

Unsupervised Learning: Clustering

Nicholas Ruozzi

University of Texas at Dallas

Announcements



- Midterm (next Wednesday in class)
 - Closed book, closed notes, etc. (just you and a pencil)
 - Try to arrive as early as possible so as to maximize your exam taking time
 - Covers everything up to the **end of boosting**
 - Be prepared for theoretical questions! A practice exam will be made available on eLearning.
 - The exam is worth a significant percentage of the grade, talk to other students and use Piazza to make sure that you are prepared!

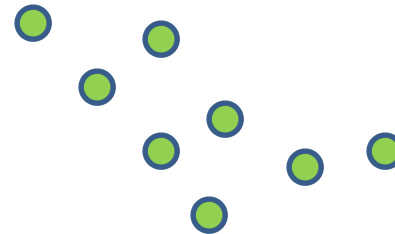
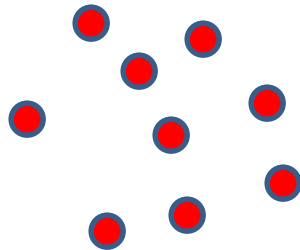
Clustering systems:

- **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

- 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



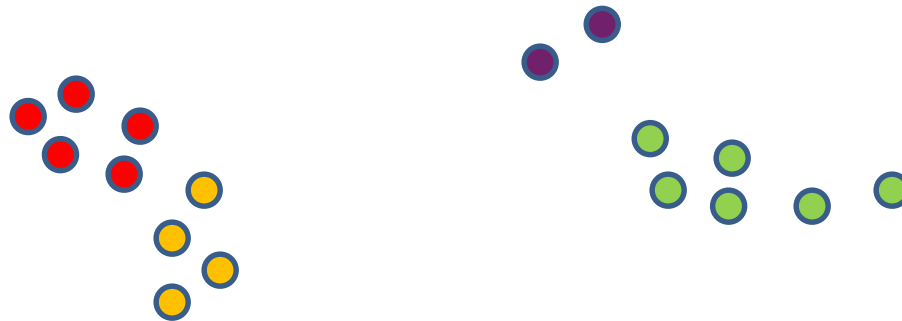
- Want to group together parts of a dataset that are close together in some metric
- Useful for finding the important parameters/features of a dataset



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



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



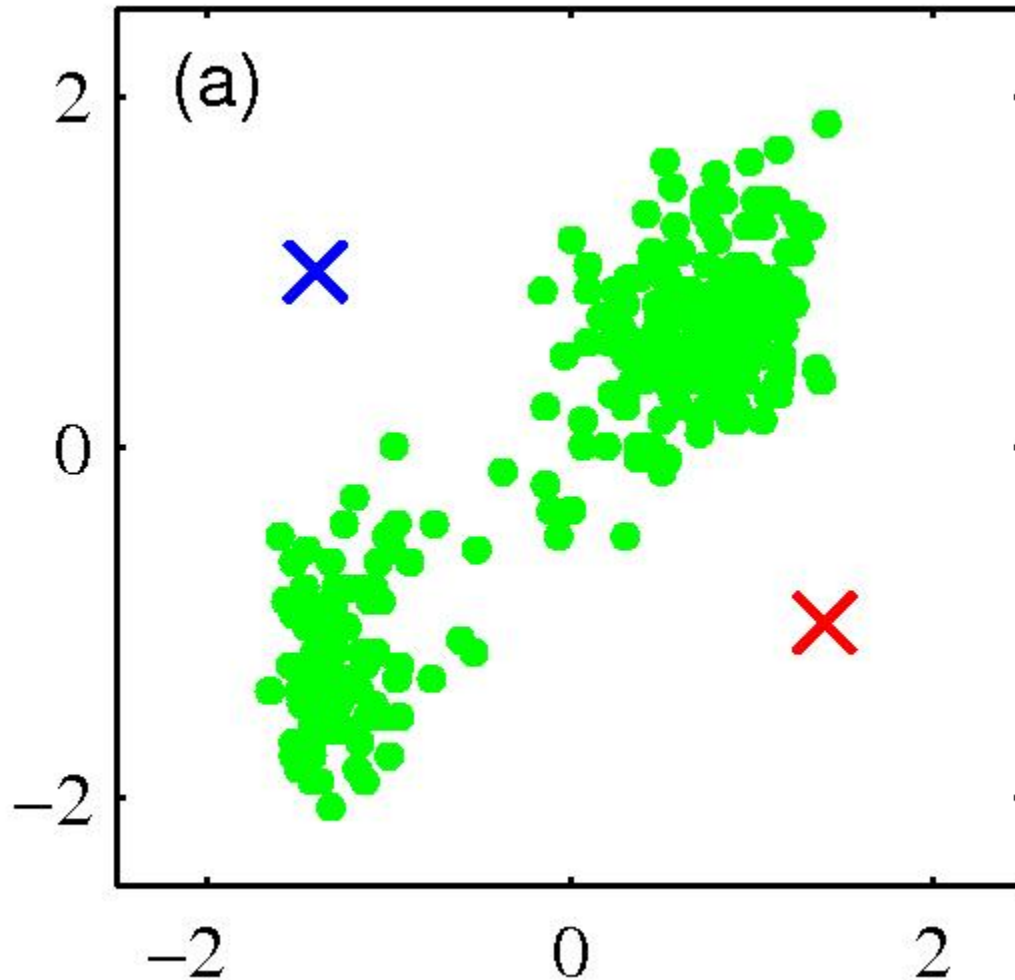
- Input: a collection of points $x^{(1)}, \dots, x^{(m)} \in \mathbb{R}^n$, an integer k
- Output: A partitioning of the input points into k sets that minimizes some metric of closeness

