Data Mining & Machine Learning

CS37300
Purdue University

November 17, 2017

Descriptive modeling: representation

Data mining components

- Task specification: **Description**
- Data representation: **Homogeneous IID data**
- Knowledge representation
- Learning technique

Descriptive models

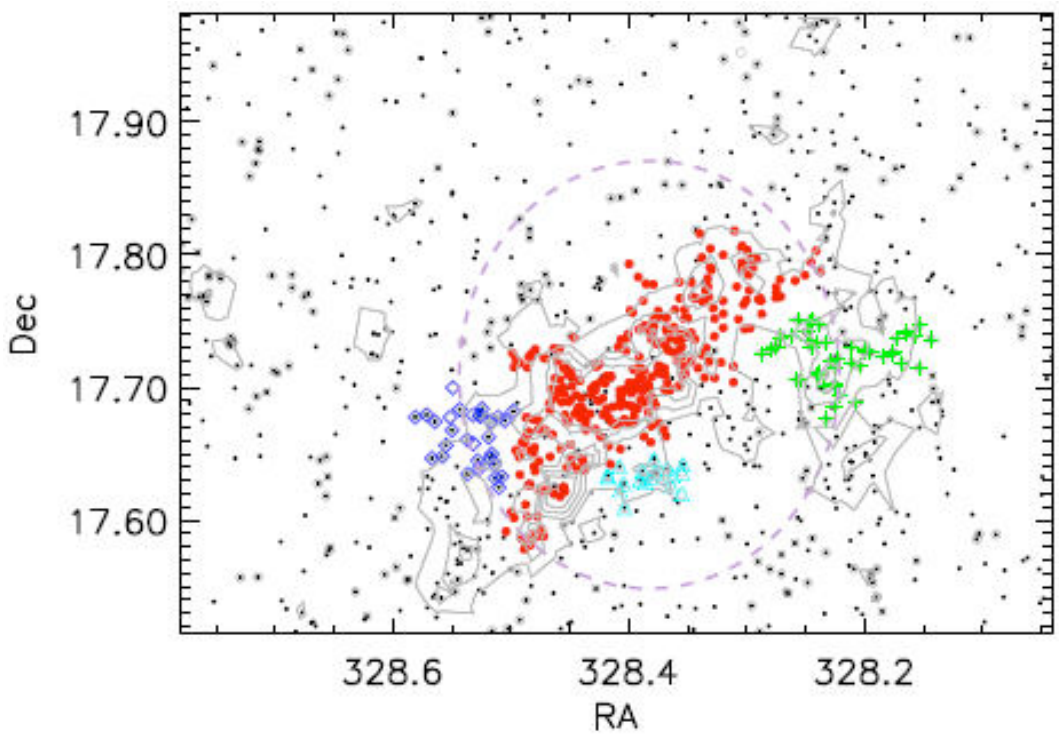
- Descriptive models summarize the data
  - Global summary
  - Model main features of the data
- Two main approaches:
  - Density estimation
    - Explicit models (e.g., data is a high-dimensional Gaussian)
    - Implicit models, which just generate more examples: RBMs, GANs
  - Cluster analysis (*we will now focus on this topic*)

Modeling task

- Data representation: training set of ***x**(i) instances*
- Task—depends on approach
  - Clustering: partition the instances into groups of similar instances
  - Density estimation: determine a compact representation of the full joint distribution $P(\mathbf{X})=P(X_1,X_2,...,X_p)$

Cluster analysis

- Decompose or partition instances into groups s.t.:
  - **Intra**-group similarity is ***high***
  - **Inter**-group similarity is ***low***
- Measure of distance/ similarity is crucial

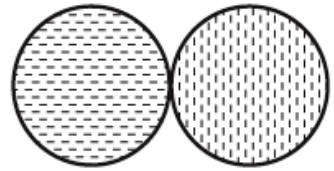


Cluster analysis

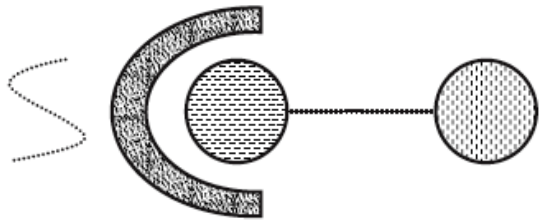
- Huge body of work
  - Also known as unsupervised learning, segmentation, etc.
- Difficult to evaluate success
  - If goal is to find “interesting” clusters, then it is difficult to quantify
  - If goal is to find “similar” clusters, then success depends on distance measure (circular)

What makes a "good" cluster?

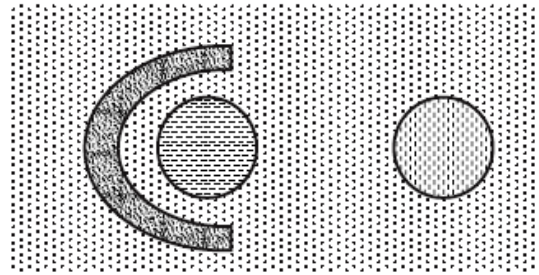
(a) Well-separated clusters. Each point is closer to all of the points in its cluster than to any point in another cluster.



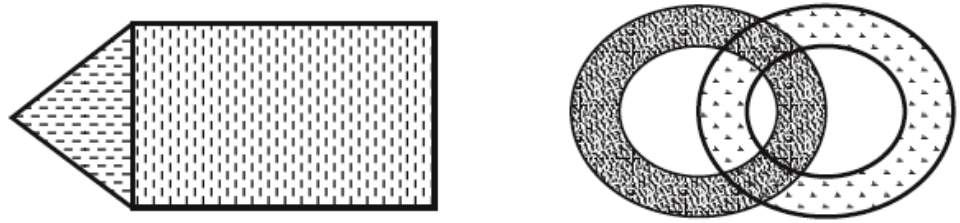
(b) Center-based clusters. Each point is closer to the center of its cluster than to the center of any other cluster.



(c) Contiguity-based clusters. Each point is closer to at least one point in its cluster than to any point in another cluster.



(d) Density-based clusters. Clusters are regions of high density separated by regions of low density.



