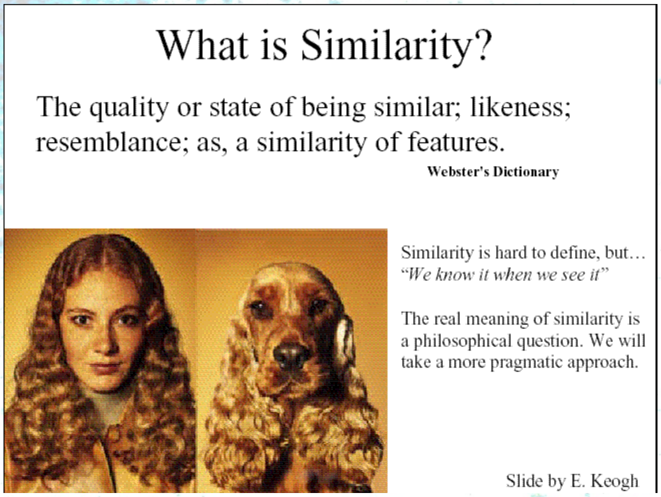
Advanced Topics: An Overview

Topics

- Clustering
- Subspace clustering
- Ensembles of classifiers and clusterings
- Semi-supervised clustering
- Learning Metrics

Clustering

- Goal: Grouping a collection of objects (data points) into subsets or “clusters”, such that those within each cluster are more closely related to one other than objects assigned to different clusters.
- Fundamental to all clustering techniques is the choice of ***distance or dissimilarity measure*** between two objects.



What is Similarity?

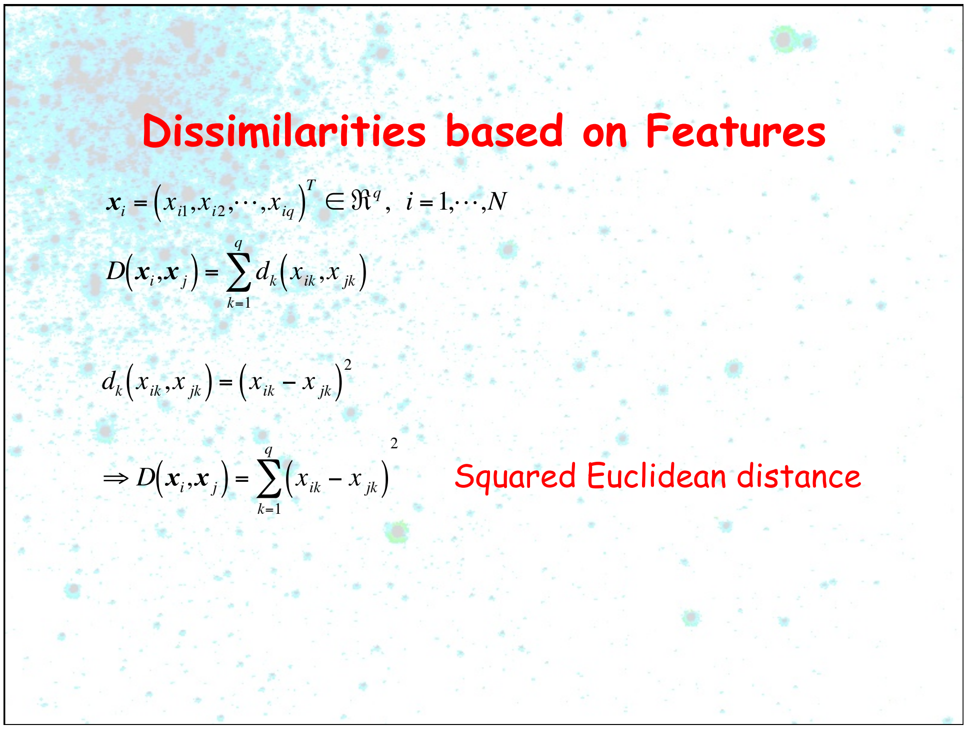
The quality or state of being similar; likeness; resemblance; as, a similarity of features.

Webster's Dictionary

Similarity is hard to define, but…
“We know it when we see it”

The real meaning of similarity is a philosophical question. We will take a more pragmatic approach.

Slide by E. Keogh



Dissimilarities based on Features

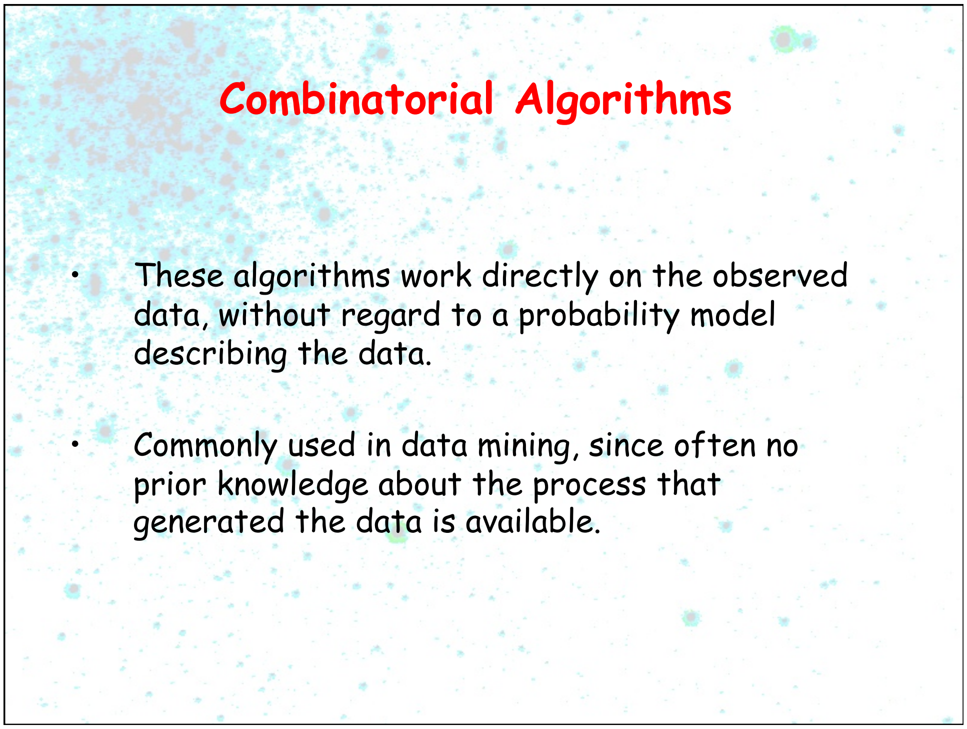
$$\boldsymbol{x}_i = \left(x_{i1}, x_{i2}, \cdots, x_{iq}\right)^T \in \Re^q, \ \ i = 1, \cdots, N$$

$$D\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \sum_{k=1}^{q} d_k\left(x_{ik}, x_{jk}\right)$$

$$d_k\left(x_{ik}, x_{jk}\right) = \left(x_{ik} - x_{jk}\right)^2$$

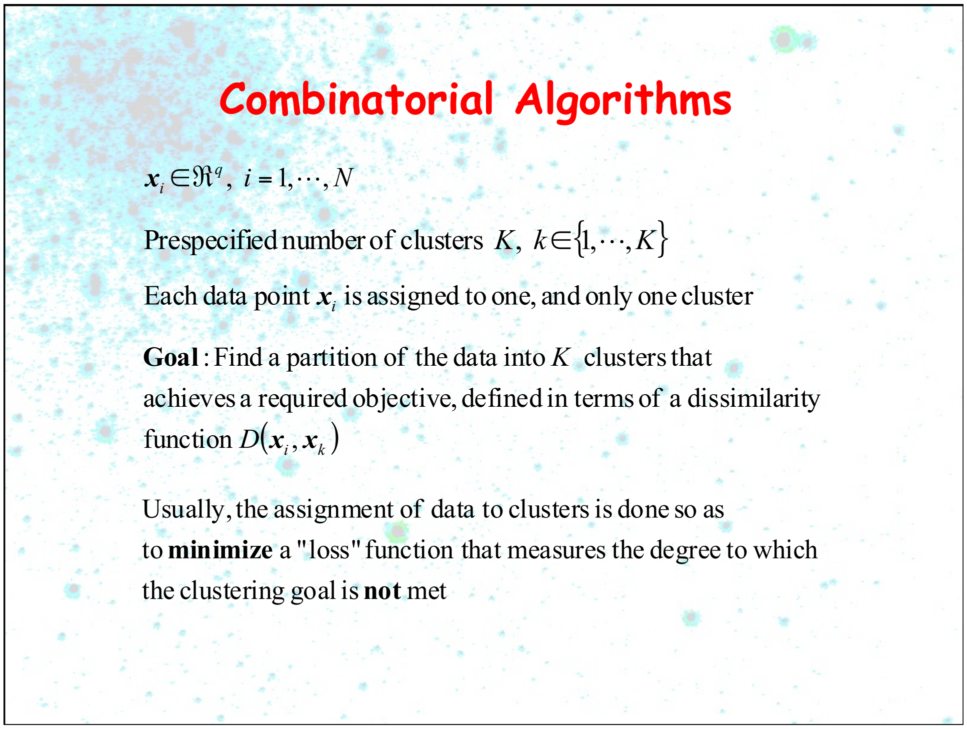
$$\Rightarrow D\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \sum_{k=1}^{q} \left(x_{ik} - x_{jk}\right)^2$$

Squared Euclidean distance



Combinatorial Algorithms

- These algorithms work directly on the observed data, without regard to a probability model describing the data.
- Commonly used in data mining, since often no prior knowledge about the process that generated the data is available.



Combinatorial Algorithms

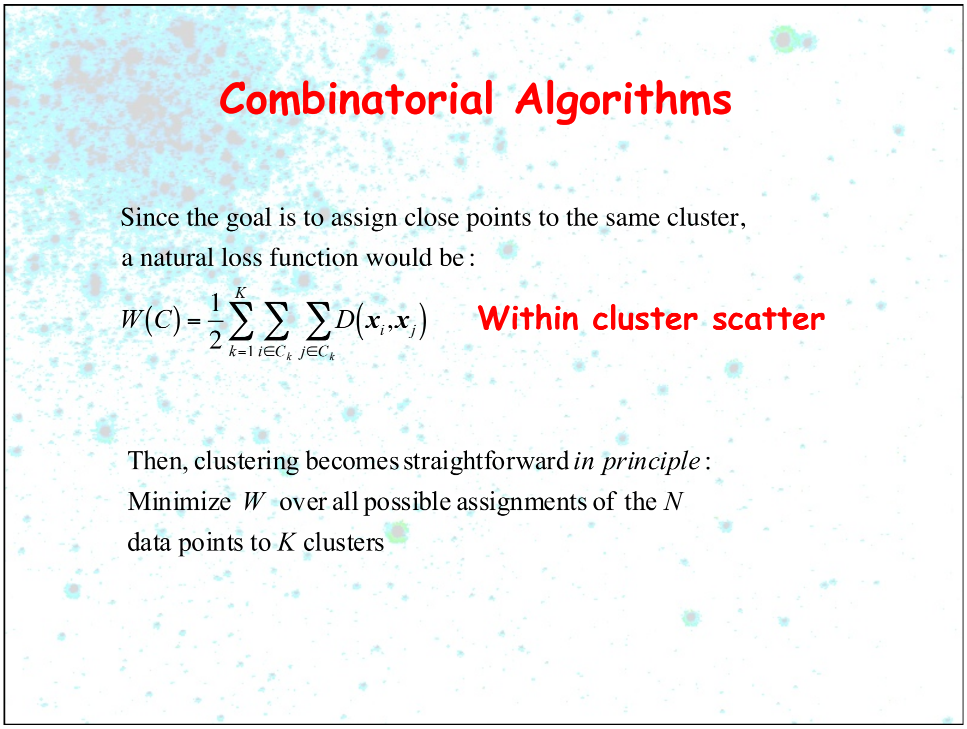
$\boldsymbol{x}_i \in \Re^q,\ i = 1, \cdots, N$

Prespecified number of clusters $K,\ k \in \{1, \cdots, K\}$

Each data point $\boldsymbol{x}_i$ is assigned to one, and only one cluster

Goal : Find a partition of the data into K clusters that achieves a required objective, defined in terms of a dissimilarity function $D(\boldsymbol{x}_i, \boldsymbol{x}_k)$

Usually, the assignment of data to clusters is done so as to **minimize** a "loss" function that measures the degree to which the clustering goal is **not** met



