Clustering

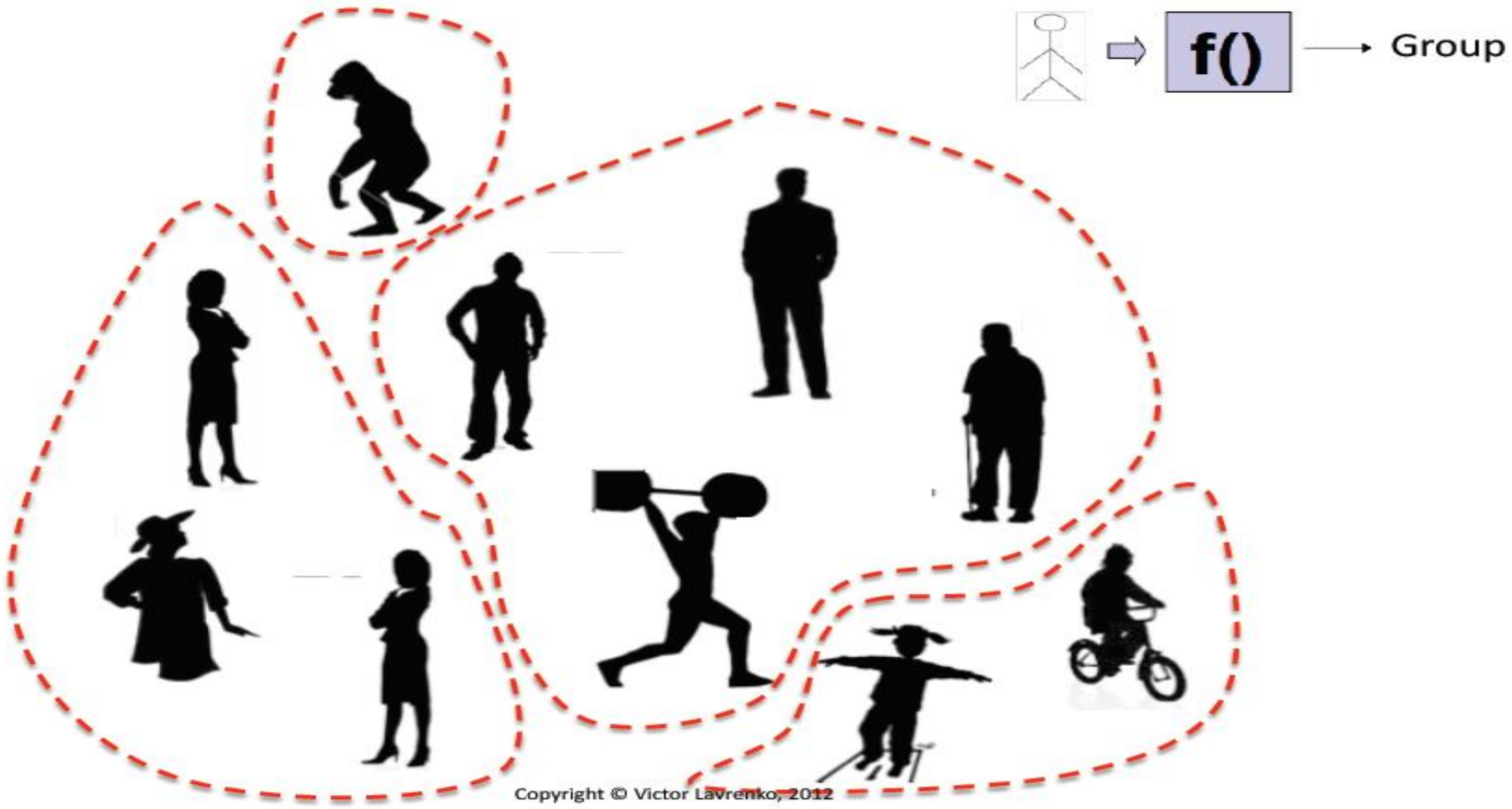
Slide credits : Vineeth N Balasubramanian

आई आई टी हैदराबाद
IIT Hyderabad

ML Problems

	Supervised Learning	Unsupervised Learning
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

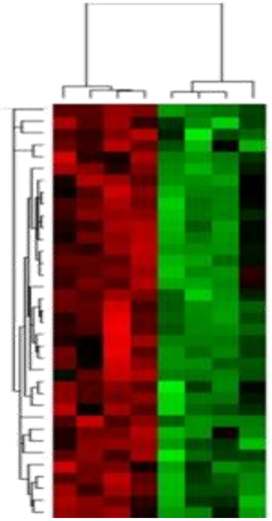
Clustering (Unsupervised Learning)

