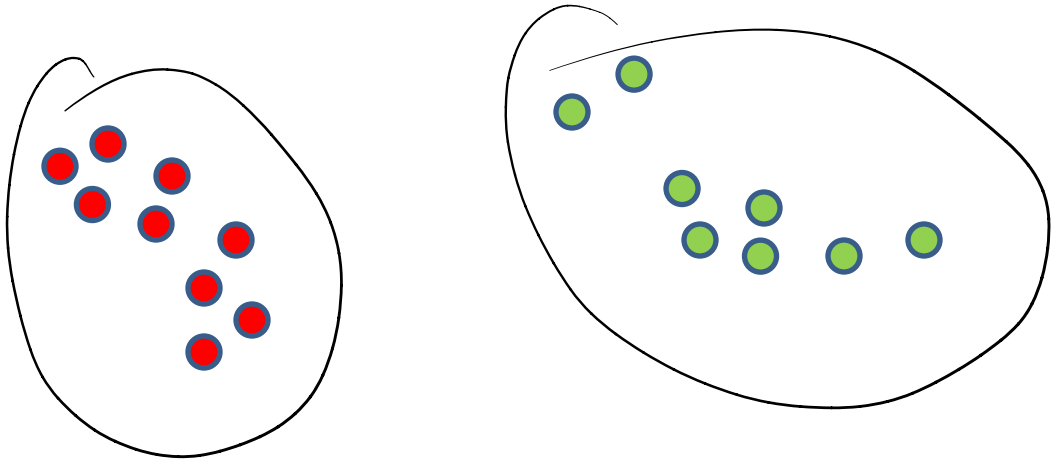
Clustering

- Want to group together parts of a dataset that are close together in some metric
  - Useful for finding the important parameters/features of a dataset



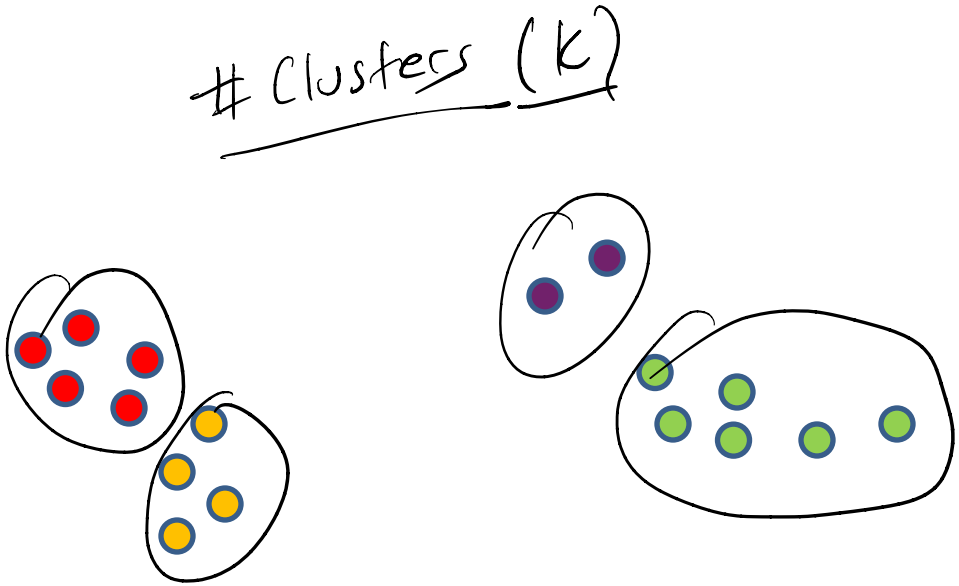
Clustering

- Intuitive notion of clustering is a somewhat ill-defined problem
  - Identification of clusters depends on the scale at which we perceive the data



Clustering

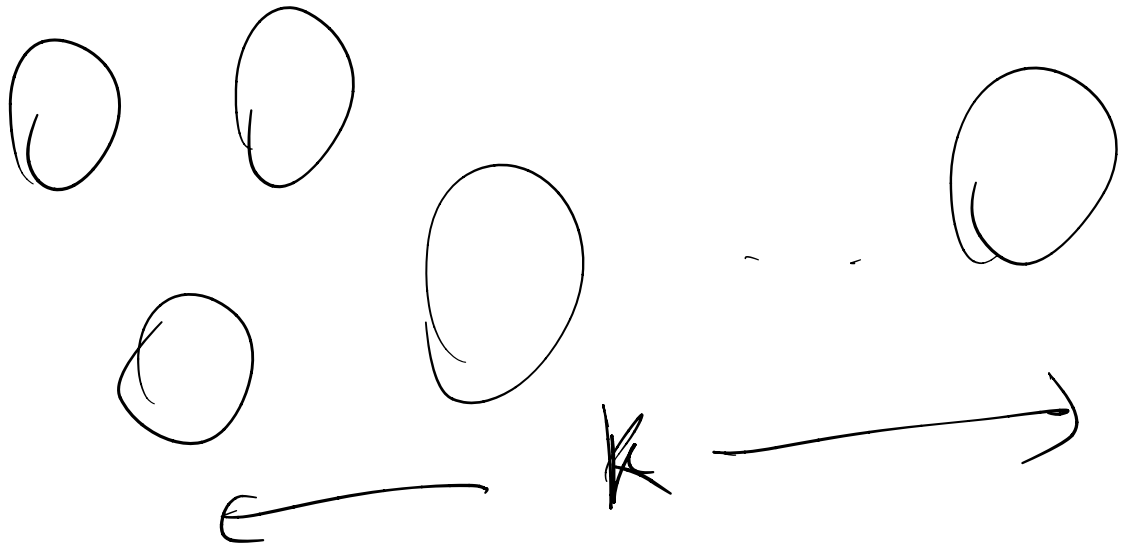
- Intuitive notion of clustering is a somewhat ill-defined problem
  - Identification of clusters depends on the scale at which we perceive the data



Clustering

- Input: a collection of points $x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^n$, an integer k

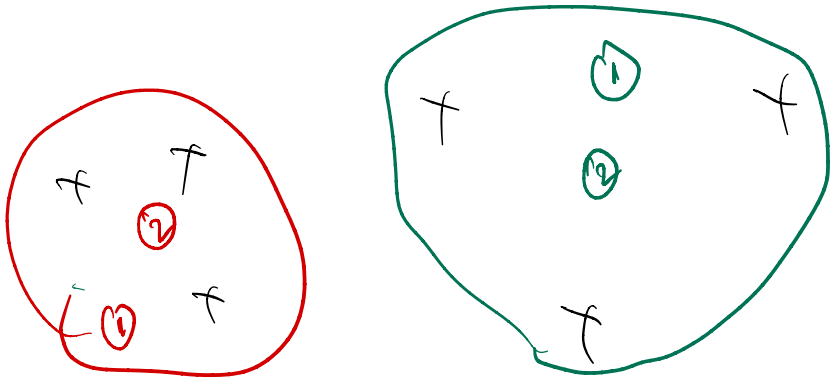
- Output: A partitioning of the input points into k sets that minimizes some metric of closeness



