In The Name of Allah

Digital Media Laboratory
Sharif University of Technology

Statistical Pattern Recognition

Clustering

Hamid R. Rabiee
Jafar Muhammadi, Nima Pourdamghani

Spring 2012
http://ce.sharif.edu/courses/90-91/2/ce725-1/

Agenda

- Unsupervised Learning
- Quality Measurement
- Similarity Measures
- Major Clustering Approaches
  - Distance Measuring
  - Partitioning Methods
  - Hierarchical Methods
  - Density Based Methods
  - Spectral Clustering
  - Other Methods
- Constraint Based Clustering
- Clustering as Optimization

Unsupervised Learning

- **Clustering or unsupervised classification is aimed at discovering natural groupings in a set of data.**
  - **Note: All samples in the training set are unlabeled.**
- **Applications for clustering:**
  - **Spatial data analysis: Create thematic maps in GIS by clustering feature space**
  - **Image processing: Segmentation**
  - **Economic science: Discover distinct groups in costumer bases**
  - **Internet: Document classification**
  - **To gain insight into the structure of the data prior to classifier design; classifier design**

Quality Measurement

- **High quality clusters must have**
  - **high intra-class similarity**
  - **low inter-class similarity**
- **Some other measures**
  - **Ability to discover hidden patterns**
    - Judged by the user
  - **Purity**
    - Suppose we know the labels of the data, assign to each cluster its most frequent class
    - Purity is the number of correctly assigned points divided by the number of data

Similarity Measures

- **Distances are normally used to measure the similarity or dissimilarity between two data objects**
  - **Some popular distances are Minkowski and Mahalanobis.**
  - **Distance between binary strings**

$$d(S_1,S_2)=|\{(s_{1,i},s_{2,i}) : s_{1,i} \neq s_{2,i}\}|$$

  - **Distance between vector objects**

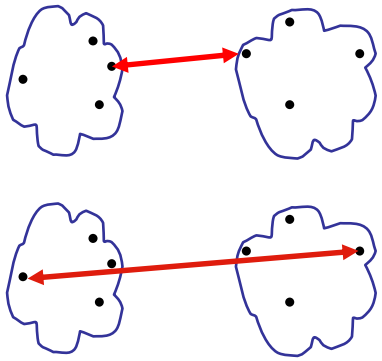
$$d(\vec{X},\vec{Y}) = \frac{\vec{X}^T.\vec{Y}}{\|\vec{X}\|\|\vec{Y}\|}$$

Major Clustering Approaches

- **Partitioning approach**
  - Construct various partitions and then evaluate them by some criterion (ex. k-means, c-means, k-medoids)
- **Hierarchical approach**
  - Create a hierarchical decomposition of the set of data using some criterion (ex. Agnes)
- **Density-based approach**
  - Based on connectivity and density functions (ex. DBSACN, OPTICS)
- **Graph-based approach (Spectral Clustering)**
  - approximately optimizing the normalized cut criterion
- **Grid-based approach**
  - based on a multiple-level granularity structure (ex. STING, WaveCluster, CLIQUE)
- **Model-based**
  - A model is hypothesized for each of the clusters and tries to find the best fit of that model to each other (ex. EM, SOM)

Distance Measuring

- **Single link**
  - **smallest distance between an element in one cluster and an element in the other**
- **Complete link**
  - **largest distance between an element in one cluster and an element in the other**



- **Average**
  - **avg distance between an element in one cluster and an element in the other**
- **Centroid**
  - **distance between the centroids of two clusters**
    - Used in k-means
- **Medoid**
  - **distance between the medoids of two clusters**
    - Medoid: A representative object whose average dissimilarity to all the objects in the cluster is minimal

Partitioning Methods

- **Construct a partition of n data into a set of k clusters, s.t., min sum of squared distance**

$$\min \sum_{m=1}^{k} \sum_{x_j \in Cluster_m} (x_j - C_m)^2$$

where C_ms are clusters representatives.