(e) Conceptual clusters. Points in a cluster share some general property that derives from the entire set of points. (Points in the intersection of the circles belong to both.)

Figure 8.2. Different types of clusters as illustrated by sets of two-dimensional points.

Application examples

- **Sense-making**: Understand the data and its dynamics. Are the website users interacting with the functionalities the way they should?
- **Marketing**: discover distinct groups in customer base to develop targeted marketing programs
- **Land use**: identify areas of similar use in an earth observation database to understand geographic similarities
- **City-planning**: group houses according to house type, value, and location to identify “neighborhoods”
- **Earth-quake studies**: Group observed earthquakes to see if they cluster along continent faults

Clustering algorithms

- Types:
  - Partition-based methods
  - Hierarchical clustering (divisive/agglomerative)
  - Probabilistic model-based methods

- Different algorithms find clusters of different “shapes”
  - Appropriate shape will depend on application, match method to objectives

Algorithm examples

- K-means clustering (partition-based)
- Spectral clustering (hierarchical-divisive)
- Nearest neighbor clustering (hierarchical-agglomerative)
- Mixture models (probabilistic model-based)

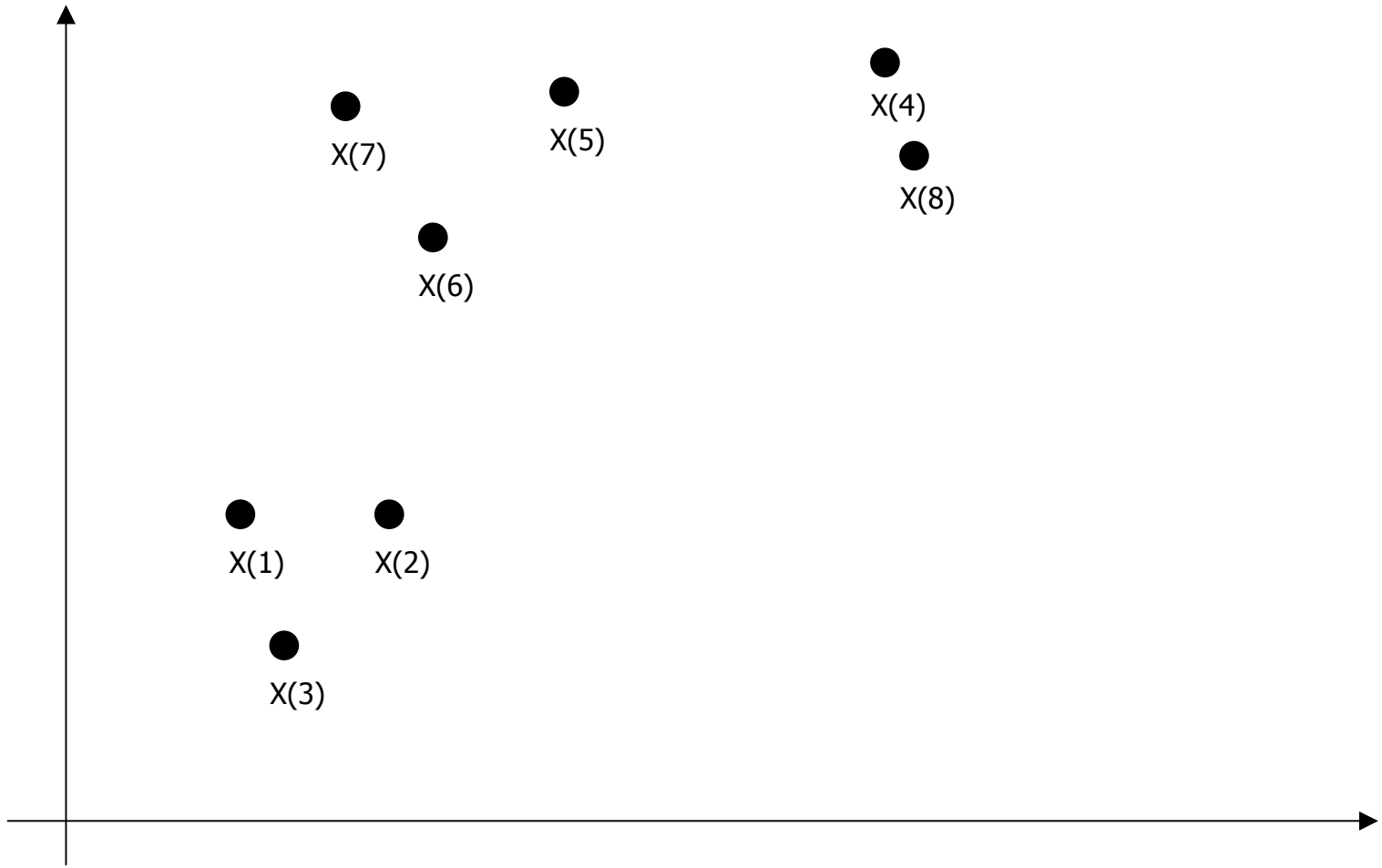
Partition-based clustering

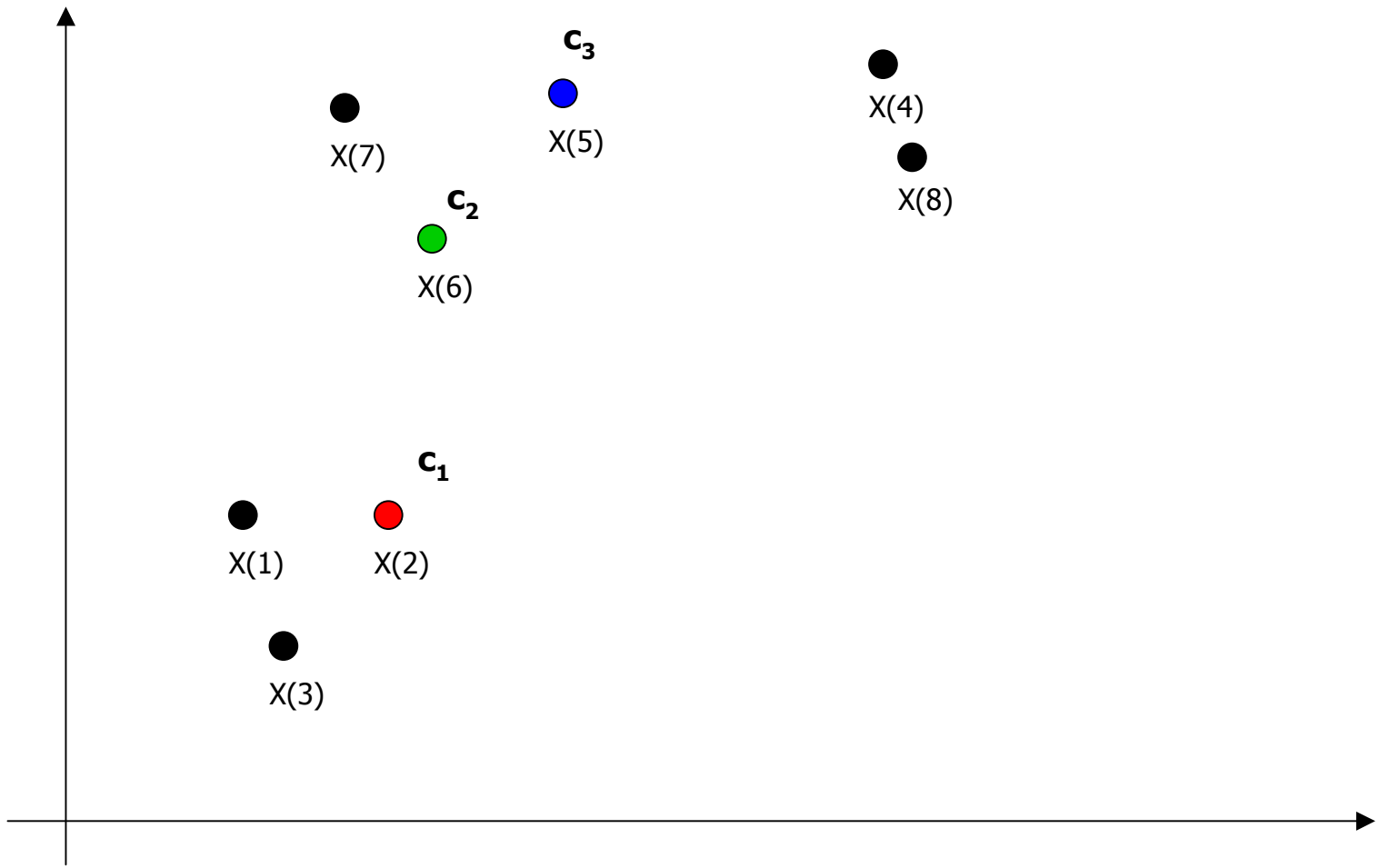
Partition-based

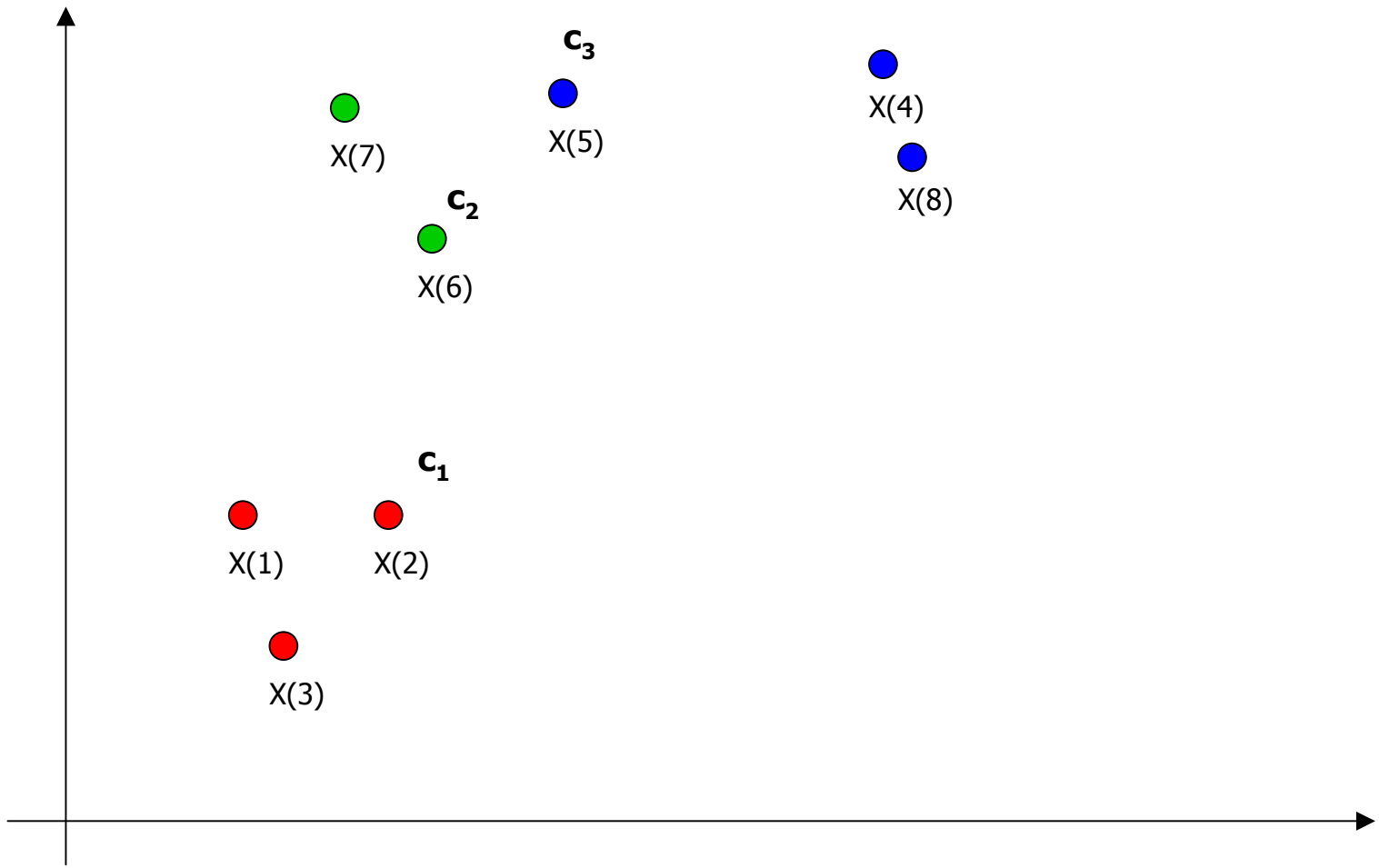
- Input: data D={**x**(1),**x**(2),...,**x**(n)}
- Output: k clusters C={C_1,...,C_k} such that each x(i) is assigned to a unique C_j
- Evaluation: Score(C,D) is maximized/minimized
  - Combinatorial optimization: search among n^k allocations of n objects into k classes to maximize score function
  - Exhaustive search is intractable
  - Most approaches use iterative improvement algorithms

