Semi-supervised Learning

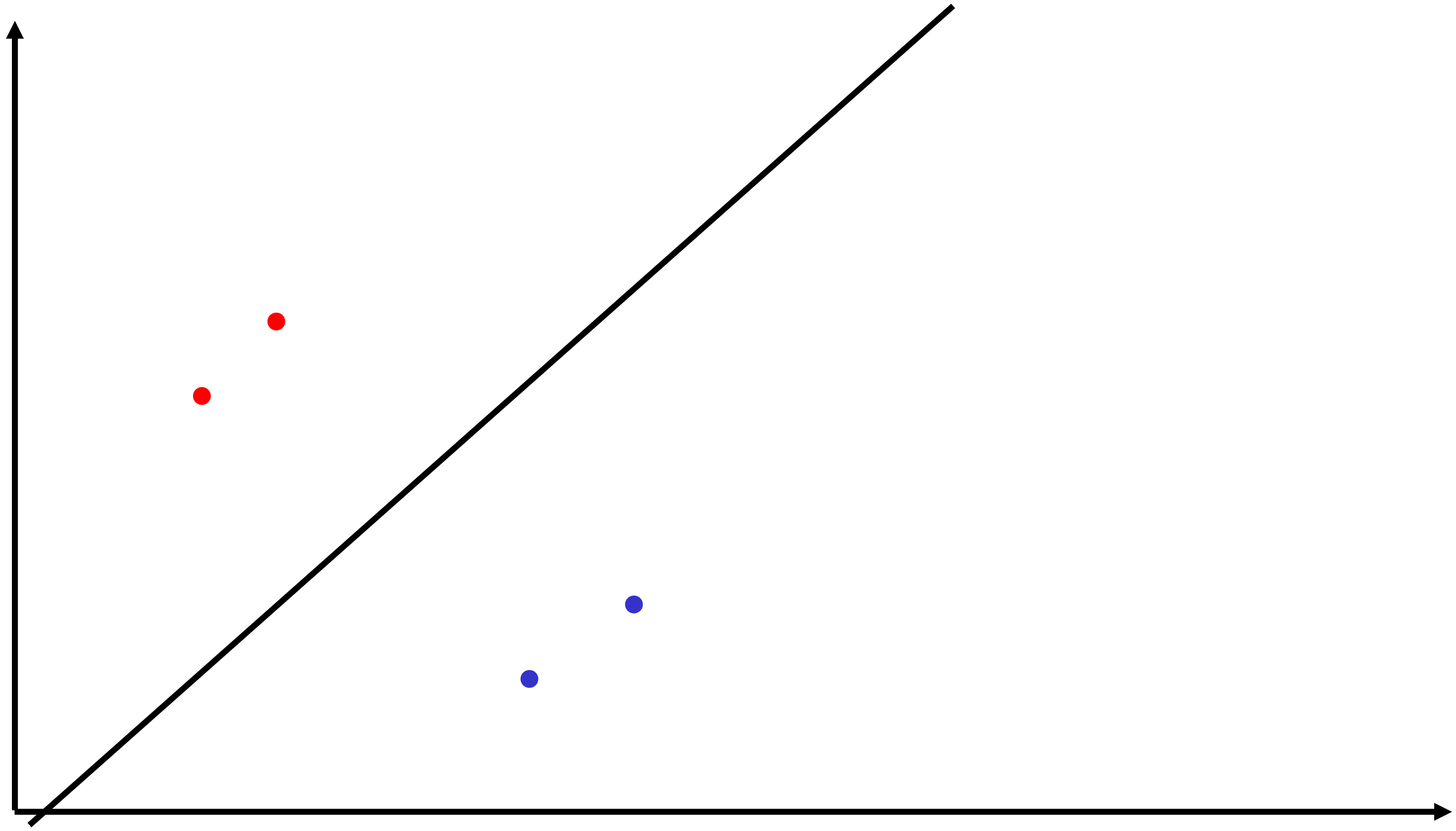
CS 145

Fall 2015

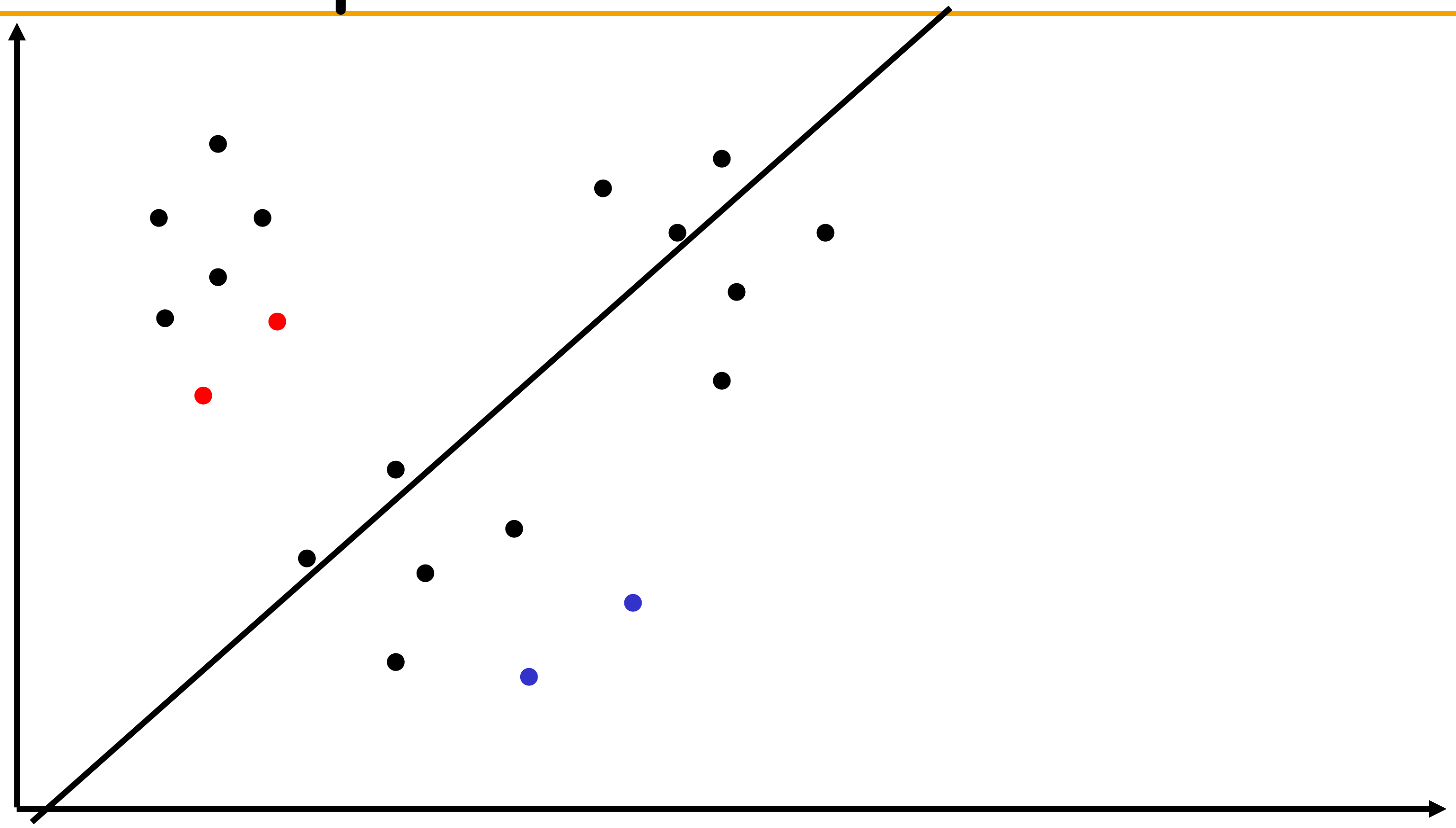
Overview

- Semi-supervised learning
  - Semi-supervised clustering
  - Semi-supervised classification
- Semi-supervised clustering
  - Search based methods
    - Cop K-mean
    - Seeded K-mean
    - Constrained K-mean
  - Similarity based methods