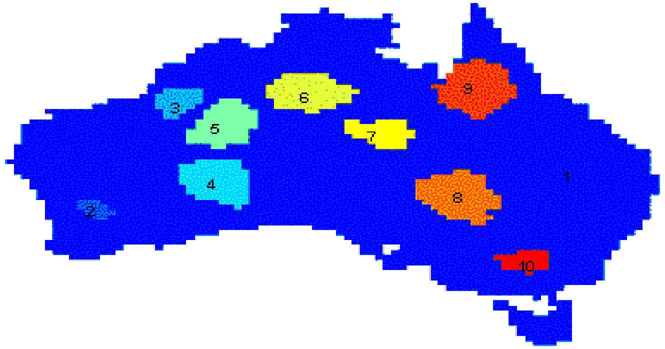


Where is Clustering used?

- **Understanding**
  - Group genes and proteins that have similar functionality, or group stocks with similar price fluctuations
- **Summarization**
  - Reduce the size of large data sets

More real-world applications?





Clustering precipitation in Australia

Where is Clustering Used?

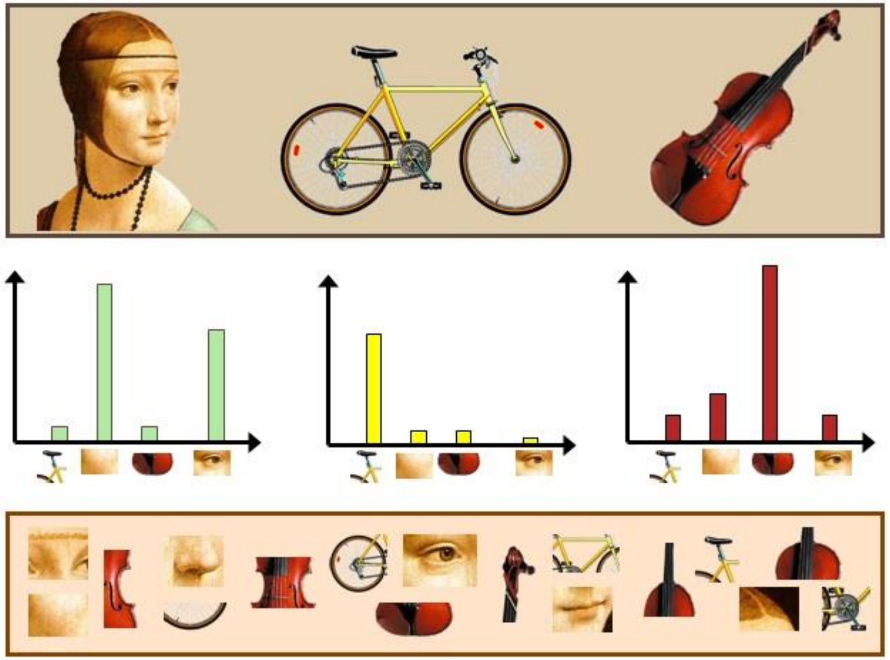
- Bank/Internet Security: fraud/spam pattern discovery
- Biology: taxonomy of living things such as kingdom, phylum, class, order, family, genus and species
- City-planning: Identifying groups of houses according to their house type, value, and geographical location
- Climate change: understanding earth climate, find patterns of atmospheric and ocean
- Finance: stock clustering analysis to uncover correlation among underlying shares
- Image Compression/segmentation: coherent pixels grouped
- Information retrieval/organization: Google search, topic-based news
- Land use: Identification of areas of similar land use in an earth observation database
- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- Social network mining: special interest group automatic discovery

Clustering: Objectives

- Discover underlying structure of data
- What sub-populations exist in the data?
  - How many are there?
  - What are their sizes?
  - Do elements in a sub-population have any common properties?
  - Are sub-populations cohesive? Can they be further split?
  - Are there outliers?