k -means Clustering



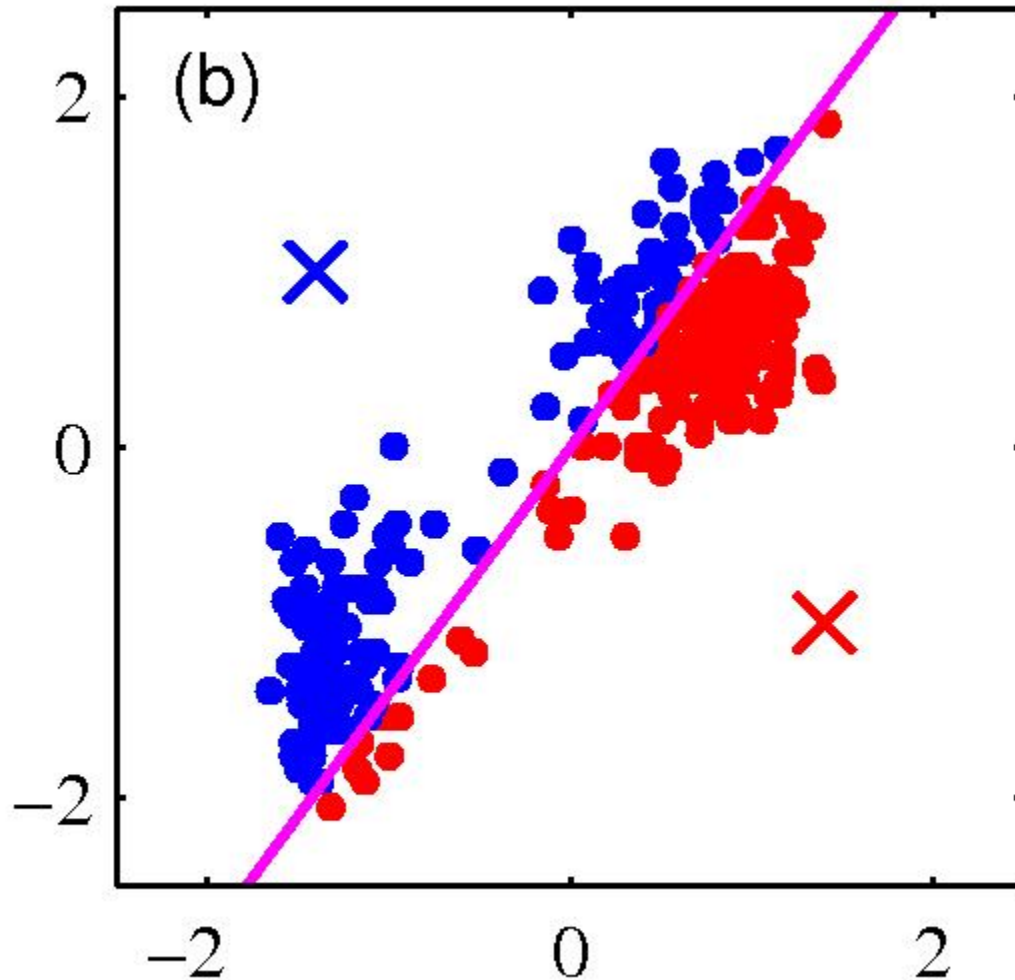
- Pick an initial set of k means (usually at random)
- Repeat until the clusters do not change:
 - Partition the data points, assigning each data point to a cluster based on the mean that is closest to it
 - Update the cluster means so that the i^{th} mean is equal to the average of all data points assigned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

k -means clustering: Example



Pick k random points
as cluster centers
(means)