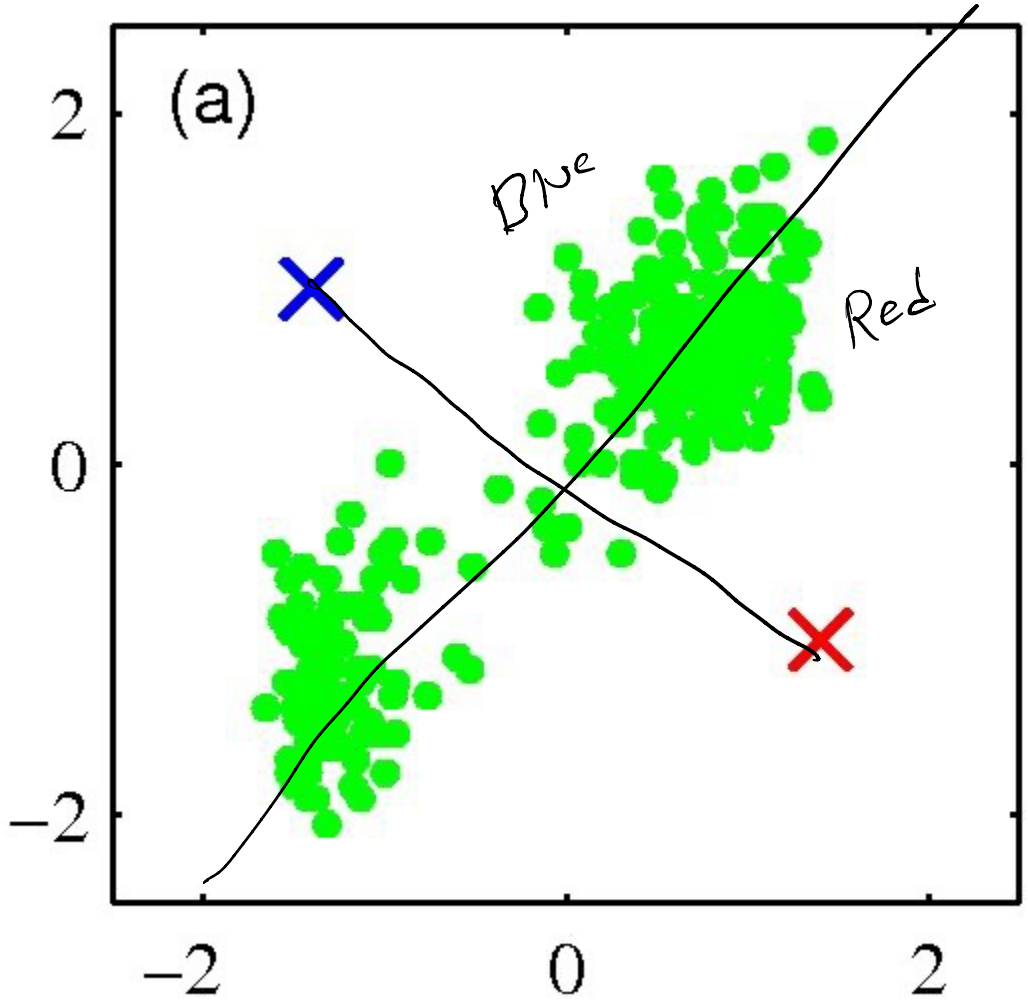
k-means Clustering

centers / centroids

- Pick an initial set of k means (usually at random) ← Choose k points from n at random
- Repeat until the clusters do not change:
  - Partition the data points, assigning each data point to a cluster based on the mean that is closest to it
  - Update the cluster means so that the i^{th} mean is equal to the average of all data points assigned to cluster i



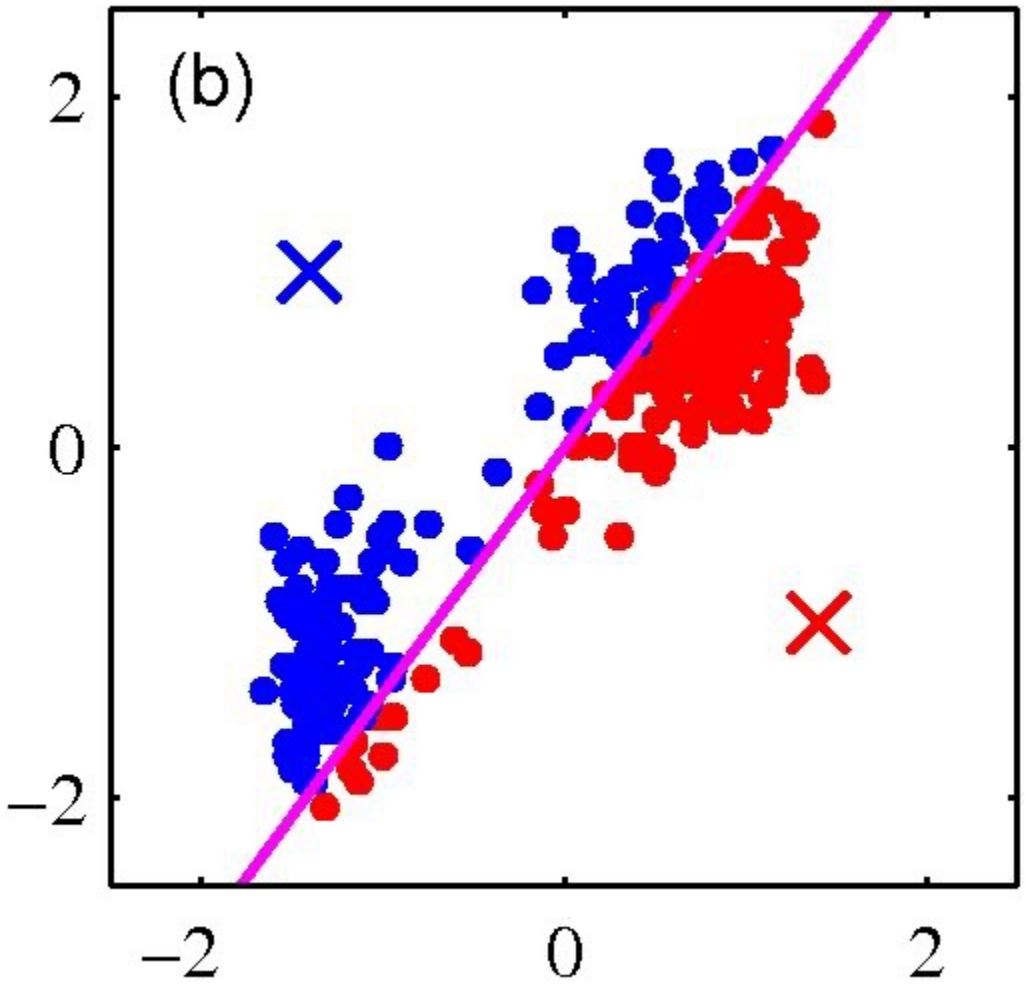
k-means clustering: Example



$k = 2$

Pick k random points as cluster centers (means)