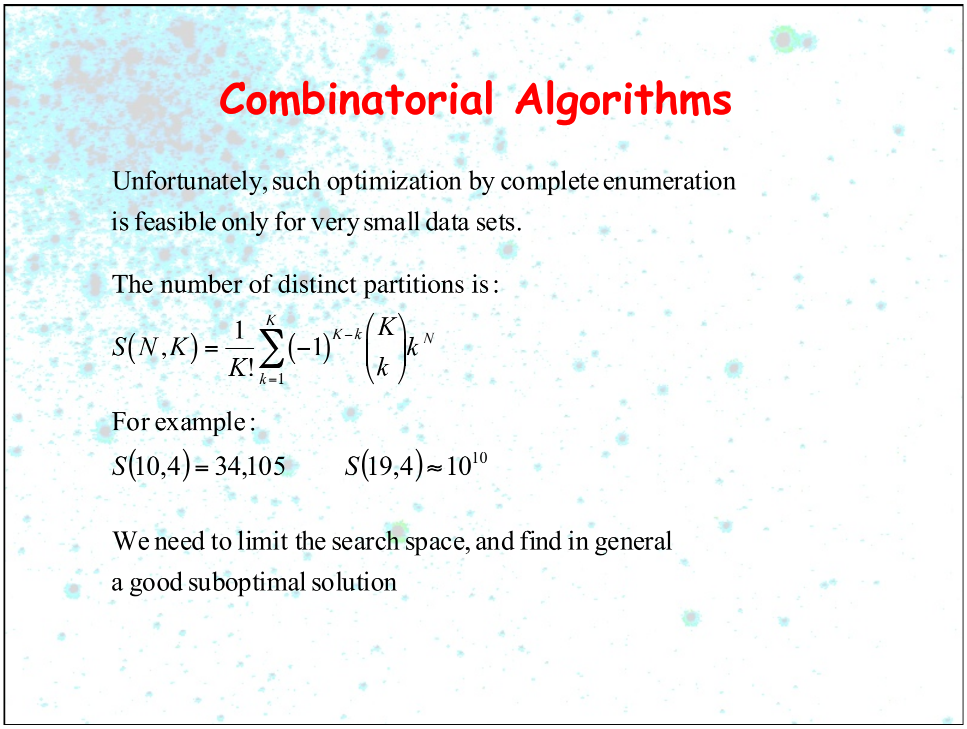
Combinatorial Algorithms

Since the goal is to assign close points to the same cluster, a natural loss function would be :

$$W(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in C_k} D(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

Within cluster scatter

Then, clustering becomes straightforward *in principle* : Minimize W over all possible assignments of the N data points to K clusters



Combinatorial Algorithms

Unfortunately, such optimization by complete enumeration is feasible only for very small data sets.

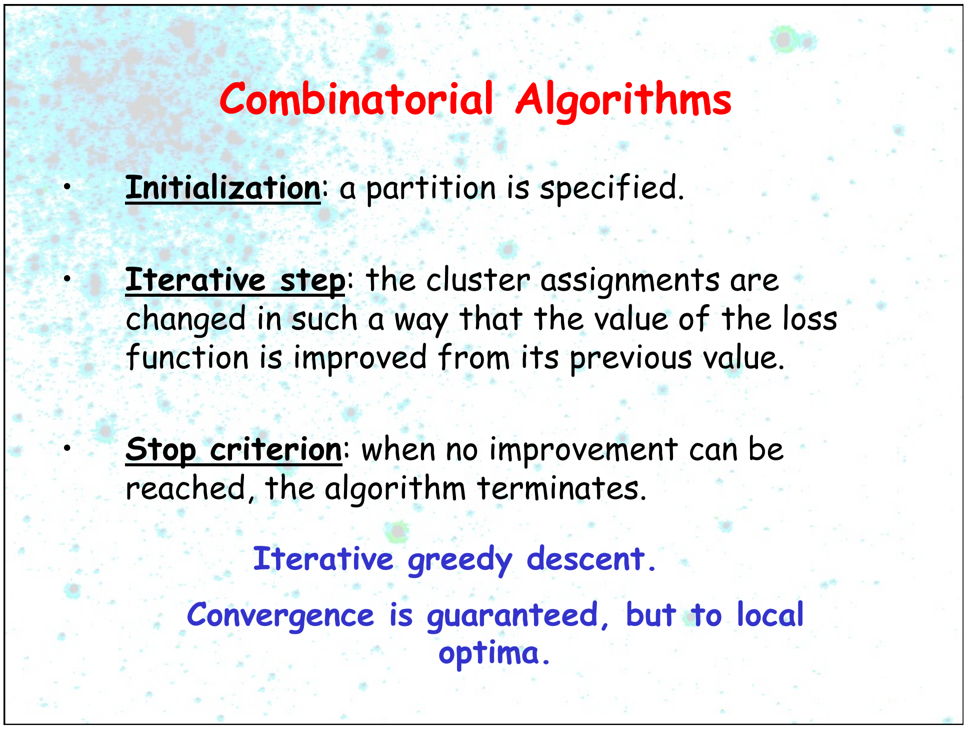
The number of distinct partitions is:

$$S(N,K) = \frac{1}{K!}\sum_{k=1}^{K}(-1)^{K-k}\binom{K}{k}k^{N}$$

For example:

$S(10,4) = 34{,}105 \qquad S(19,4) \approx 10^{10}$

We need to limit the search space, and find in general a good suboptimal solution

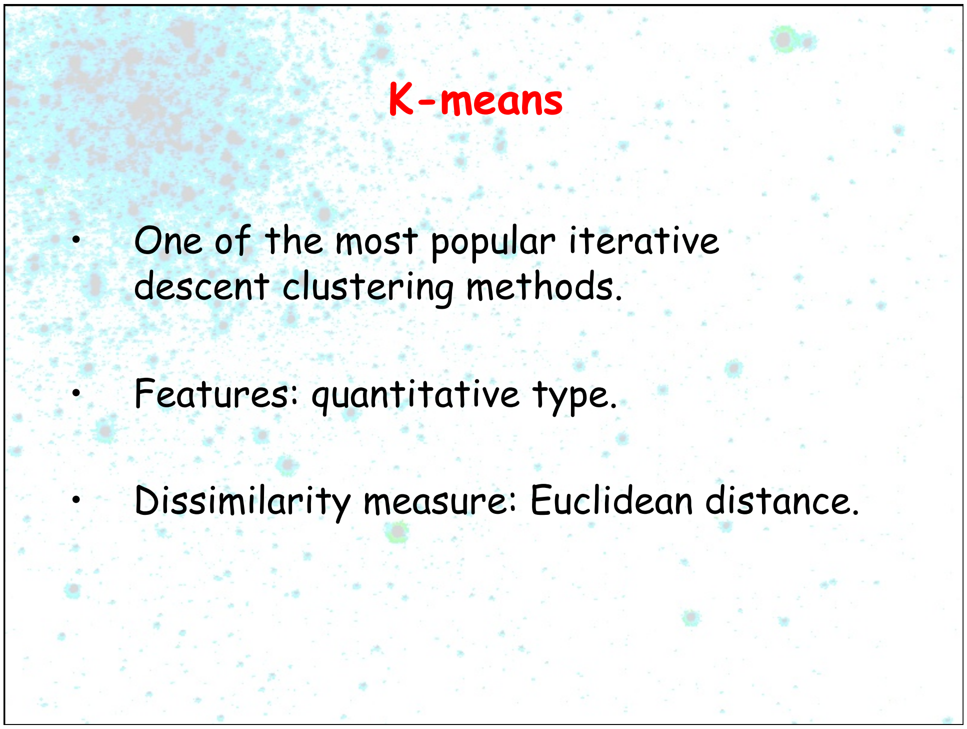


Combinatorial Algorithms

- **Initialization**: a partition is specified.
- **Iterative step**: the cluster assignments are changed in such a way that the value of the loss function is improved from its previous value.
- **Stop criterion**: when no improvement can be reached, the algorithm terminates.

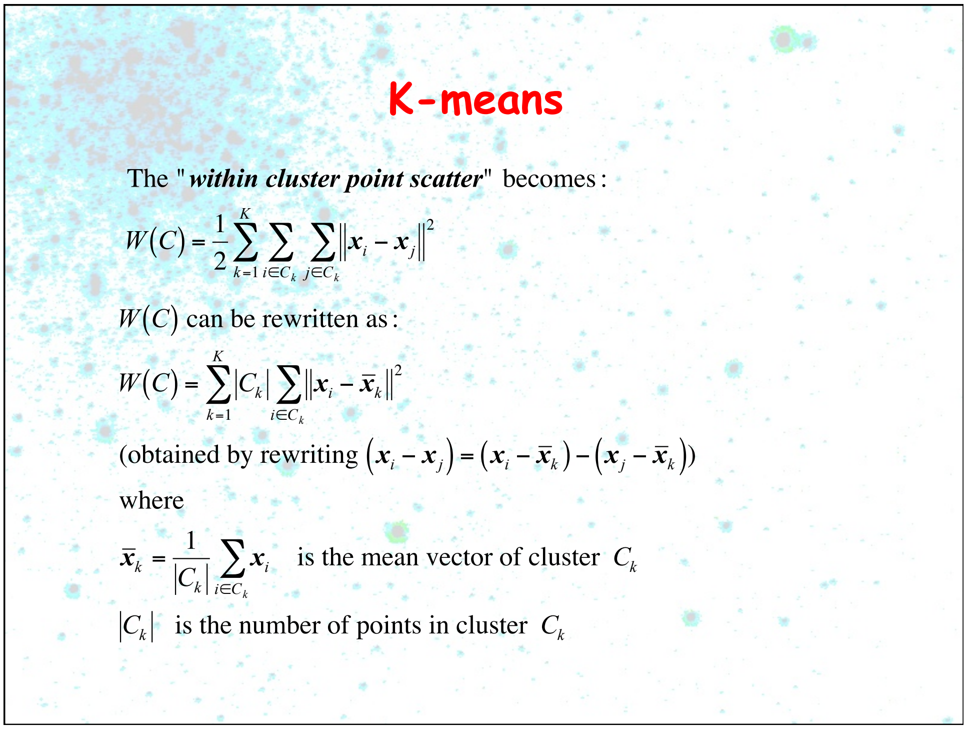
Iterative greedy descent.

Convergence is guaranteed, but to local optima.



K-means

- One of the most popular iterative descent clustering methods.
- Features: quantitative type.
- Dissimilarity measure: Euclidean distance.



K-means

The "***within cluster point scatter***" becomes:

$$W(C)=\frac{1}{2}\sum_{k=1}^{K}\sum_{i\in C_k}\sum_{j\in C_k}\left\|\boldsymbol{x}_i-\boldsymbol{x}_j\right\|^2$$

$W(C)$ can be rewritten as:

$$W(C)=\sum_{k=1}^{K}\left|C_k\right|\sum_{i\in C_k}\left\|\boldsymbol{x}_i-\bar{\boldsymbol{x}}_k\right\|^2$$

(obtained by rewriting $\left(\boldsymbol{x}_i-\boldsymbol{x}_j\right)=\left(\boldsymbol{x}_i-\bar{\boldsymbol{x}}_k\right)-\left(\boldsymbol{x}_j-\bar{\boldsymbol{x}}_k\right)$)

where

$\bar{\boldsymbol{x}}_k=\frac{1}{\left|C_k\right|}\sum_{i\in C_k}\boldsymbol{x}_i$ is the mean vector of cluster C_k

$\left|C_k\right|$ is the number of points in cluster C_k

K-means

The objective is:

$$\min_C \sum_{k=1}^{K} |C_k| \sum_{i \in C_k} \|\boldsymbol{x}_i - \bar{\boldsymbol{x}}_k\|^2$$

We can solve this problem by noticing:

for any set of data S

$$\bar{\boldsymbol{x}}_S = \arg\min_{\boldsymbol{m}} \sum_{i \in S} \|\boldsymbol{x}_i - \boldsymbol{m}\|^2$$

(this is obtained by setting $\dfrac{\partial \sum_{i \in S} \|\boldsymbol{x}_i - \boldsymbol{m}\|^2}{\partial \boldsymbol{m}} = 0$)

So we can solve the enlarged optimization problem:

$$\min_{C, \boldsymbol{m}_k} \sum_{k=1}^{K} |C_k| \sum_{i \in C_k} \|\boldsymbol{x}_i - \boldsymbol{m}_k\|^2$$

K-means: The Algorithm

1. Given a cluster assignment C, the total within cluster scatter

$\sum_{k=1}^{K} |C_k| \sum_{i \in C_k} \|\boldsymbol{x}_i - \boldsymbol{m}_k\|^2$ is minimized with respect to the $\{\boldsymbol{m}_1, \cdots, \boldsymbol{m}_K\}$