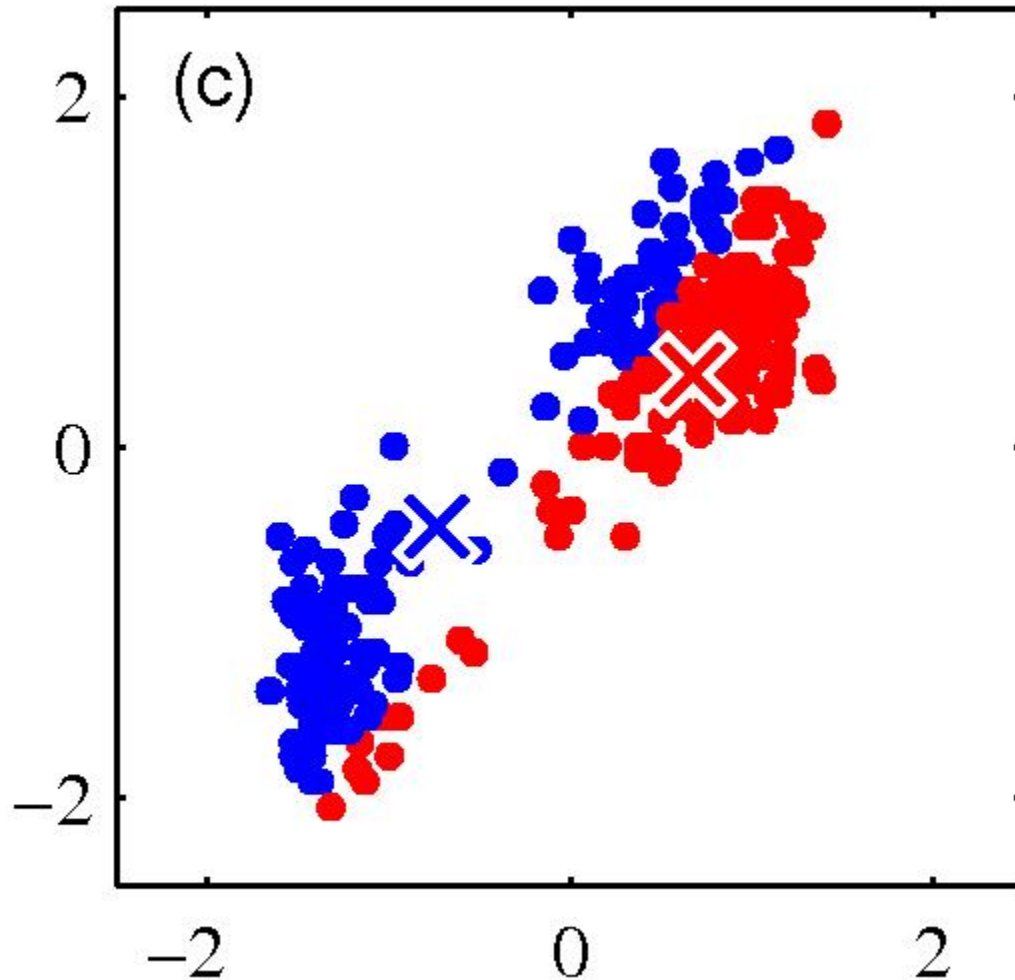
k -means clustering: Example



Iterative Step 1:

Assign data instances
to closest cluster
center

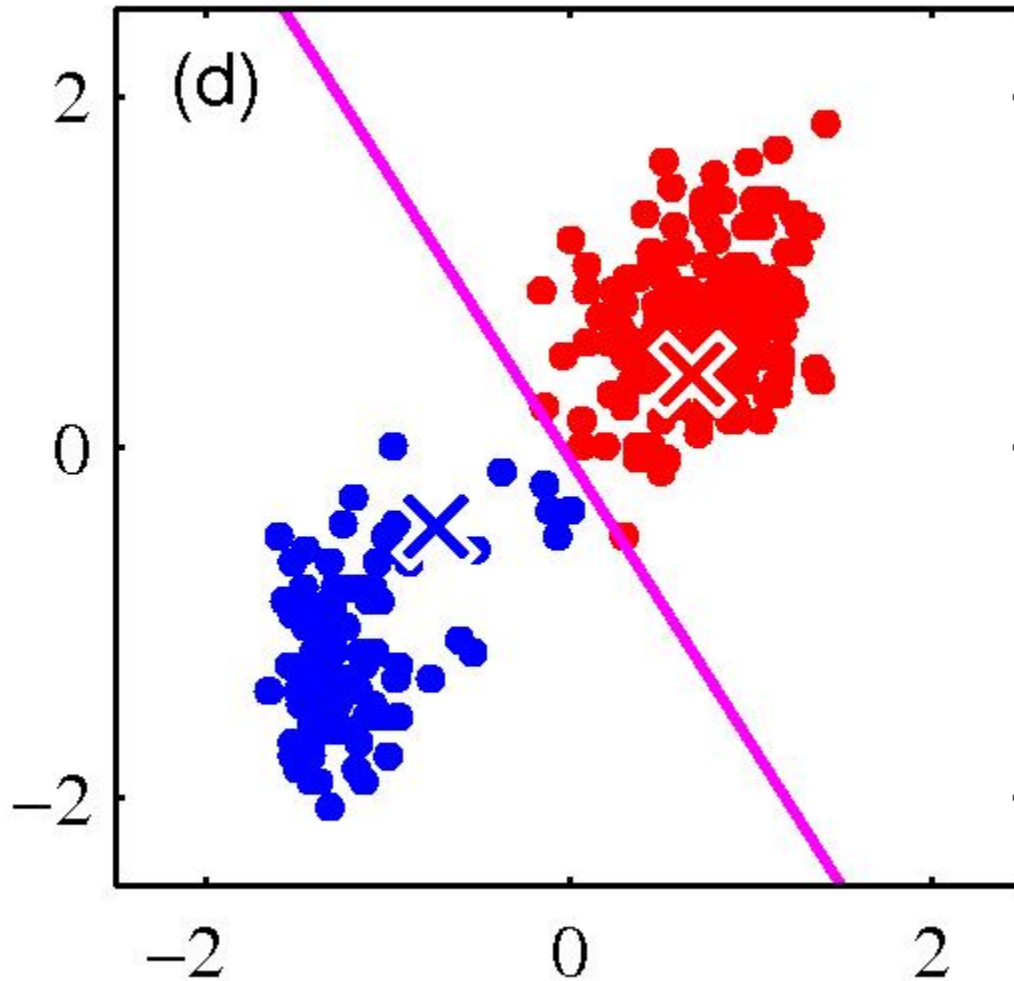
k -means clustering: Example



Iterative Step 2:

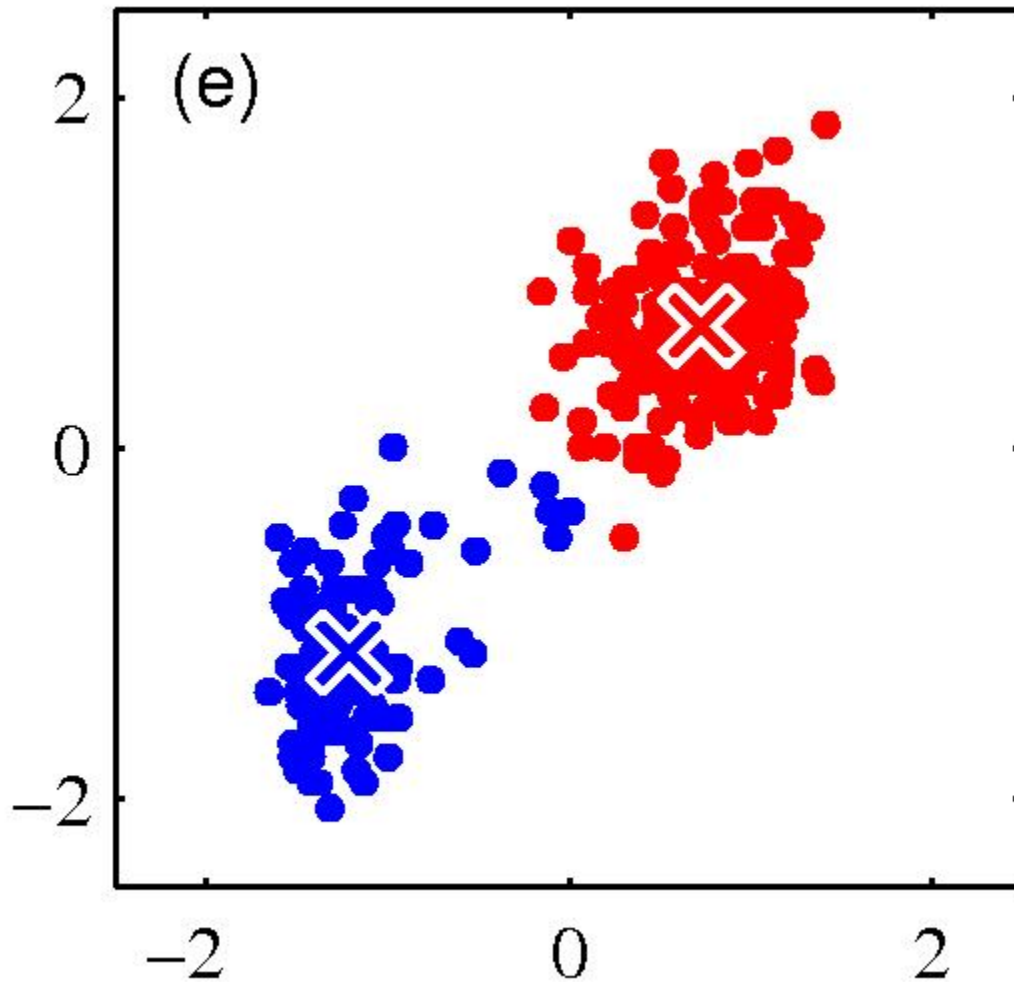
Change the cluster center to the average of the assigned points