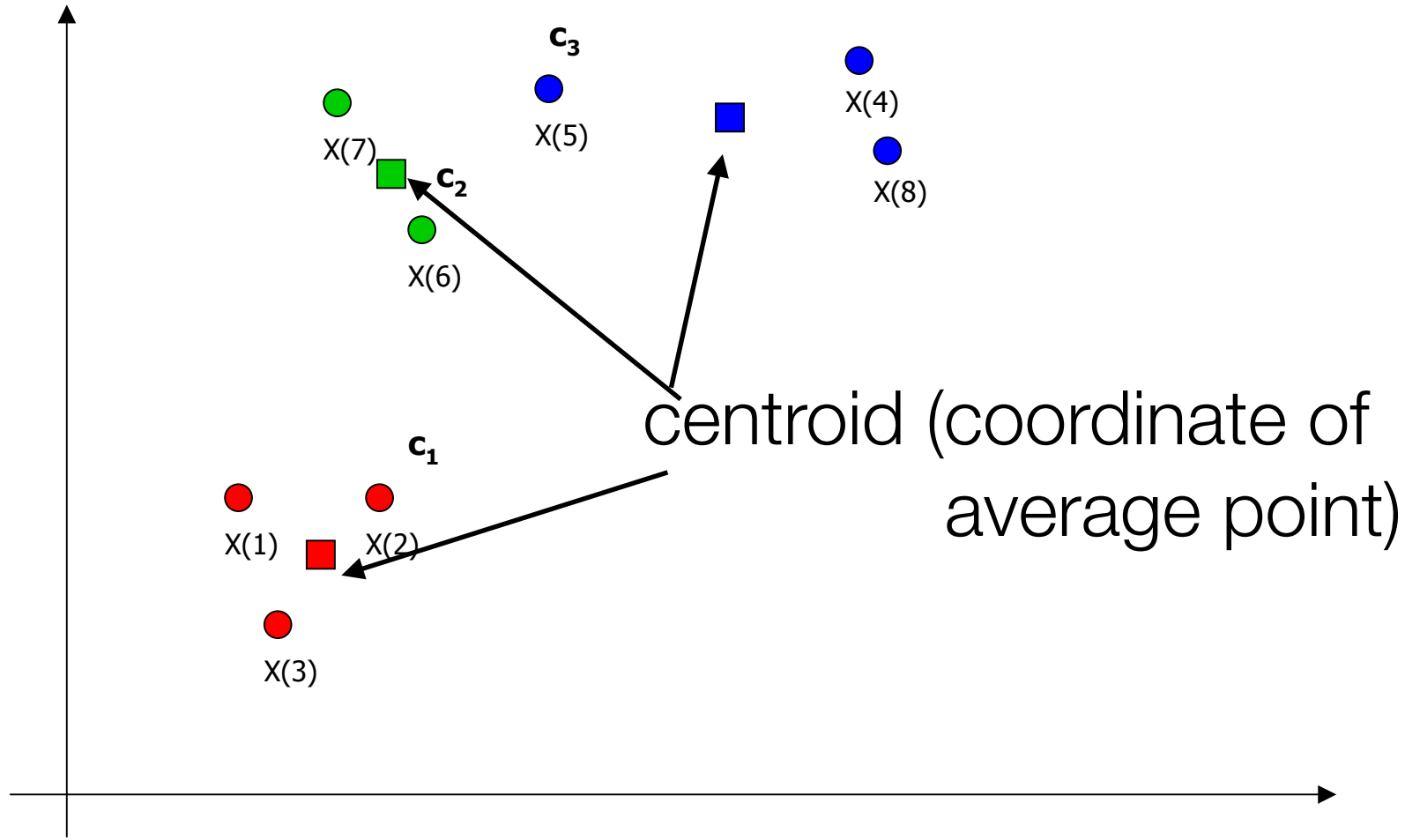
Example: K-means

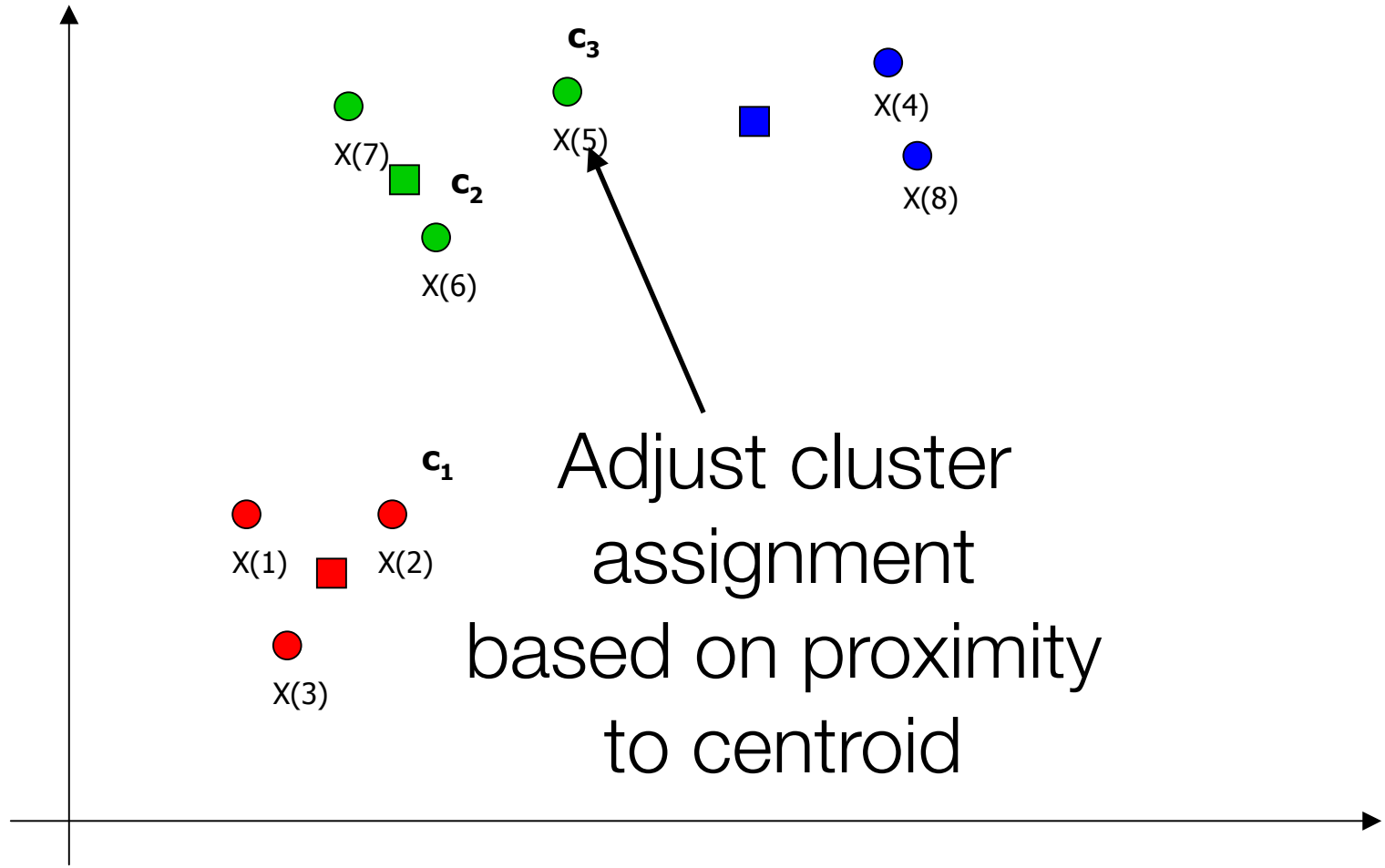
- Algorithm idea:
  - Start with k randomly chosen centroids
  - Repeat until no changes in assignments