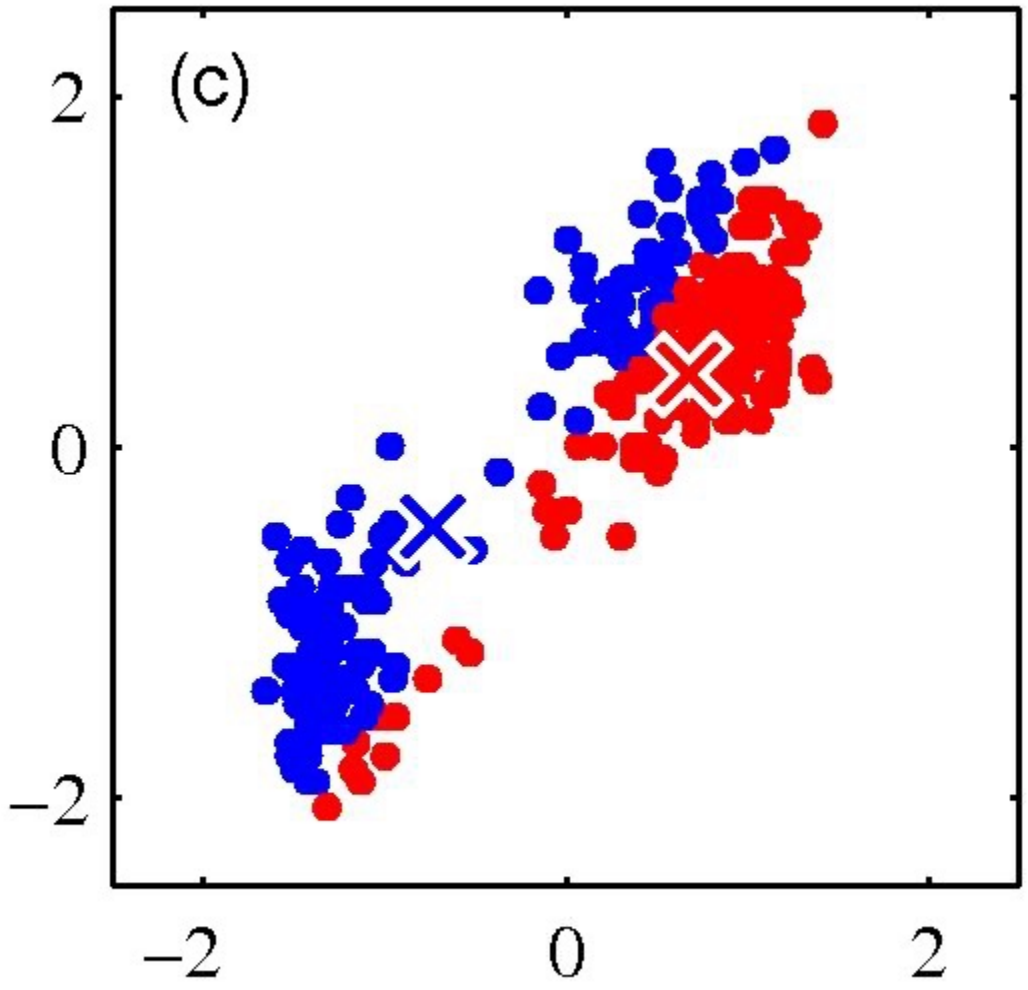
k-means clustering: Example



Iterative Step 1:

Assign data instances to closest cluster center

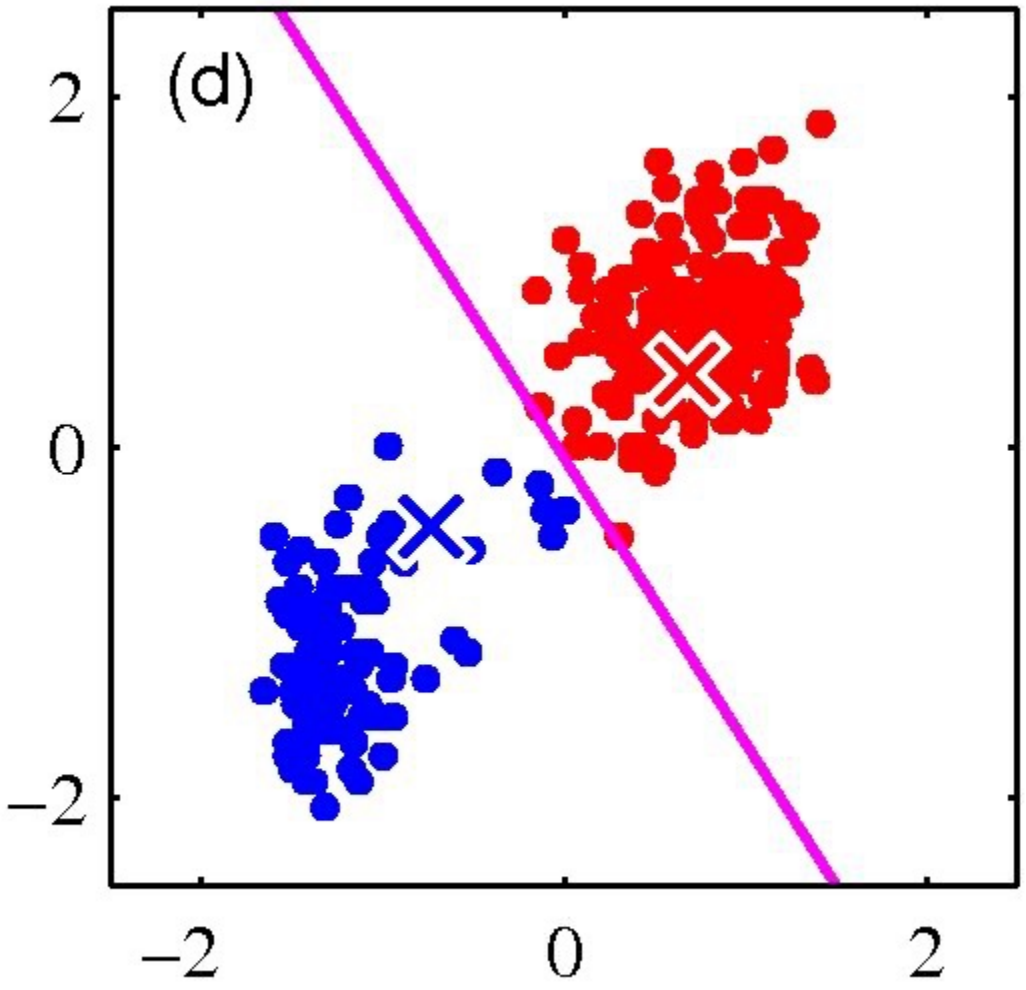
k-means clustering: Example



Iterative Step 2:

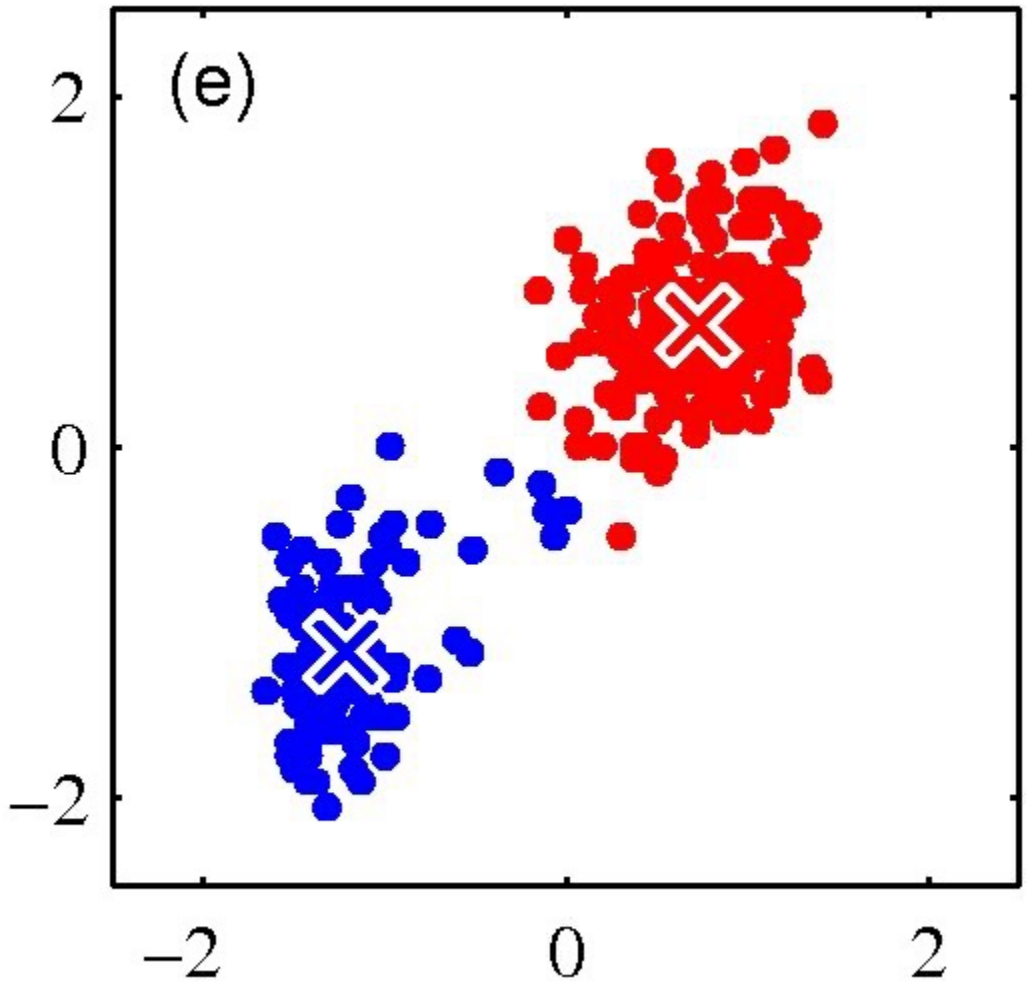
Change the cluster center to the average of the assigned points