Clustering as Preprocessing

- Popular application of clustering
- Estimated group labels h_j (soft) or b_j (hard) may be seen as the dimensions of a new k dimensional space, where we can then learn our discriminant or regressor
- E.g. Bag-of-words representation in images

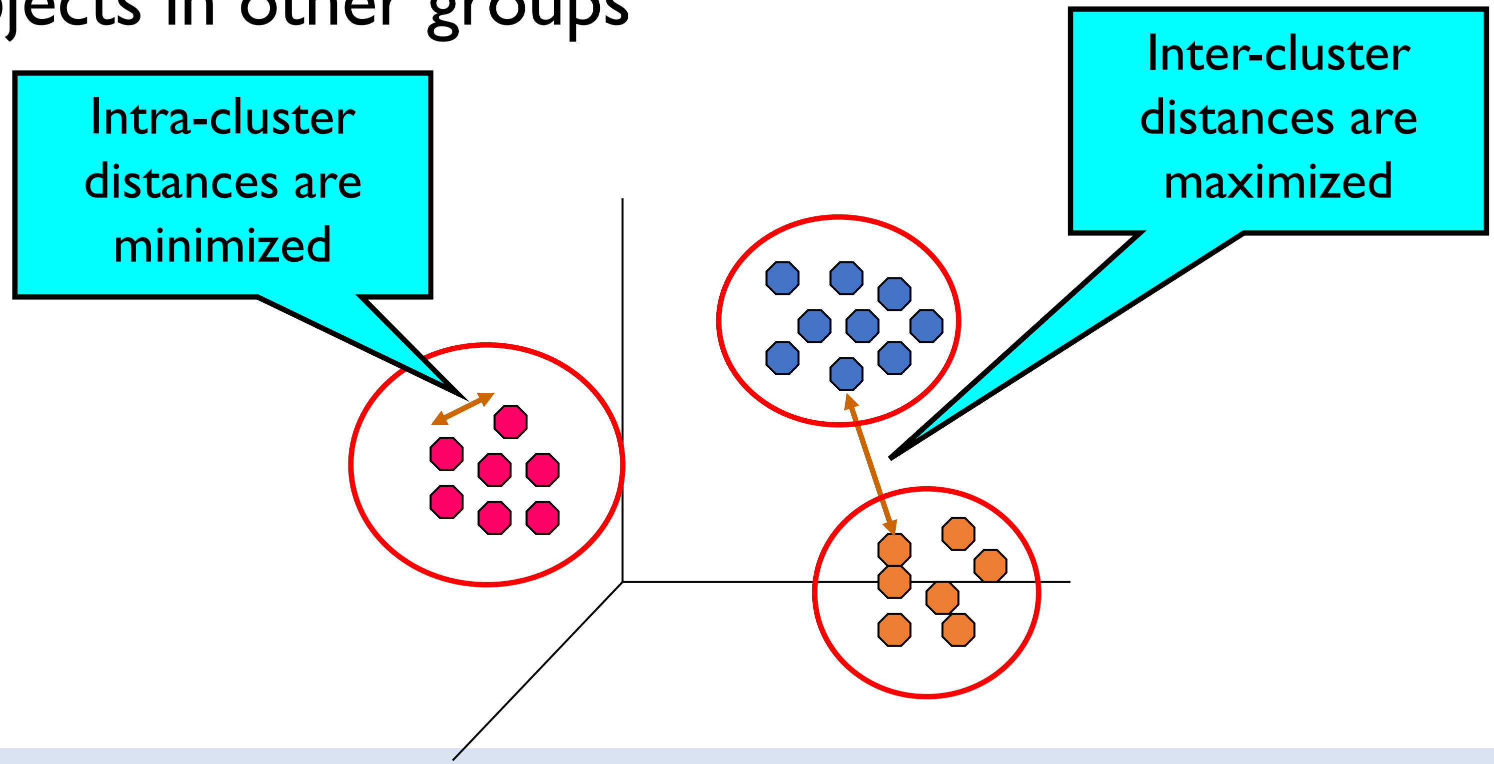


Types of Clustering Methods

- **In terms of objective:**
  - Monothetic: cluster members have some common property
    - E.g. All are males aged 20-35, or all have X% response to test B
  - Polythetic: cluster members are similar to each other
    - Distance between elements defines membership
- **In terms of overlap of clusters**
  - Hard clustering: clusters do not overlap
  - Soft clustering: clusters may overlap
    - “Strength of association” between element and cluster
- **In terms of methodology**
  - Flat/partitioning (vs) hierarchical: Set of groups (vs) taxonomy
  - Density-based (vs) Model/Distribution-based: DBSCAN vs GMMs
  - Connectionist (vs) Centroid-based: k-means vs Hierarchical clustering

Clustering Methods

- Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups
- How?