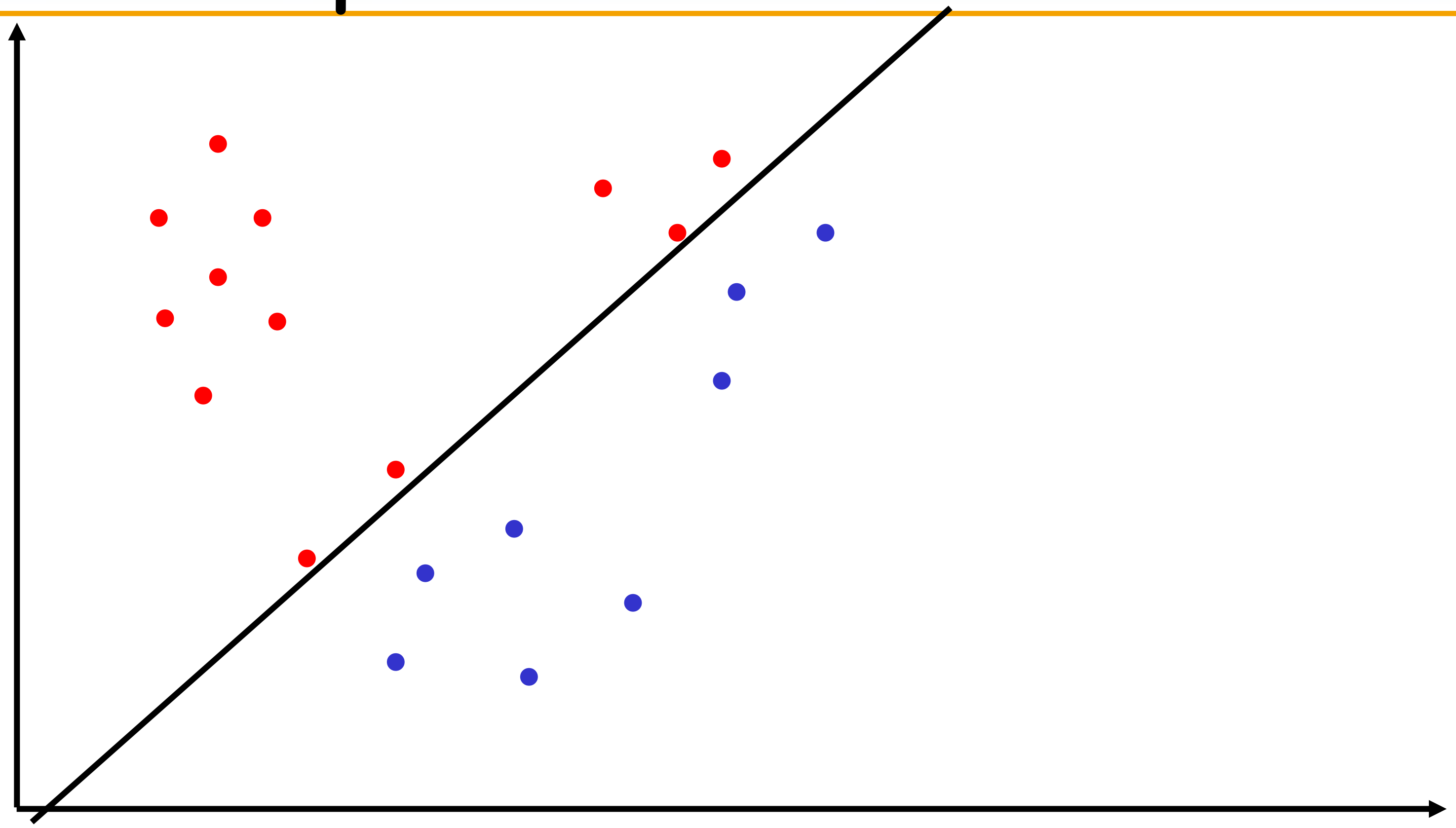
Supervised Classification Example



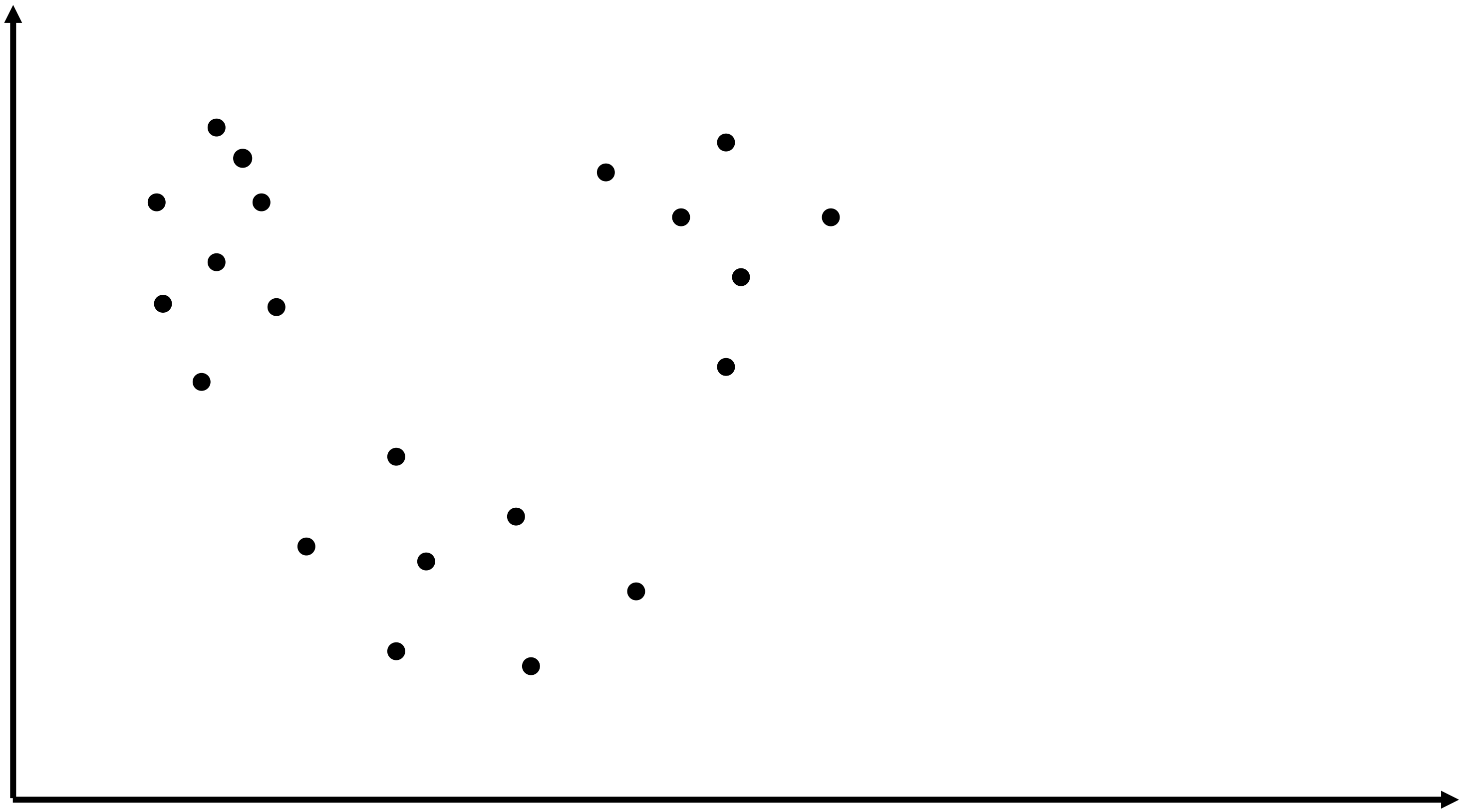
Supervised Classification Example



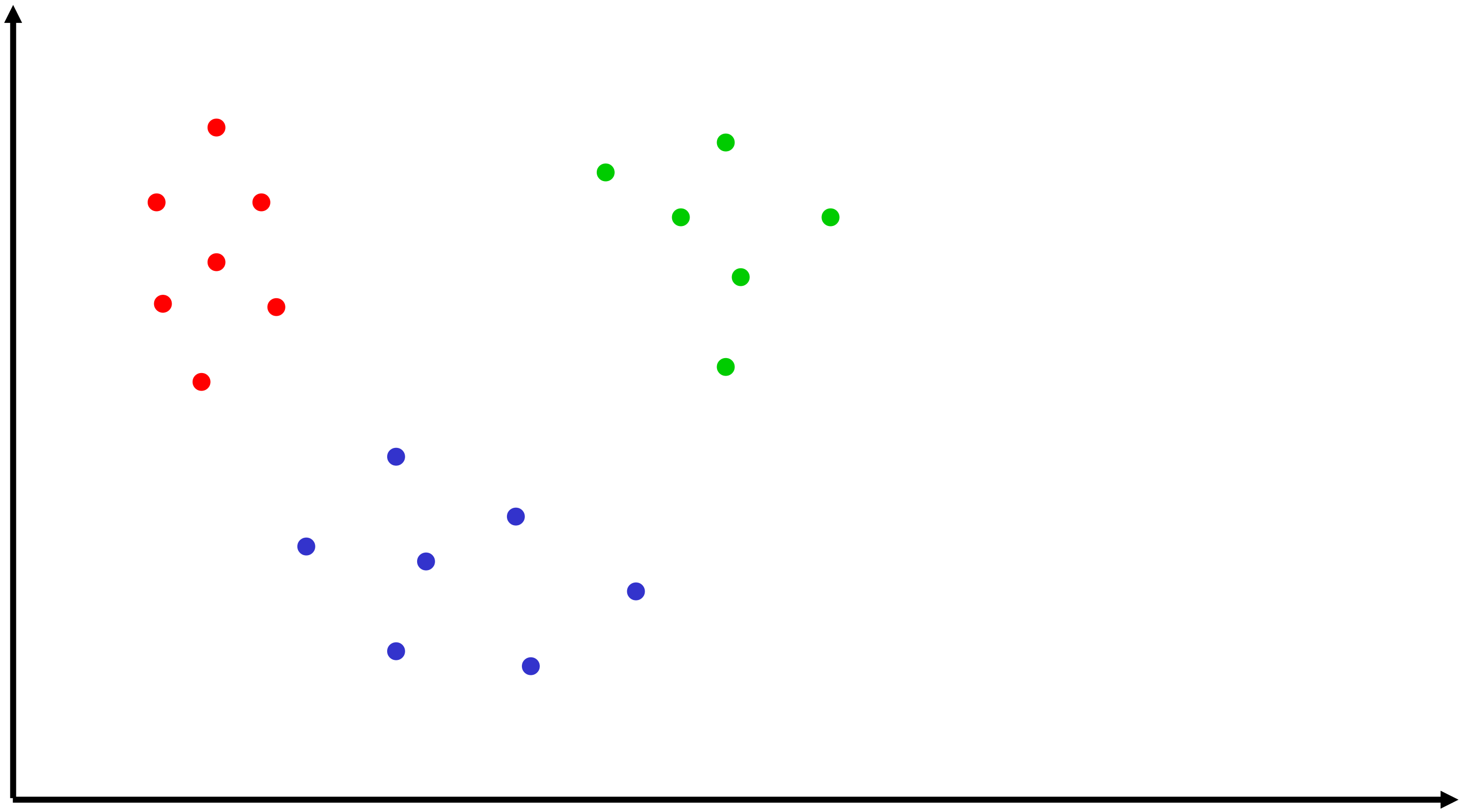
Supervised Classification Example



Unsupervised Clustering Example



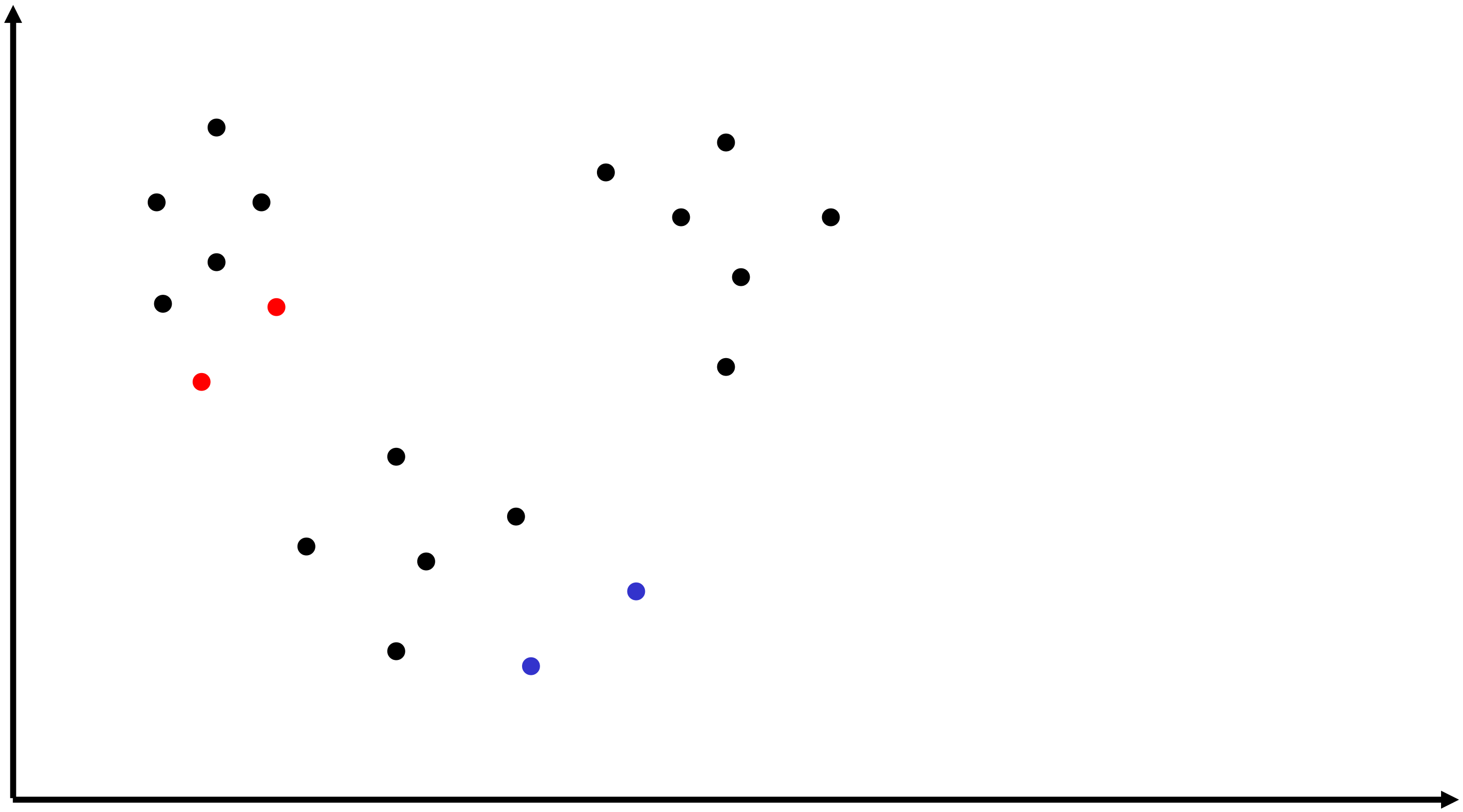
Unsupervised Clustering Example



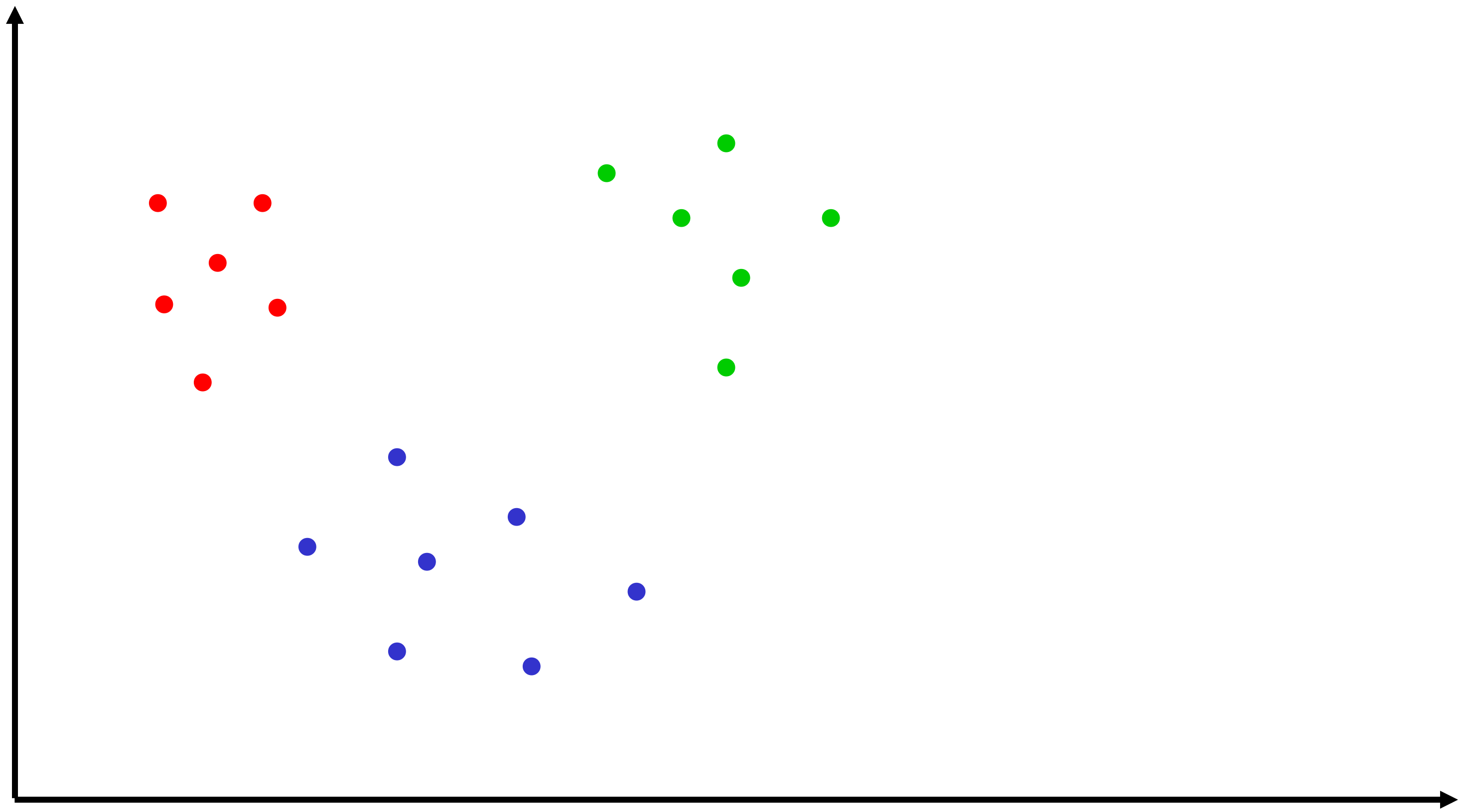
Semi-Supervised Learning

- Combines labeled and unlabeled data during training to improve performance:
  - ***Semi-supervised clustering:*** Uses small amount of labeled data to aid and bias the clustering of unlabeled data.
  - ***Semi-supervised classification***: Training on labeled data exploits additional unlabeled data, frequently resulting in a more accurate classifier.

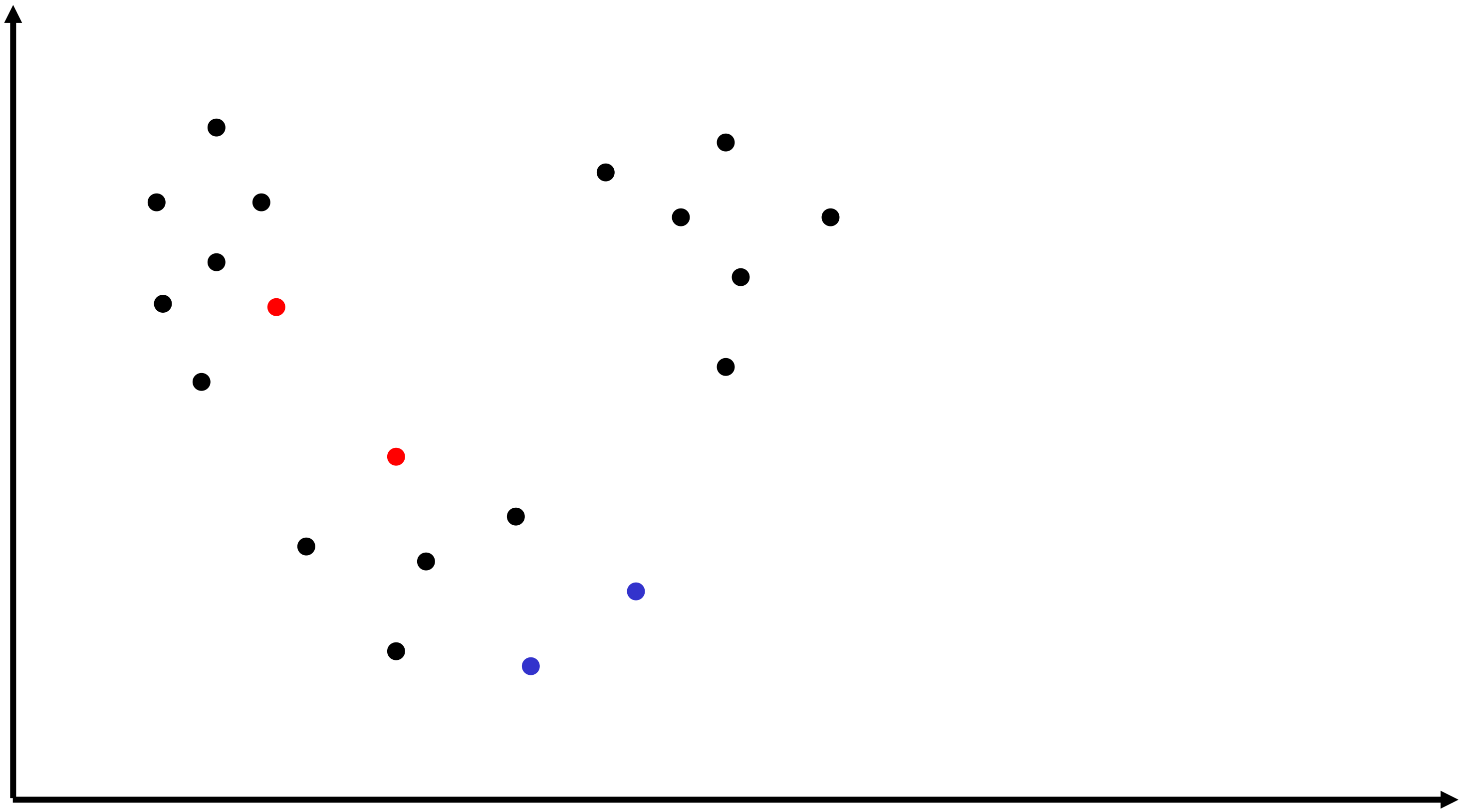
Semi-Supervised Clustering Example



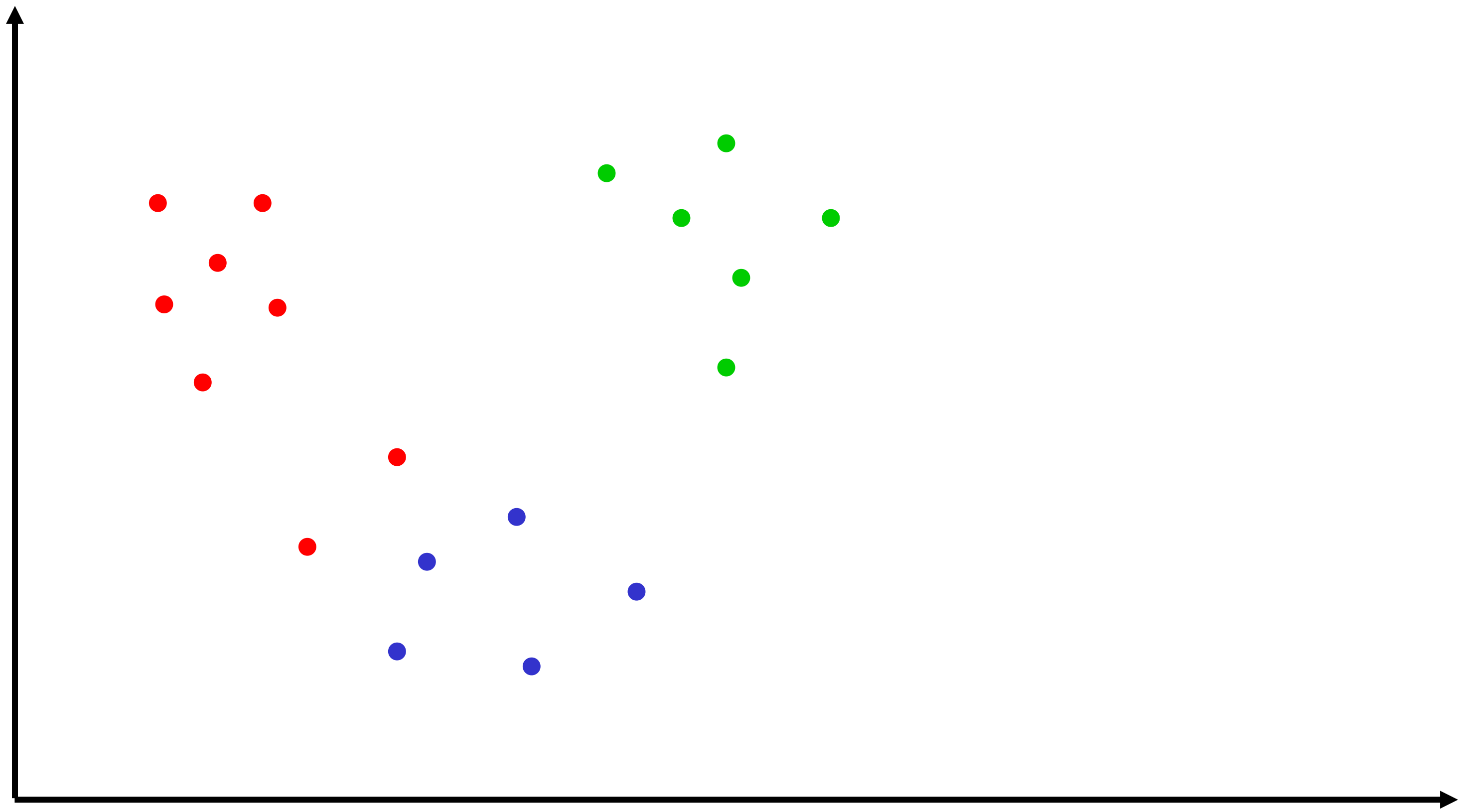
Semi-Supervised Clustering Example



Second Semi-Supervised Clustering Example



Second Semi-Supervised Clustering Example



Semi-Supervised Clustering

- Can group data using the categories in the initial labeled data.
- Can also extend and modify the existing set of categories as needed to reflect other regularities in the data.
- Can cluster a disjoint set of unlabeled data using the labeled data as a "guide" to the type of clusters desired.

Problem definition

- Input:
  - A set of unlabeled objects
  - Some ***domain knowledge***
- Output:
  - A partitioning of the objects into clusters
- Objective:
  - Maximum intra-cluster similarity
  - Minimum inter-cluster similarity
  - ***High consistency between the partitioning and the domain knowledge***

What is Domain Knowledge?

- Must-link and cannot-link
- Class labels
- Ontology

Why semi-supervised clustering?

- Why not clustering?
  - Could not incorporate prior knowledge into clustering process
- Why not classification?
  - Sometimes there are insufficient labeled data.
- Potential applications
  - Bioinformatics (gene and protein clustering)
  - Document hierarchy construction
  - News/email categorization
  - Image categorization

Semi-Supervised Clustering

- Approaches
  - Search-based Semi-Supervised Clustering
    - Alter the clustering algorithm using the constraints
  - Similarity-based Semi-Supervised Clustering
    - Alter the similarity measure based on the constraints
  - Combination of both

Search-Based Semi-Supervised Clustering

- Alter the clustering algorithm that searches for a good partitioning by:
  - Modifying the objective function to give a reward for obeying labels on the supervised data [Demeriz:ANNIE99].
  - Enforcing constraints (*must-link, cannot-link*) on the labeled data during clustering [Wagstaff:ICML00, Wagstaff:ICML01].
  - Use the labeled data to initialize clusters in an iterative refinement algorithm (kMeans, EM) [Basu:ICML02].

Unsupervised KMeans Clustering

- KMeans iteratively partitions a dataset into K clusters.