k-means clustering: Example

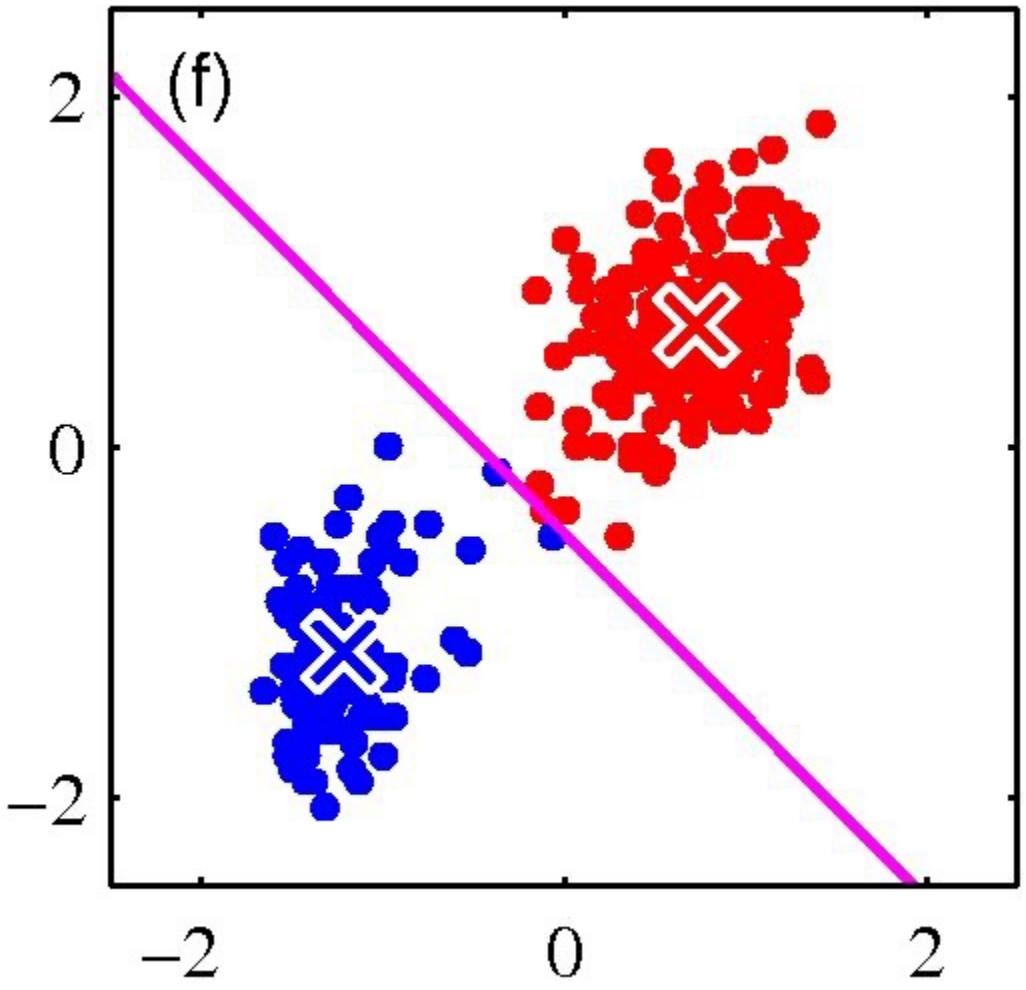


Repeat until convergence

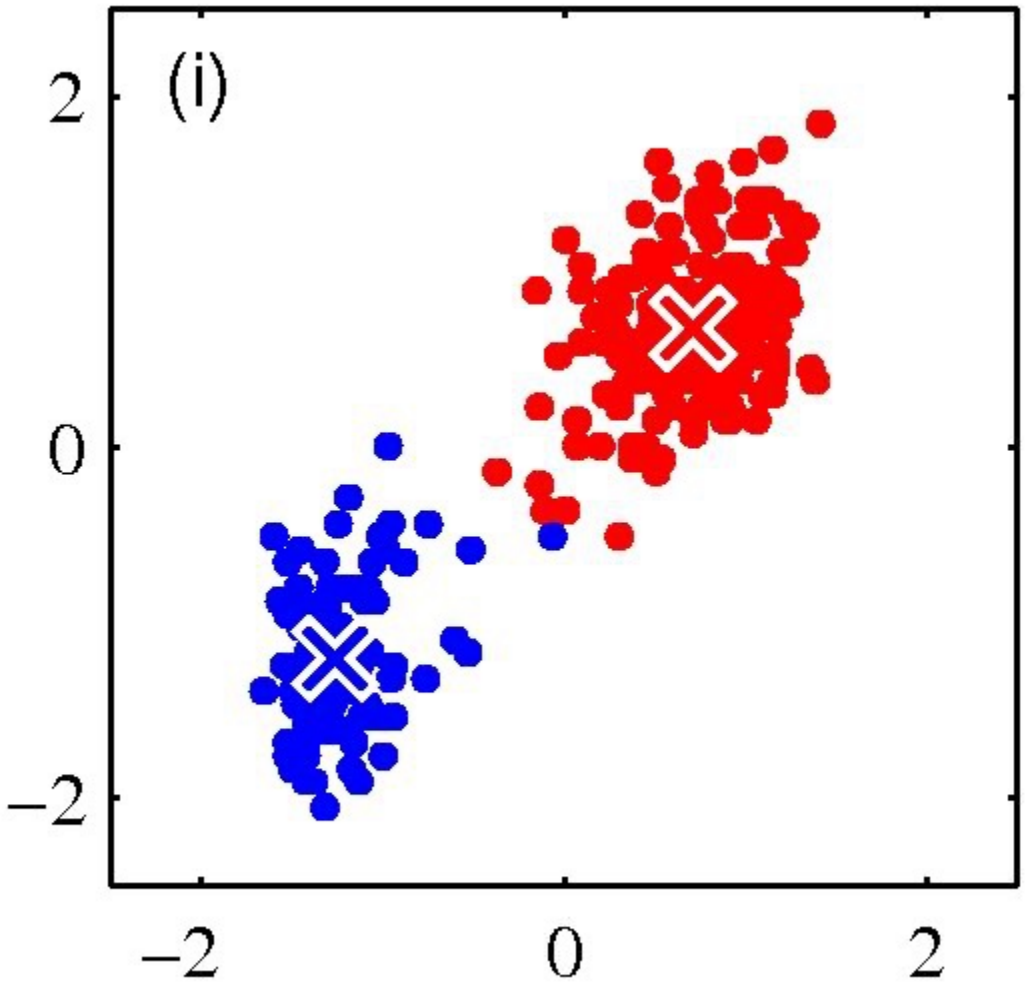
k-means clustering: Example



k-means clustering: Example



k-means clustering: Example



Proceed until Convergence

No Change to Assignment / Means

k-Means for Segmentation

$k = 2$

Goal of segmentation is to partition an image into regions, each of which has reasonably homogenous visual appearance