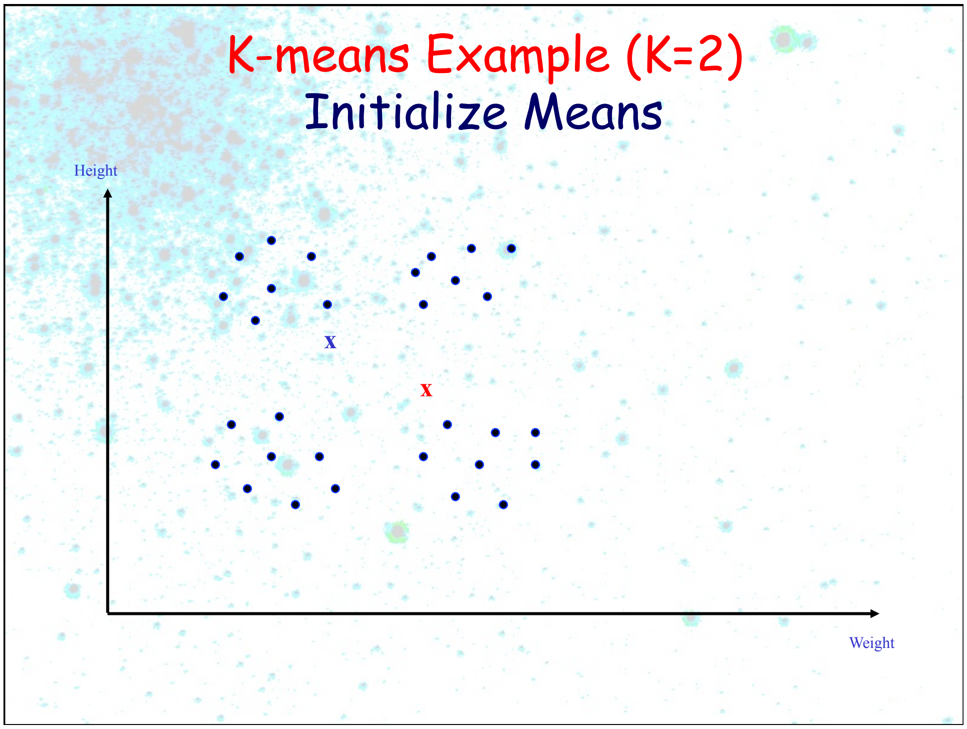
giving the means of the currently assigned clusters;

2. Given a current set of means $\{\boldsymbol{m}_1, \cdots, \boldsymbol{m}_K\}$,

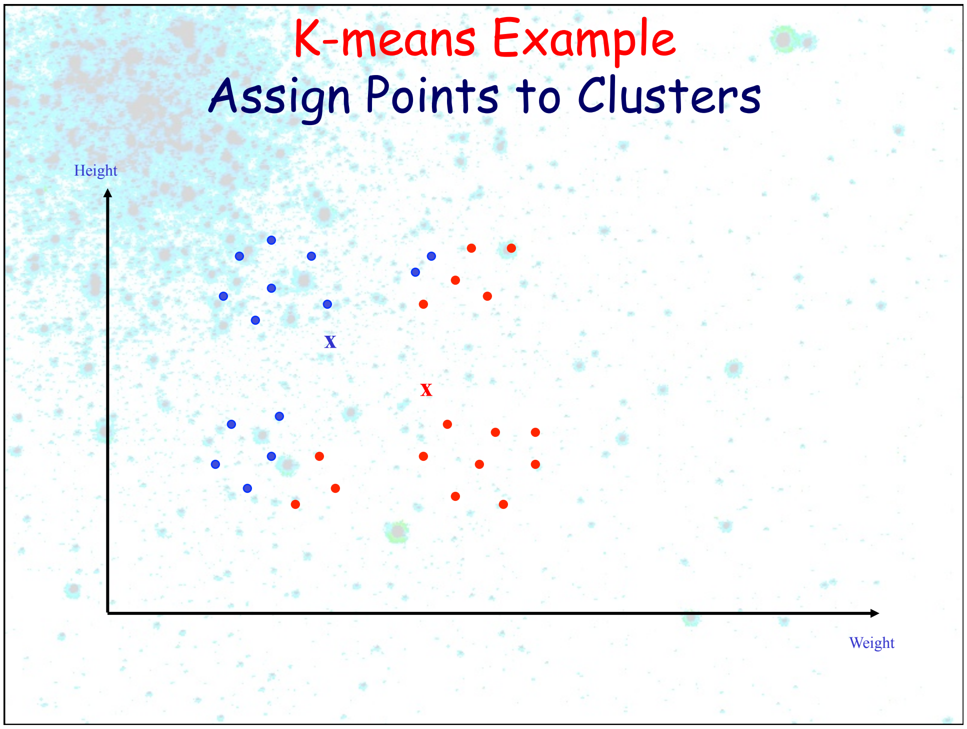
$\sum_{k=1}^{K} |C_k| \sum_{i \in C_k} \|\boldsymbol{x}_i - \boldsymbol{m}_k\|^2$ is minimized with respect to C

by assigning each point to the closest current cluster mean;

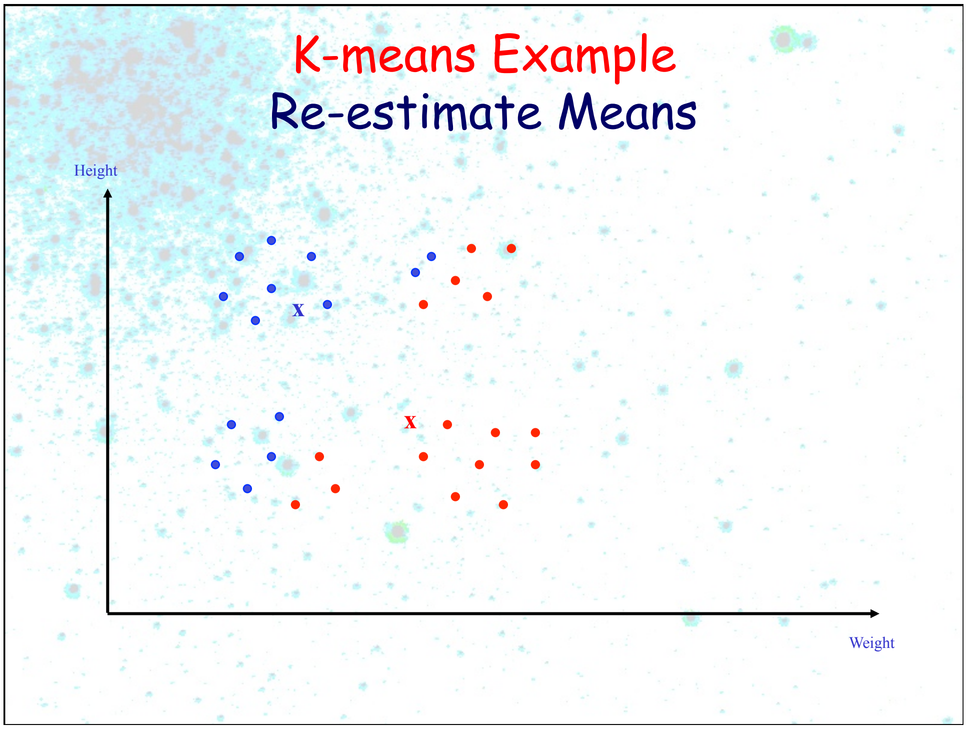
3. Steps 1 and 2 are iterated until the assignments do not change.



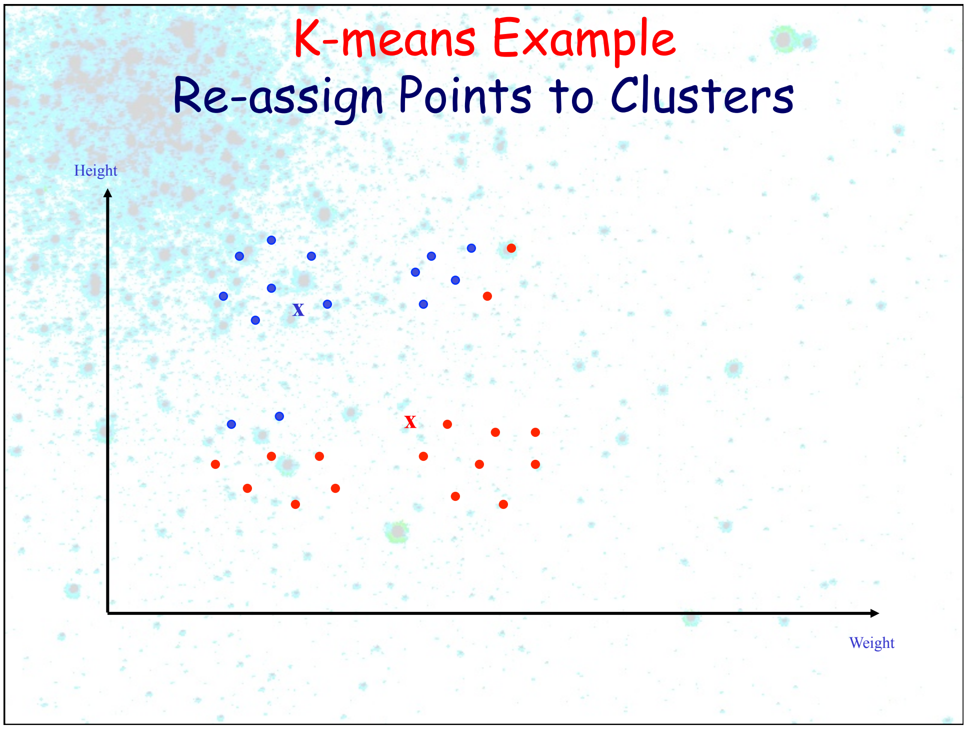
K-means Example (K=2)
Initialize Means



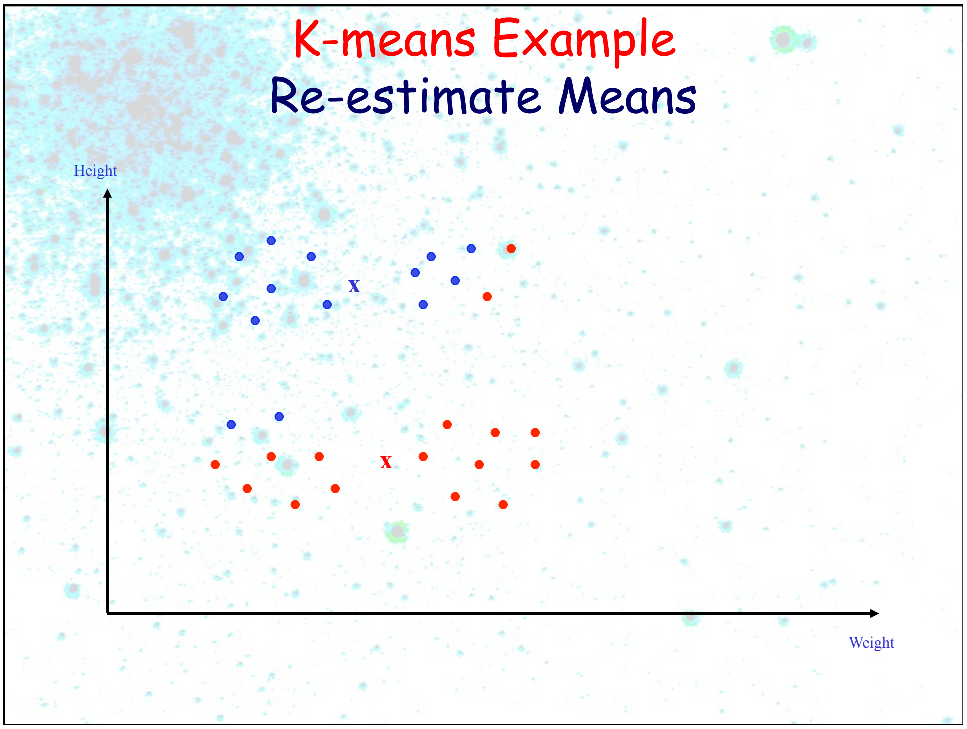
K-means Example
Assign Points to Clusters