Algorithm:

Initialize K cluster centers $\{\mu_l\}_{l=1}^{K}$ randomly. Repeat until *convergence:*

- **Cluster Assignment Step**: Assign each data point x to the cluster X_l, such that L_2 distance of x from μ_l (center of X_l) is minimum
- **Center Re-estimation Step**: Re-estimate each cluster center μ_l as the mean of the points in that cluster

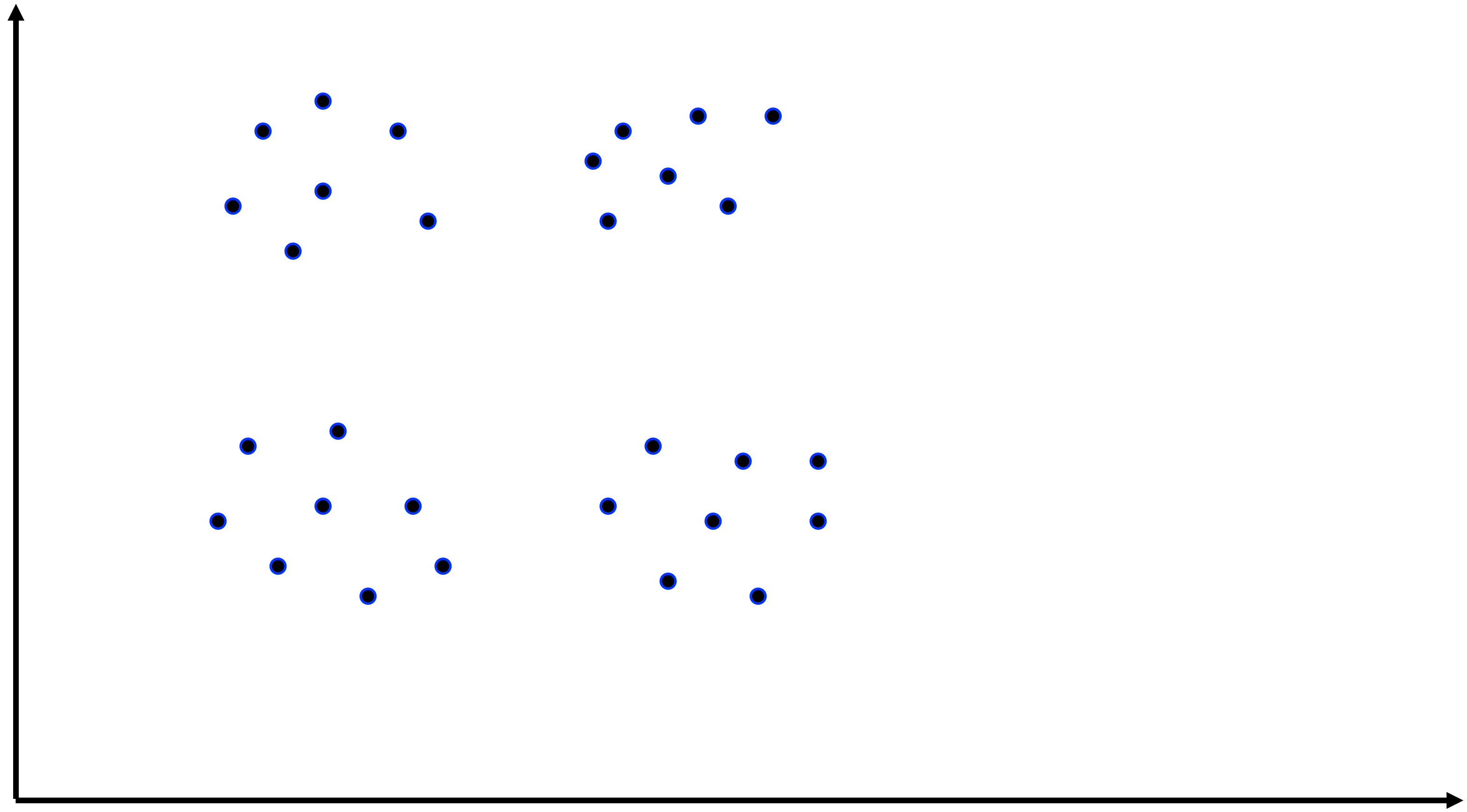
KMeans Objective Function

- Locally minimizes sum of squared distance between the data points and their corresponding cluster centers:

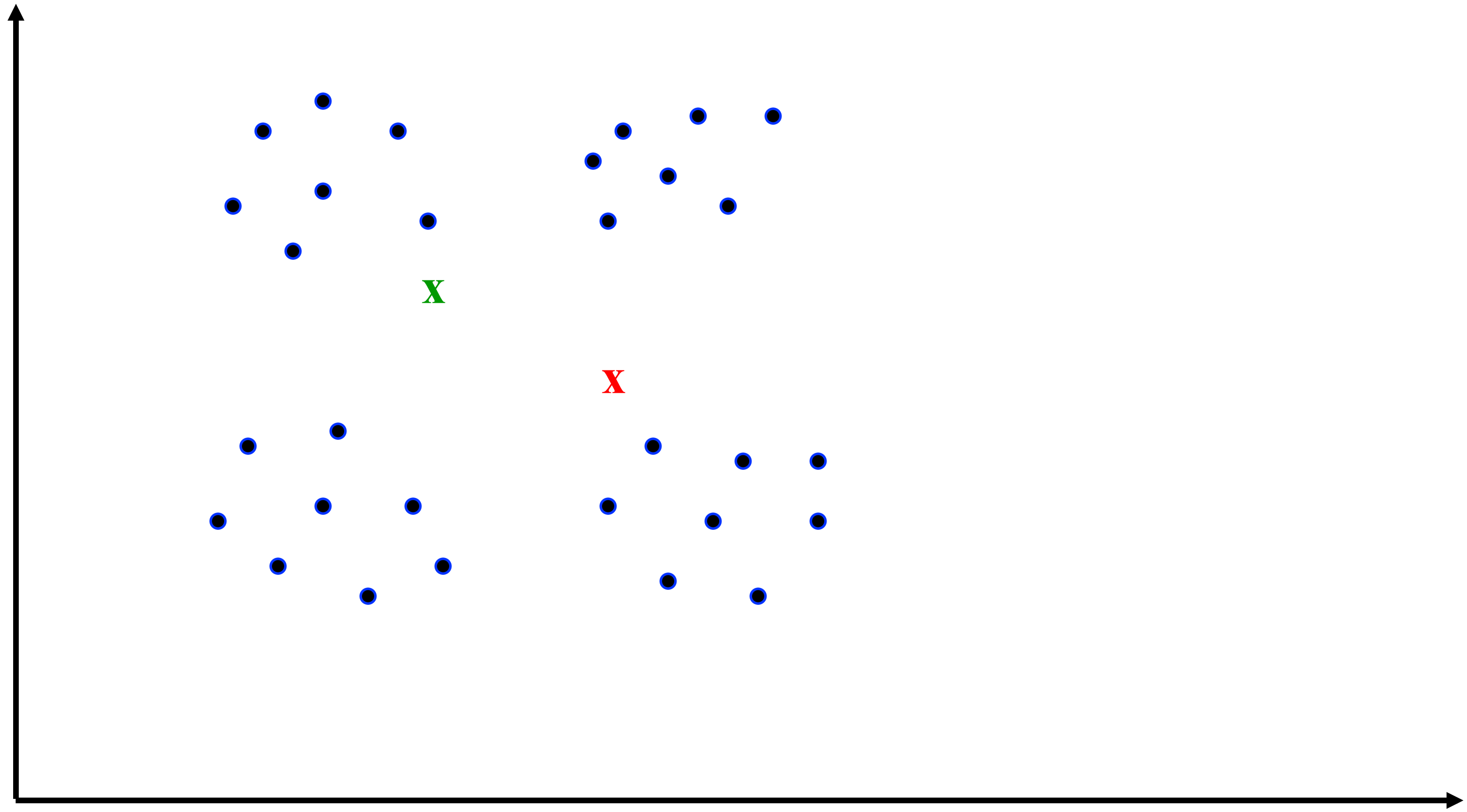
$$\sum_{l=1}^{K} \sum_{x_i \in X_l} \| x_i - \mu_l \|^2$$

- Initialization of K cluster centers:
  - Totally random
  - Random perturbation from global mean
  - Heuristic to ensure well-separated centers etc.

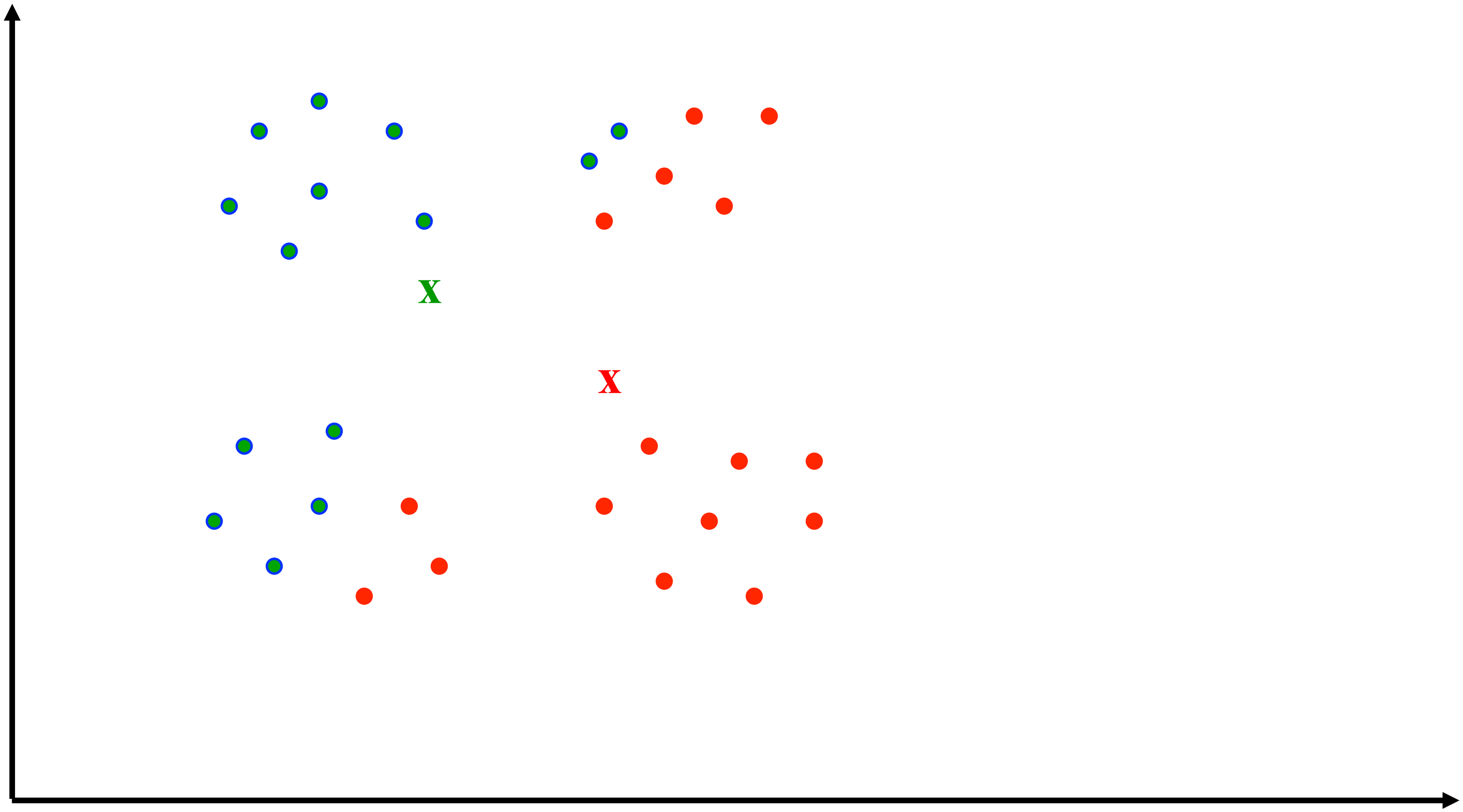
K Means Example



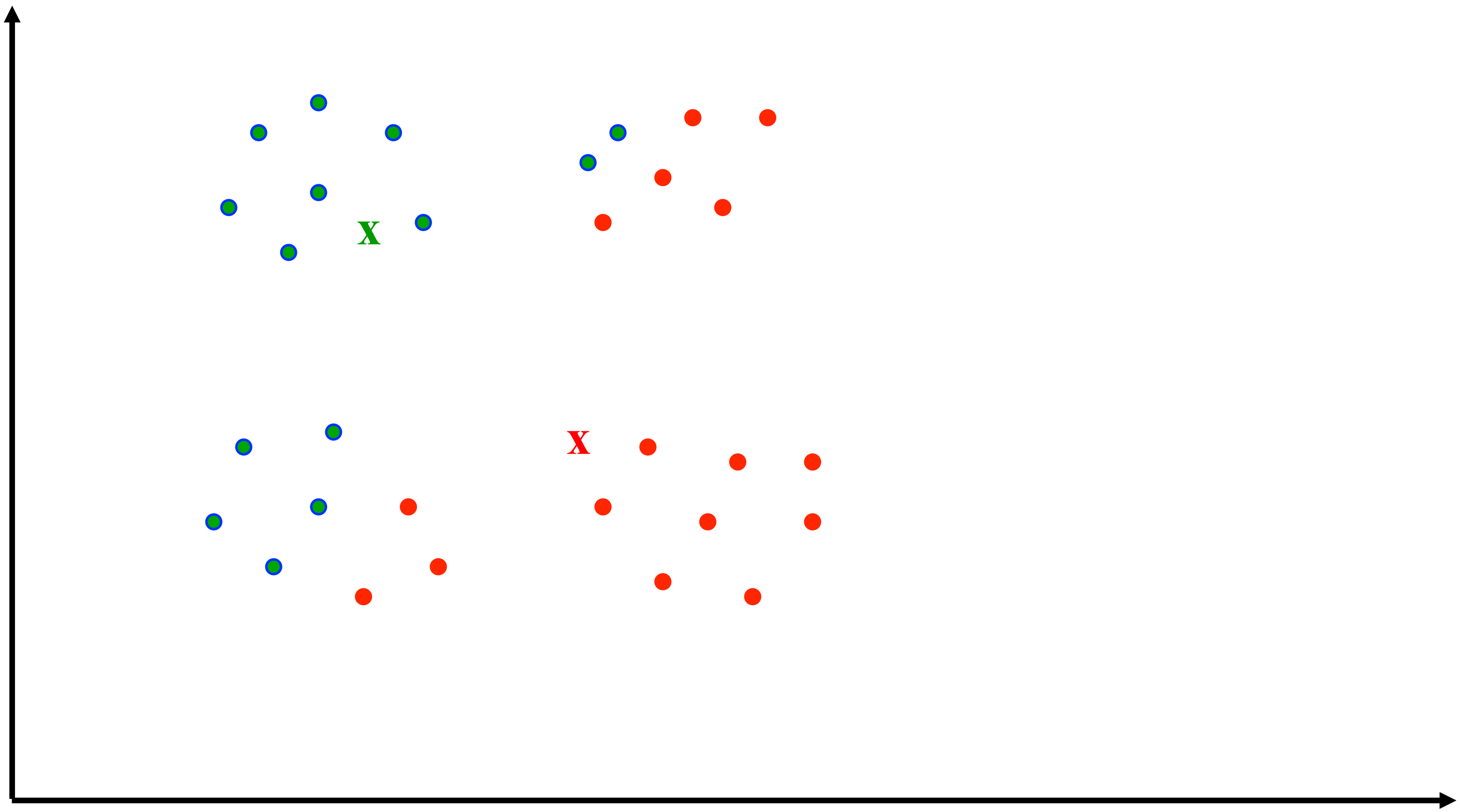
K Means Example
Randomly Initialize Means



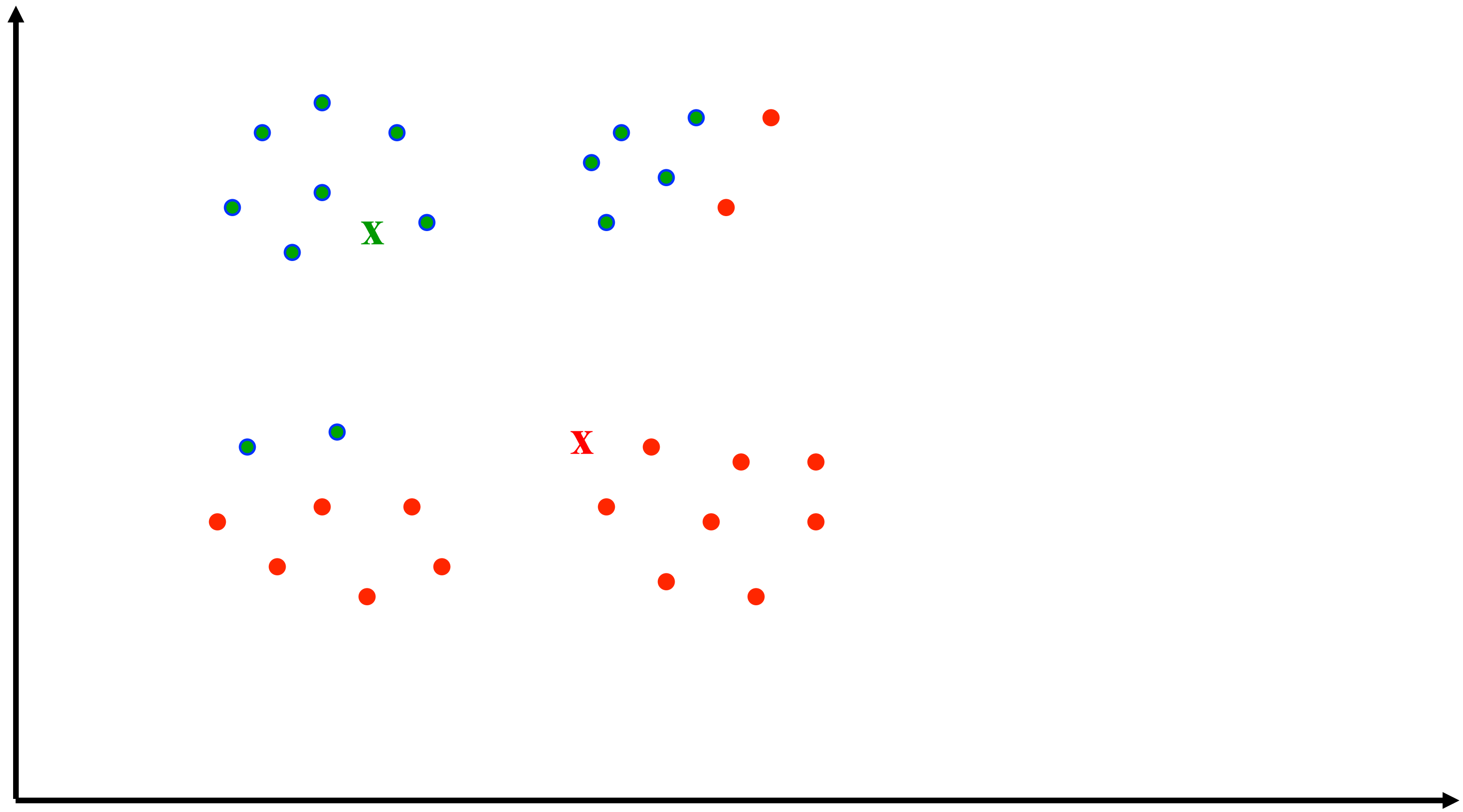
K Means Example
Assign Points to Clusters