k -means clustering: Example

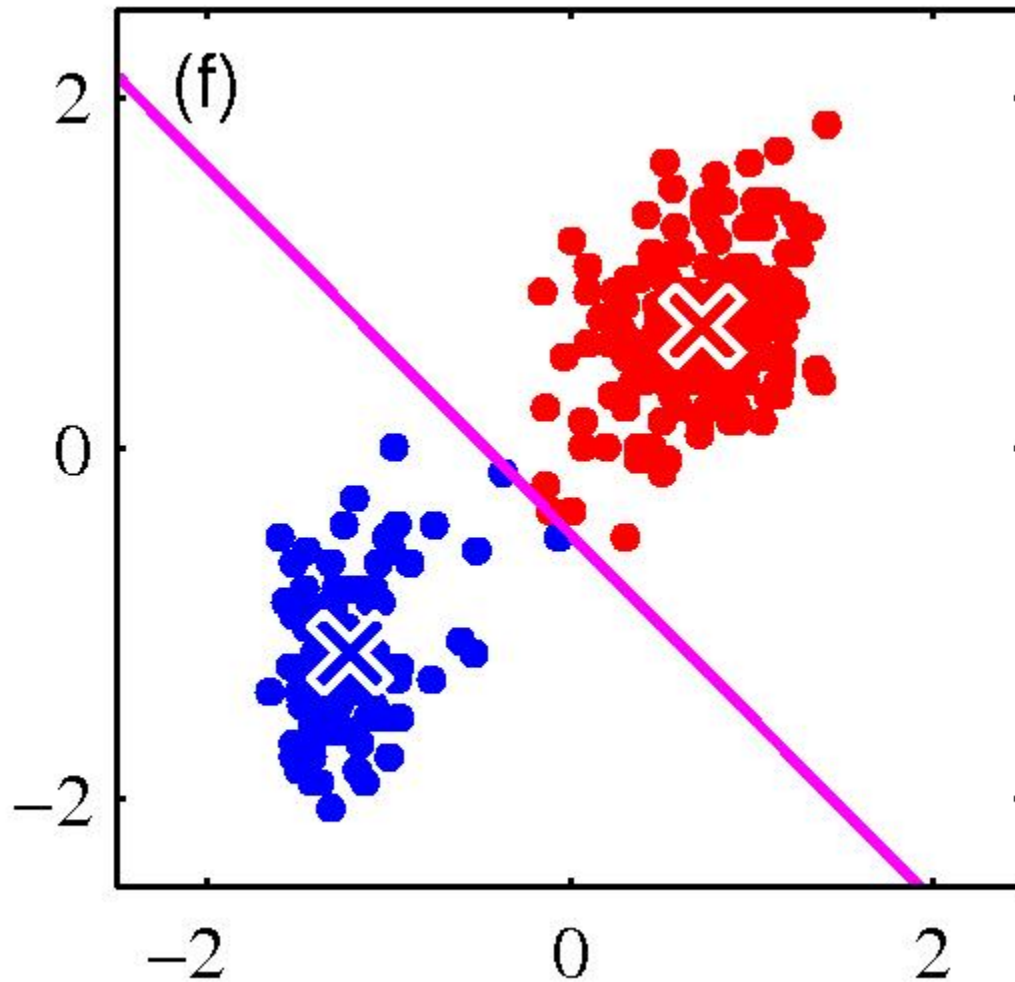


Repeat until
convergence

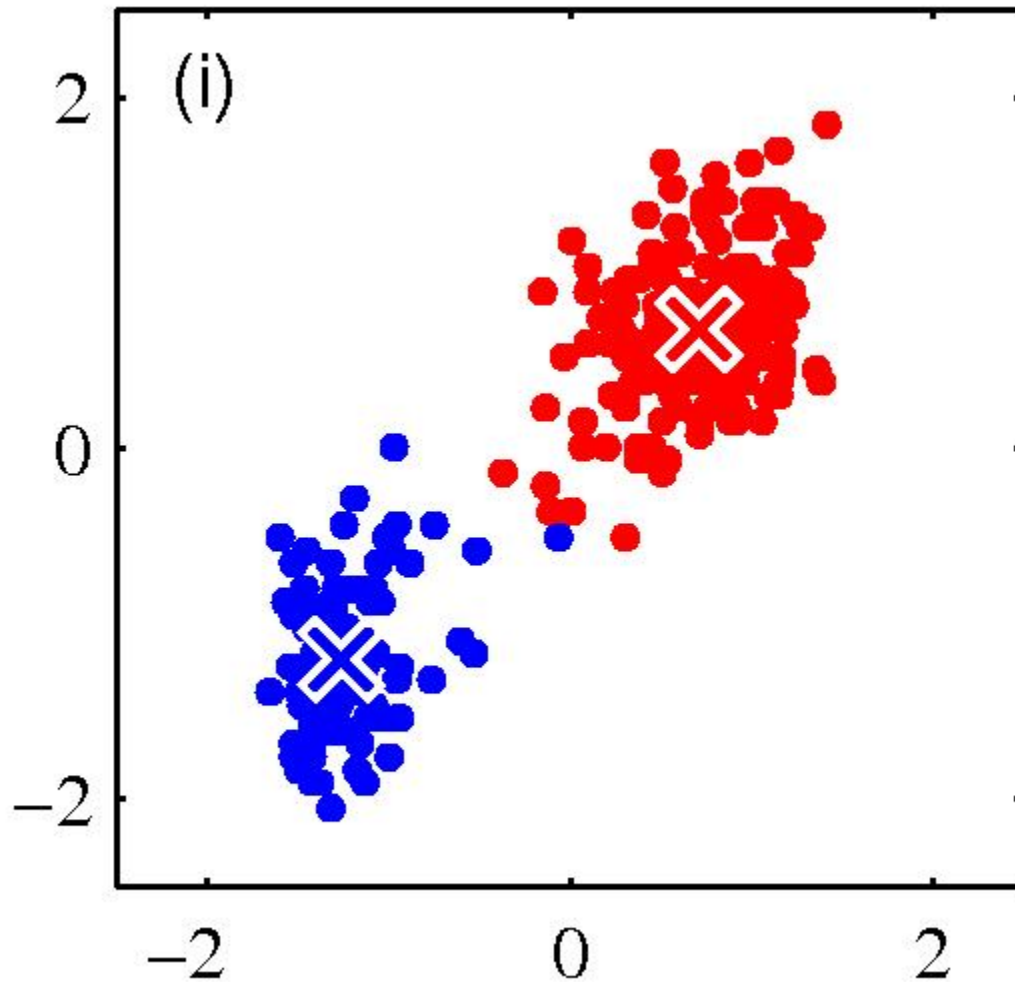
k -means clustering: Example



k -means clustering: Example



k -means clustering: Example



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



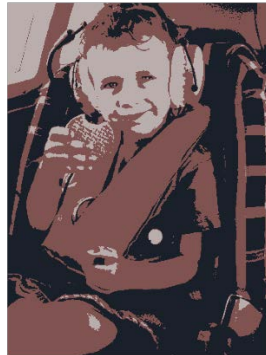
$k = 2$



$k = 3$



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

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

where

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

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

where

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

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

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

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