- **Given a k, find a partition of k clusters that optimizes the chosen partitioning criterion**
- **Global optimal: exhaustively enumerate all partitions**
- **Heuristic methods: k-means, c-means and k-medoids algorithms**
  - k-means: Each cluster is represented by the center of the cluster
  - c-means: The fuzzy version of k-means
  - k-medoids: Each cluster is represented by one of the samples in the cluster

Partitioning Methods: k-means

- **k-means**
  - Suppose we know there are K categories and each category is represented by its sample mean
  - Given a set of unlabeled training samples, how to estimate the means?
- **Algorithm k-means (k)**
  1. Partition samples into k non-empty subsets (random initialization)
  2. Compute mean points of the clusters of the current partition
  3. Assign each sample to the cluster with the nearest mean point
  4. Go back to Step 2, stop when no more new assignment

Partitioning Methods: k-means

- **Some notes on k-means**
  - **Need to specify k, the number of clusters, in advance**
  - **Unable to handle noisy data and outliers (Why?)**
  - **Not suitable to discover clusters with non-convex shapes (Why?)**
  - **Algorithm is sensitive to**
    - number of cluster centers,
    - choice of initial cluster centers
    - sequence in which data are processed (Why?)
  - **Convergence not guaranteed, but results acceptable if there are well-separated clusters**

Partitioning Methods: c-means

- **The membership function μ_{il} expresses to what degree x_l belongs to class C_i.**
- **Crisp clustering: x_l can belong to one class only**

$$\mu_{il} = \begin{cases} 1 & if\ x_l \in C_i \\ 0 & if\ x_l \notin C_i \end{cases}$$

- **Fuzzy clustering: x_l belongs to all classes simultaneously with varying degrees of membership**

$$\mu_{il} = \frac{\left(\frac{1}{d(z_i^{(m)}, x_l)}\right)^{\frac{1}{q-1}}}{\sum_{i=1}^{k}\left(\frac{1}{d(z_i^{(m)}, x_l)}\right)^{\frac{1}{q-1}}}$$

  - **where $z^{(m)}$s are cluster means**
  - **q is a fuzziness index with 1<q<2**
    - Fuzzy clustering becomes crisp clustering when q→1
  - **Observe that** $\sum_{i=1}^{k}\mu_{il} = 1,\ for\ l = 1,2,...,N$.
  - **C-mean minimizes** $J_e = \sum_{i=1}^{k} J_i^f,\ J_i^f = \sum_{l=1}^{N}(\mu_{il})^q \left\| z_i^{(m)} - x_l \right\|^2$

Partitioning Methods: k-medoids

- **k-medoids**
  - **Instead of taking the mean value of the samples in a cluster as a reference point, medoids can be used**
  - **Note that choosing the new medoids is slightly different with choosing the new means in k-means algorithm**
- **Algorithm k-medoids (k)**
  1. **Select k representative samples arbitrarily**
  2. **Associate each data point to the closest medoid**
  3. **For each medoid** m **and data point** o

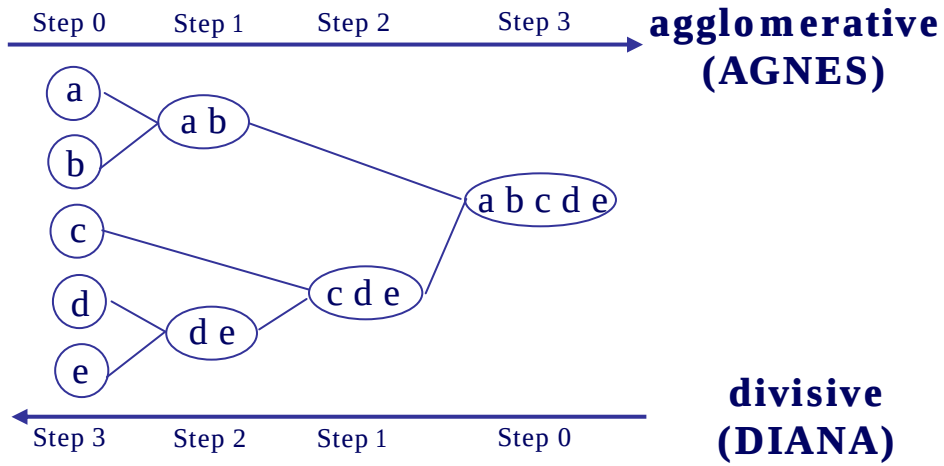
     Swap m and o and compute the total cost of configuration
  4. **Select the configuration with the lowest cost**
  5. **repeat steps 2-5 until there is no change**

Partitioning Methods: k-medoids

- **Some notes on k-medoids**
  - **k-medoids is more robust than k-means in the presence of noise and outliers (Why?)**
  - **works effectively for small data sets, but does not scale well for large data sets**
    - For Large data sets we can use sampling based methods (How?)