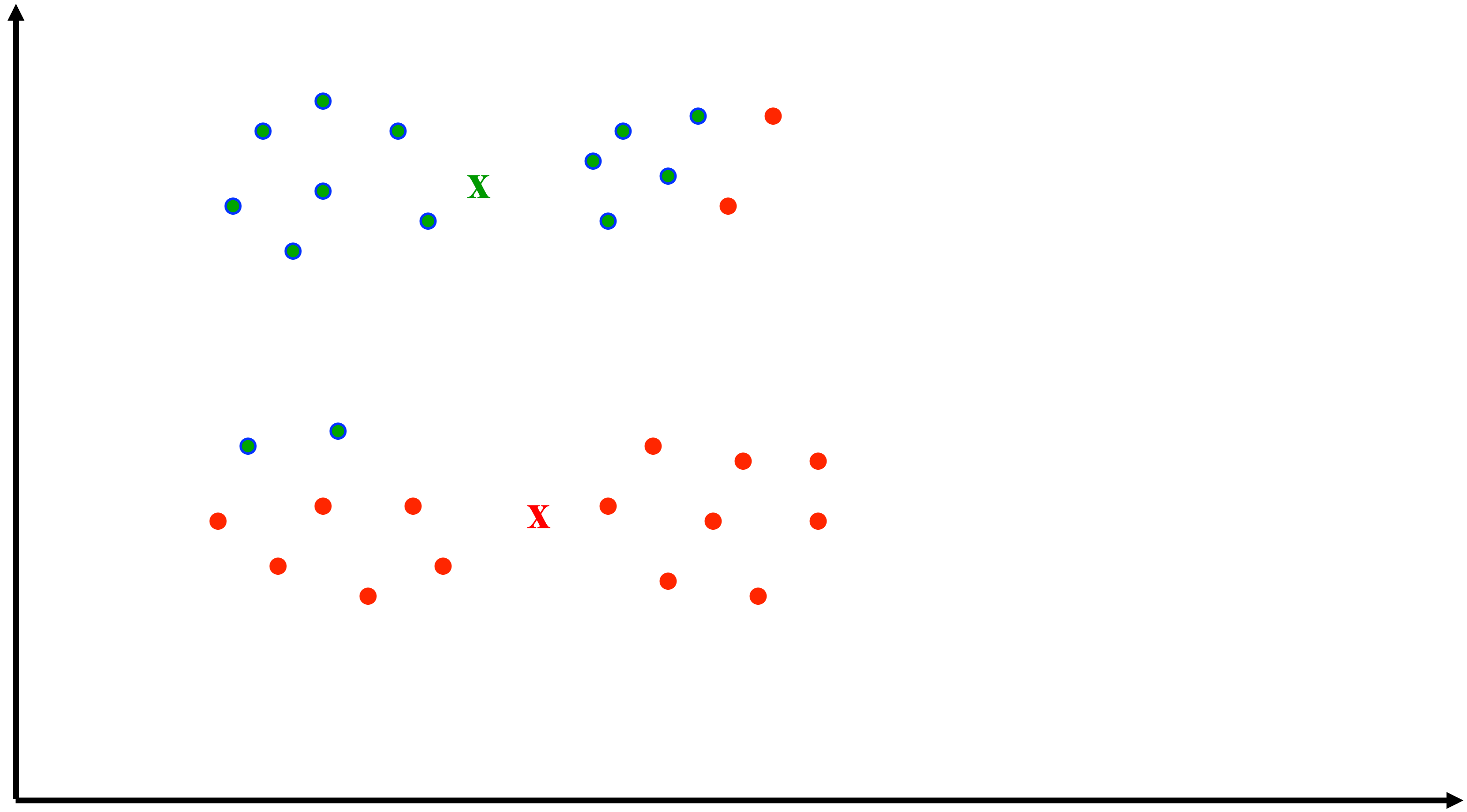
K Means Example
Re-estimate Means



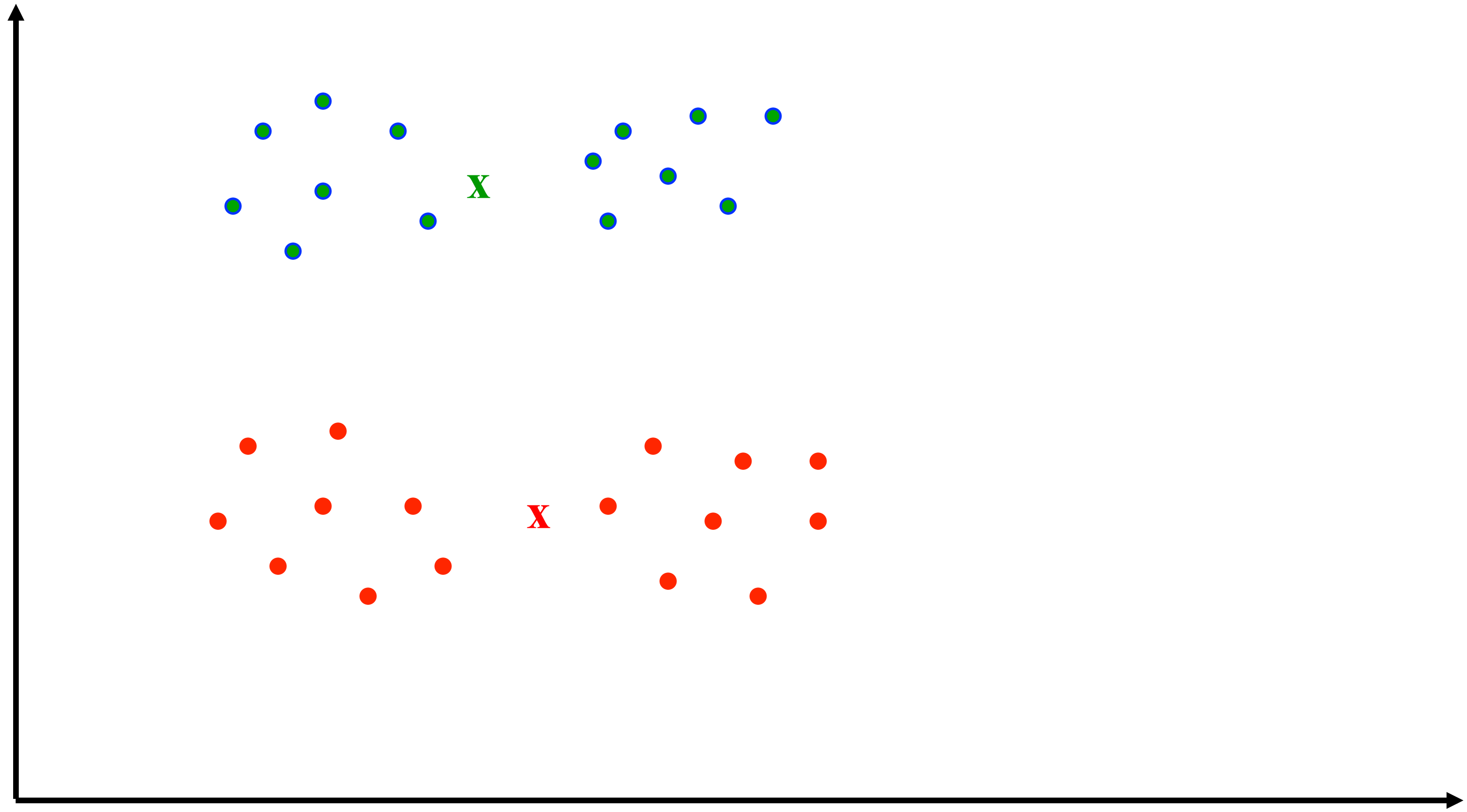
K Means Example
Re-assign Points to Clusters



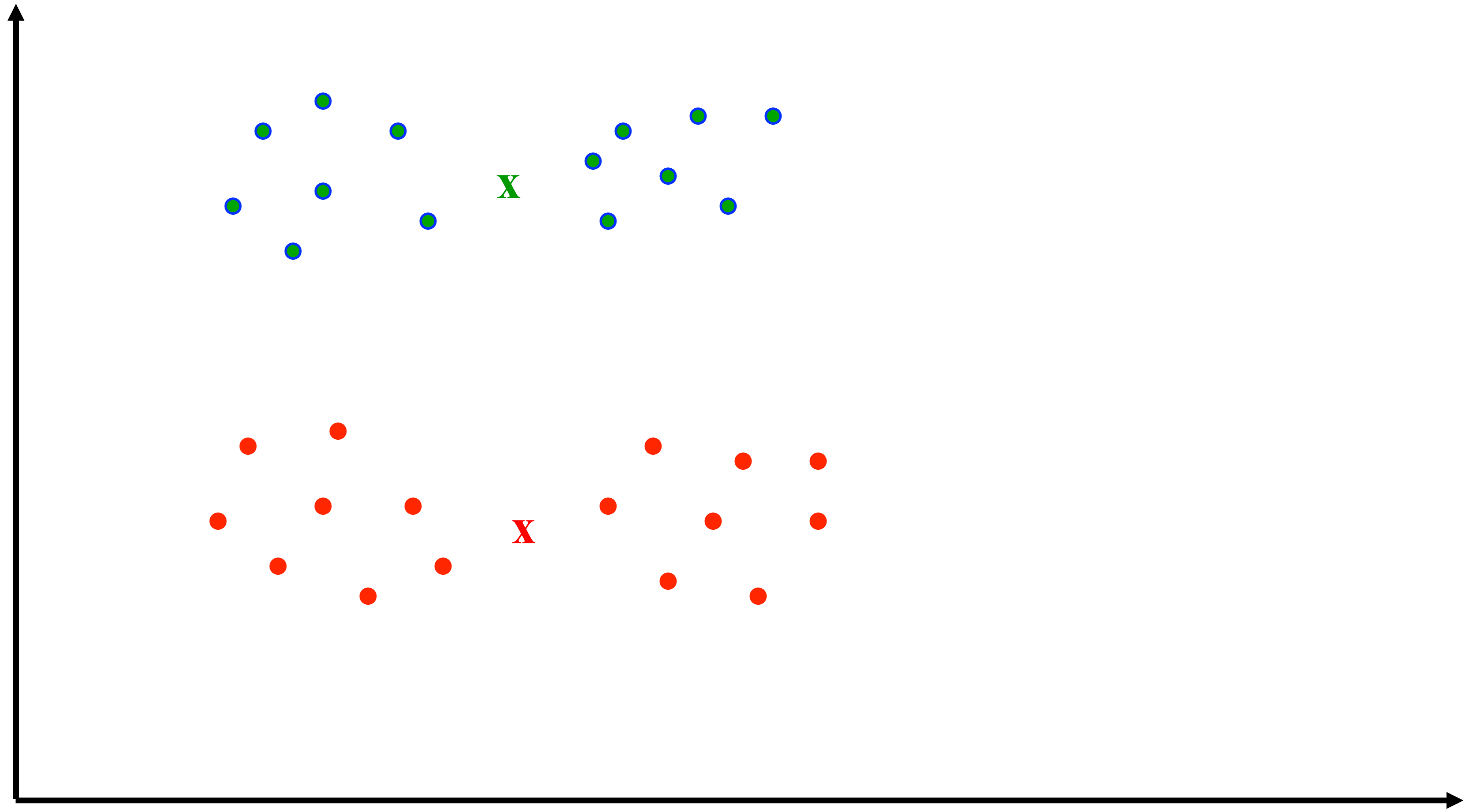
K Means Example
Re-estimate Means



K Means Example
Re-assign Points to Clusters



K Means Example
Re-estimate Means and Converge



Semi-Supervised K-Means

- Constraints (Must-link, Cannot-link)
  - COP K-Means
- Partial label information is given
  - Seeded K-Means (Basu, ICML'02)
  - Constrained K-Means

COP K-Means

- COP K-Means is K-Means with must-link (must be in same cluster) and cannot-link (cannot be in same cluster) constraints on data points.
- Initialization: Cluster centers are chosen randomly but no must-link constraints that may be violated
- Algorithm: During cluster assignment step in COP-K-Means, a point is assigned to its nearest cluster without violating any of its constraints. If no such assignment exists, abort.
- Based on Wagstaff *et al*.: ICML01

COP K-Means Algorithm

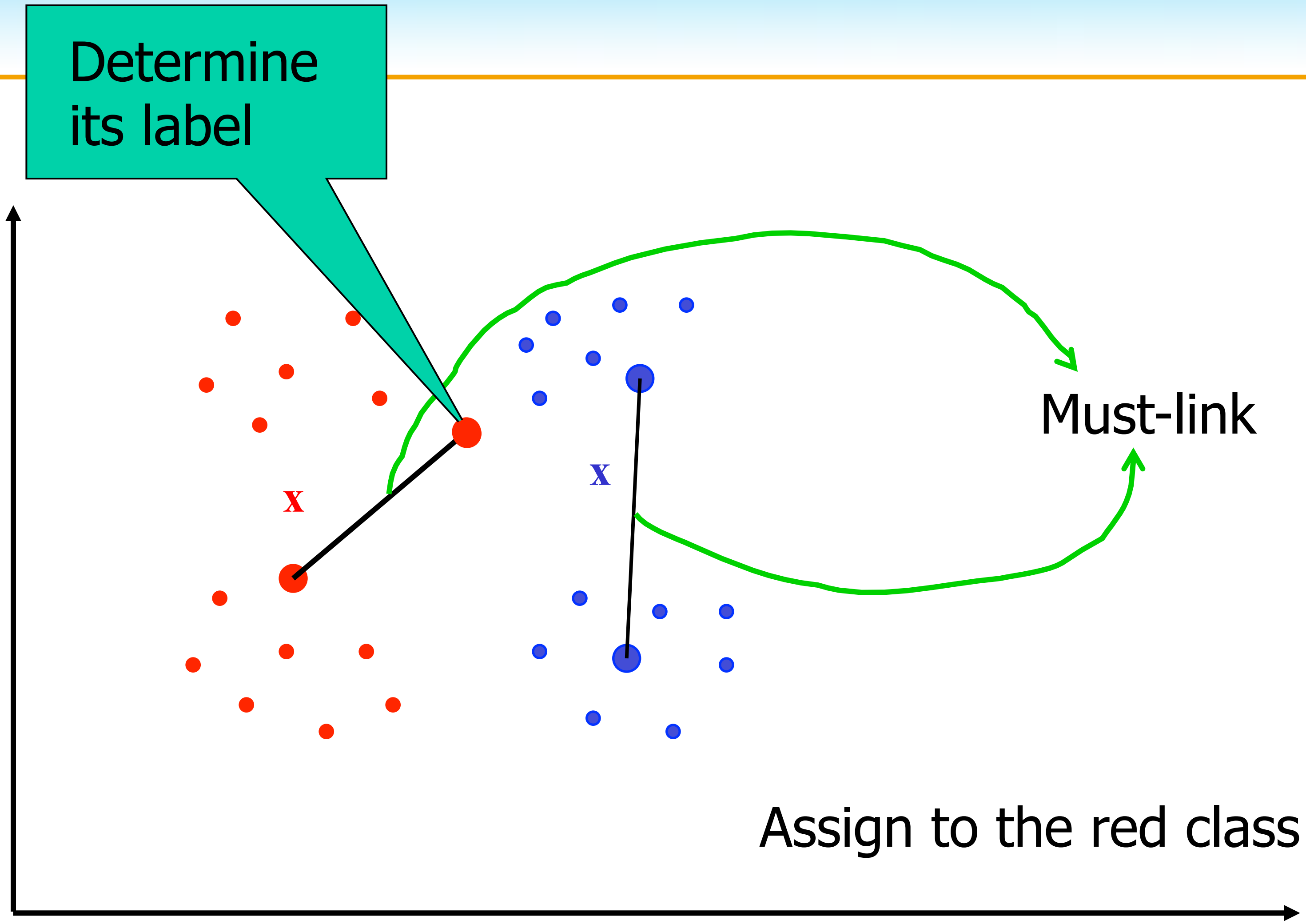
COP-KMEANS(data set D, must-link constraints $Con_= \subseteq D \times D$, cannot-link constraints $Con_{\neq} \subseteq D \times D$)

1. Let $C_1 \ldots C_k$ be the initial cluster centers.
2. For each point d_i in D, assign it to the closest cluster C_j **such that** VIOLATE-CONSTRAINTS(d_i, C_j, $Con_=$, $Con_{\neq}$) **is false. If no such cluster exists, fail (return {})**.
3. For each cluster C_i, update its center by averaging all of the points d_j that have been assigned to it.
4. Iterate between (2) and (3) until convergence.
5. Return $\{C_1 \ldots C_k\}$.

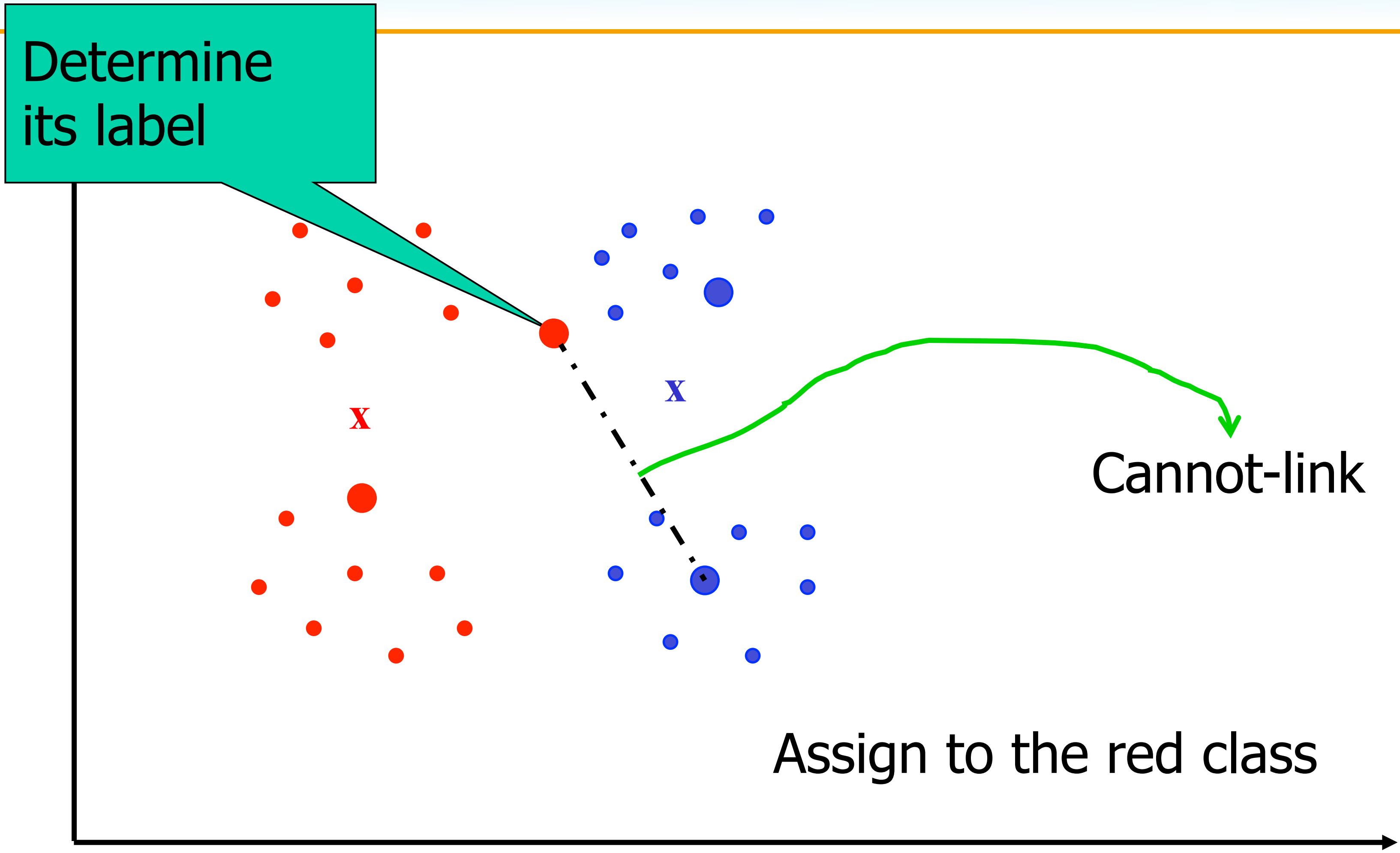
VIOLATE-CONSTRAINTS(data point d, cluster C, must-link constraints $Con_= \subseteq D \times D$, cannot-link constraints $Con_{\neq} \subseteq D \times D$)

1. For each $(d, d_=) \in Con_=$: If $d_= \notin C$, return true.
2. For each $(d, d_{\neq}) \in Con_{\neq}$: If $d_{\neq} \in C$, return true.
3. Otherwise, return false.