K-means Example
Re-estimate Means

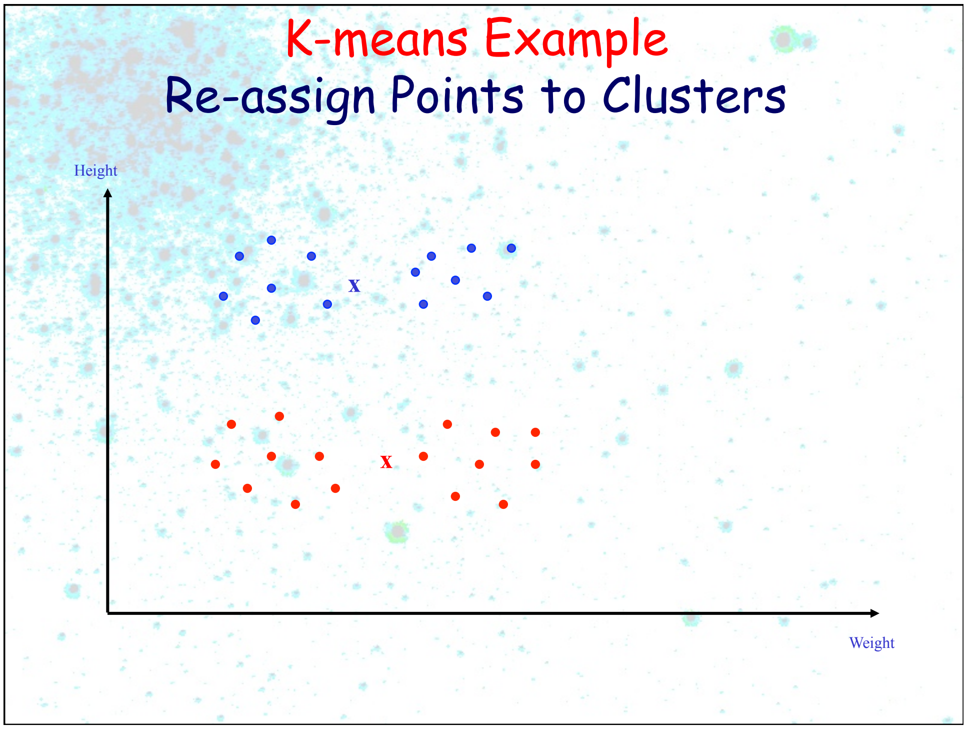


K-means Example
Re-assign Points to Clusters



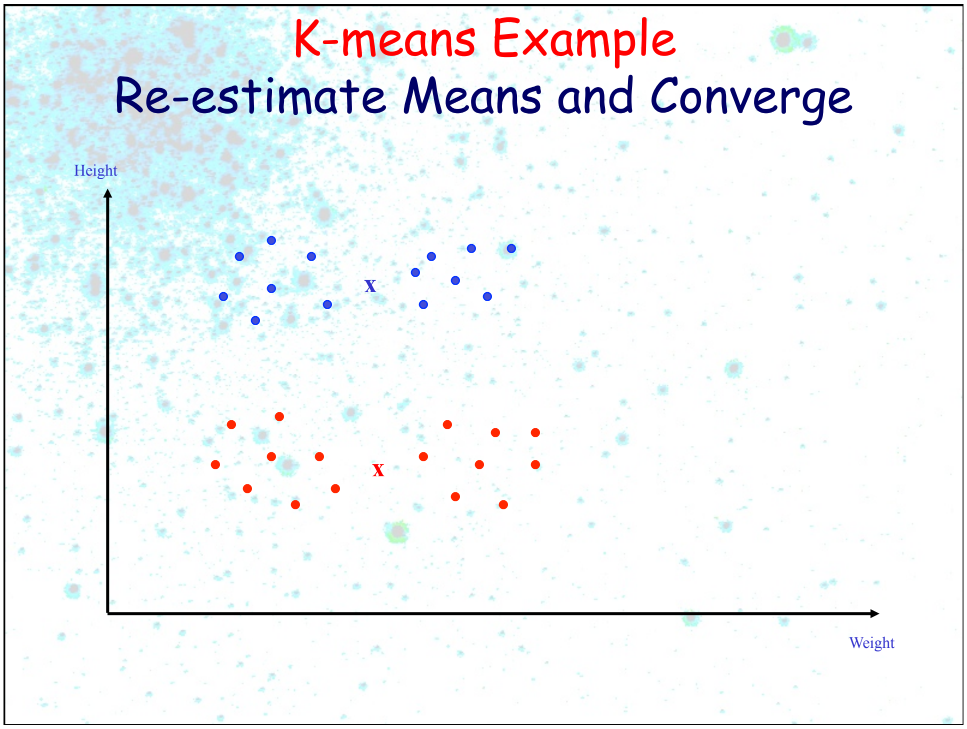
K-means Example

Re-estimate Means

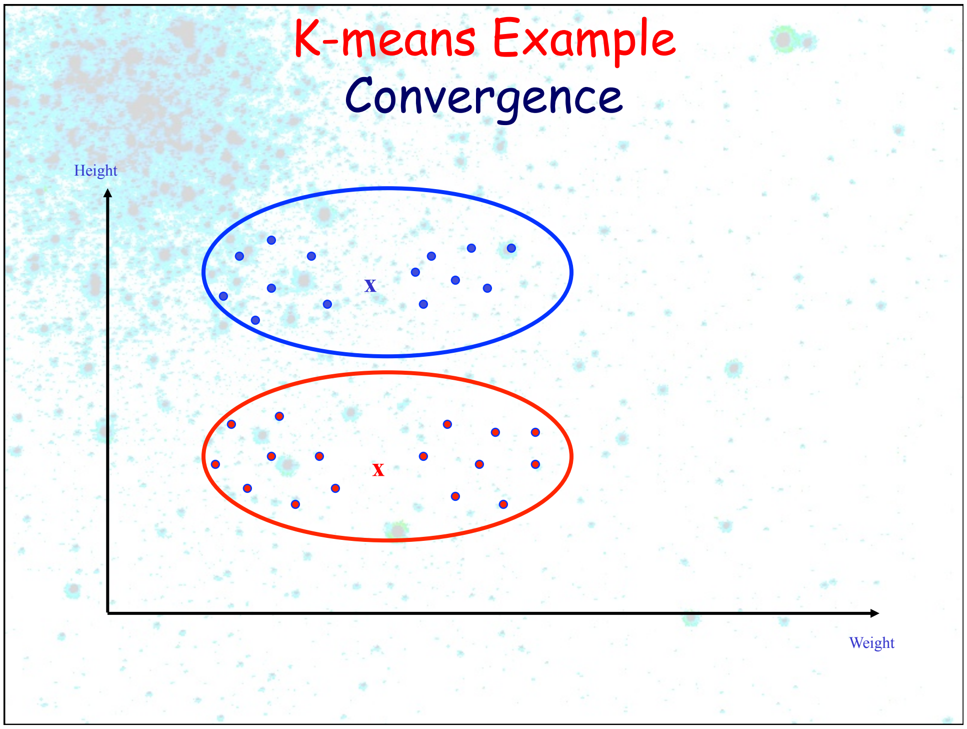


K-means Example

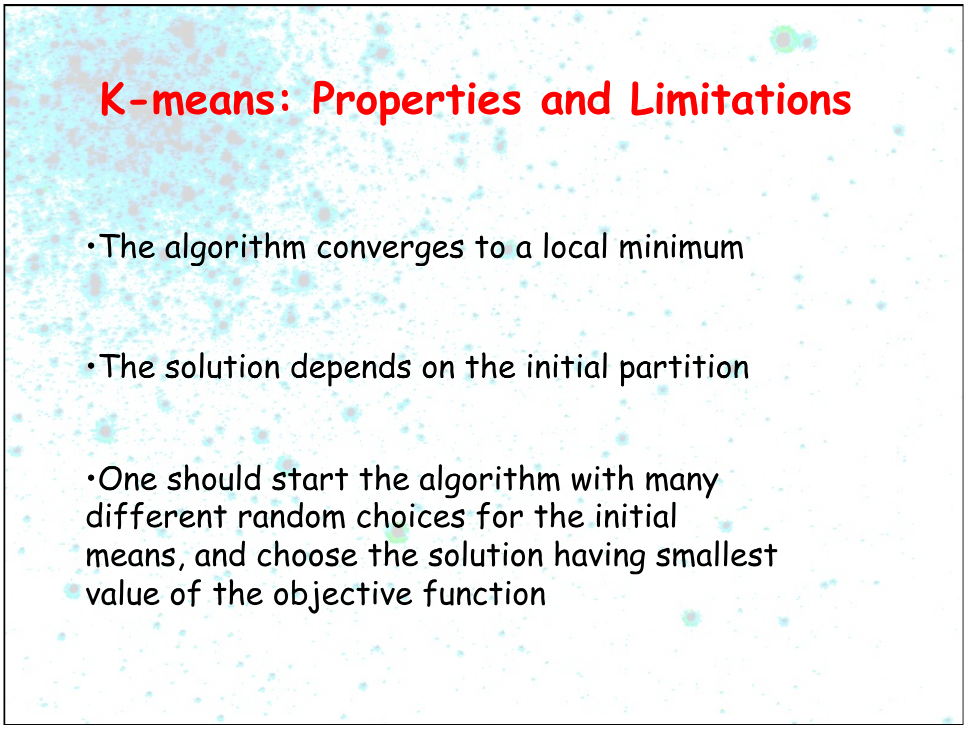
Re-assign Points to Clusters



K-means Example
Re-estimate Means and Converge

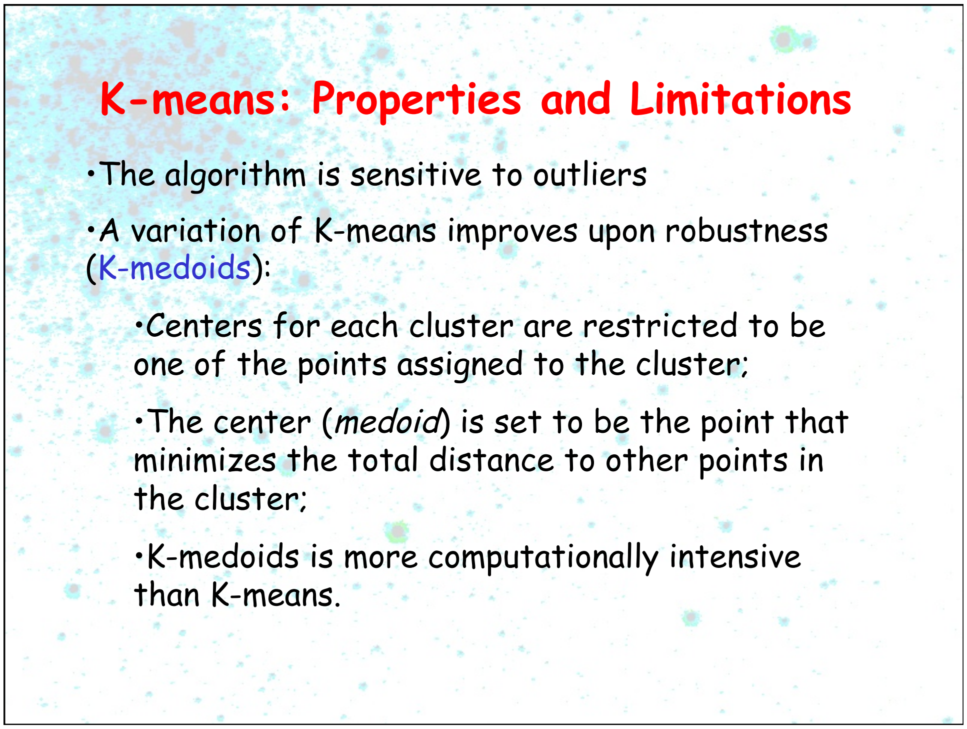


K-means Example
Convergence



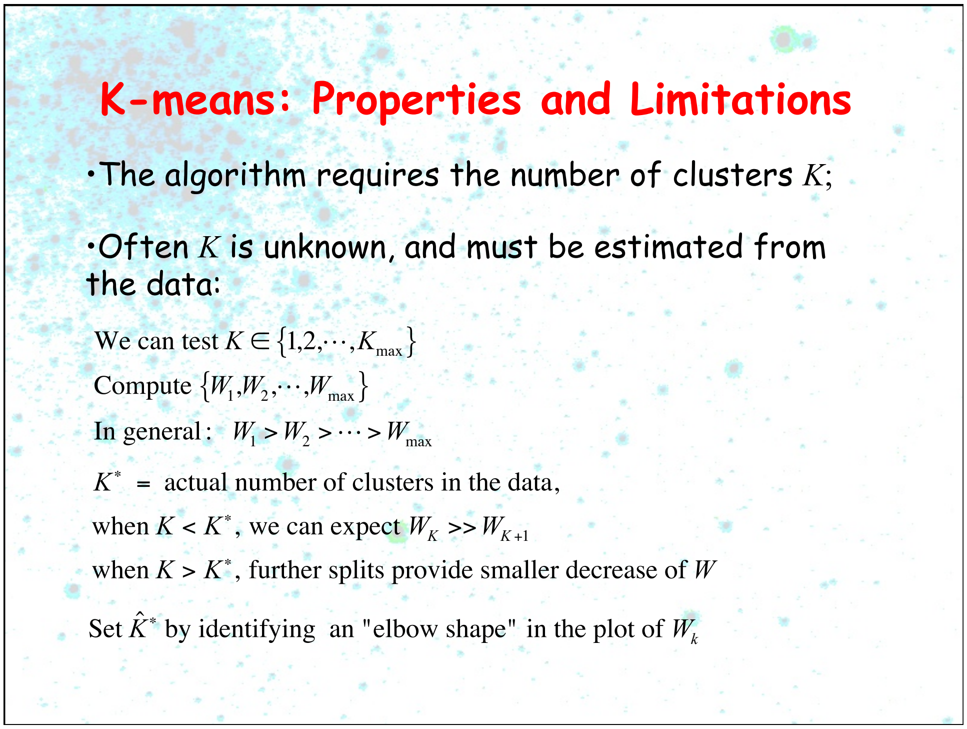
K-means: Properties and Limitations

- The algorithm converges to a local minimum
- The solution depends on the initial partition
- One should start the algorithm with many different random choices for the initial means, and choose the solution having smallest value of the objective function



K-means: Properties and Limitations

- The algorithm is sensitive to outliers
- A variation of K-means improves upon robustness (K-medoids):
  - Centers for each cluster are restricted to be one of the points assigned to the cluster;
  - The center (*medoid*) is set to be the point that minimizes the total distance to other points in the cluster;
  - K-medoids is more computationally intensive than K-means.