    - Assign instances to closest centroid
    - Recompute cluster centroids

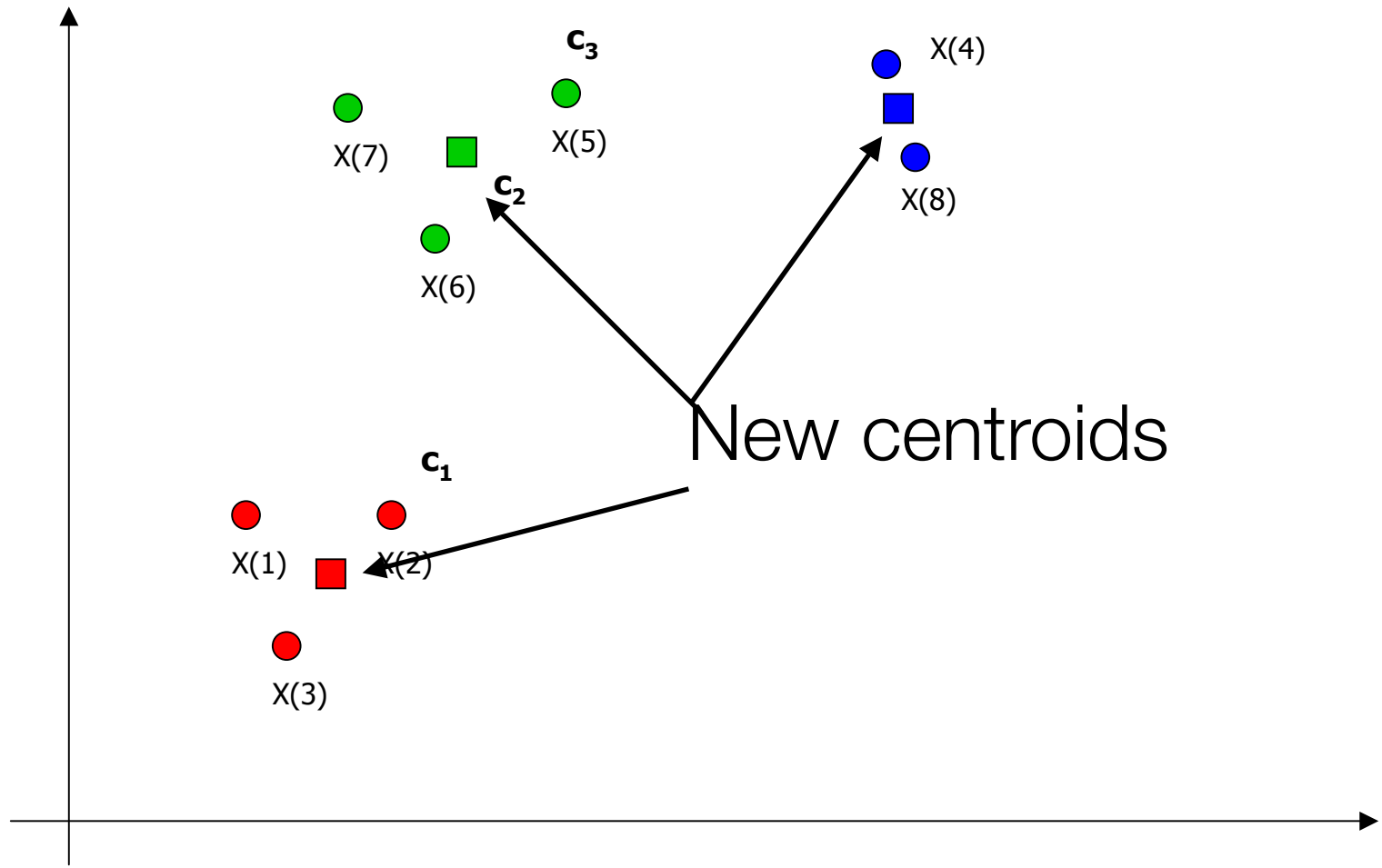

X(7)
X(5)
X(4)
X(8)
X(6)
X(1)
X(2)
X(3)


c_3
X(7)
X(5)
X(4)
X(8)
c_2
X(6)
c_1
X(1)
X(2)
X(3)