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

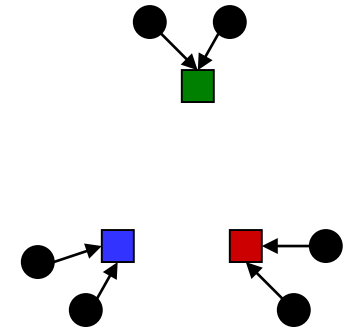
k -Means as Optimization



- Consider the k -means objective function

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

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

Phase I: Update Assignments

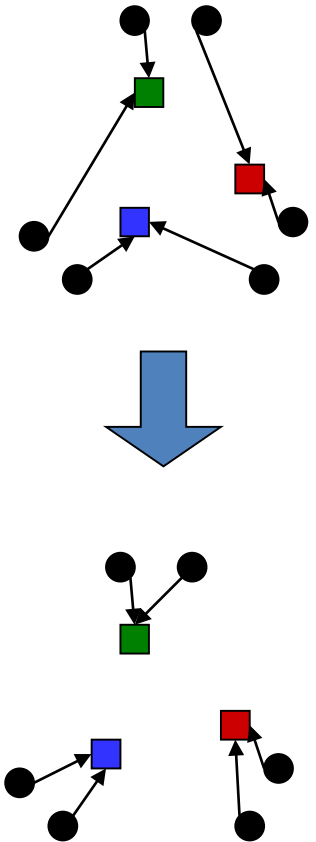


- For each point, re-assign to closest mean, $x^{(j)} \in S_i$ if

$$j \in \arg \min_i \|x^{(j)} - \mu_i\|^2$$

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

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



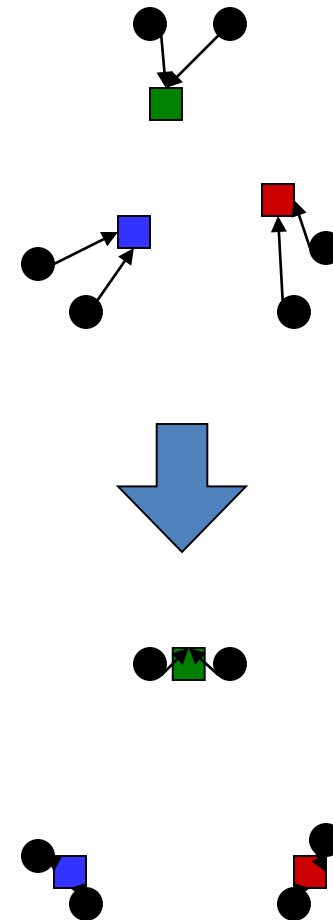
Phase II: Update Means



- 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...
 - Why?



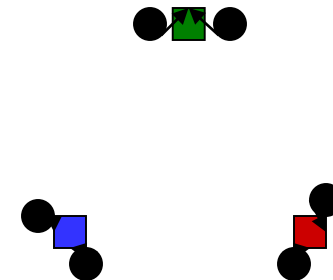
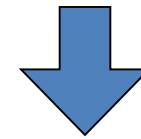
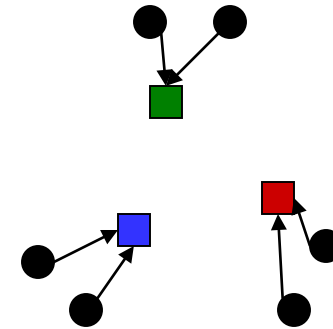
Phase II: Update Means