K-means: Properties and Limitations

- The algorithm requires the number of clusters K;
- Often K is unknown, and must be estimated from the data:

We can test $K \in \{1,2,\cdots,K_{\max}\}$

Compute $\{W_1,W_2,\cdots,W_{\max}\}$

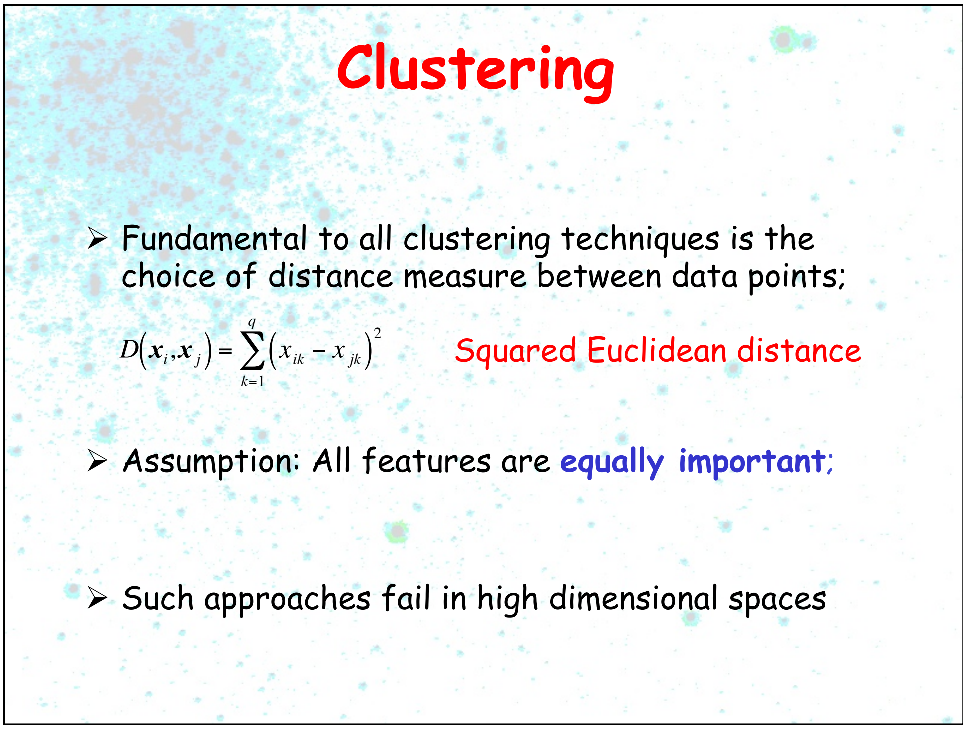
In general: $W_1 > W_2 > \cdots > W_{\max}$

K^* = actual number of clusters in the data,

when $K < K^*$, we can expect $W_K >> W_{K+1}$

when $K > K^*$, further splits provide smaller decrease of W

Set $\hat{K}^*$ by identifying an "elbow shape" in the plot of W_k



Clustering

- Fundamental to all clustering techniques is the choice of distance measure between data points;

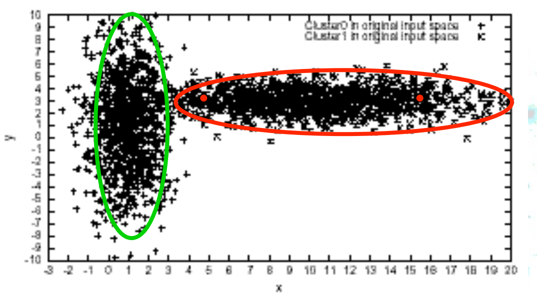
$$D(\boldsymbol{x}_i,\boldsymbol{x}_j) = \sum_{k=1}^{q}\left(x_{ik} - x_{jk}\right)^2 \qquad \text{Squared Euclidean distance}$$

- Assumption: All features are **equally important**;
- Such approaches fail in high dimensional spaces

Clustering: The Curse of Dimensionality

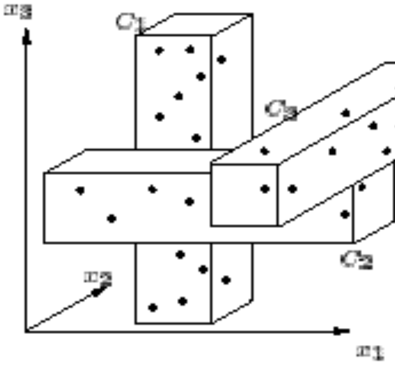
- A full-dimensional distance is often irrelevant, as the farthest point is expected to be almost as close as the nearest point;
- In high dimensional spaces, it is likely that, for any given pair of points within the same cluster, there exist at least a few dimensions on which the points are far apart from each other.

Example



Clustering

- Clusters may exist in different subspaces, comprised of different combinations of features:



Each dimension is relevant to at least one cluster

Global Dimensionality Reduction

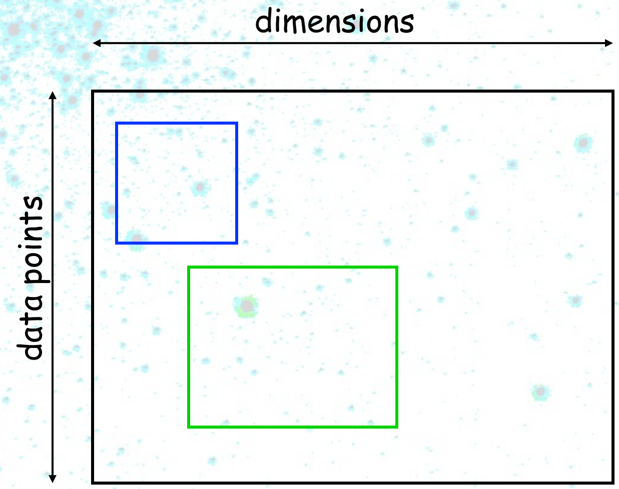
- We cannot prune off dimensions without incurring a loss of crucial information;
- Global dimensionality reduction techniques, e.g. PCA, do not handle well situations where different clusters are dense in different subspaces;
- The data presents local structure

Local Dimensionality Reduction

- To capture the local correlations of data, a proper feature selection procedure should operate locally;
- A local operation would allow to embed different distance measures in different regions;

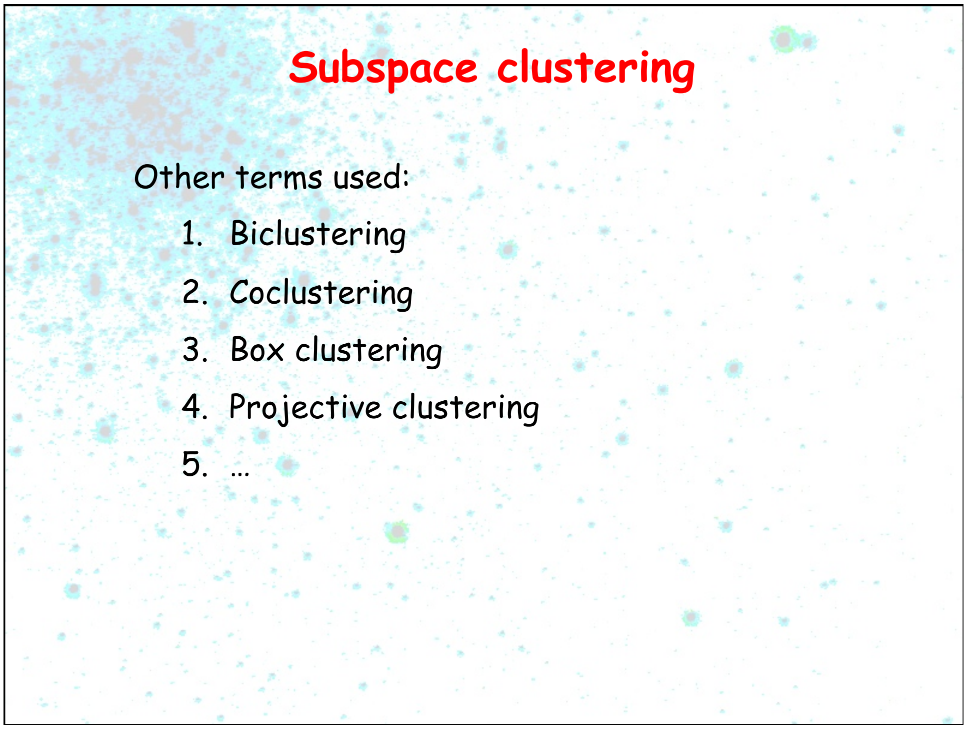
Subspace clustering

Simultaneous clustering of both row and column sets in a data matrix



dimensions

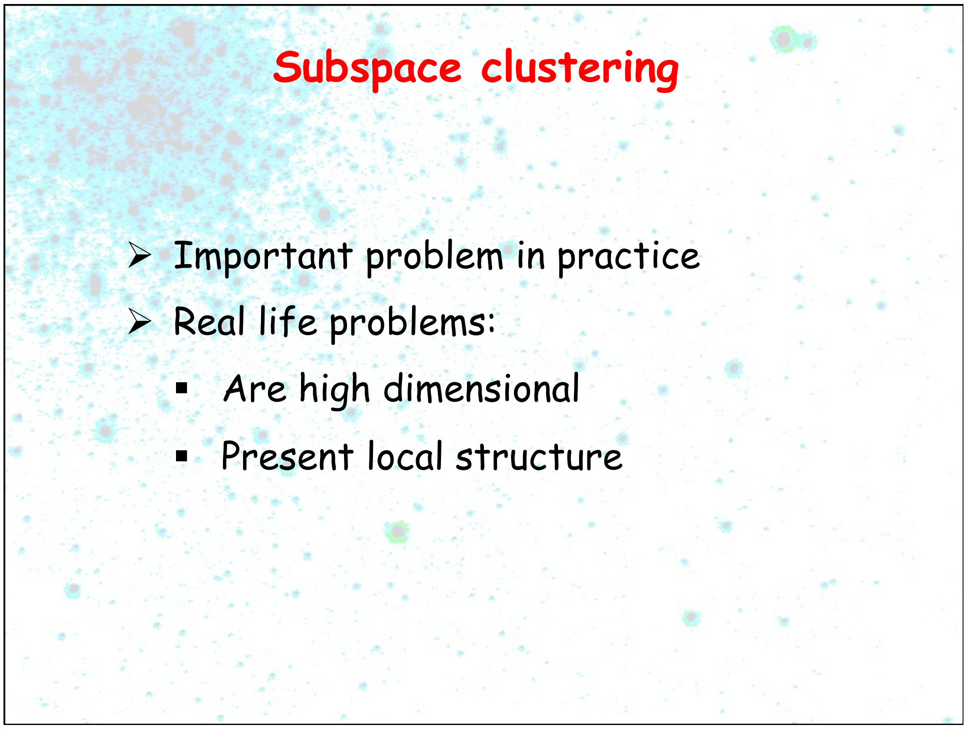
data points



Subspace clustering

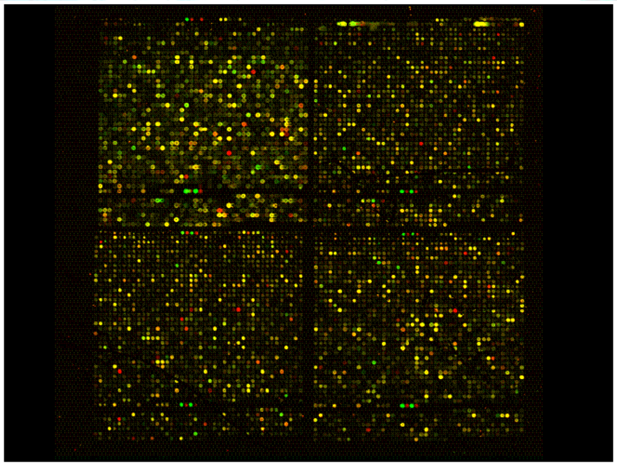
Other terms used:

1. Biclustering
2. Coclustering
3. Box clustering
4. Projective clustering
5. …



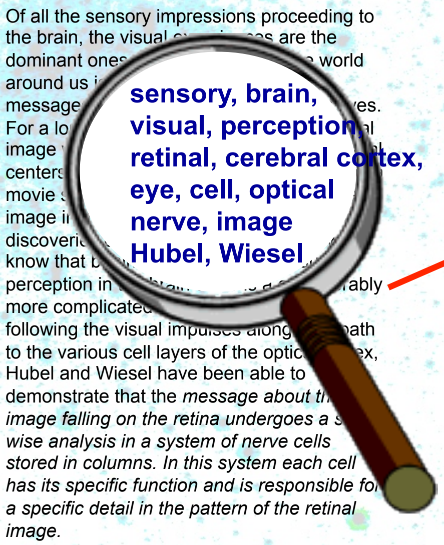
Subspace clustering

- Important problem in practice
- Real life problems:
  - Are high dimensional
  - Present local structure

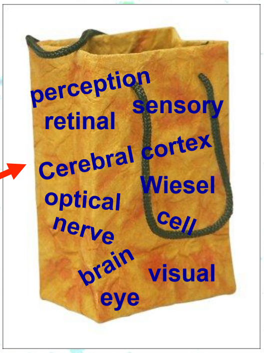


Clustering of Microarray data:

- Different conditions may have different importance for a given set of genes;
- The relevance of one condition may vary from gene to gene

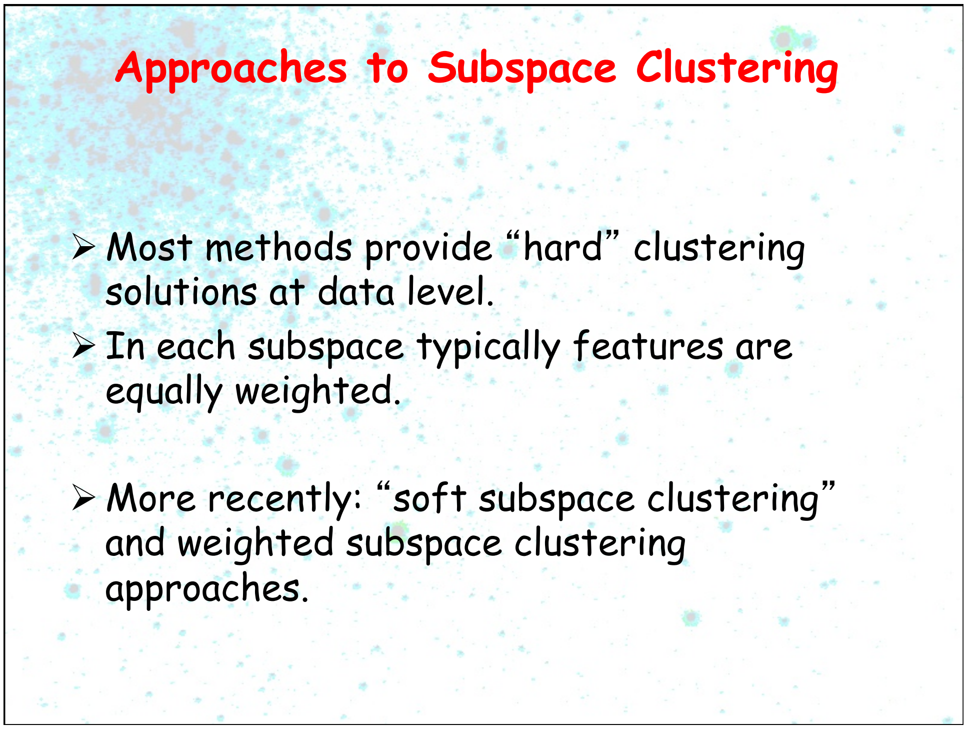


Of all the sensory impressions proceeding to
the brain, the visual ex... ces are the
dominant ones ... world
around us i...
message ... es.
For a lo... al
image ...
centers
movie ...
image i...
discoveri...
know that b...
perception in ... ably
more complicated...
following the visual impulses along ... path
to the various cell layers of the optic... ex,
Hubel and Wiesel have been able to
demonstrate that the *message about th...*
image falling on the retina undergoes a s...
wise analysis in a system of nerve cells
stored in columns. In this system each cell
has its specific function and is responsible fo...
a specific detail in the pattern of the retinal
image.



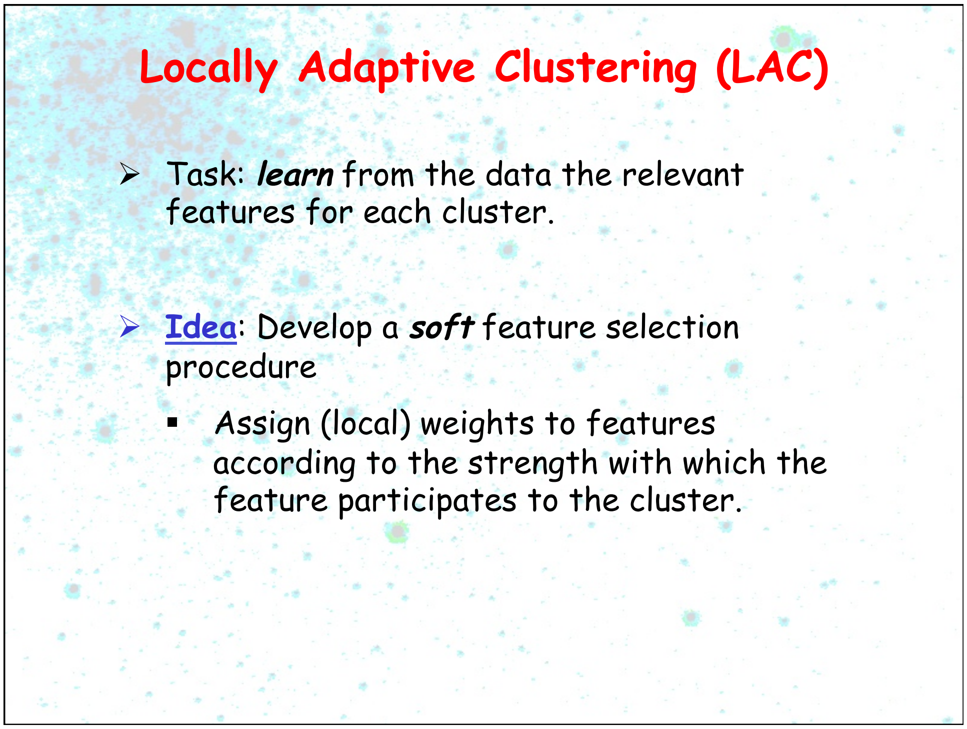
Bag-of-words representation of a document

Text classification: Different words may have different degrees of relevance for a given category of documents;
A single word may have a different importance across different categories.



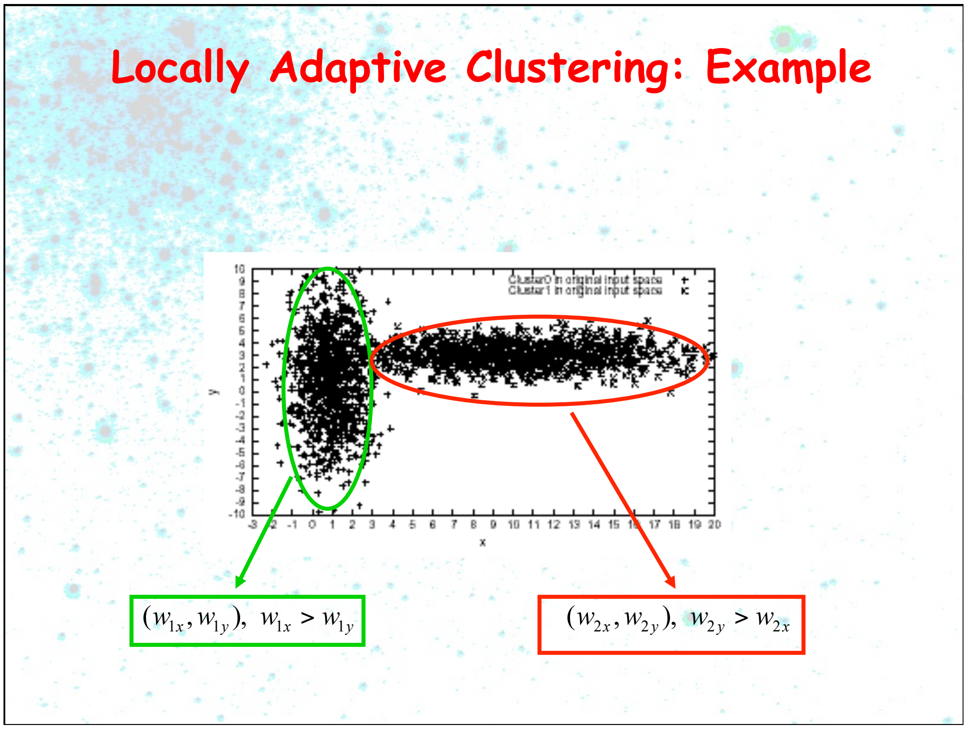
Approaches to Subspace Clustering

- Most methods provide "hard" clustering solutions at data level.
- In each subspace typically features are equally weighted.
- More recently: "soft subspace clustering" and weighted subspace clustering approaches.



Locally Adaptive Clustering (LAC)

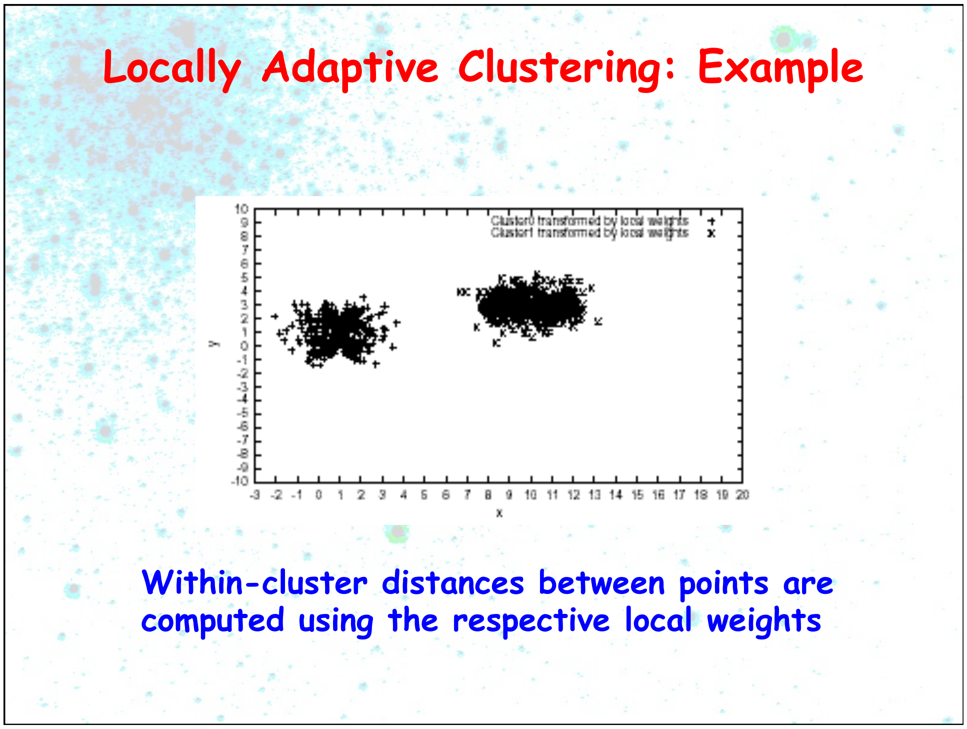
- Task: ***learn*** from the data the relevant features for each cluster.
- Idea: Develop a ***soft*** feature selection procedure
  - Assign (local) weights to features according to the strength with which the feature participates to the cluster.