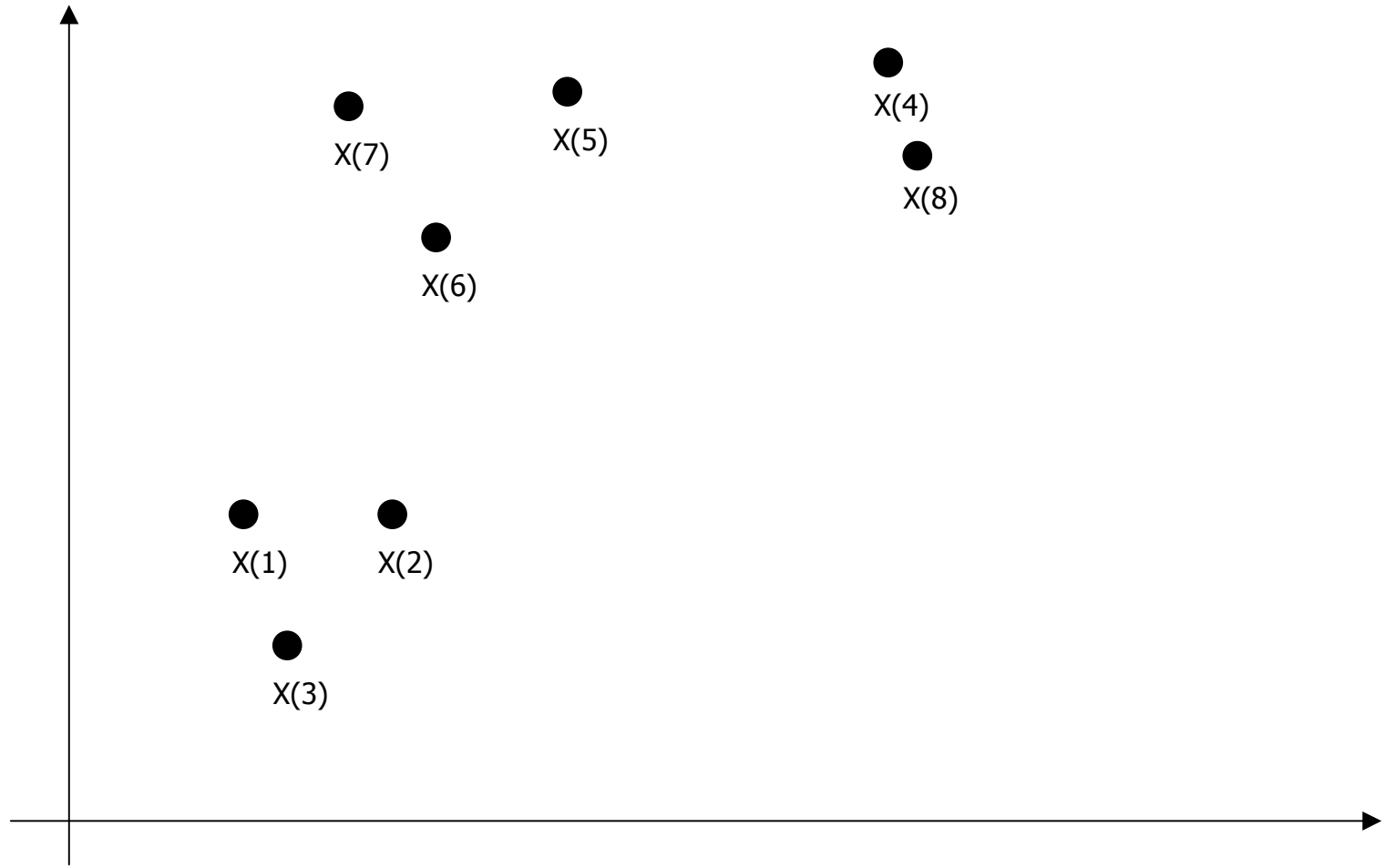

c_3
X(4)
X(5)
X(7)
X(8)
c_2
X(6)
c_1
X(1)
X(2)
X(3)