Hierarchical Methods

- Clusters have sub-clusters and sub-clusters can have sub-sub-clusters, …
- Use distance matrix as clustering criteria.



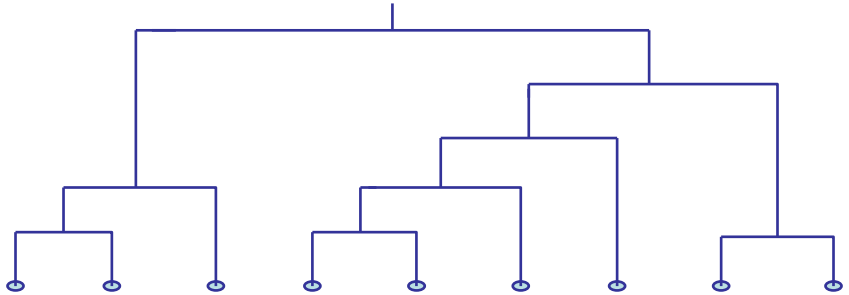
- This method does not require the number of clusters k as an input, but needs a termination condition

Hierarchical Methods

- **Agglomerative Hierarchical Clustering**
  - **AGNES (Agglomerative Nesting)**
    - Uses the Single-Link method
    - Merge nodes (clusters) that have the maximum similarity
- **divisive Hierarchical Clustering**
  - **DIANA (Divisive Analysis)**
    - Inverse order of AGNES
    - Eventually each node forms a cluster on its own

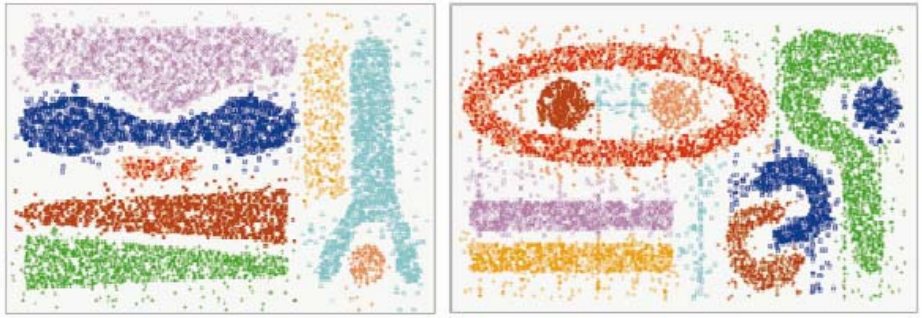
Hierarchical Methods

- **Dendrogram**
  - **Shows How the Clusters are Merged**
  - **Decompose samples into a several levels of nested partitioning (tree of clusters), called a dendrogram.**
  - **A clustering of the samples is obtained by cutting the dendrogram at the desired level, then each connected component forms a cluster.**



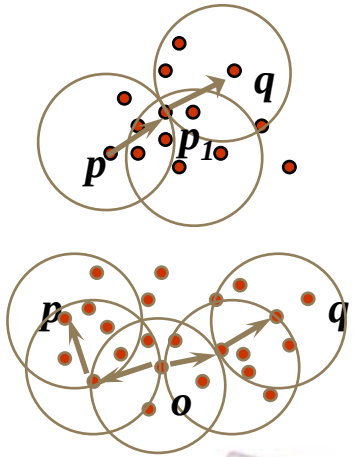
Density Based Methods

- Clustering based on density (local cluster criterion), such as density-connected points
- Major features:
  - Discover clusters of arbitrary shapes
  - Handle noise
  - Need density parameters as termination condition



Density Based Methods

- **Main Concepts:**
  - **parameters:**
    - Eps: Maximum radius of the neighborhood
    - MinPts: Minimum number of points in an Eps-neighbourhood of that point
  - **Sample q is directly density-reachable from sample p, if d(p,q)<=Eps and p has MinPts points in its neighborhood.**
  - **Sample q is density-reachable from a sample p if there is a chain of points p_1, …, p_n, $p_1 = p$, $p_n = q$ such that p_{i+1} is directly density-reachable from p_i**
  - **Sample p is density-connected to sample q if there is a sample o such that both, p and q are density-reachable from o.**