Locally Adaptive Clustering: Example

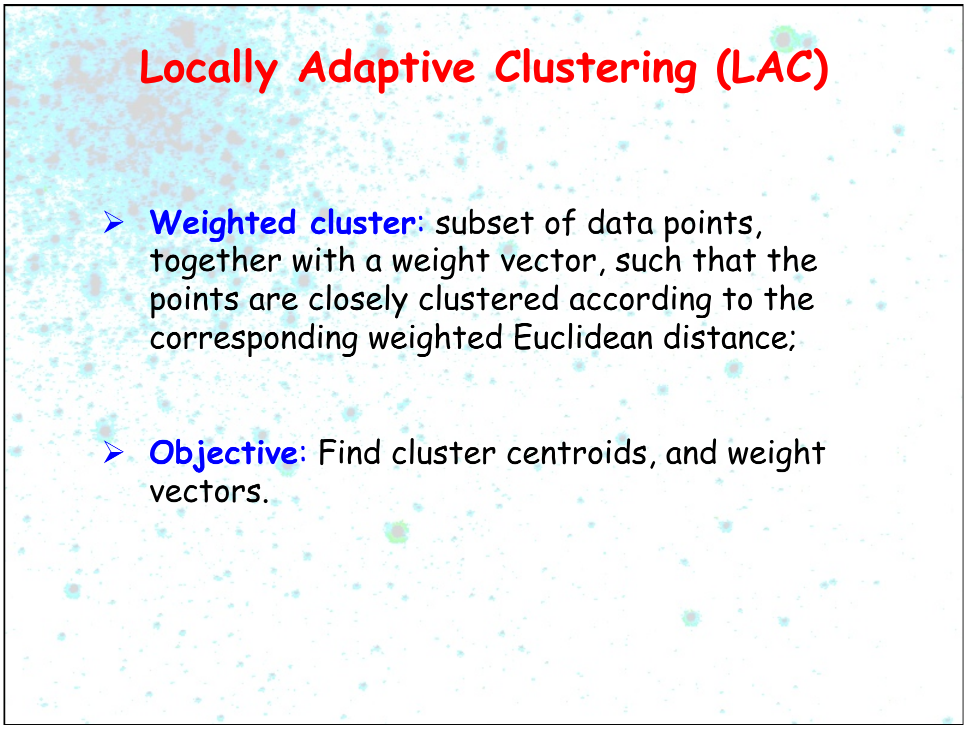
$(w_{1x}, w_{1y}),\ w_{1x} > w_{1y}$

$(w_{2x}, w_{2y}),\ w_{2y} > w_{2x}$



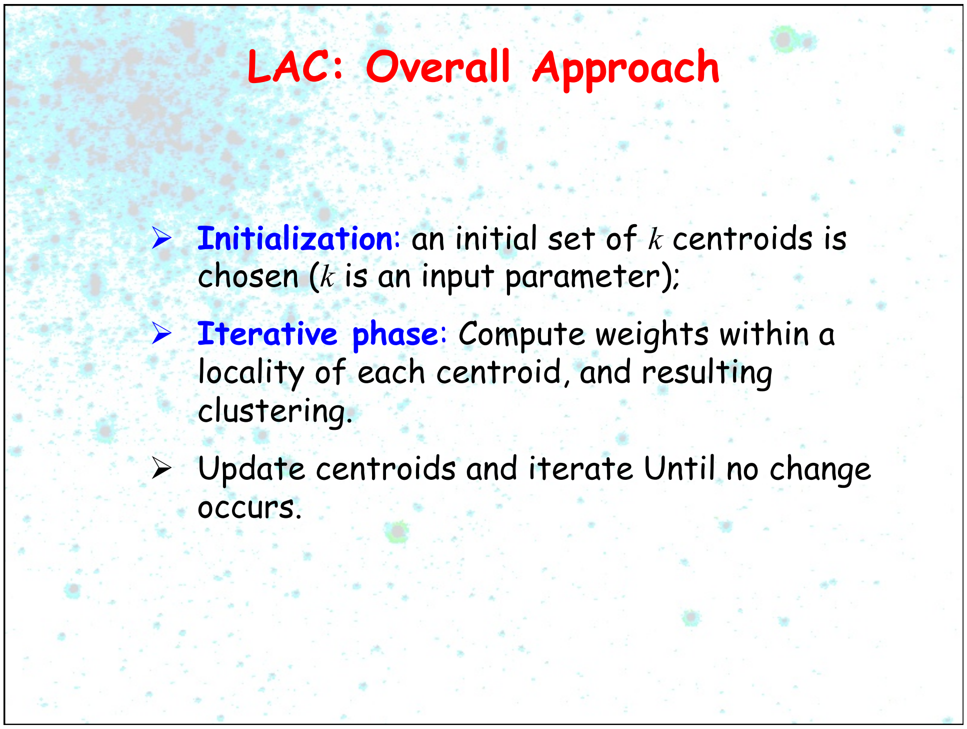
Locally Adaptive Clustering: Example

Within-cluster distances between points are computed using the respective local weights



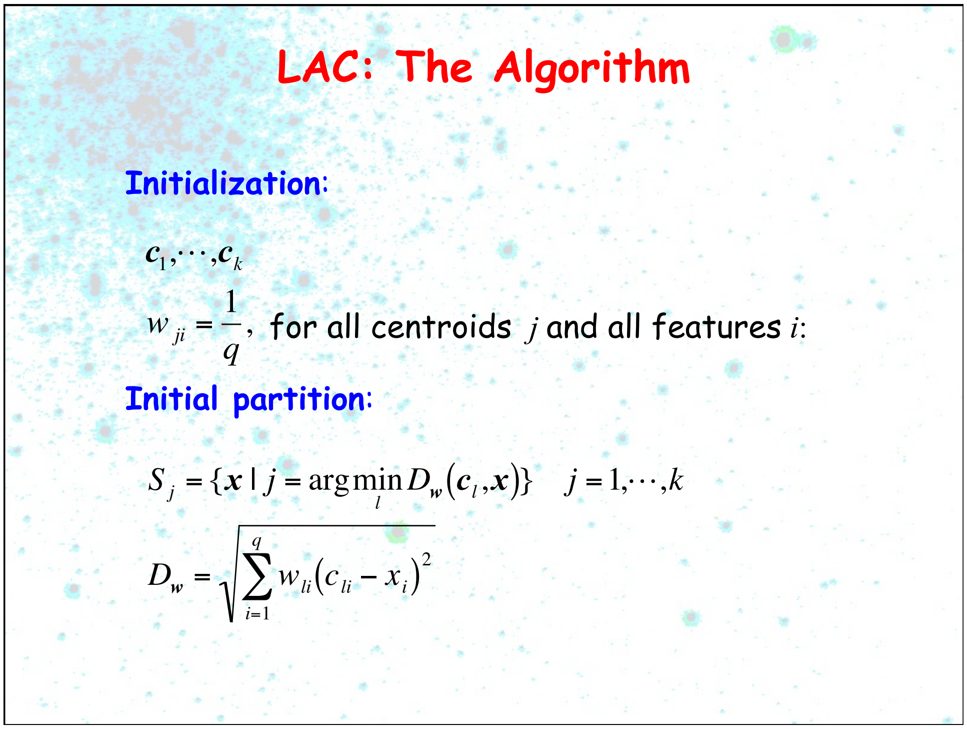
Locally Adaptive Clustering (LAC)

- **Weighted cluster:** subset of data points, together with a weight vector, such that the points are closely clustered according to the corresponding weighted Euclidean distance;
- **Objective:** Find cluster centroids, and weight vectors.



LAC: Overall Approach

- **Initialization:** an initial set of k centroids is chosen (k is an input parameter);
- **Iterative phase:** Compute weights within a locality of each centroid, and resulting clustering.
- Update centroids and iterate Until no change occurs.



LAC: The Algorithm

Initialization:

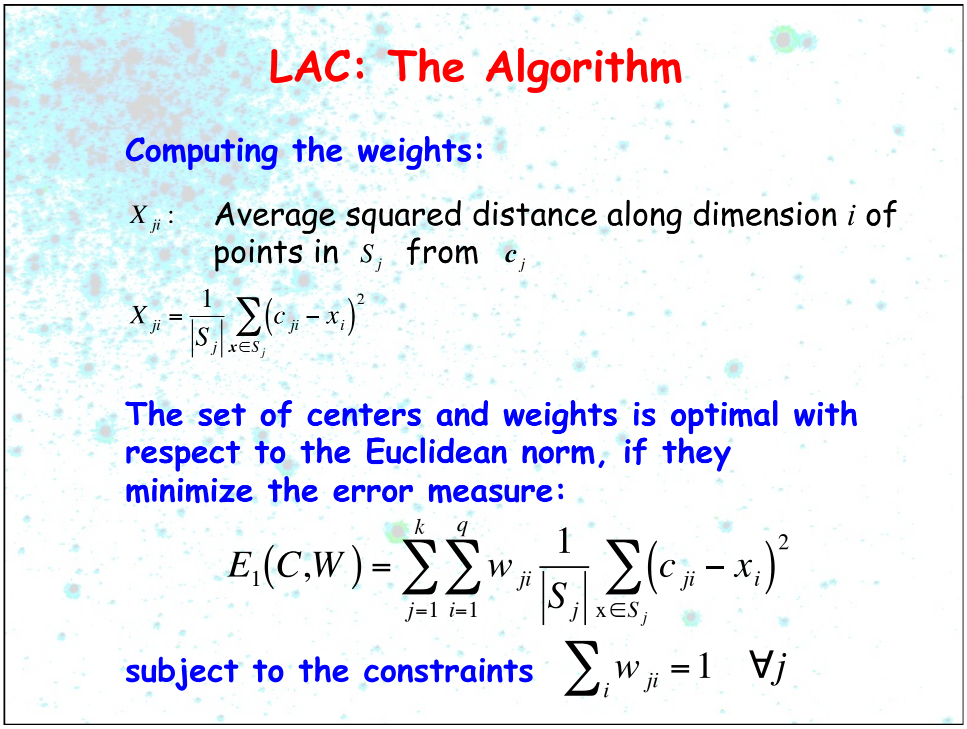
$\boldsymbol{c}_1,\cdots,\boldsymbol{c}_k$

$w_{ji} = \frac{1}{q}$, for all centroids j and all features i:

Initial partition:

$$S_j = \{\boldsymbol{x} \mid j = \arg\min_l D_w(\boldsymbol{c}_l,\boldsymbol{x})\} \quad j = 1,\cdots,k$$

$$D_w = \sqrt{\sum_{i=1}^{q} w_{li}(c_{li} - x_i)^2}$$



LAC: The Algorithm

Computing the weights:

X_{ji}: Average squared distance along dimension i of points in S_j from $\boldsymbol{c}_j$

$$X_{ji} = \frac{1}{|S_j|}\sum_{x \in S_j}(c_{ji} - x_i)^2$$

The set of centers and weights is optimal with respect to the Euclidean norm, if they minimize the error measure:

$$E_1(C,W) = \sum_{j=1}^{k}\sum_{i=1}^{q} w_{ji}\frac{1}{|S_j|}\sum_{\mathrm{x} \in S_j}(c_{ji} - x_i)^2$$

subject to the constraints $\sum_i w_{ji} = 1 \quad \forall j$

LAC: The Algorithm

$$E_1(C,W)=\sum_{j=1}^{k}\sum_{i=1}^{q} w_{ji}\frac{1}{|S_j|}\sum_{\mathrm{x}\in S_j}\left(c_{ji}-x_i\right)^2$$

The solution:

$$\left(C^*,W^*\right)=\arg\min_{(C,W)} E_1(C,W)$$

discovers one dimensional clusters.