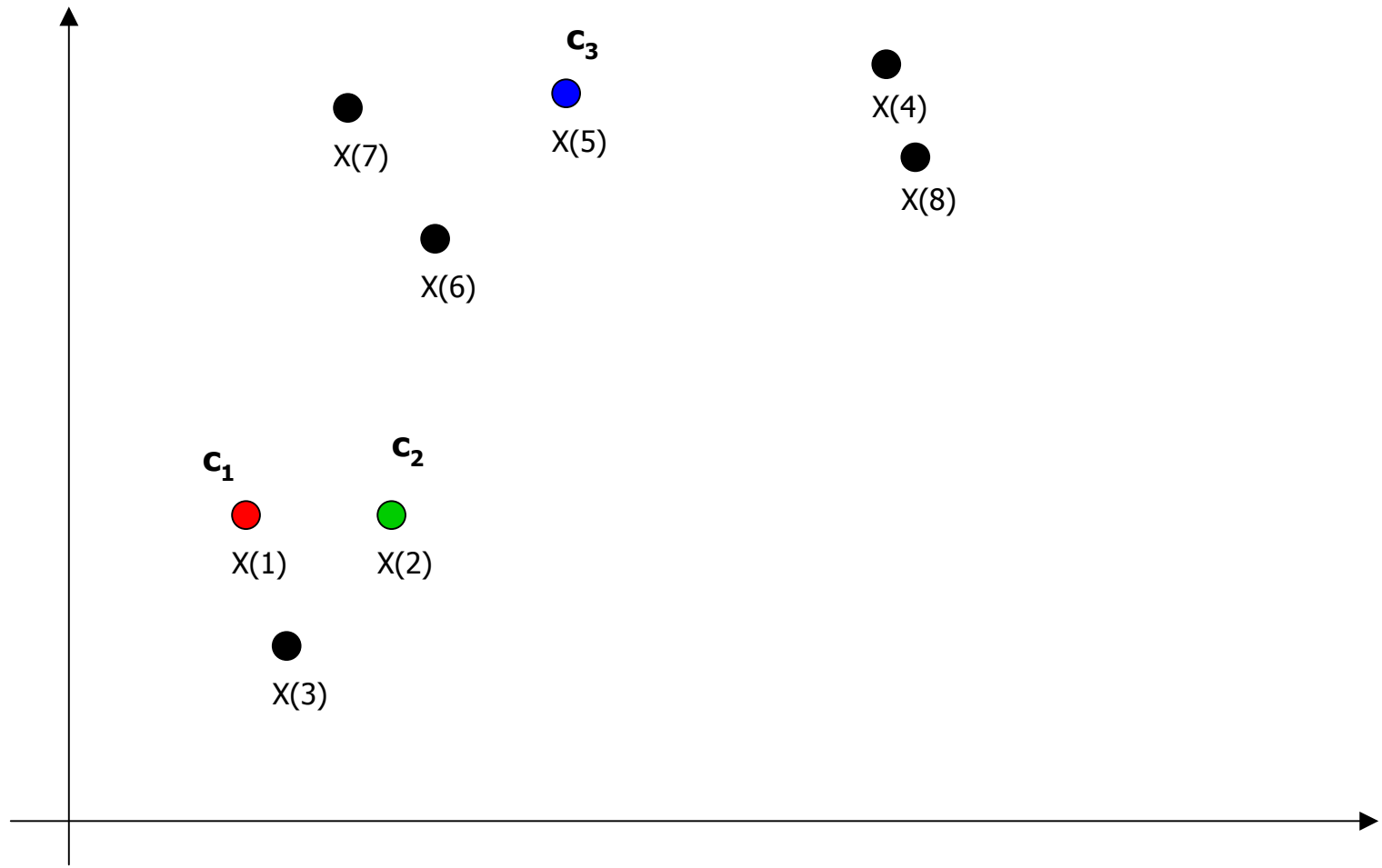

c3
X(5)
X(4)
X(7)
c2
X(8)
X(6)
centroid (coordinate of
average point)
c1
X(1)
X(2)
X(3)

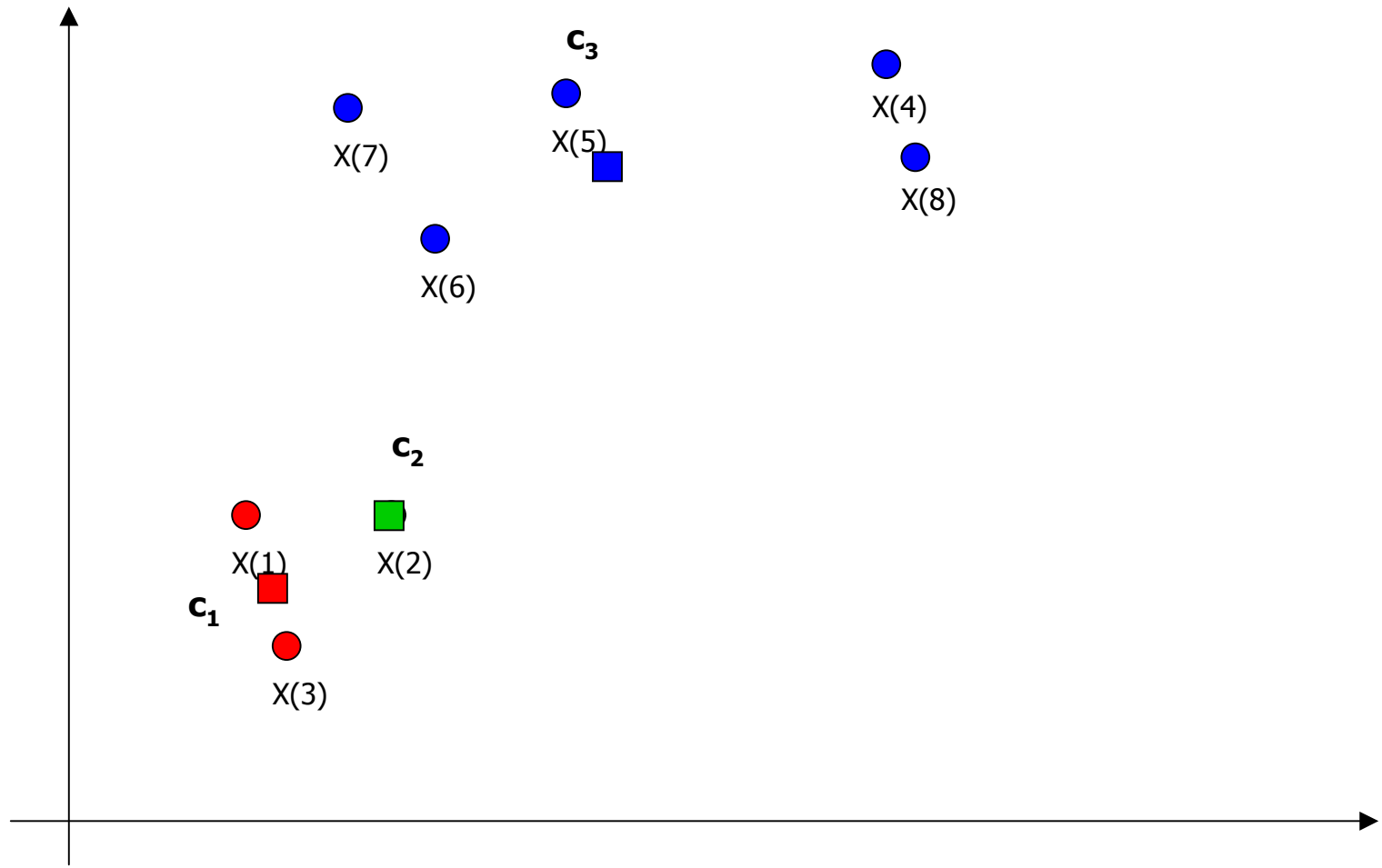

c3
X(7)
X(5)
X(4)
c2
X(8)
X(6)
c1
X(1)
X(2)
X(3)
Adjust cluster assignment based on proximity to centroid