Original

k-Means for Segmentation

$k = 2$

$k = 3$

Original

k-Means for Segmentation

$k = 2$	$k = 3$	$k = 10$	Original
			
			

k-means Clustering as Optimization

- Minimize the distance of each input point to the mean of the cluster/partition that contains it

$$\min_{S_1,\ldots,S_k} \sum_{i=1}^{k} \sum_{j \in S_i} \left\| x^{(j)} - \mu_i \right\|^2$$

$S_1 \rightarrow$ cluster 1

$S_2 \rightarrow$ cluster 2

$\vdots$

$S_k \rightarrow$ cluster k.

where

- $S_i \subseteq \{1, \ldots, M\}$ is the i^{th} cluster
- $S_i \cap S_j = \emptyset$ for $i \neq j$, $\cup_i S_i = \{1, \ldots, n\}$
- μ_i is the centroid of the i^{th} cluster

k-means Clustering as Optimization

- Minimize the distance of each input point to the mean of the cluster/partition that contains it

$$\min_{S_1,\ldots,S_k} \sum_{i=1}^{k} \sum_{j \in S_i} \left\| x^{(j)} - \mu_i \right\|^2$$

where

- $S_i \subseteq \{1, \ldots, M\}$ is the i^{th} cluster
- $S_i \cap S_j = \emptyset$ for $i \neq j$, $\cup_i S_i = \{1, \ldots, n\}$
- μ_i is the centroid of the i^{th} cluster

Exactly minimizing this function is NP-hard (even for $k = 2$)

k-means Clustering

- The k-means clustering algorithm performs a block coordinate descent on the objective function

$$\sum_{i=1}^{k} \sum_{j \in S_i} \left\| x^{(j)} - \mu_i \right\|^2$$

- This is not a convex function: could get stuck in local minima

k-Means as Optimization

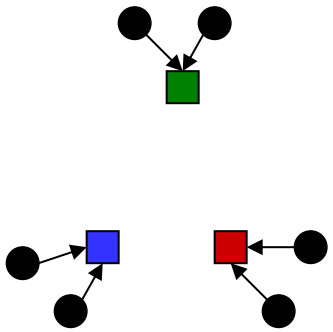
- Consider the k-means objective function

$$\phi(x, S, \mu) = \sum_{i=1}^{k} \sum_{j \in S_i} \left\| x^{(j)} - \mu_i \right\|^2$$

points

cluster assignments

cluster means



- Two stages each iteration
  - Update cluster assignments: fix means μ, change assignments S
  - Update means: fix assignments S, change means μ

$$\min_{X,Y} F(X,Y)$$

For $t = 1:T$

$$X^{t+1} \leftarrow \min_{X} F(X, Y^{t})$$

$$Y^{t+1} \leftarrow \min_{Y} F(X^{t+1}, Y)$$

Phase I: Update Assignments

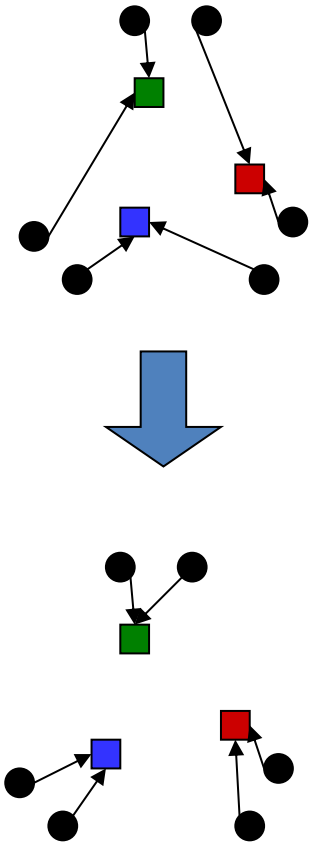
- For each point, re-assign to closest mean, $x^{(j)} \in S_i$ if

$$j \in \arg\min_{i=1,\dots,k} \left\| x^{(j)} - \mu_i \right\|^2$$

- Can only decrease ϕ as the sum of the distances of all points to their respective means must decrease

$$\phi(x, S, \mu) = \sum_{i=1}^{k} \sum_{j \in S_i} \left\| x^{(j)} - \mu_i \right\|^2$$

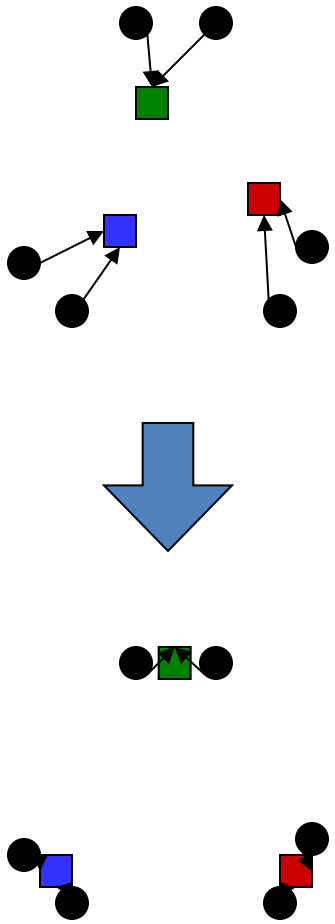
$\mu_1, \dots, \mu_k$



Phase II: Update Means

- Move each mean to the average of its assigned points

$$\mu_i = \sum_{j \in S_i} \frac{x^{(j)}}{|S_i|}$$ ← Average



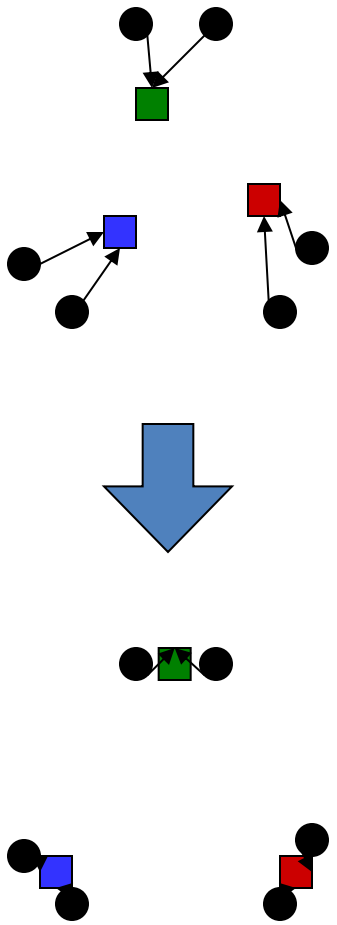
- Also can only decrease total distance…
  - Why?