Outline

- K-Means
- Hierarchical Clustering
- Model-based Clustering (GMM and Expectation Maximization)
- Evaluation of Clustering Algorithms

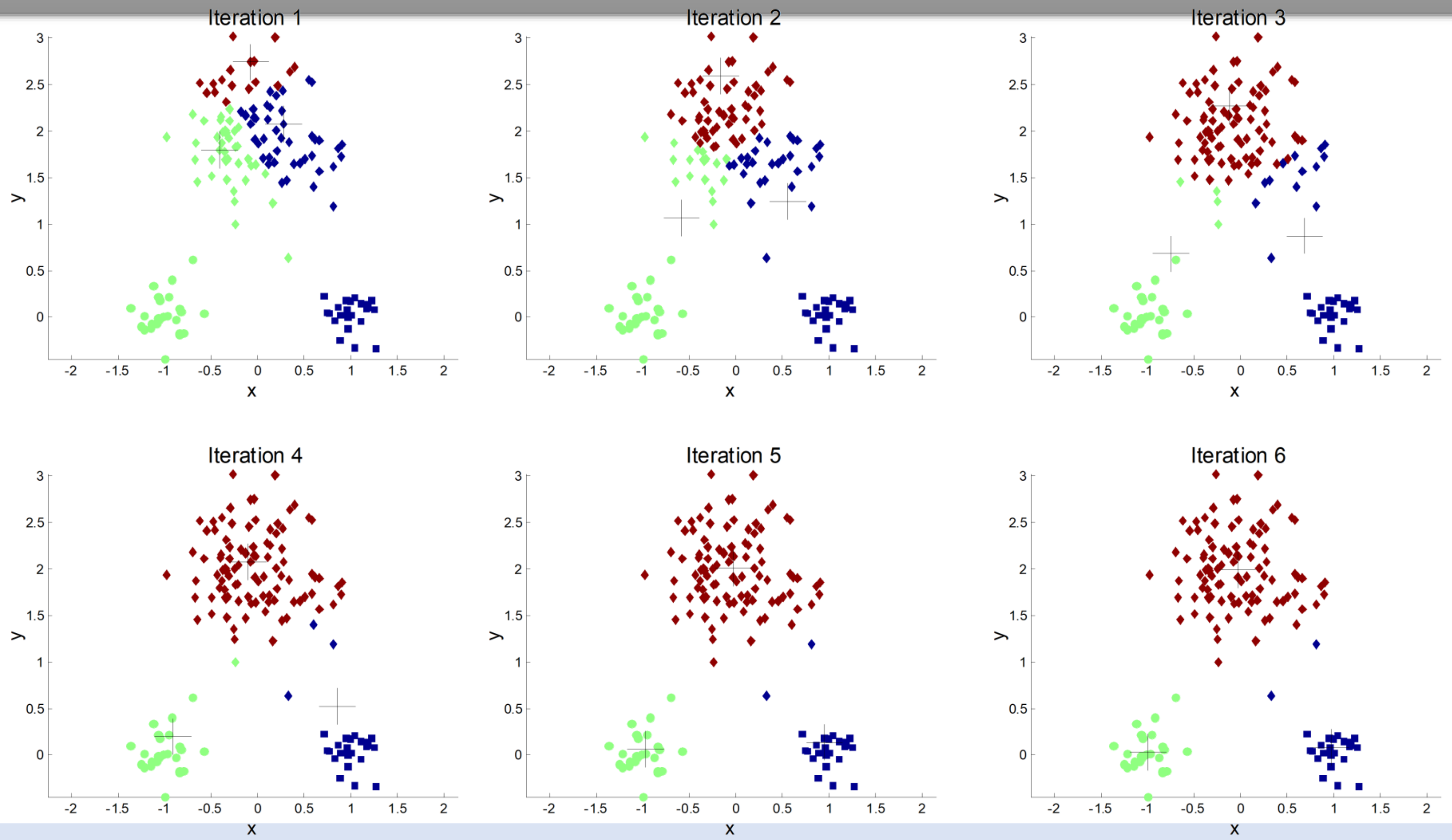
k-Means Clustering

- Partitional clustering approach
- Each cluster is associated with a centroid (center point)
- Each point is assigned to the cluster with the closest centroid
- chicken-and-egg problem
- Number of clusters, K, must be specified
- The basic algorithm is very simple

Select K points as the initial centroids.
repeat
    Form K clusters by assigning all points to the closest centroid.
    Recompute the centroid of each cluster.
until The centroids don't change