- Move each mean to the average of its assigned points

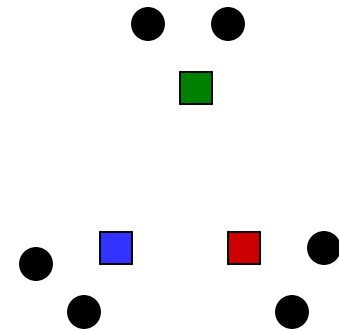
$$\mu_i = \sum_{j \in S_i} \frac{x^{(j)}}{|S_i|}$$

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

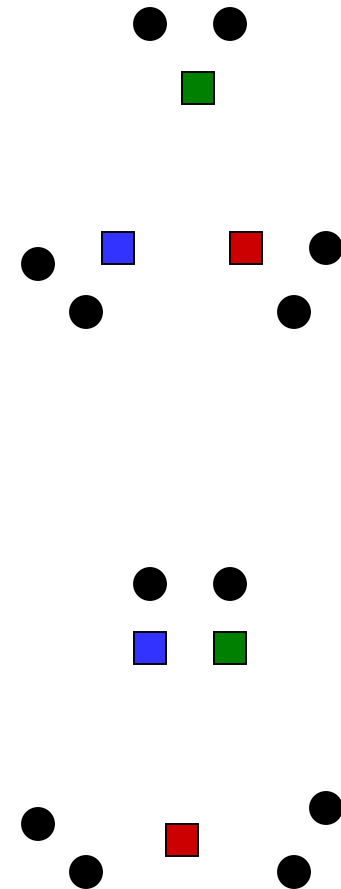


- K-means is sensitive to initialization
 - It does matter what you pick!
 - What can go wrong?

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



- 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

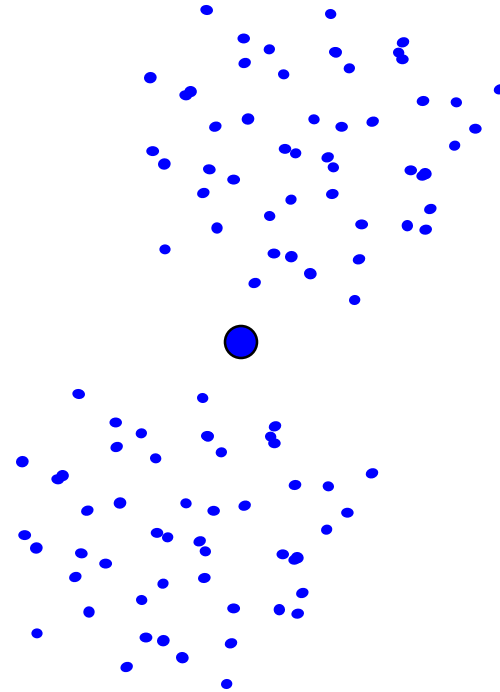
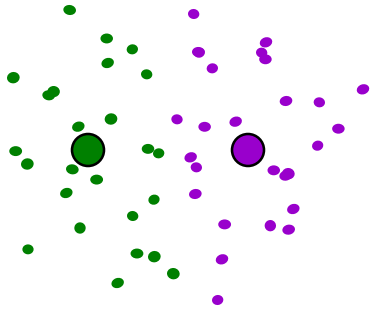


k -means Clustering



- 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

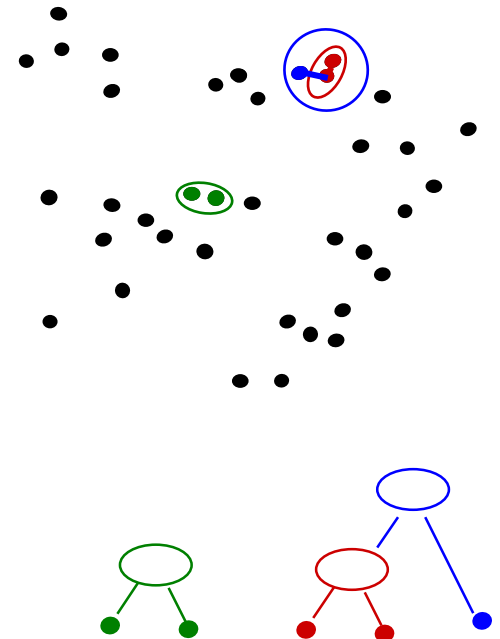


- Guaranteed to converge
 - But not to a global optimum
- Choice of k and initialization can greatly affect the outcome
- Runtime: $O(kM)$ per iteration
- Popular because it is fast, though there are other clustering methods that may be more suitable depending on your data

Hierarchical Clustering



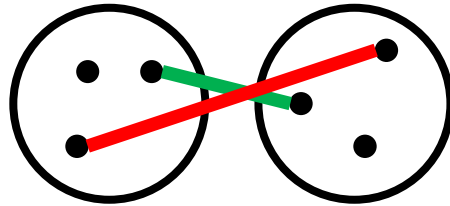
- Agglomerative clustering
 - Incrementally build larger clusters out of smaller clusters
- Algorithm:
 - Maintain a set of clusters
 - Initially, each instance in its own cluster
 - Repeat:
 - Pick the two closest clusters
 - Merge them into a new cluster
 - Stop when there is only one cluster left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