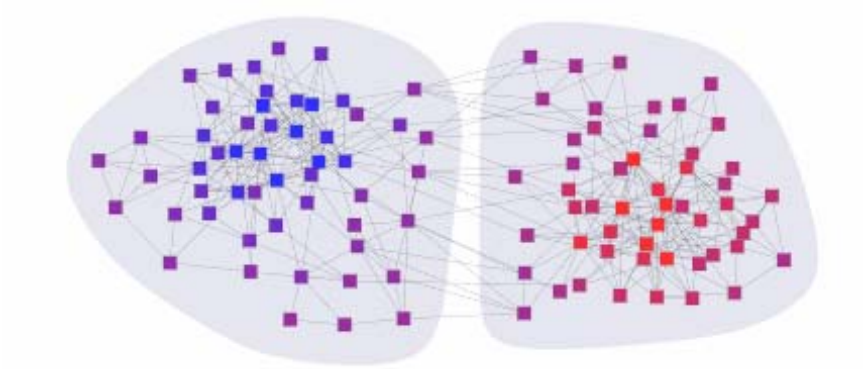


Density Based Methods: DBSCAN

- **DBSCAN (Density Based Spatial Clustering of Applications with Noise)**
  - Relies on a *density-based* notion of cluster: A *cluster* is defined as a maximal set of density-connected points
  - Discovers clusters of arbitrary shape in spatial data with noise
- **Algorithm DBSCAN (Eps, MinPts)**
  - Arbitrary select a sample *p*
  - Retrieve all samples density-reachable from *p* w.r.t. *Eps* and *MinPts*.
  - If *p* is a core sample (some samples are density-reachable from p), a cluster is formed.
  - If *p* is a border sample (no samples are density-reachable from *p),* DBSCAN visits the next sample of the database.
  - Continue the process until all of the samples have been processed.

Graph-based Clustering

- **Represent data points as the vertices *V* of a graph *G*.**
- **All pairs of vertices are connected by an edge *E*.**
- **Edges have weights *W*.**
  - **Large weights mean that the adjacent vertices are very similar; small weights imply dissimilarity.**

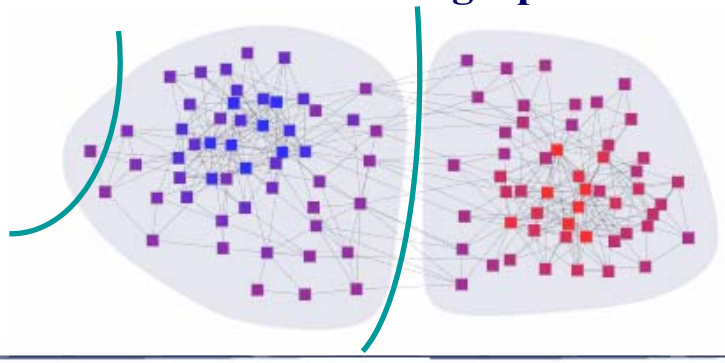


Graph-based Clustering

- **Clustering on a graph is equivalent to partitioning the vertices of the graph.**
- **A loss function for a partition of V into sets A and B**

$$cut(A,B) = \sum_{u \in A, v \in B} W_{u,v}$$

- **In a good partition, vertices in different partitions will be dissimilar.**
  - **Mincut criterion: Find a partition *A, B* that minimizes *cut(A,B)***
- **Mincut criterion ignores the size of the subgraphs formed**



Graph-based Clustering

- **Normalized cut criterion favors balanced partitions.**

$$Ncut(A,B) = \frac{cut(A,B)}{\sum_{u \in A, v \in V} W_{u,v}} + \frac{cut(A,B)}{\sum_{u \in B, v \in V} W_{u,v}}$$

- **Minimizing the normalized cut criterion exactly is NP-hard.**
- **One way of approximately optimizing the normalized cut criterion leads to *spectral clustering*.**

Spectral Clustering

- **Spectral clustering**
  - **Looks for a new representation of the original data points, such that**
    - Preserve the edge weights.
    - The convex clusters' shapes in the new space represents non-convex ones in the original space.
  - **Cluster the points in the new space using any clustering scheme (say k-means).**

- **We only describe the resulting algorithm here.**
  - **For more information about derivations, refer to *U. Luxburg, "A Tutorial on Spectral Clustering".***

Spectral Clustering

- **Inputs**
  - **Set of points** $S = \{S_1, ..., S_n\} \in R^l$ **and number of clusters k**
- **Algorithm**
  - **Form the edge weights matrix** $W \in R^{nxn}$
    - For example:

$$W_{ij} = \begin{cases} e^{-\|s_i - s_j\|^2 / 2\sigma^2} & if\ i \neq j \\ 0 & else \end{cases}$$