c_3
X(4)
X(7)
X(5)
c_2
X(8)
X(6)
New centroids
c_1
X(1)
X(2)
X(3)

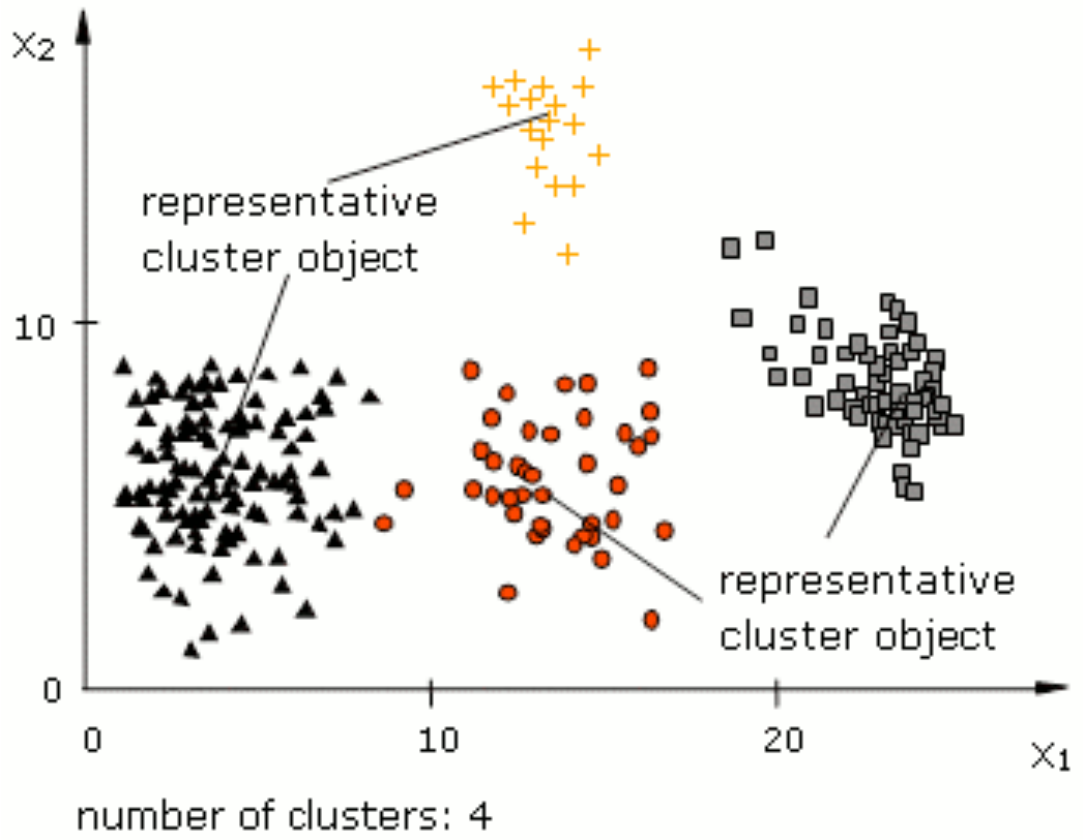
K-means example II


X(7)
X(5)
X(4)
X(8)
X(6)
X(1)
X(2)
X(3)


C3
X(7)
X(5)
X(4)
X(8)
X(6)
C1
C2
X(1)
X(2)
X(3)


c_3
X(7)
X(5)
X(4)
X(8)
X(6)
c_2
X(1)
X(2)
c_1
X(3)

K-means



Groups represented by *canonical* item description(s)

Clustering score functions

- Goal:
  - Compact clusters: minimize within cluster distance
  - Separated clusters: maximize between cluster distance
- Score(C,D) = *f(wc(C), bc(C))*
  - Score measures quality of clustering C for dataset D
  - *Many score functions are a combination of within-cluster (wc) and between-cluster (bc) distance measures*

Clustering score functions

- Score(C,D) = *f(wc(C), bc(C))*

cluster centroid: $$r_k = \frac{1}{n_k} \sum_{x(i) \in C_k} x(i)$$

between-cluster distance: $$bc(C) = \sum_{1 \le j < k \le K} d(r_j, r_k)^2$$

within-cluster distance: $$wc(C) = \sum_{k=1}^{K} wc(C_k) = \sum_{k=1}^{K} \sum_{x(i) \in C_k} d(x(i), r_k)^2$$

Clustering search

- Most learning algorithms involve iterative search over assignments due to score functions which require combinatorial optimization

K-means clustering

Algorithm 2.1 The `k-means` algorithm

Input: Dataset D, number clusters k

Output: Set of cluster representatives C, cluster membership vector $\mathbf{m}$

```
    /* Initialize cluster representatives C */
    Randomly choose k data points from D
    Use these k points as initial set of cluster representatives C
    repeat
        /* Data Assignment */
        Reassign points in D to closest cluster mean
        Update m such that m_i is cluster ID of ith point in D
        /* Relocation of means */
        Update C such that c_j is mean of points in jth cluster
    until convergence of objective function sum_{i=1}^{N} (argmin_j ||x_i - c_j||_2^2)
```

Score function: $$wc(C) = \sum_{k=1}^{K} wc(C_k) = \sum_{k=1}^{K} \sum_{x(i) \in C_k} d(x(i), r_k)^2$$