---

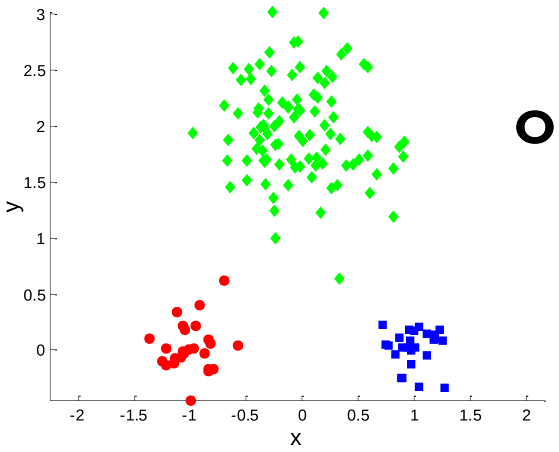
k-Means: Illustration



k-Means Clustering

- Initial centroids are often chosen randomly.
  - Clusters produced can vary from one run to another.
  - The centroid is (typically) the mean of the points in the cluster.
- '**Closeness**' is measured by Euclidean distance, cosine similarity, correlation, etc.
- K-means will converge for common similarity measures mentioned above (**local minimum** though)
- Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to 'Until relatively few points change clusters'
- Nearby points may not end up in the same cluster! Example?

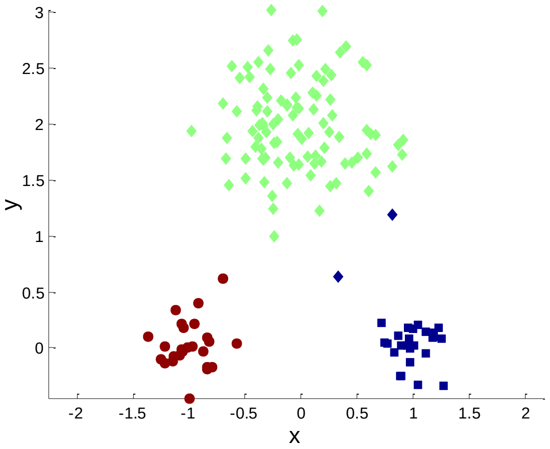
Two different k-Means clusterings

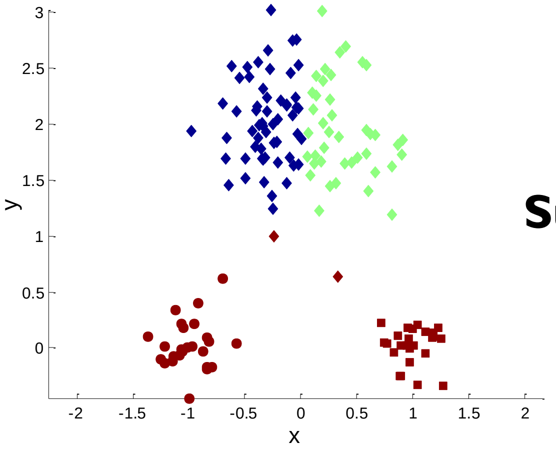


Original Points

What's the problem?

Optimal Clustering





Sub-optimal Clustering

Selecting Initial Centroids

- If there are K 'real' clusters then the chance of selecting one centroid from each cluster is small.
  - Chance is relatively small when K is large
  - If clusters are the same size, n, then

$$P = \frac{\text{number of ways to select one centroid from each cluster}}{\text{number of ways to select } K \text{ centroids}} = \frac{K!n^K}{(Kn)^K} = \frac{K!}{K^K}$$

  - For example, if K = 10, then probability = $10!/10^{10}$ = 0.00036
  - Sometimes the initial centroids will readjust themselves in 'right' way, and sometimes they don't

Possible Solutions

- Multiple runs
  - Helps, but probability is not on your side
- Sample and use hierarchical clustering to determine initial centroids
- Select more than k initial centroids and then select among these initial centroids
  - Select most widely separated
- Bisecting K-means
  - Not as susceptible to initialization issues

Evaluating k-Means Clusters

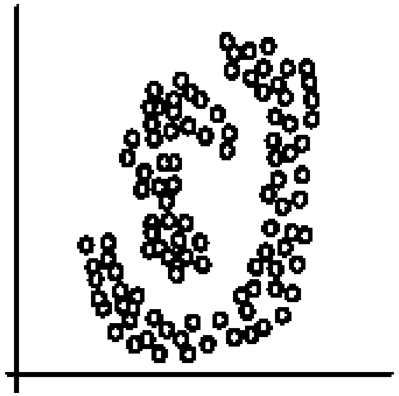
- Most common measure is Sum of Squared Error (SSE)
  - For each point, the error is the distance to the nearest cluster
  - To get SSE, we square these errors and sum them.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