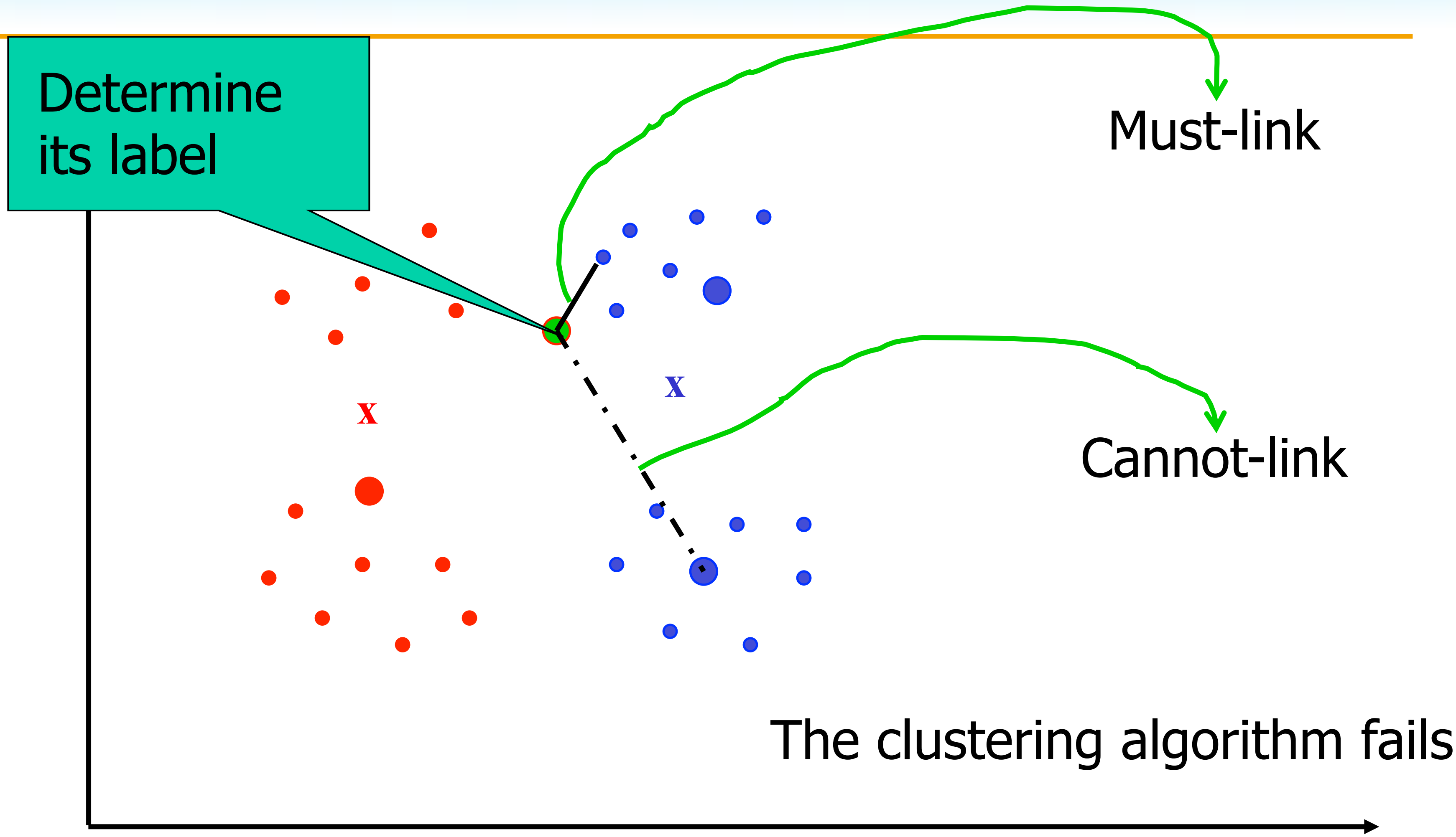
Illustration



Illustration



Illustration



Semi-Supervised K-Means

- Seeded K-Means:
  - Labeled data provided by user are used for initialization: initial center for cluster *i* is the mean of the seed points having label *i*.
  - Seed points are only used for initialization, and not in subsequent steps.
- Constrained K-Means:
  - Labeled data provided by user are used to initialize K-Means algorithm.
  - Cluster labels of seed data are kept unchanged in the cluster assignment steps, and only the labels of the non-seed data are re-estimated.
- Based on Basu et al., ICML'02.

Seeded K-Means

Algorithm: `Seeded-KMeans`

Input: Set of data points $\mathcal{X} = \{x_1, \cdots, x_N\}, x_i \in \mathbb{R}^d$, number of clusters K, set $\mathcal{S} = \cup_{l=1}^{K} \mathcal{S}_l$ of initial seeds

Output: Disjoint K partitioning $\{\mathcal{X}_l\}_{l=1}^{K}$ of $\mathcal{X}$ such that KMeans objective function is optimized

Method:

1. `intialize`: $\mu_h^{(0)} \leftarrow \frac{1}{|\mathcal{S}_h|} \sum_{x \in \mathcal{S}_h} x$, for $h = 1, \ldots, K; t \leftarrow 0$
2. Repeat until *convergence*

2a. `assign_cluster`: Assign each data point x to the cluster h^* (i.e. set $\mathcal{X}_{h^*}^{(t+1)}$), for $h^* = \arg\min_h \|x - \mu_h^{(t)}\|^2$

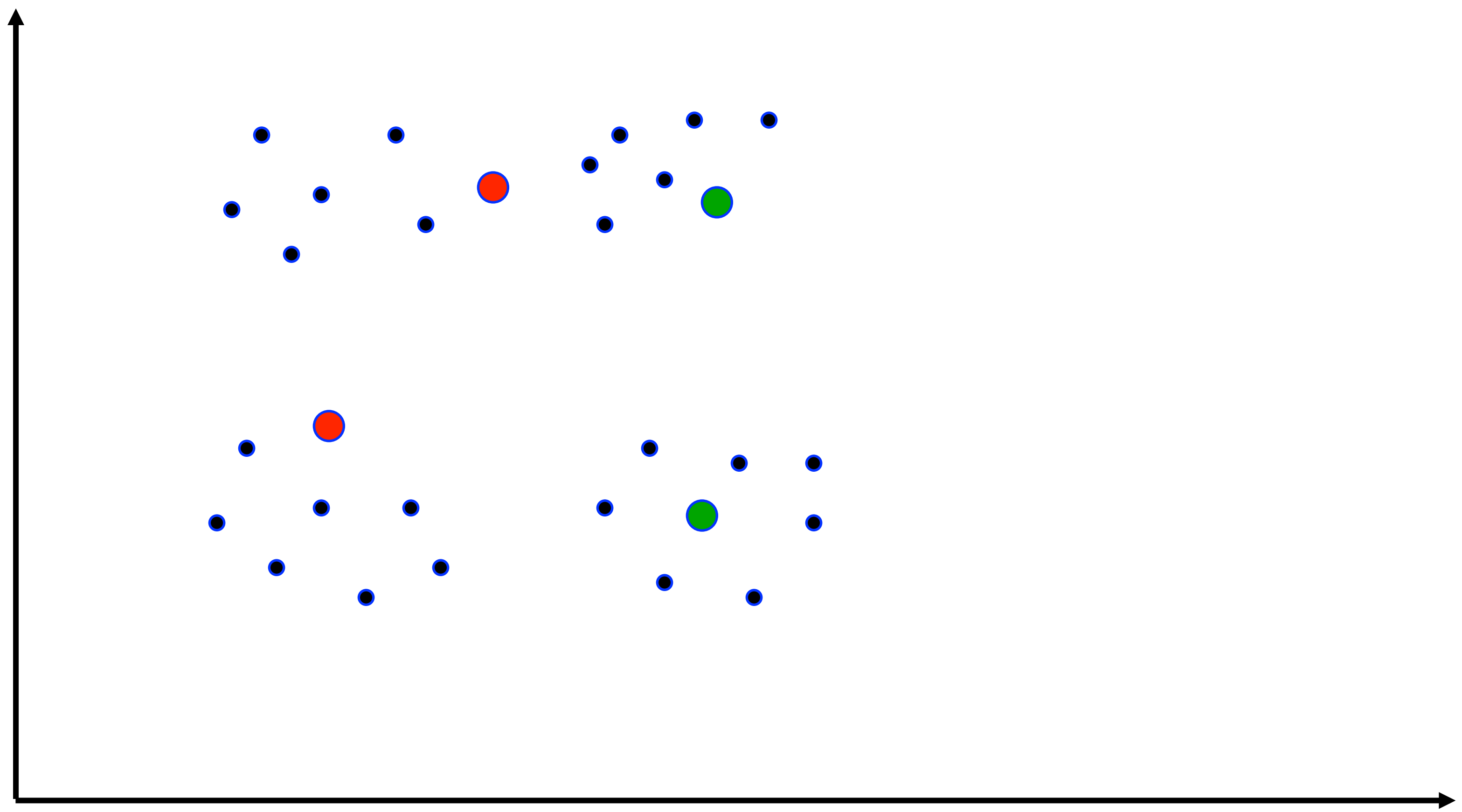
2b. `estimate_means`: $\mu_h^{(t+1)} \leftarrow \frac{1}{|\mathcal{X}_h^{(t+1)}|} \sum_{x \in \mathcal{X}_h^{(t+1)}} x$

2c. $t \leftarrow (t+1)$

Use labeled data to find the initial centroids and then run K-Means.

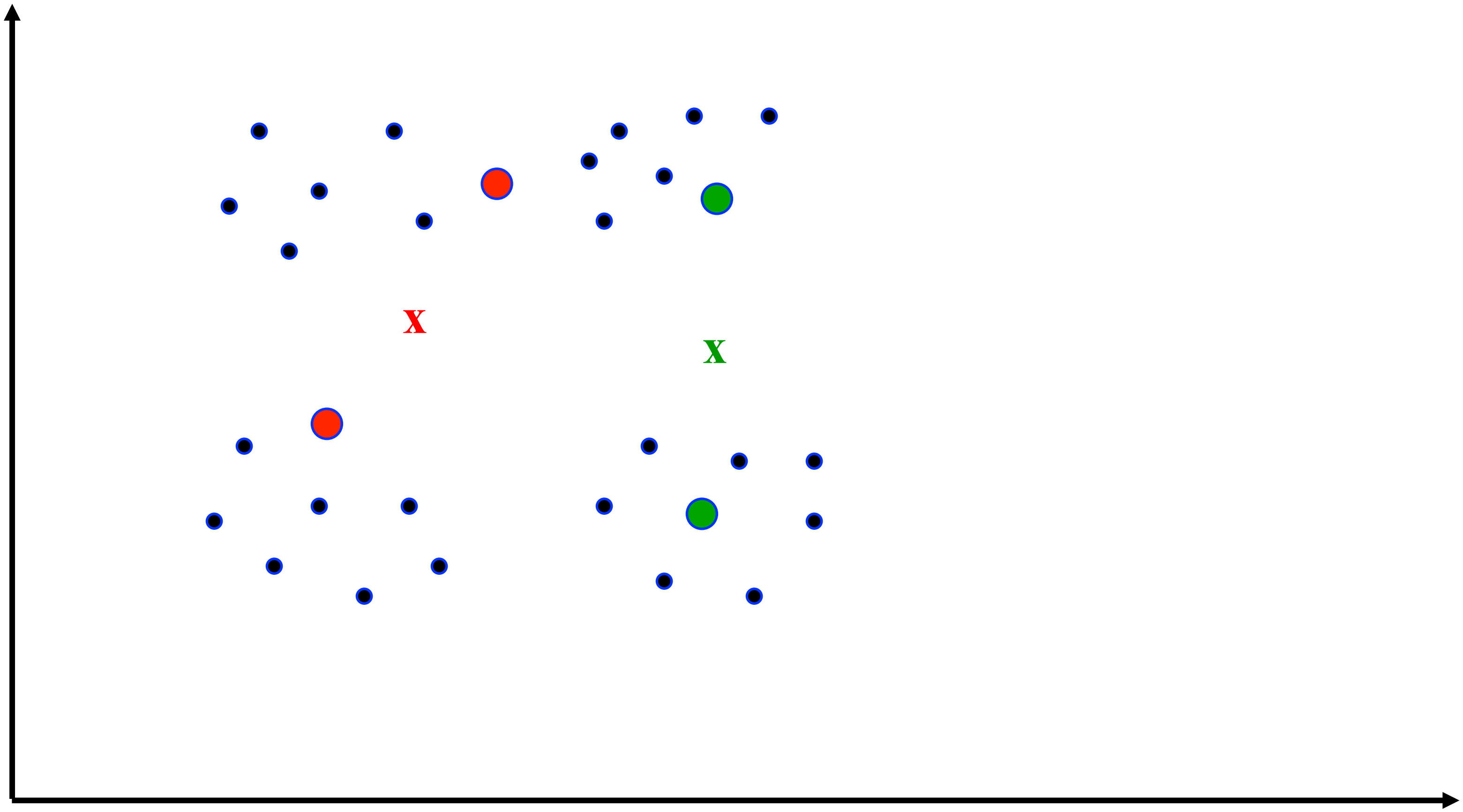
The labels for seeded points may change.