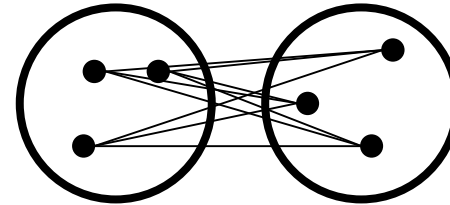
Agglomerative Clustering



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



Closest / farthest pair



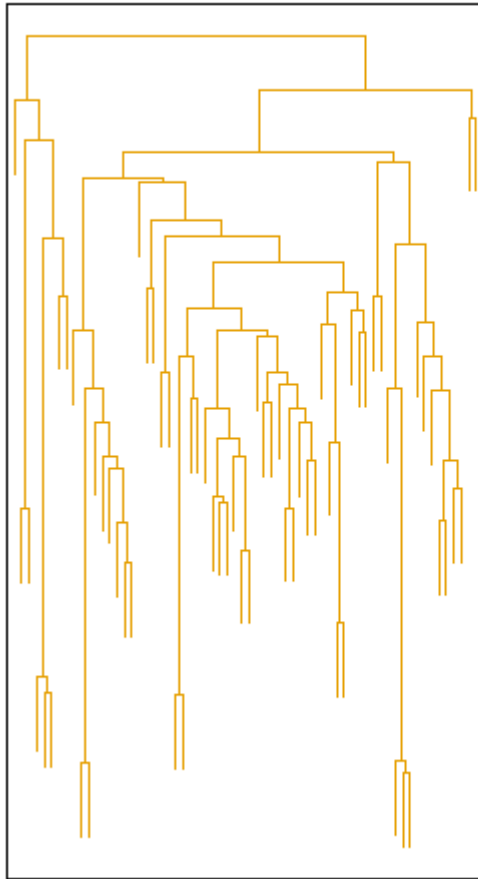
Average of all pairs

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

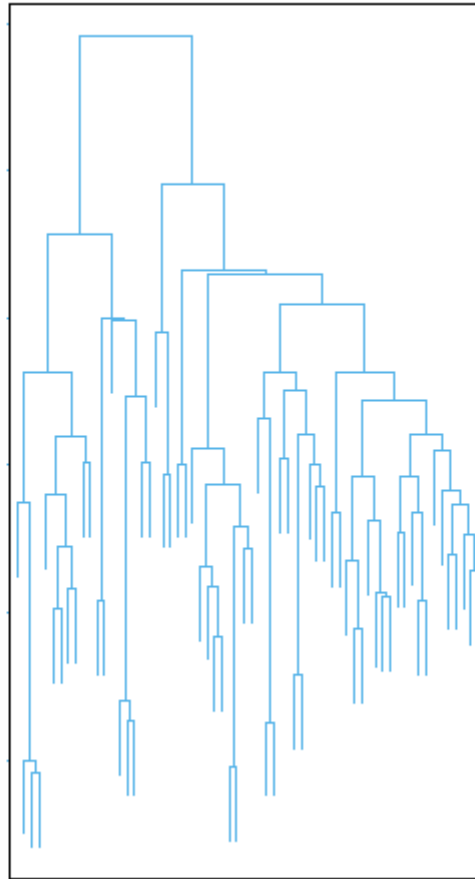
Clustering Behavior



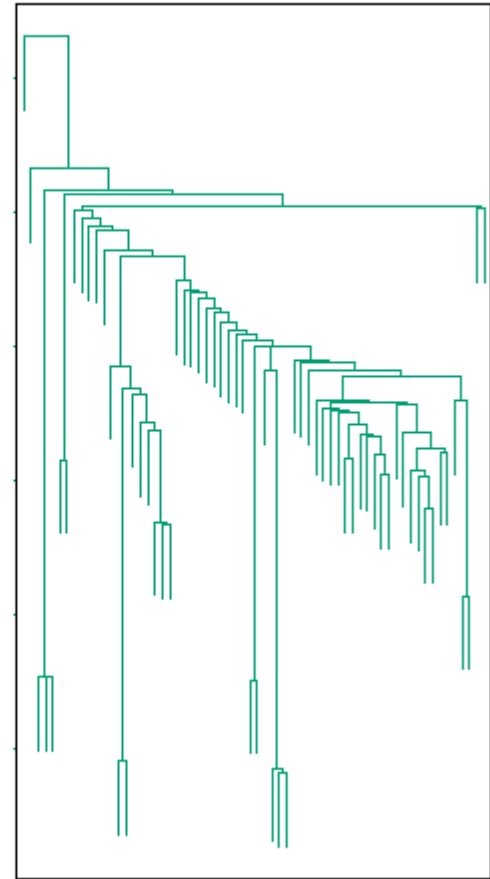
Average



Farthest



Nearest



Mouse tumor data from [Hastie]