      - **Scaling parameter chosen by user**
  - **Define D a diagonal matrix whose (i,i) element is the sum of W's row i**
  - **Form the matrix** $L = D^{-1/2} W D^{-1/2}$
  - **Find the k largest eigenvectors of L to form the matrix** X_{nxk}

Spectral Clustering

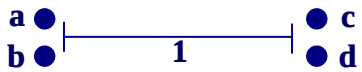
- **Algorithm (cont.)**
  - **Normalized the matrix X_{nxk} and form matrix Y_{nxk}**

$$Y_{ij} = X_{ij} / \sqrt{\sum_j X_{ij}^2}$$

  - **Treat each row of Y as a point in R^k (data dimensionality reduction from n to k)**
  - **Cluster the new data into k clusters via K-means**

Spectral Clustering

✧ Example

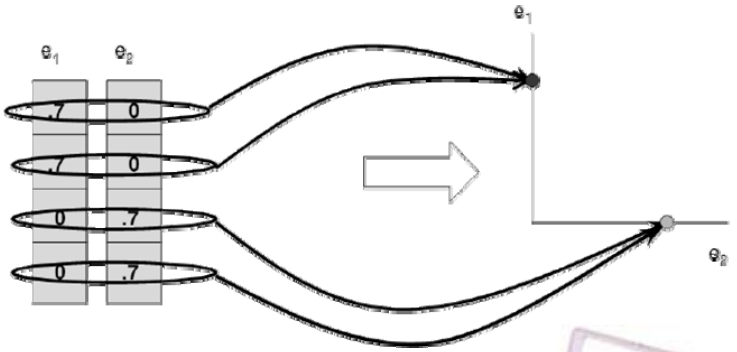


- ✧ **simple edge weights matrix ($d(x_i,x_j)$ denotes Euclidean distance between points x_i, x_j and $\theta=1$)**

$$W(i,j)=W(j,i)=\begin{cases}1 & if\ d(x_i,x_j)<\theta\\ 0 & otherwise\end{cases}$$

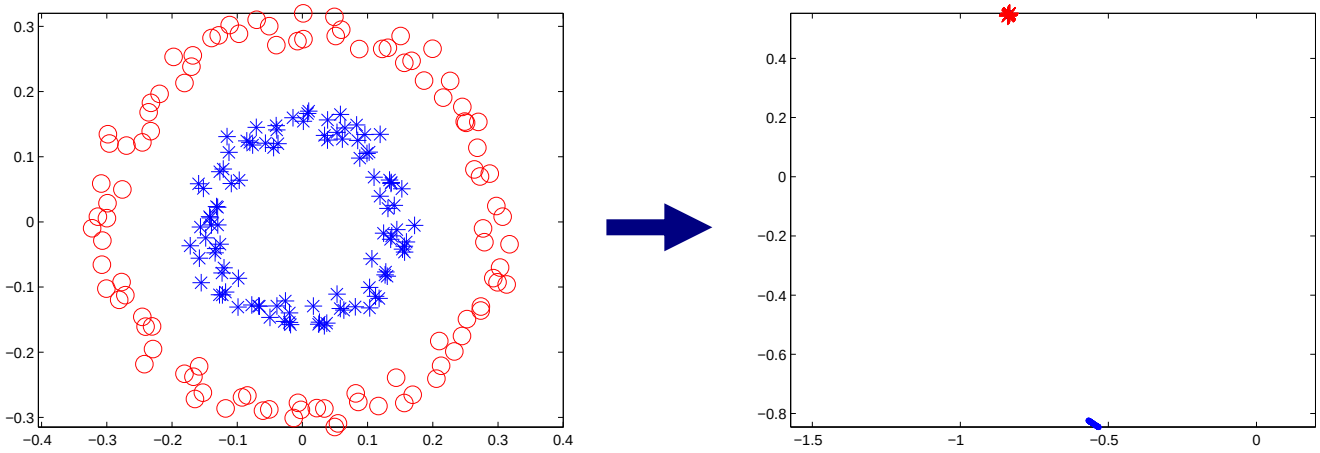
$$W=\begin{array}{c|cccc} & a & b & c & d\\ \hline a & 1 & 1 & 0 & 0\\ b & 1 & 1 & 0 & 0\\ c & 0 & 0 & 1 & 1\\ d & 0 & 0 & 1 & 1\end{array} \qquad \tilde{W}=\begin{array}{c|cccc} & a & c & b & d\\ \hline a & 1 & 0 & 1 & 0\\ c & 0 & 1 & 0 & 1\\ b & 1 & 0 & 1 & 0\\ d & 0 & 1 & 0 & 1\end{array}$$

$$e_1=(0.7,0.7,0,0)^T \qquad e_1=(0.7,0,0.7,0)^T$$
$$e_2=(0,0,0.7,0.7)^T \qquad e_2=(0,0.7,0,0.7)^T$$