Seeded K-Means Example

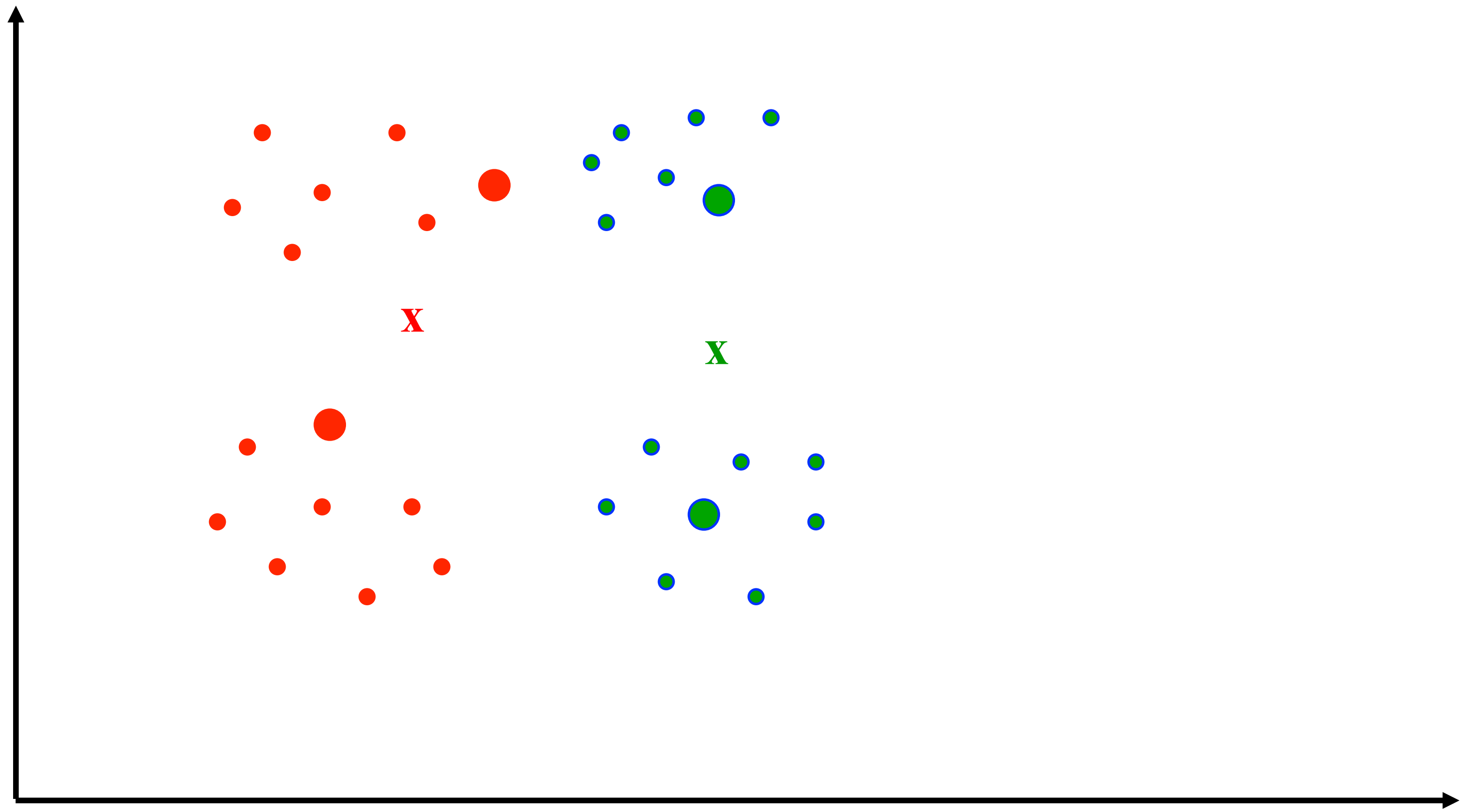


Seeded K-Means Example

Initialize Means Using Labeled Data

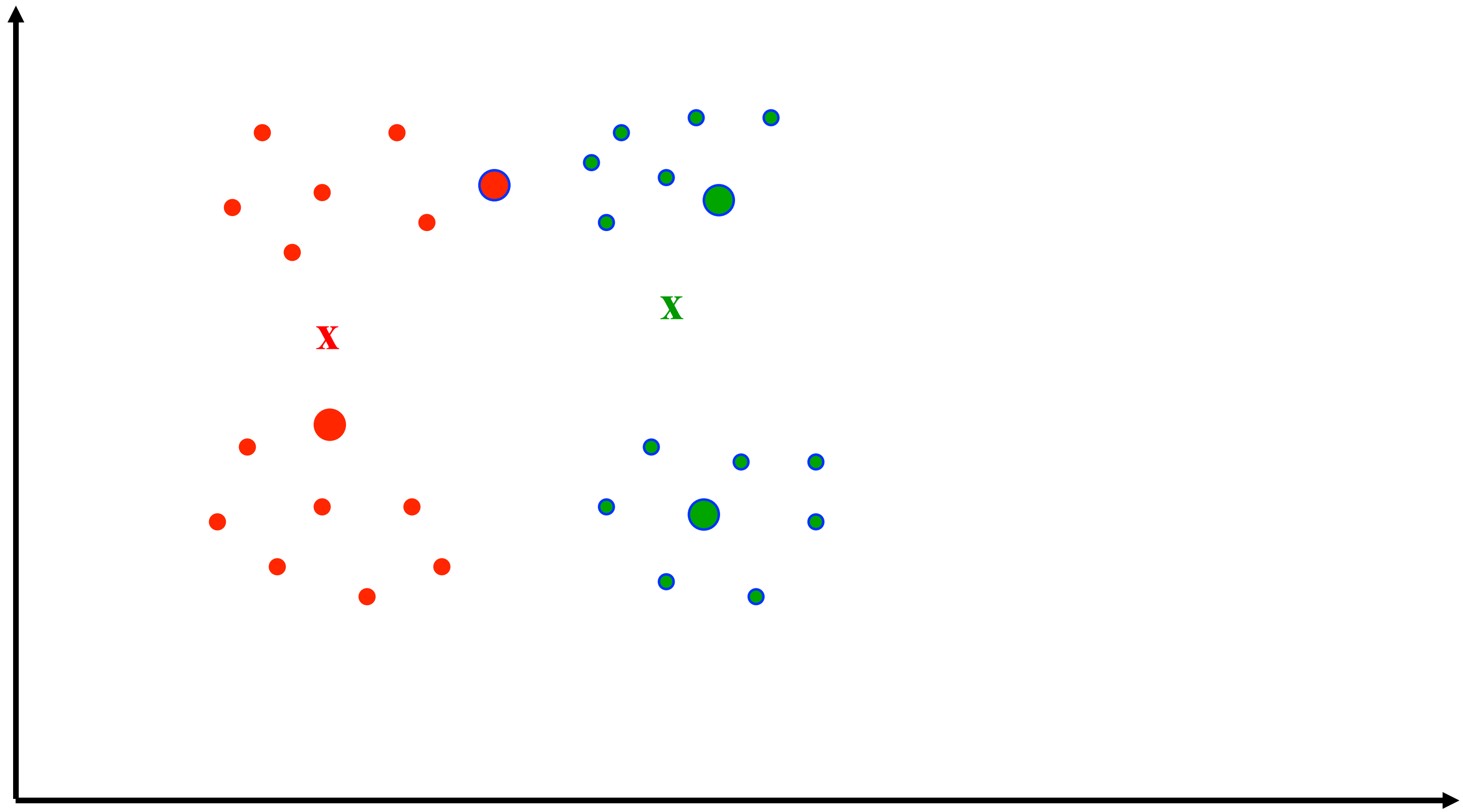


Seeded K-Means Example
Assign Points to Clusters



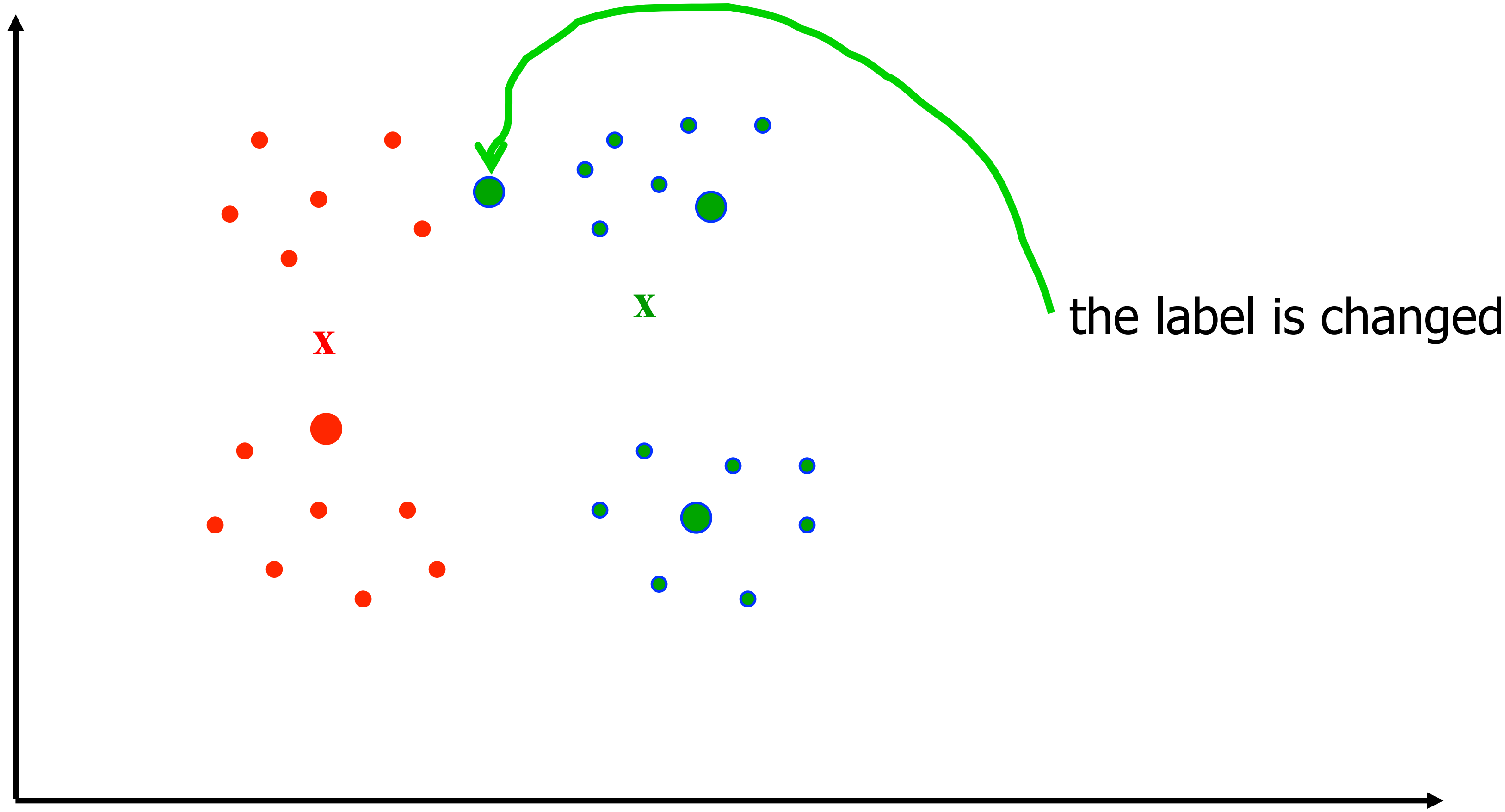
Seeded K-Means Example

Re-estimate Means



Seeded K-Means Example

Assign points to clusters and Converge



Constrained K-Means

Algorithm: `Constrained-KMeans`
Input: Set of data points $\mathcal{X} = \{x_1, \cdots, x_N\}, x_i \in \mathbb{R}^d$, number of clusters K, set $\mathcal{S} = \cup_{l=1}^{K} \mathcal{S}_l$ of initial seeds
Output: Disjoint K partitioning $\{\mathcal{X}_l\}_{l=1}^{K}$ of $\mathcal{X}$ such that the KMeans objective function is optimized
Method:
1. `intialize`: $\mu_h^{(0)} \leftarrow \frac{1}{|\mathcal{S}_h|} \sum_{x \in \mathcal{S}_h} x$, for $h = 1, \ldots, K; t \leftarrow 0$
2. Repeat until *convergence*

2a. `assign_cluster`: For $x \in \mathcal{S}$, if $x \in \mathcal{S}_h$ assign x to the cluster h (i.e., set $\mathcal{X}_h^{(t+1)}$). For $x \notin \mathcal{S}$, assign x to the cluster h^* (i.e. set $\mathcal{X}_{h^*}^{(t+1)}$), for $h^* = \arg\min_h \|x - \mu_h^{(t)}\|^2$

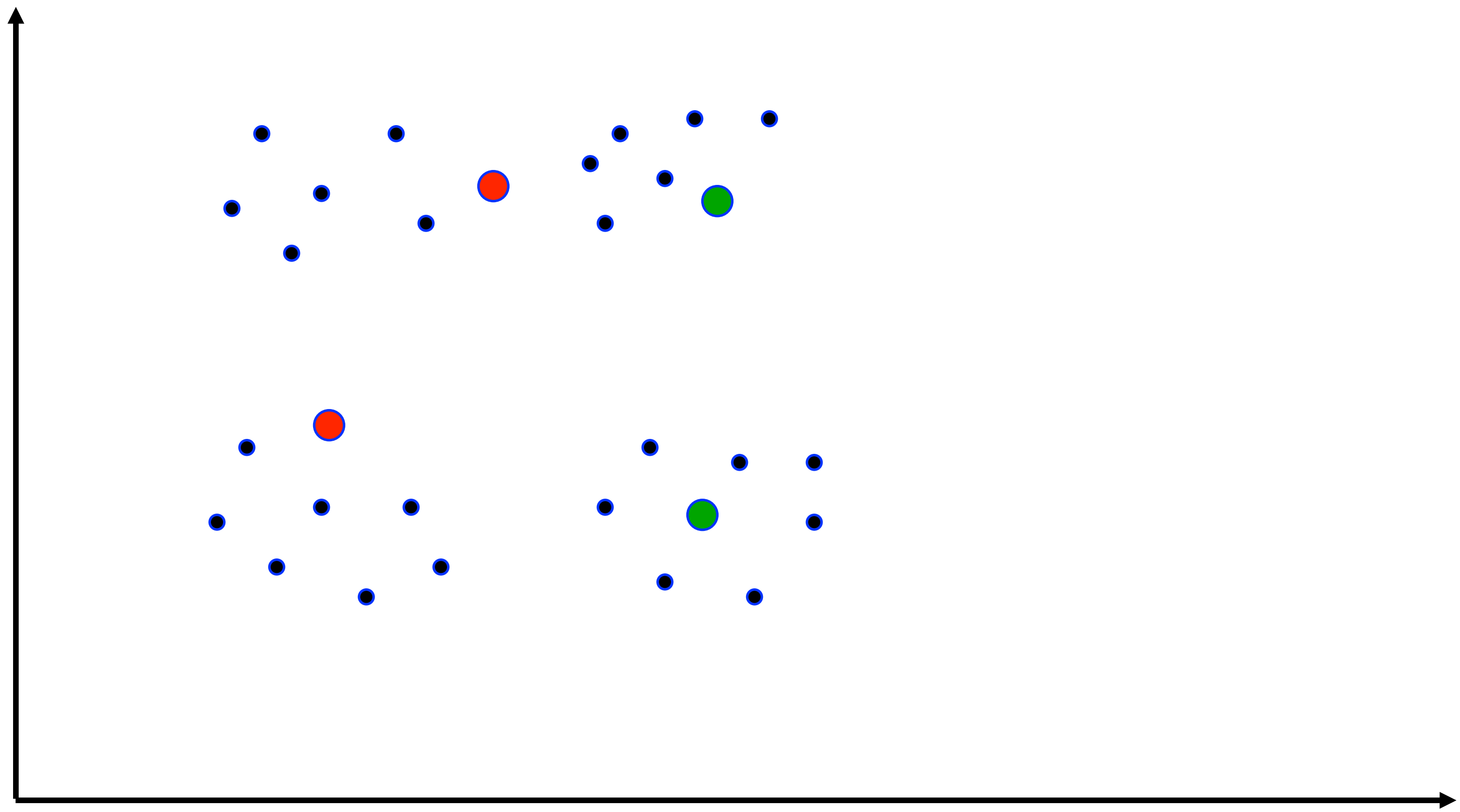
2b. `estimate_means`: $\mu_h^{(t+1)} \leftarrow \frac{1}{|\mathcal{X}_h^{(t+1)}|} \sum_{x \in \mathcal{X}_h^{(t+1)}} x$

2c. $t \leftarrow (t+1)$

Use labeled data to find the initial centroids and then run K-Means.

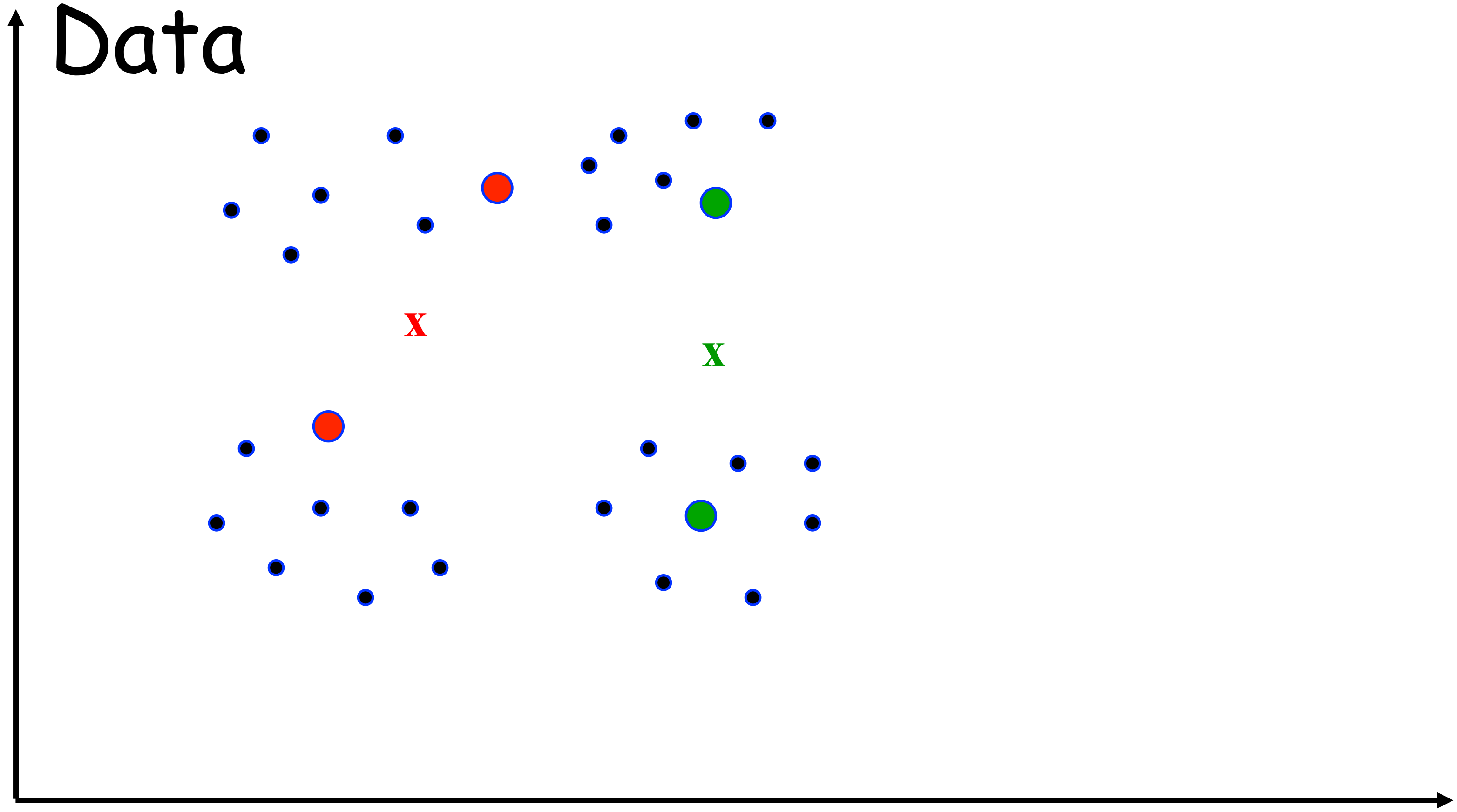
The labels for seeded points will not change.