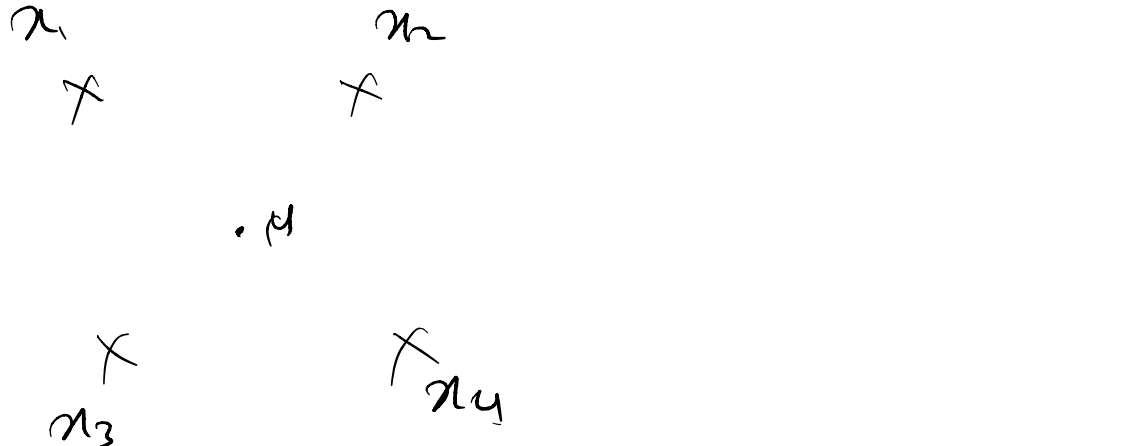
Phase II: Update Means

- Move each mean to the average of its assigned points

$$\mu_i = \sum_{j \in S_i} \frac{x^{(j)}}{|S_i|}$$

- Also can only decrease total distance…
  - The point y with minimum squared Euclidean distance to a set of points is their mean (Average)





$$\min_{\mu} \sum_{i=1}^{n} \|x^{(i)} - \mu\|^2 = F(\mu)$$

$$\frac{\partial F(\mu)}{\partial \mu} = \sum_{i=1}^{n} 2\left(x^{(i)} - \mu\right) = 0$$

$$\Rightarrow n\mu = \sum_{i=1}^{n} x^{(i)}$$

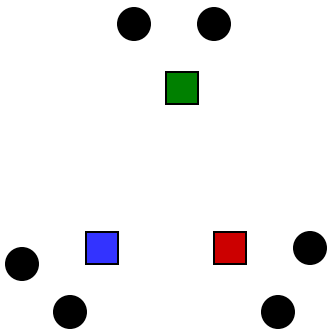
$$\Rightarrow \mu = \frac{\sum_{i=1}^{n} x^{(i)}}{n}$$

Initialization

- K-means is sensitive to initialization
  - It does matter what you pick!
  - What can go wrong?

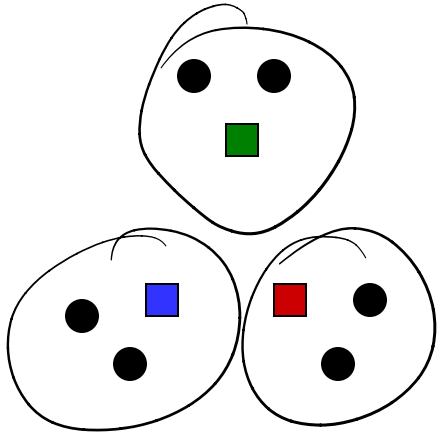
Initialization

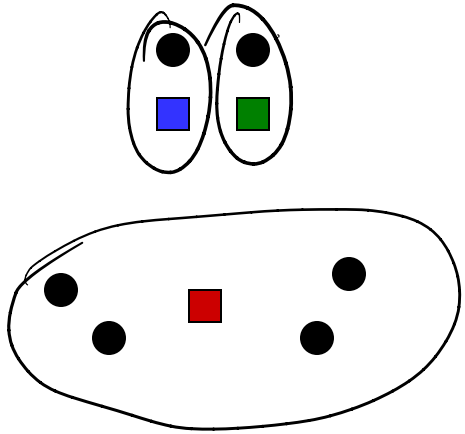
- K-means is sensitive to initialization
  - It does matter what you pick!
  - What can go wrong?



Initialization

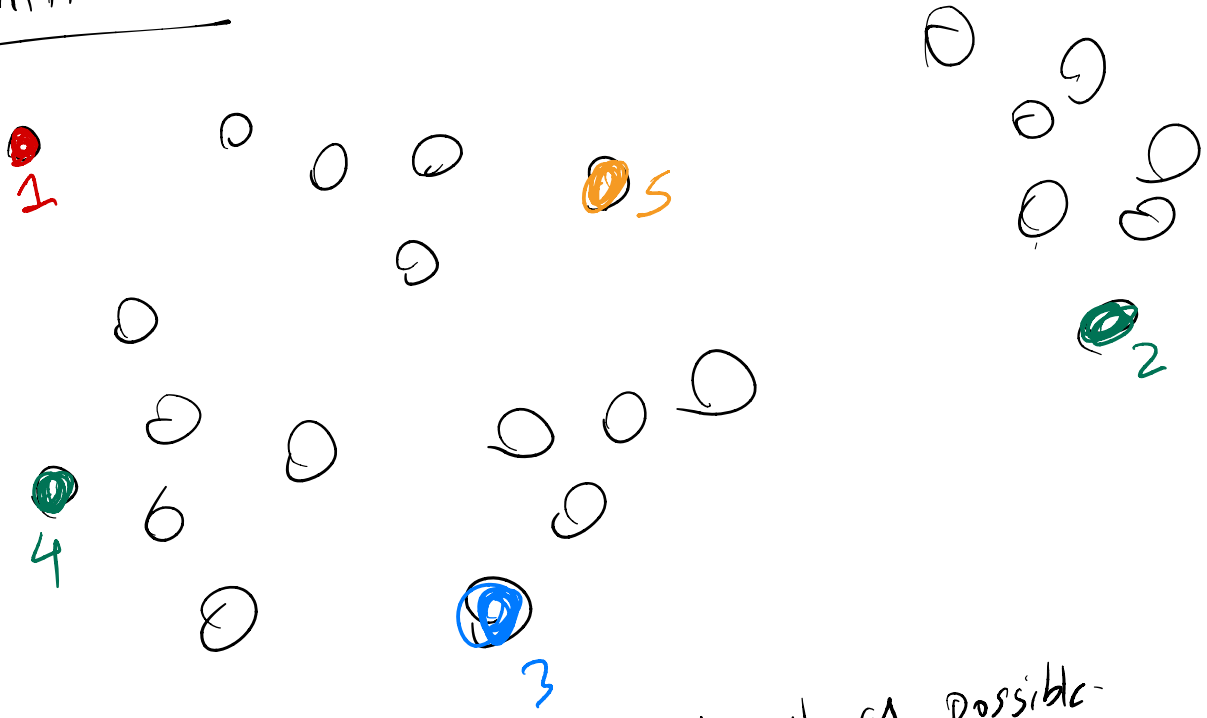
- K-means is sensitive to initialization
  - It does matter what you pick!
  - What can go wrong?
    - Various schemes to help alleviate this problem: initialization heuristics