Spectral Clustering

- **Another example**



Other Methods

- **Grid based methods**
  - **Using multi-resolution grid data structure.**
    1. Create the grid structure, i.e., partition the data space into a finite number of cells
    2. Calculate the cell density for each cell
    3. Sort the cells according to their densities
    4. Identify cluster centers
    5. Traverse the neighbor cells

Other Methods

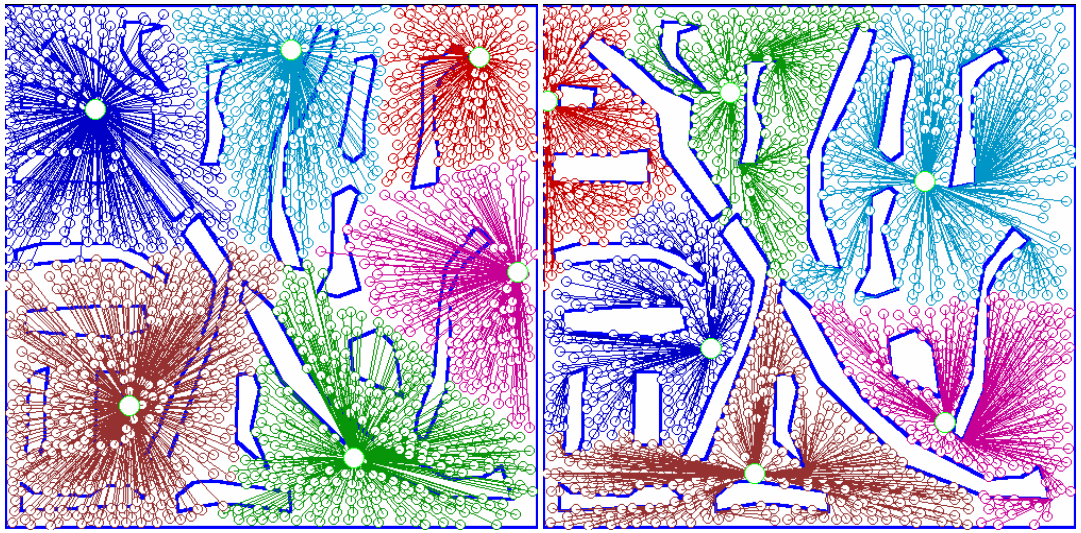
- **Model based methods**
  - **Attempt to optimize the fit between the given data and some mathematical model**
  - **Based on the assumption: Data are generated by a mixture of underlying probability distribution**
  - **Typical methods**
    - Statistical approach: EM (Expectation maximization) – will be discussed later
    - Neural network approach: SOM (Self-Organizing Feature Map)

Constraint Based Clustering

- **Why constraint based clustering?**
  - **Need user feedback: Users know their applications the best**
  - **Less parameters but more user-desired constraints, e.g., an ATM allocation problem: obstacle & desired clusters**
- **Different constraints in cluster analysis:**
  - **Constraints on individual samples (do selection first)**
    - Cluster on samples which …
  - **Constraints on distance or similarity functions**
    - Weighted functions, obstacles
  - **Constraints on the selection of clustering parameters**
    - Number of clusters, limitation of each cluster size
  - **User-specified constraints**
    - Some samples must be in cluster and some others not!
  - **Semi-supervised: giving small training sets as "constraints" or hints**

Constraint Based Clustering

- **A sample data and two answers (taking the constraints into account and not taking the constraints into account)**
- **Constraints: The data in different sides of each "wall" should be in different clusters**



Clustering as Optimization

- **Clustering can be posted as an optimization of a criterion function**
  - **The sum-of-squared-error criterion**
  - **Scatter criteria**
- **The given criterion function is optimized through iterative optimization**

Any Question?

End of Lecture 13

Thank you!

Spring 2012

http://ce.sharif.edu/courses/90-91/2/ce725-1/