Constrained K-Means Example



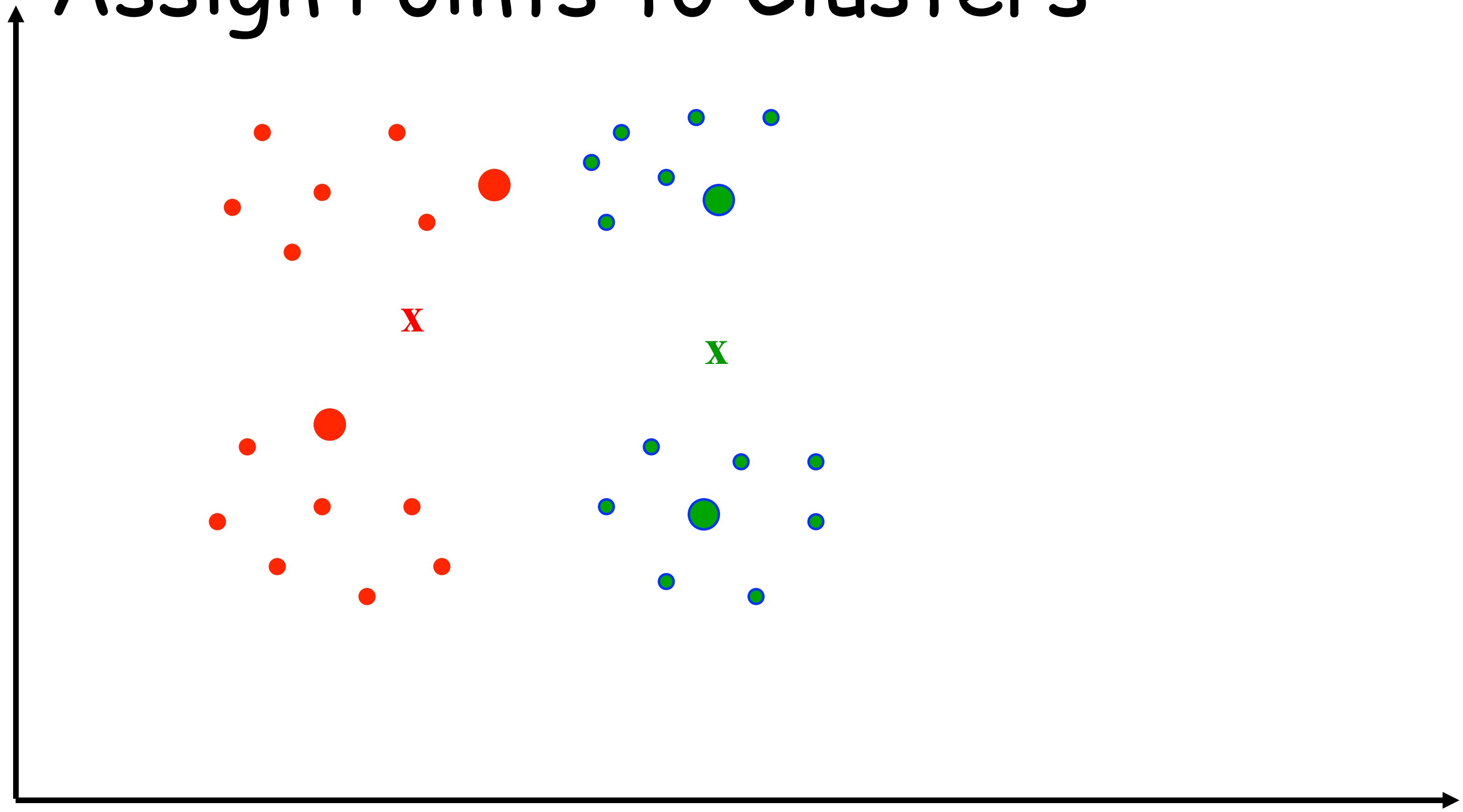
Constrained K-Means Example

Initialize Means Using Labeled Data



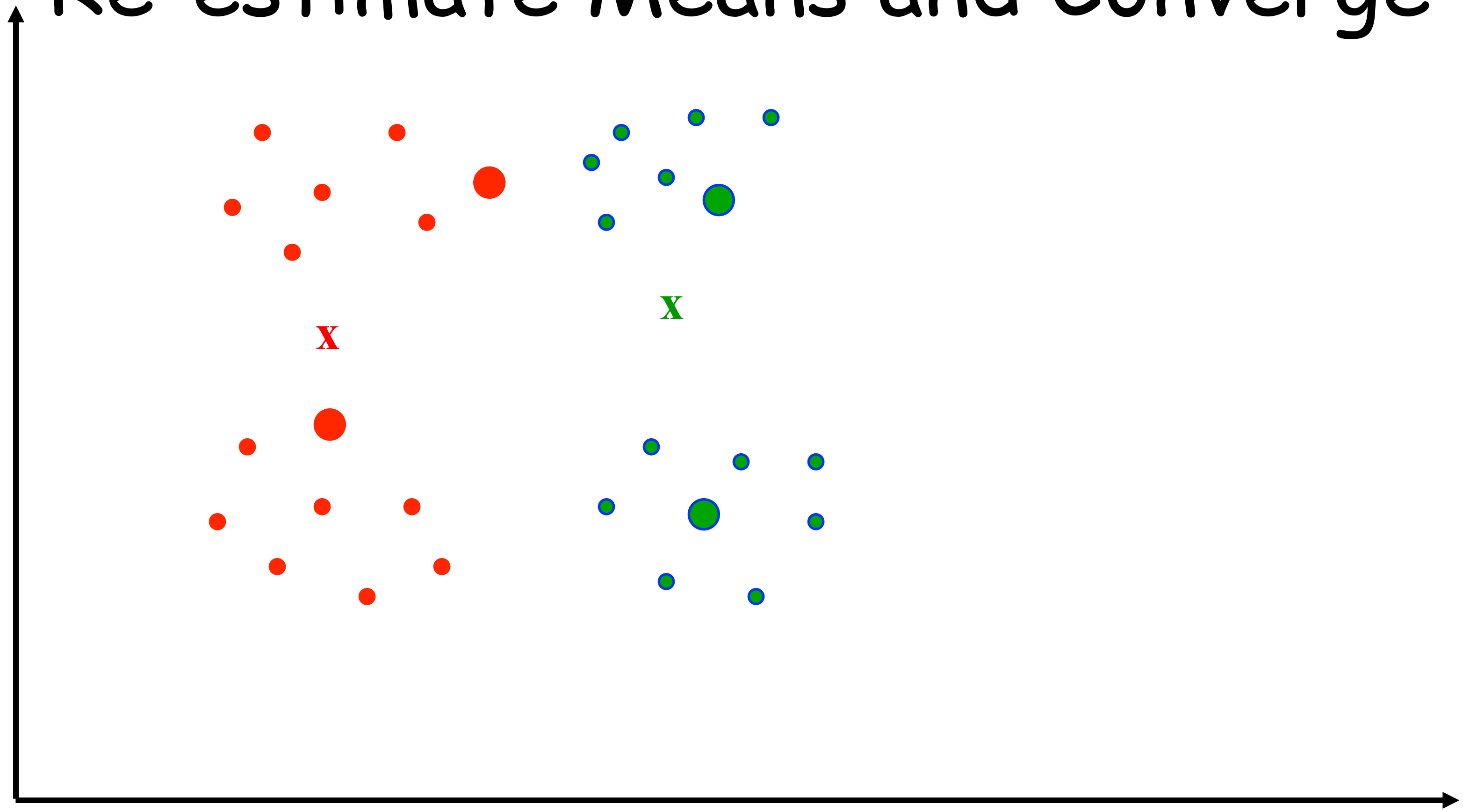
Constrained K-Means Example

Assign Points to Clusters



Constrained K-Means Example

Re-estimate Means and Converge



Datasets

- Data sets:
  - UCI Iris (3 classes; 150 instances)
  - CMU 20 Newsgroups (20 classes; 20,000 instances)
  - Yahoo! News (20 classes; 2,340 instances)
- Data subsets created for experiments:
  - **Small-20 newsgroup**: random sample of 100 documents from each newsgroup, created to study effect of datasize on algorithms.
  - **Different-3 newsgroup**: 3 very different newsgroups (*alt.atheism, rec.sport.baseball, sci.space*), created to study effect of data separability on algorithms.
  - **Same-3 newsgroup**: 3 very similar newsgroups (*comp.graphics, comp.os.ms-windows, comp.windows.x*).

Evaluation

- Objective function

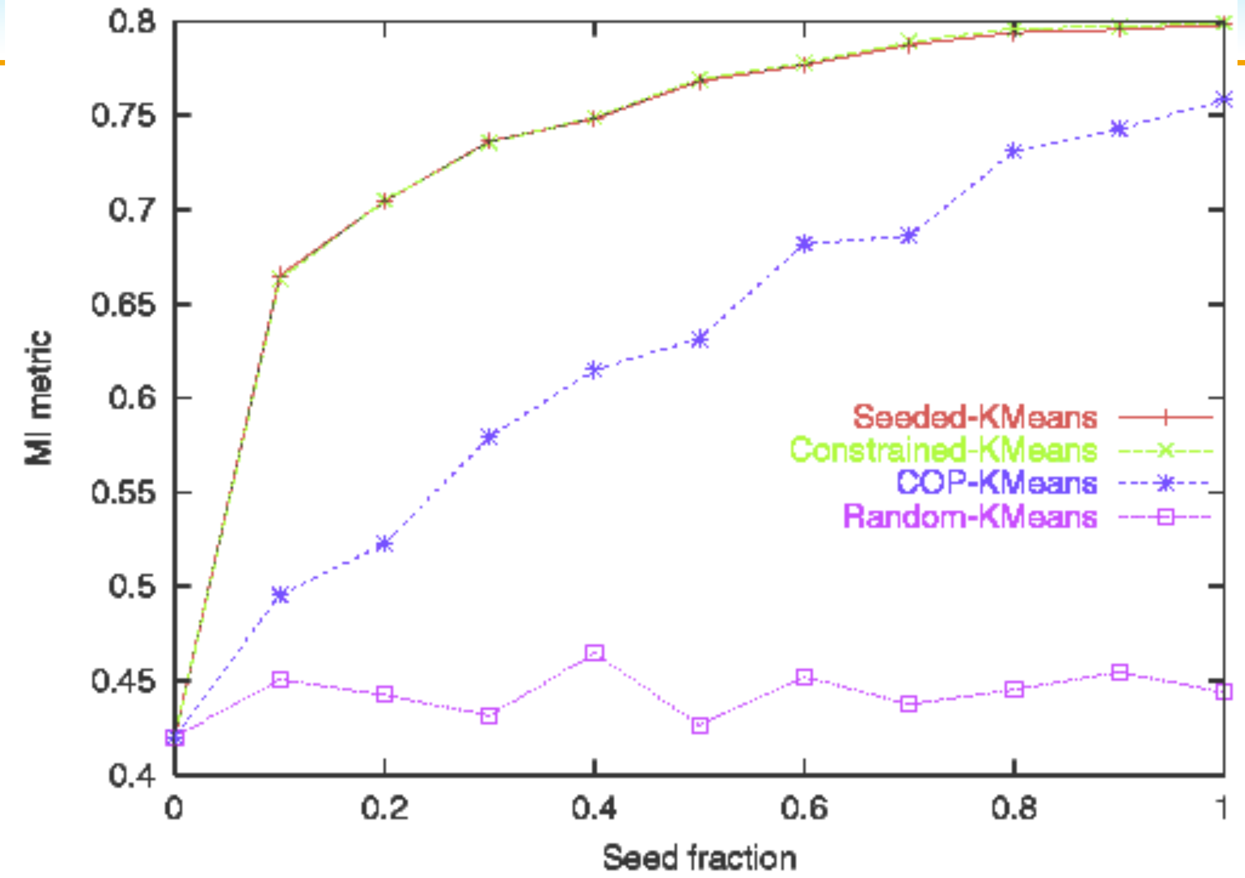
$$\|x_i - \mu_l\|^2 = 2 - 2x_i^T \mu_l$$

$$\mathcal{J}_{\text{spkmeans}} = \sum_{l=1}^{K} \sum_{x_i \in \mathcal{X}_l} x_i^T \mu_l$$

- Mutual information

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log\left(\frac{p(x,y)}{p_1(x)\,p_2(y)}\right),$$

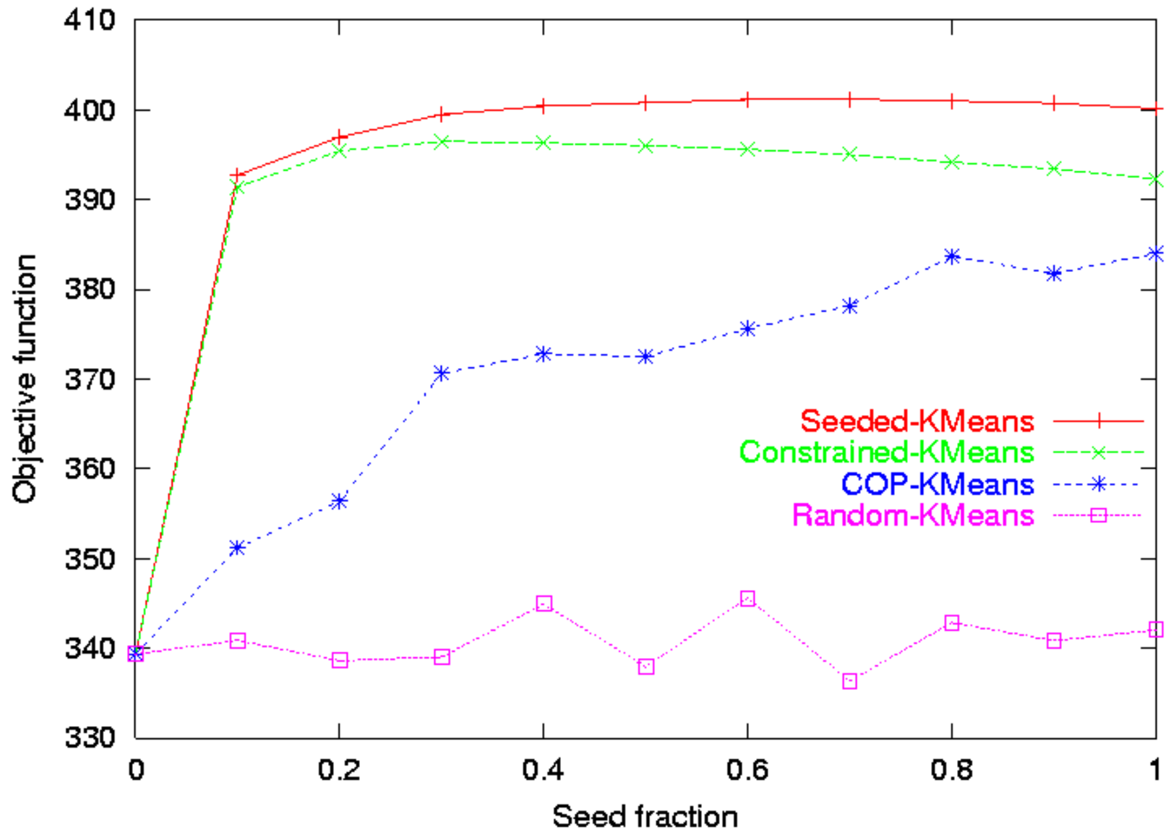
Results: MI and Seeding



Zero noise in seeds [Small-20 NewsGroup]

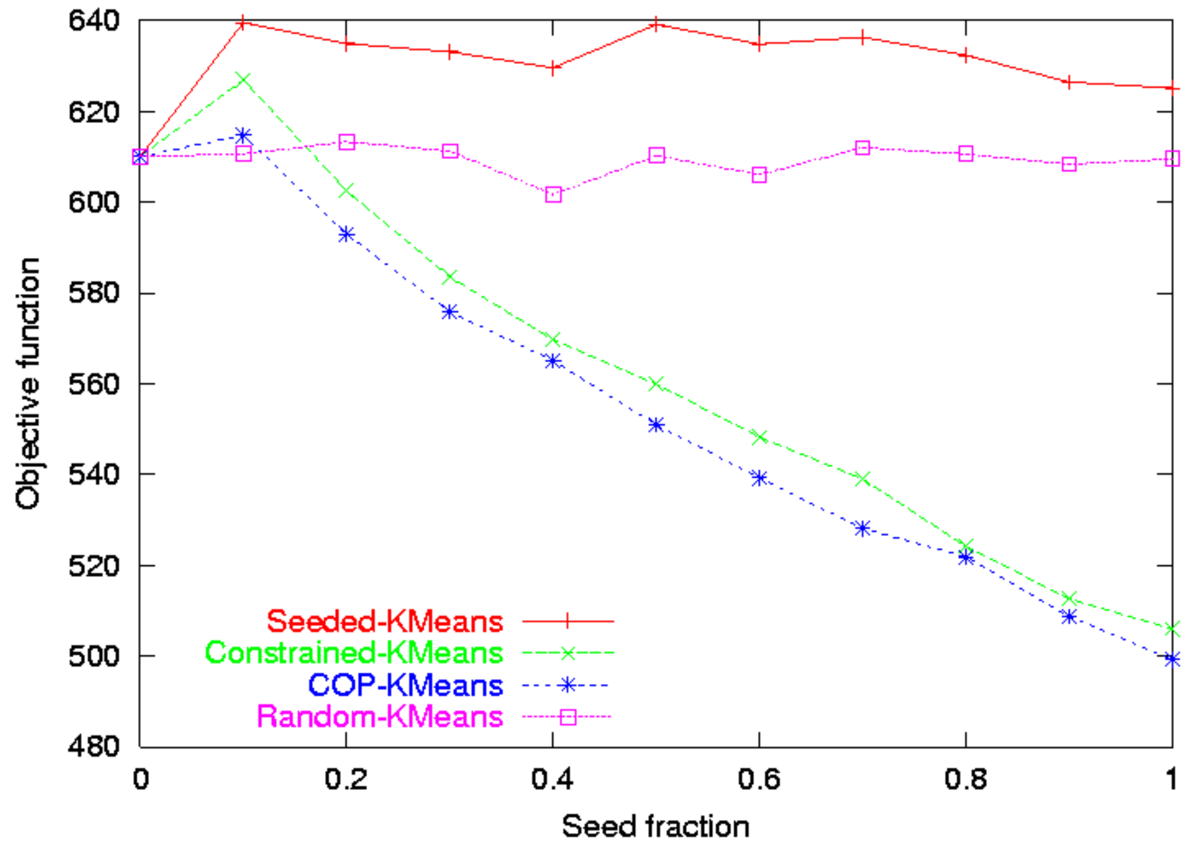
- Semi-Supervised KMeans substantially better than unsupervised KMeans

Results: Objective function and Seeding



User-labeling consistent with KMeans assumptions
[Small-20 NewsGroup] Obj. function of data partition increases exponentially with seed fraction

Results: Objective Function and Seeding



User-labeling inconsistent with KMeans assumptions [Yahoo! News] Objective function of constrained algorithms decreases with seeding

Similarity Based Methods

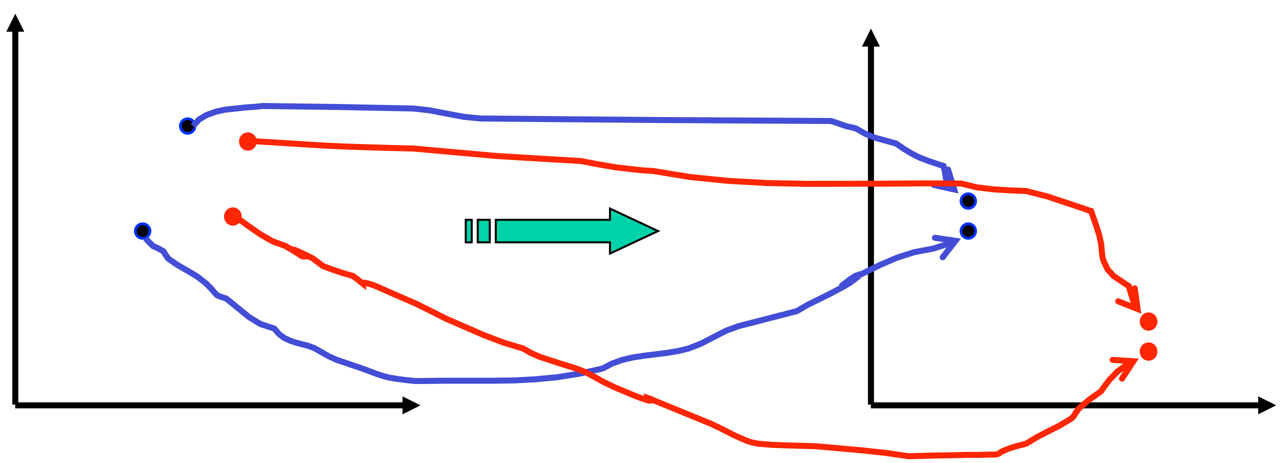
- Questions: given a set of points and the class labels, can we learn a distance matrix such that intra-cluster distance are minimized and inter-cluster distance are maximized?