k-means ++

Initialize

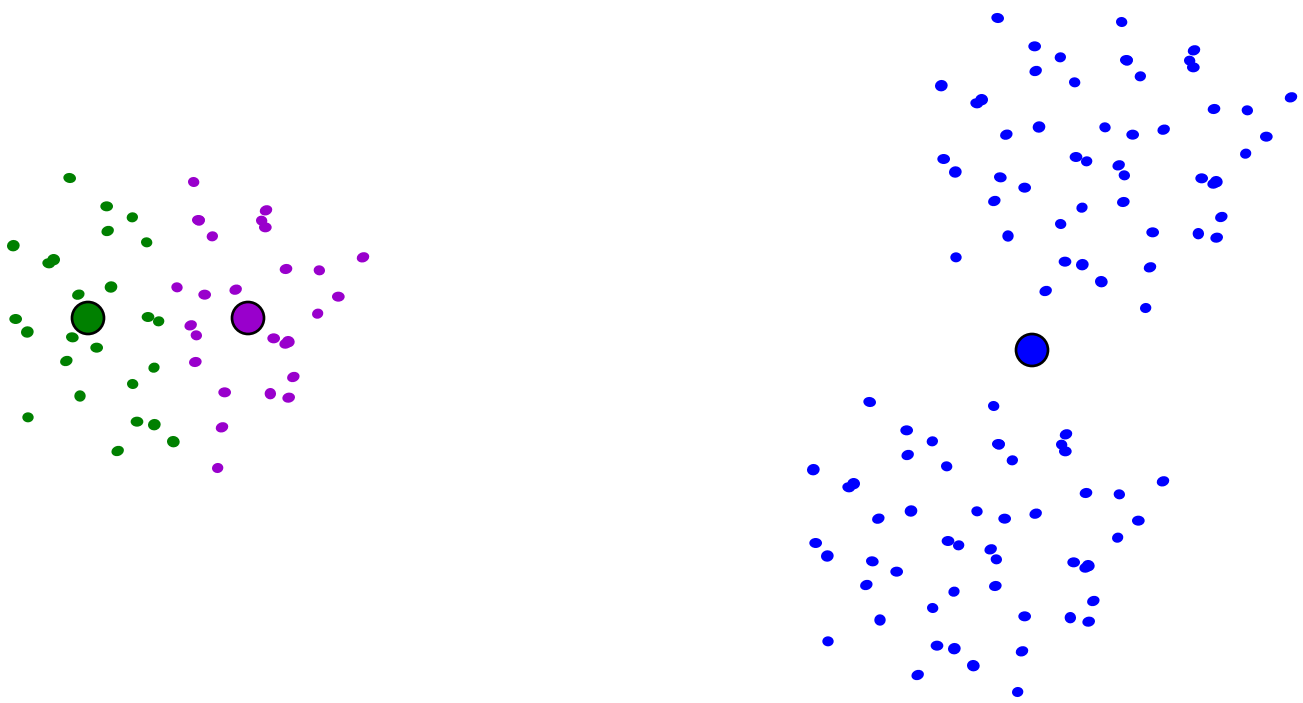


Greedy: Find points as spread out as possible.

k-means Clustering

- Not clear how to figure out the "best" k in advance
- Want to choose k to pick out the interesting clusters, but not to overfit the data points
  - Large k doesn't necessarily pick out interesting clusters
  - Small k can result in large clusters than can be broken down further

Local Optima



k-Means Summary

M = # Data
n = # Features
k = # Clusters

- Guaranteed to converge
  - But not to a global optimum
- Choice of k and initialization can greatly affect the outcome
- Runtime: $O(kMn)$ per iteration
- Popular because it is fast, though there are other clustering methods that may be more suitable depending on your data

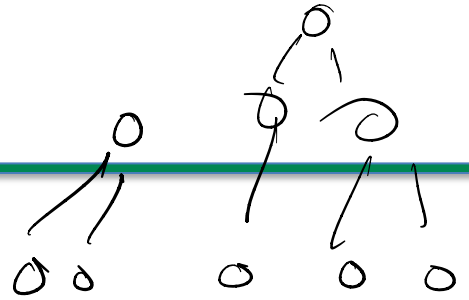
- Normalization of Features/Attributes
  ↑
  scaling

K-mediods Clustering and Extensions

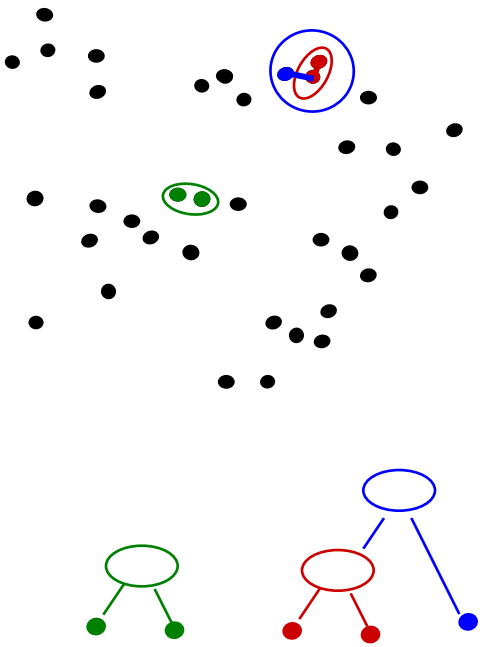
- Very similar to k-means, except that the centroid is one of the examples from the cluster
- The update means step therefore involves finding the point within the cluster which has the minimum average distance to the rest

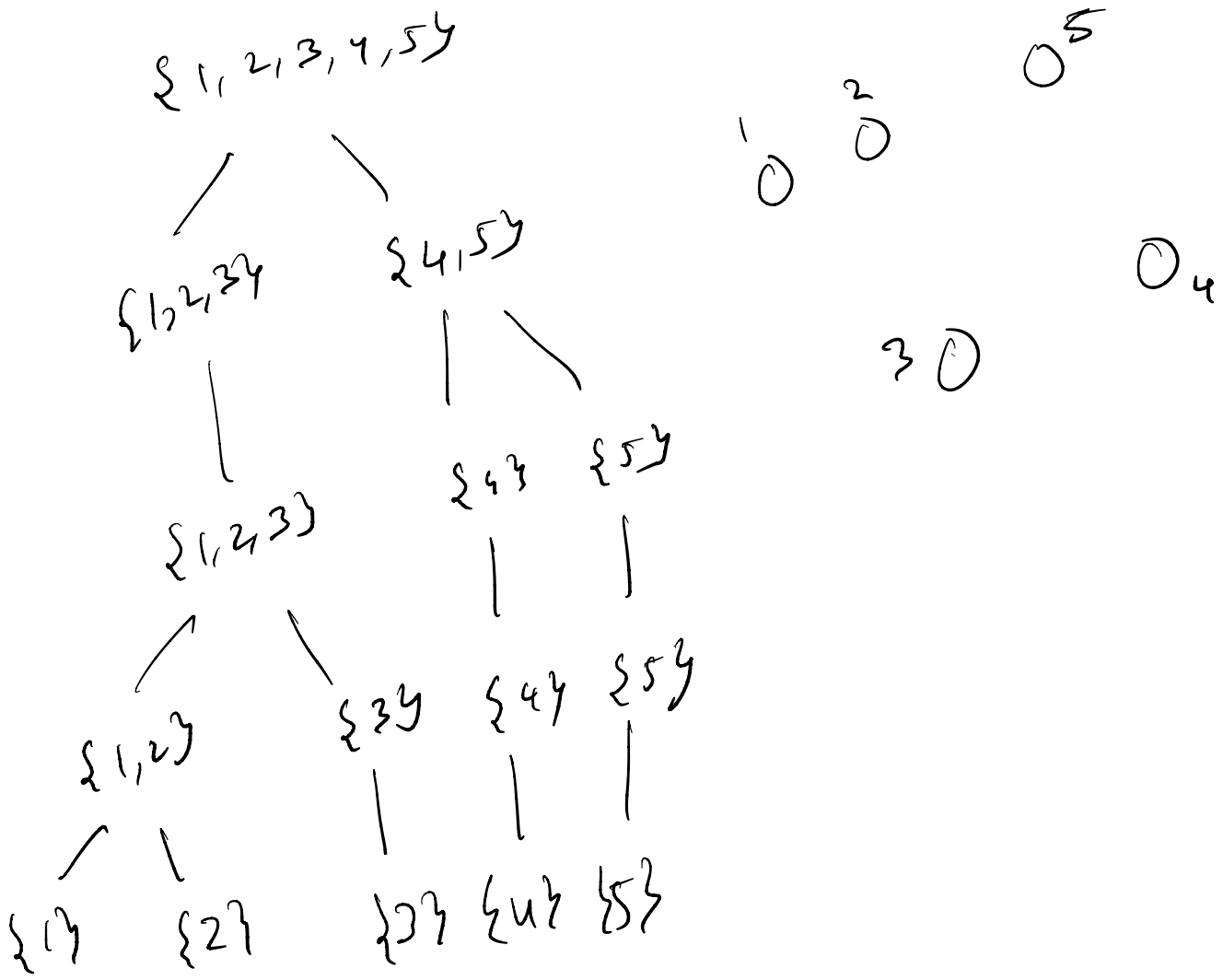
- Extensions of k-means: Other distance measures, e.g. the Bregman divergence