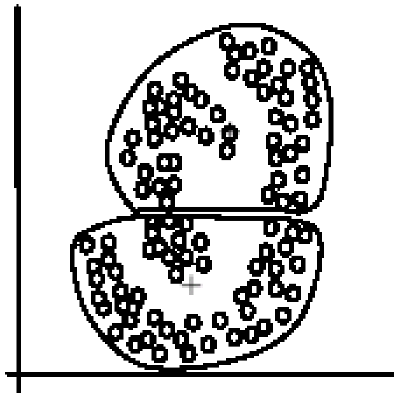
  - x is a data point in cluster Ci and m_i is the representative point for cluster Ci
  - Can show that m_i corresponds to the center (mean) of the cluster
- Given two clusterings, we can choose the one with the smaller error
- One easy way to reduce SSE is to increase K, the number of clusters
- A good clustering with smaller K can have a lower SSE than a poor clustering with higher K
- Relatively faster than other clustering methods: O(# iterations * # clusters * # instances * # dimensions)

Limitations

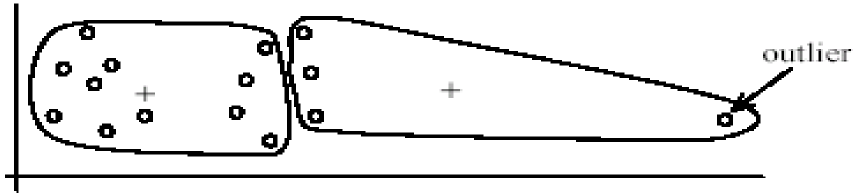
- k-Means has problems when clusters are of differing
  - Sizes, Densities, Non-globular shapes
- Sensitive to outliers
- The number of clusters (K) is difficult to determine



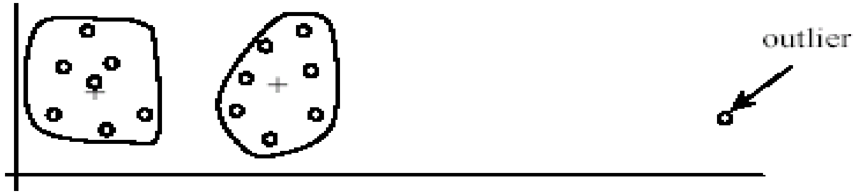
(A): Two natural clusters



(B): *k*-means clusters



(A): Undesirable clusters



(B): Ideal clusters

Extensions

- Use of various distance metrics
  - Euclidean distance $$d(\mathbf{x},\mathbf{y}) = \sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2 \cdots + |x_n - y_n|^2}$$
  - Manhattan (city-block) distance $$d(\mathbf{x},\mathbf{y}) = |x_1 - y_1| + |x_2 - y_2| \cdots + |x_n - y_n|$$
  - Cosine distance $$\cos(\mathbf{x},\mathbf{y}) = \frac{x_1 y_1 + \cdots + x_n y_n}{\sqrt{x_1^2 + \cdots + x_n^2}\sqrt{y_1^2 + \cdots + y_n^2}}$$ $$d(\mathbf{x},\mathbf{y}) = 1 - \cos(\mathbf{x},\mathbf{y})$$
  - Chebyshev distance $$dist(\mathbf{x}_i,\mathbf{x}_j) = \max(|x_{i1} - x_{j1}|, |x_{i2} - x_{j2}|, \ldots, |x_{ir} - x_{jr}|)$$

K-Means as optimization problem

- minimizes the sum of squared distances from the mean to every point in the data.

$$\mathcal{L}(z, \mu; \mathbf{D}) = \sum_n \left|\left| x_n - \mu_{z_n} \right|\right|^2 \quad = \sum_k \sum_{n: z_n = k} ||x_n - \mu_k||^2$$

Algorithm 35 K-MEANS($\mathbf{D}$, K)

```
for k = 1 to K do
    μ_k ← some random location          // randomly initialize mean for kth cluster
end for
repeat
    for n = 1 to N do
        z_n ← argmin_k ||μ_k − x_n||    // assign example n to closest center
    end for
    for k = 1 to K do
        μ_k ← MEAN({ x_n : z_n = k })   // re-estimate mean of cluster k
    end for
until converged
return z                                // return cluster assignments
```

Extensions

- k-Medioids
- Bisecting k-Means
- k-Means++