Distance metric learning

Define a new distance measure of the form:

$$d(x, y) = \|x - y\|_A = \sqrt{(x - y)^T A(x - y)} \qquad A \geq 0$$

$x \rightarrow A^{1/2}x$ Linear transformation of the original data



Distance metric learning

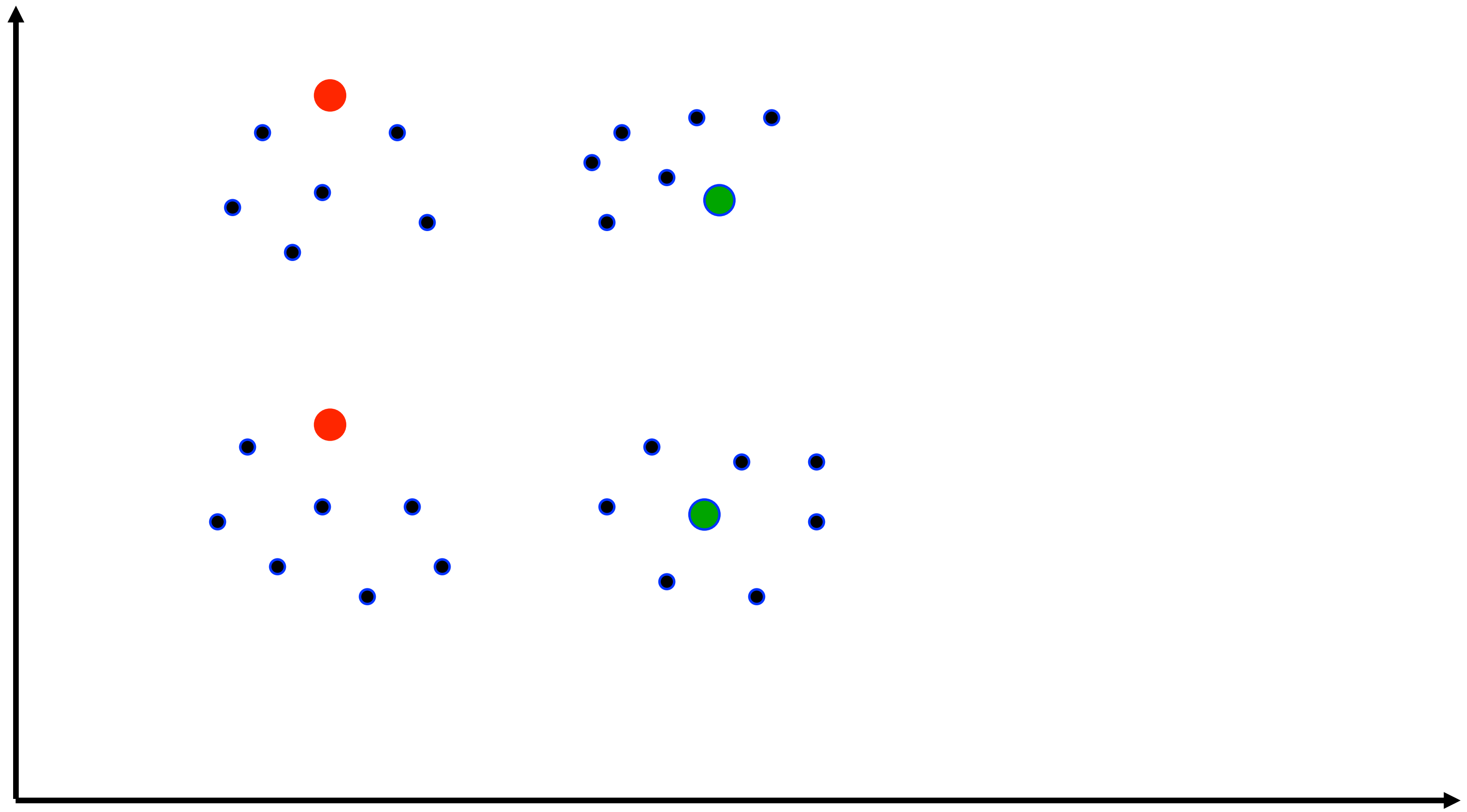
$S: (x_i, x_j) \in S$, if x_i and x_j are similar

$D: (x_i, x_j) \in D$, if x_i and x_j are disimilar

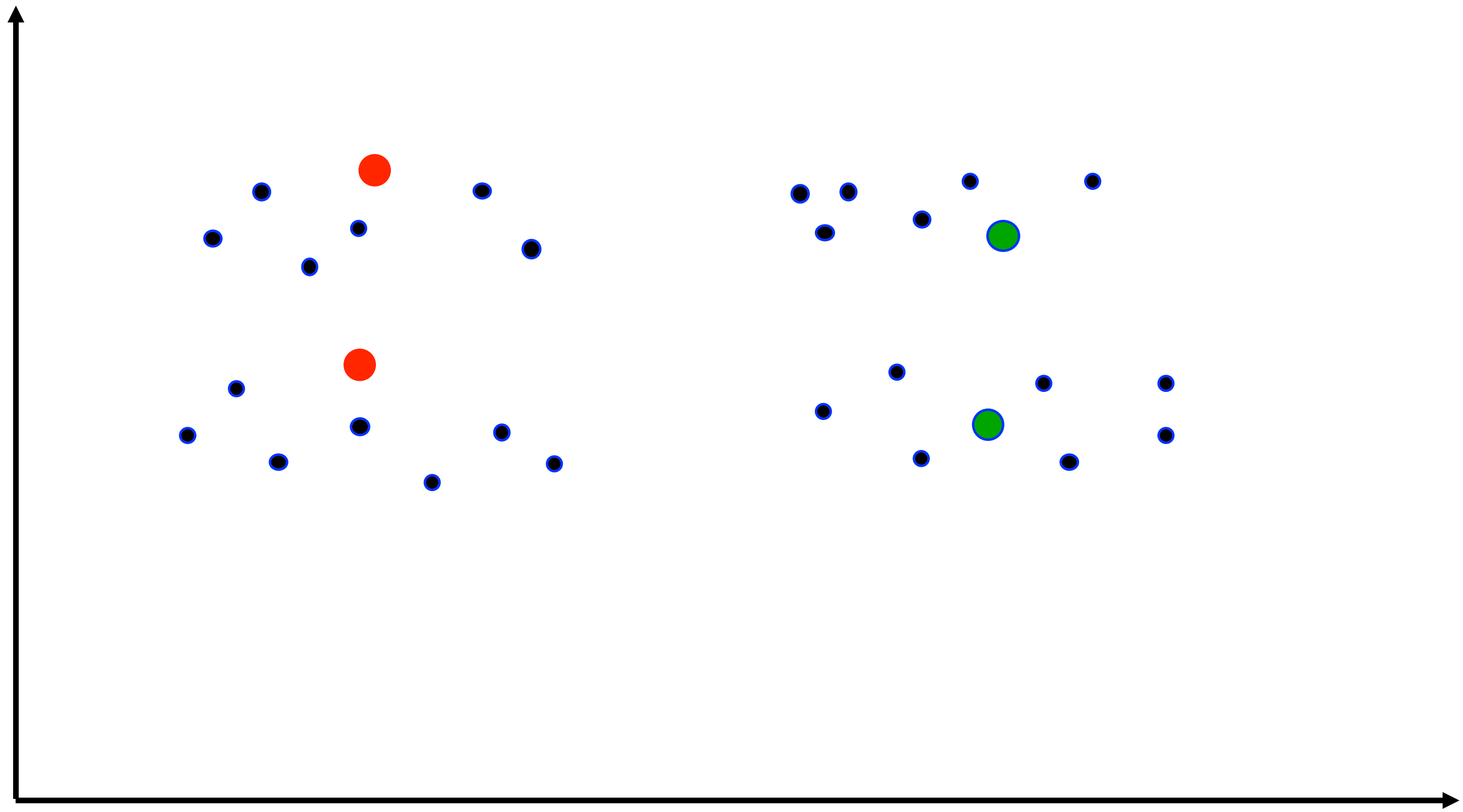
$$\left.\begin{array}{lll} (x_i, x_j) \in S, & \|x_i - x_j\|_A \text{ is small.} & \Rightarrow \sum_{(x_i, x_j) \in S} \|x_i - x_j\|_A^2 \text{ is small} \\ (x_i, x_j) \in D, & \|x_i - x_j\|_A \text{ is large.} & \Rightarrow \sum_{(x_i, x_j) \in D} \|x_i - x_j\|_A^2 \text{ is large} \end{array}\right\}$$

$$\begin{array}{ll} \min\limits_{A} & \sum_{(x_i, x_j) \in \mathcal{S}} ||x_i - x_j||_A^2 \\ \text{s.t.} & \sum_{(x_i, x_j) \in \mathcal{D}} ||x_i - x_j||_A \geq 1, \\ & A \succeq 0. \end{array}$$

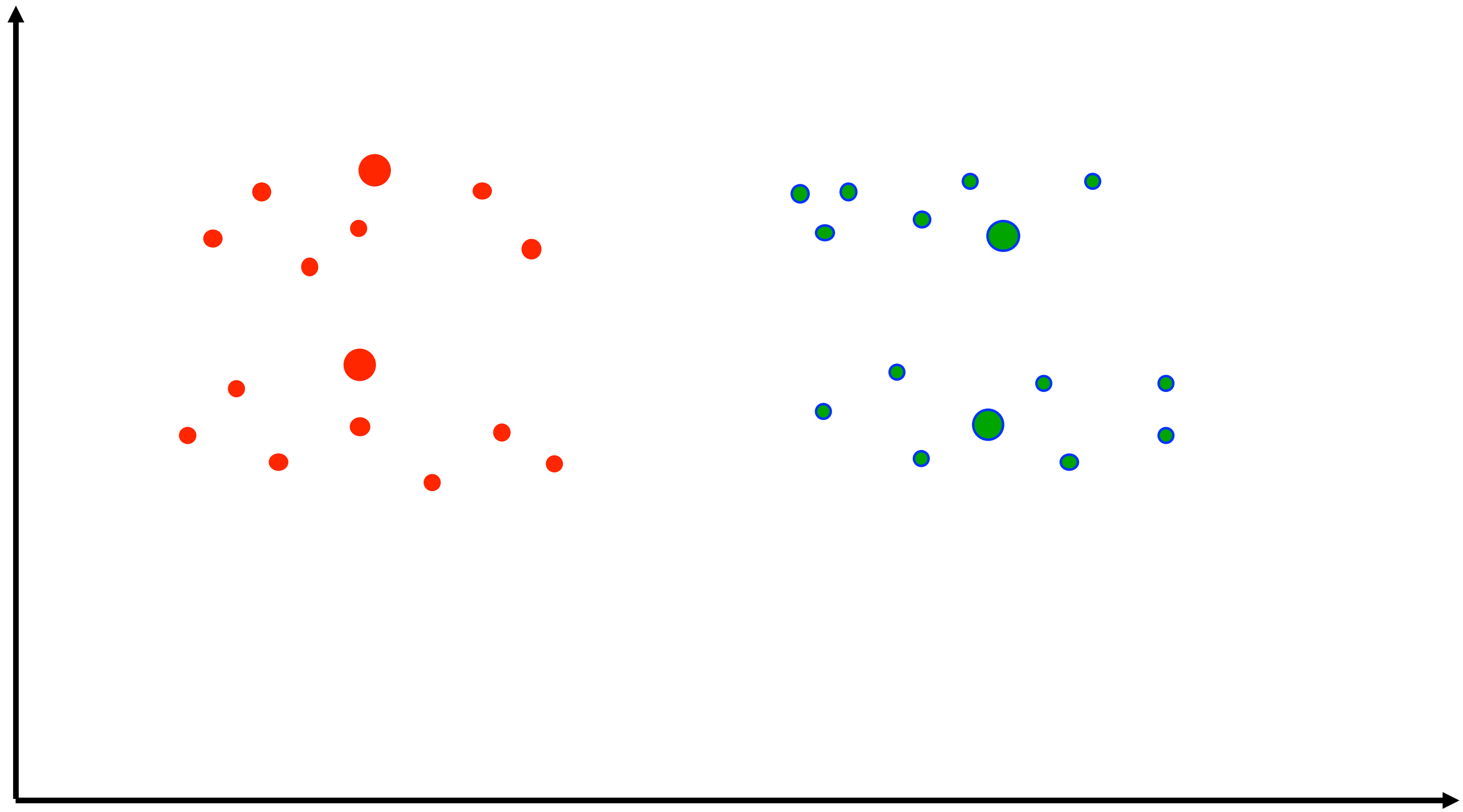
Semi-Supervised Clustering Example
Similarity Based



Semi-Supervised Clustering Example
Distances Transformed by Learned Metric



Semi-Supervised Clustering Example
Clustering Result with Trained Metric



Evaluation

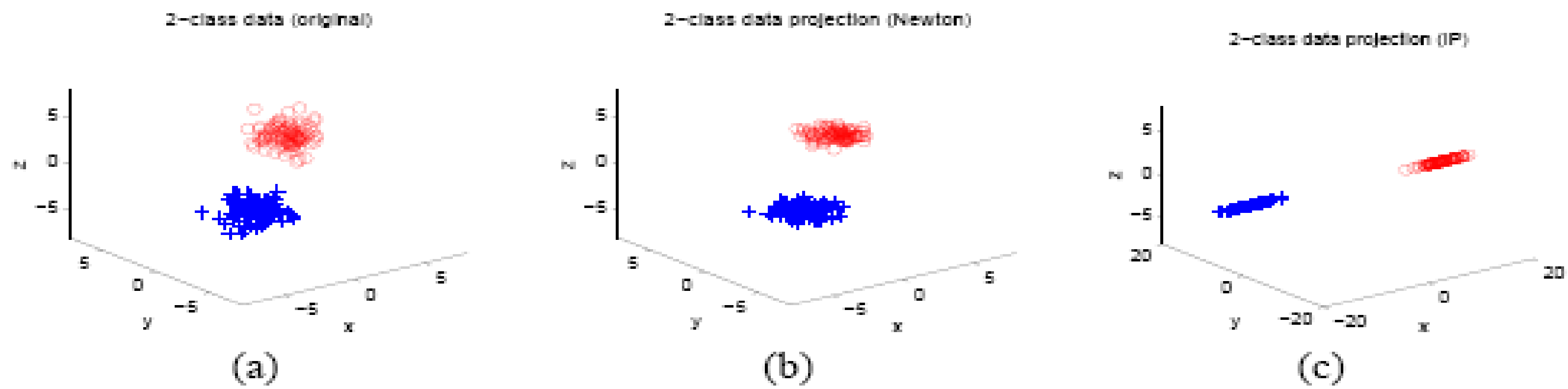


Figure 2: (a) Original data, with the different classes indicated by the different symbols (and colors, where available). (b) Rescaling of data corresponding to learned diagonal A. (c) Rescaling corresponding to full A.

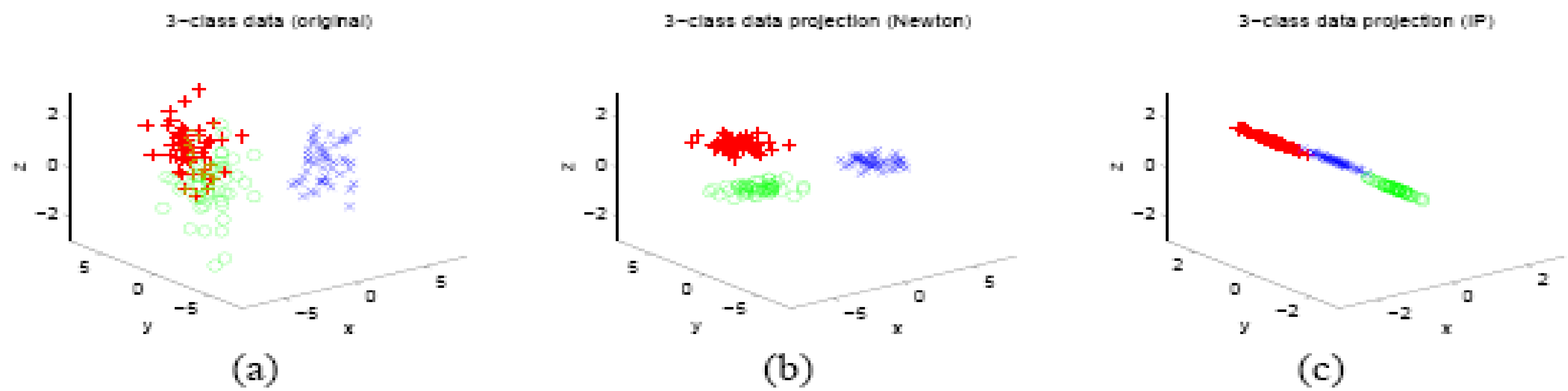


Figure 3: (a) Original data. (b) Rescaling corresponding to learned diagonal A. (c) Rescaling corresponding to full A.

Source: E. Xing, et al. Distance metric learning

Evaluation

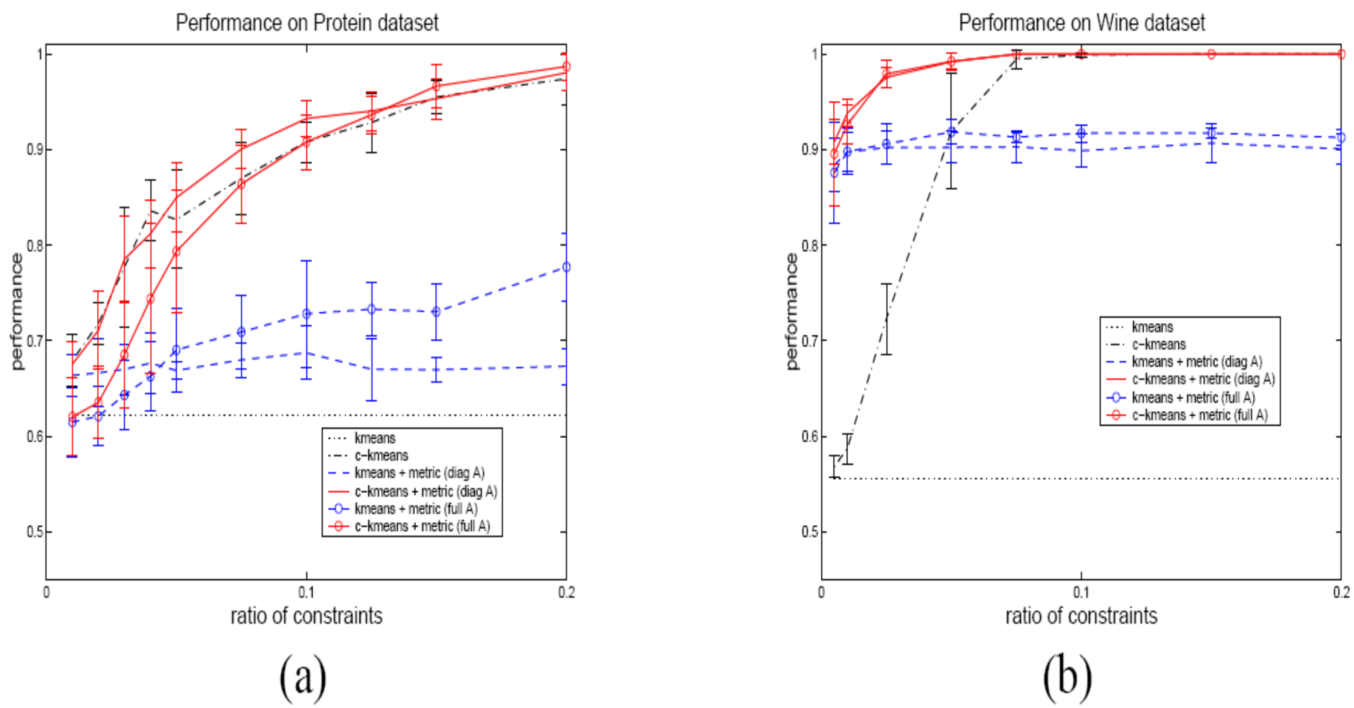


Figure 7: Plots of accuracy vs. amount of side-information. Here, the x-axis gives the fraction of all pairs of points in the same class that are randomly sampled to be included in $\mathcal{S}$.

Source: E. Xing, et al. Distance metric learning

Additional Readings

- Combining Similarity and Search-Based Semi-Supervised Clustering "Comparing and Unifying Search-Based and Similarity-Based Approaches to Semi-Supervised Clustering", Basu, *et al.*
- Ontology based semi-supervised clustering "A framework for ontology-driven subspace clustering", Liu *et al*.

References

- UT machine learning group
  - http://www.cs.utexas.edu/~ml/publication/unsupervised.html
- Semi-supervised Clustering by Seeding
  - http://www.cs.utexas.edu/users/ml/papers/semi-icml-02.pdf
- Constrained K-means clustering with background knowledge
  - http://www.litech.org/~wkiri/Papers/wagstaff-kmeans-01.pdf
- Some slides are from Jieping Ye at Arizona State