Hierarchical Clustering



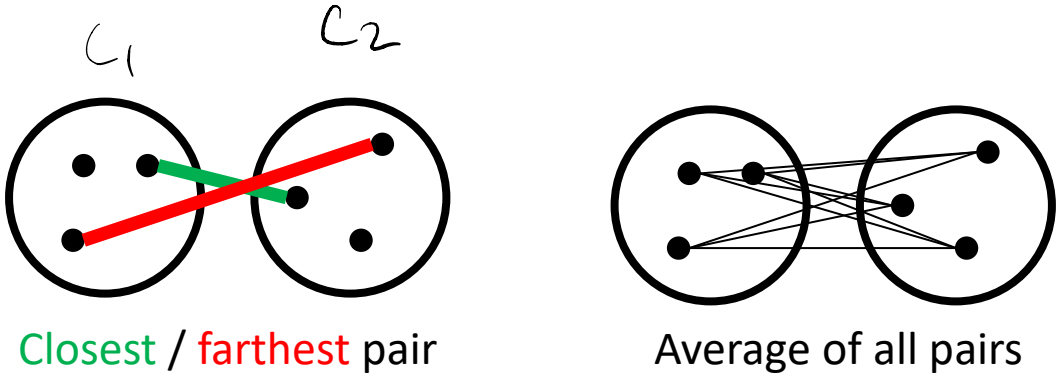
- Agglomerative clustering
  - Incrementally build larger clusters out of smaller clusters
- Algorithm:
  - Maintain a set of clusters
  - Initially, each instance in its own cluster
  - Repeat:
    - Pick the two closest clusters
    - Merge them into a new cluster
    - Stop when there is only one cluster left
- Produces not one clustering, but a family of clusterings represented by a dendrogram




{1,2,3,4,5}
{1,2,3}
{4,5}
{1,2}
{3}
{4}
{5}
{1}
{2}
1
2
3
4
5

Agglomerative Clustering

- How should we define "closest" for clusters with multiple elements?



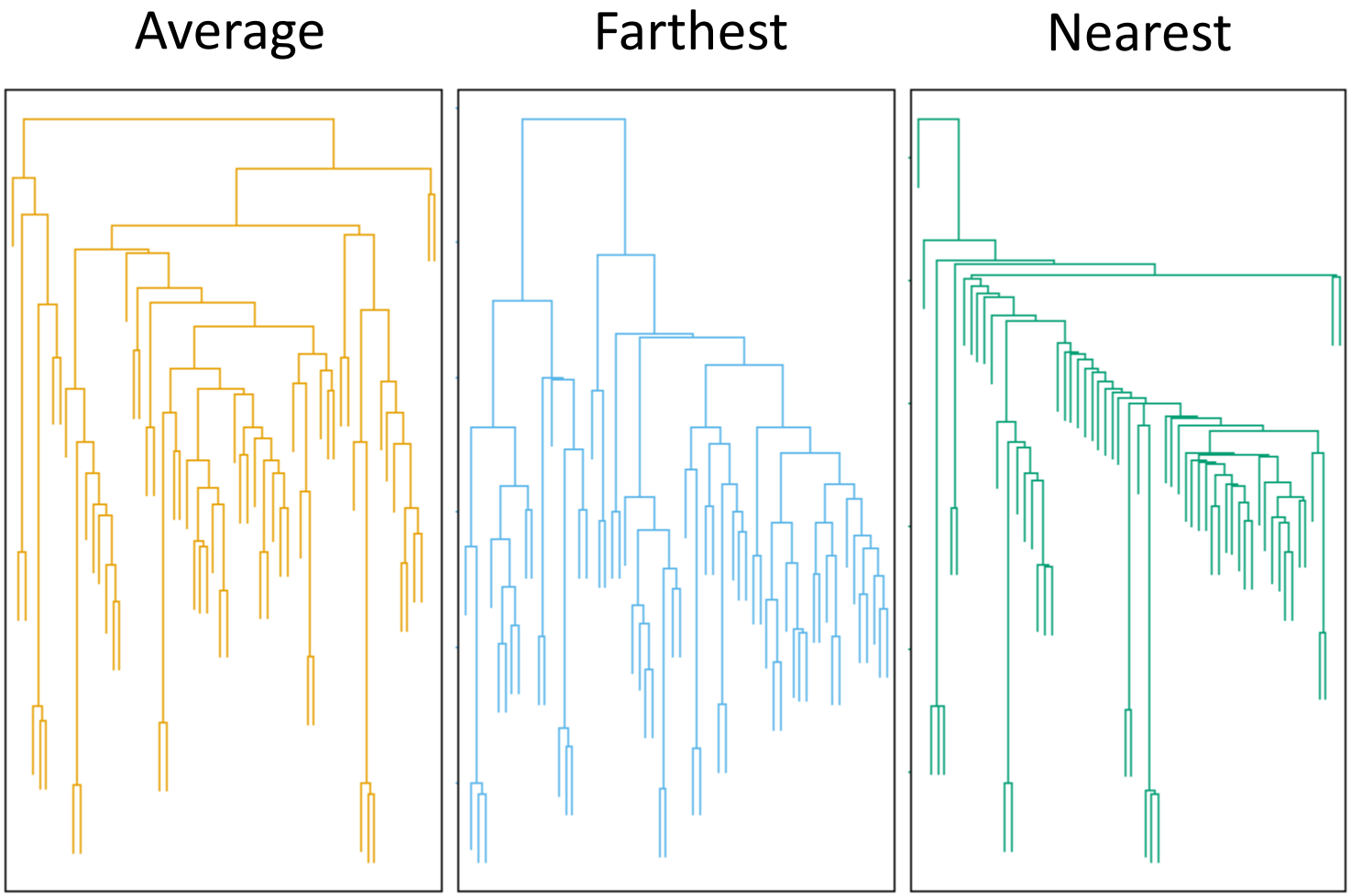
Closest / farthest pair

Average of all pairs

- Many more choices, each produces a different clustering...

$d(C_1, C_2)$

Clustering Behavior



Mouse tumor data from [Hastie]