LAC: The Algorithm

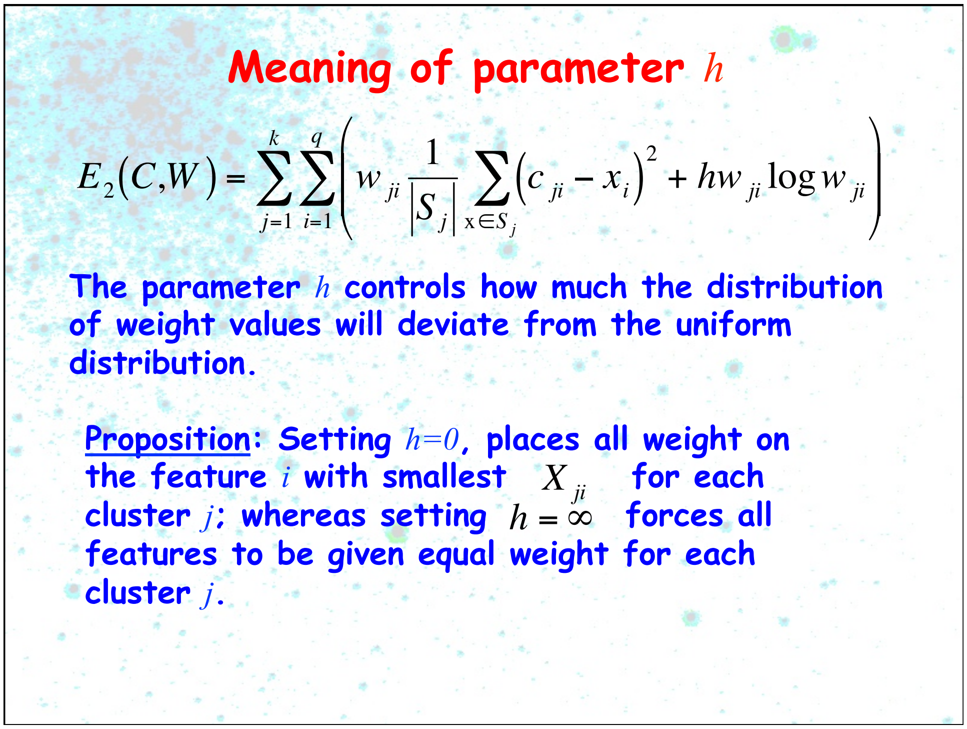
Modified error function:

$$E_2(C,W)=\sum_{j=1}^{k}\sum_{i=1}^{q}\left(w_{ji}\frac{1}{|S_j|}\sum_{\mathrm{x}\in S_j}\left(c_{ji}-x_i\right)^2+hw_{ji}\log w_{ji}\right)$$

subject to the same constraints $\sum_i w_{ji}=1 \quad \forall j$

Optimal solution:

$$w^*_{ji}=\frac{\exp\left(-X_{ji}/h\right)}{\sum_{i=1}^{q}\exp\left(-X_{ji}/h\right)} \qquad c^*_{ji}=\frac{1}{|S_j|}\sum_{\mathrm{x}\in S_j}x_i$$

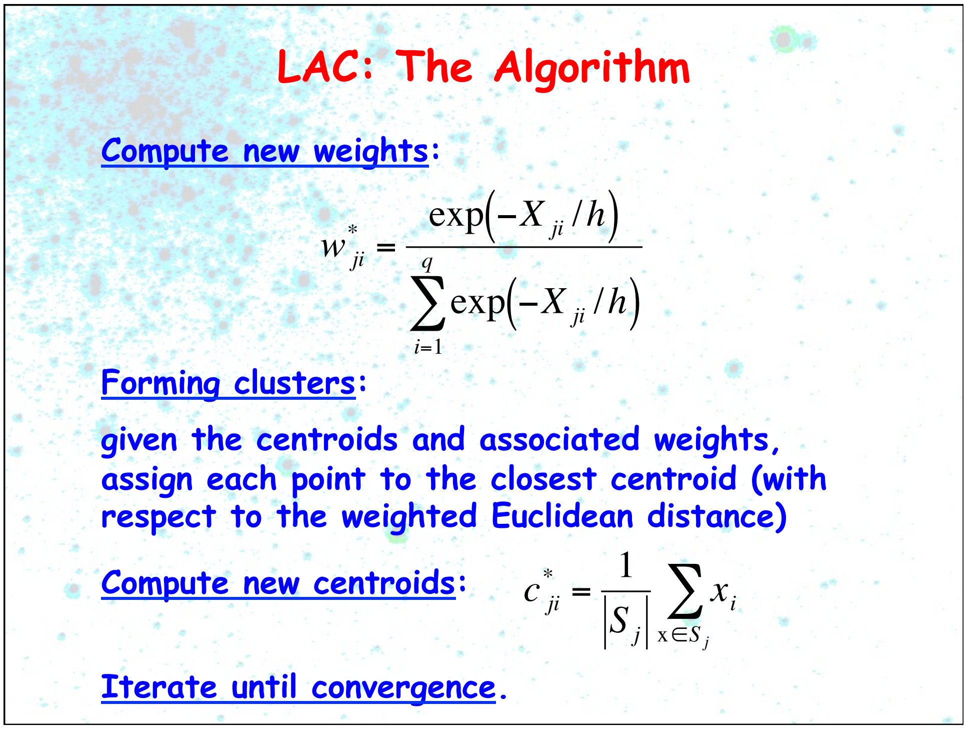


Meaning of parameter h

$$E_2(C,W)=\sum_{j=1}^{k}\sum_{i=1}^{q}\left(w_{ji}\frac{1}{|S_j|}\sum_{x\in S_j}(c_{ji}-x_i)^2+hw_{ji}\log w_{ji}\right)$$

The parameter h controls how much the distribution of weight values will deviate from the uniform distribution.

Proposition: Setting $h=0$, places all weight on the feature i with smallest X_{ji} for each cluster j; whereas setting $h=\infty$ forces all features to be given equal weight for each cluster j.



LAC: The Algorithm

Compute new weights:

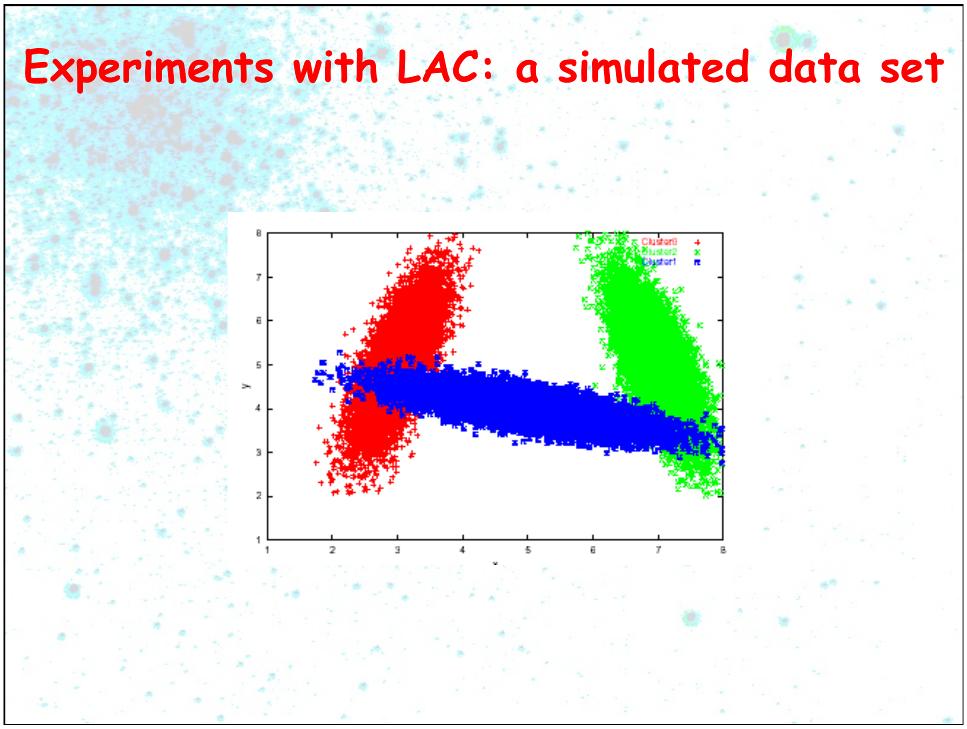
$$w^*_{ji}=\frac{\exp(-X_{ji}/h)}{\sum_{i=1}^{q}\exp(-X_{ji}/h)}$$

Forming clusters:

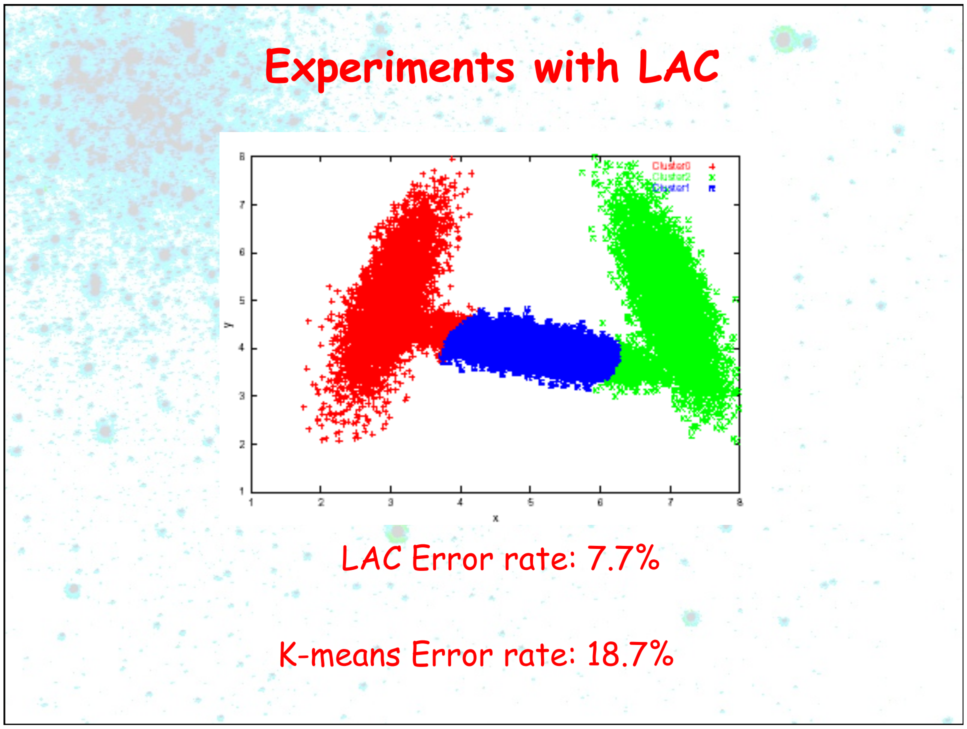
given the centroids and associated weights, assign each point to the closest centroid (with respect to the weighted Euclidean distance)

Compute new centroids: $c^*_{ji}=\frac{1}{|S_j|}\sum_{x\in S_j}x_i$

Iterate until convergence.



Experiments with LAC: a simulated data set

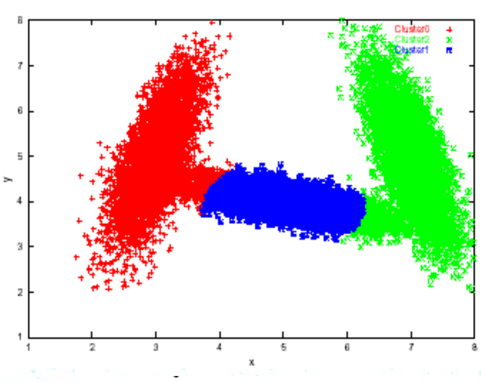


Experiments with LAC

LAC Error rate: 7.7%

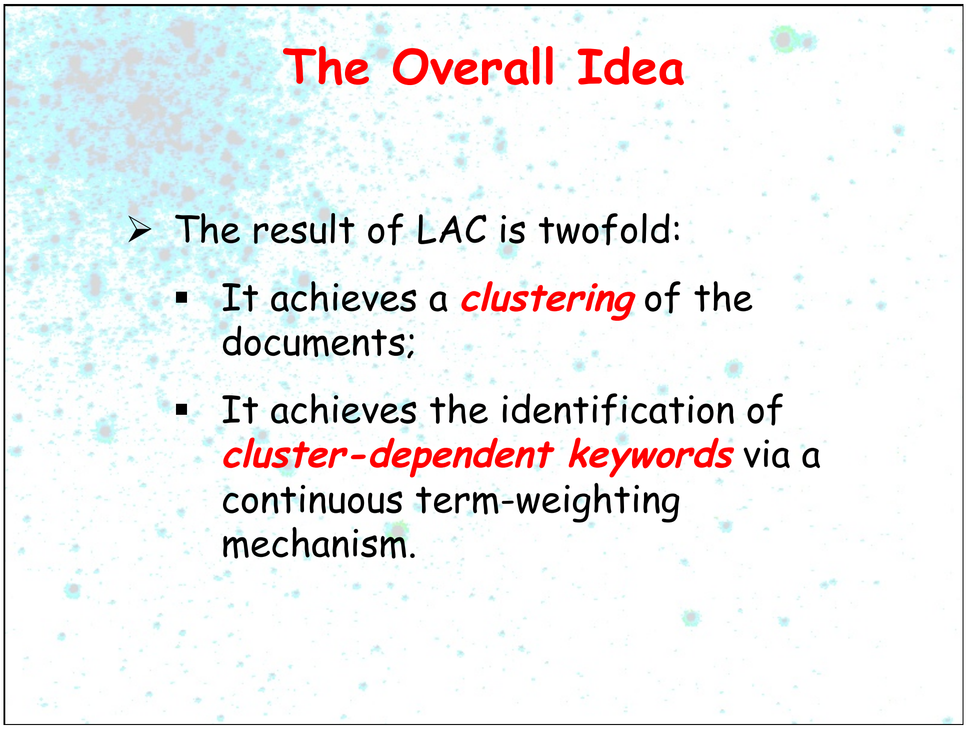
K-means Error rate: 18.7%

Experiments with LAC



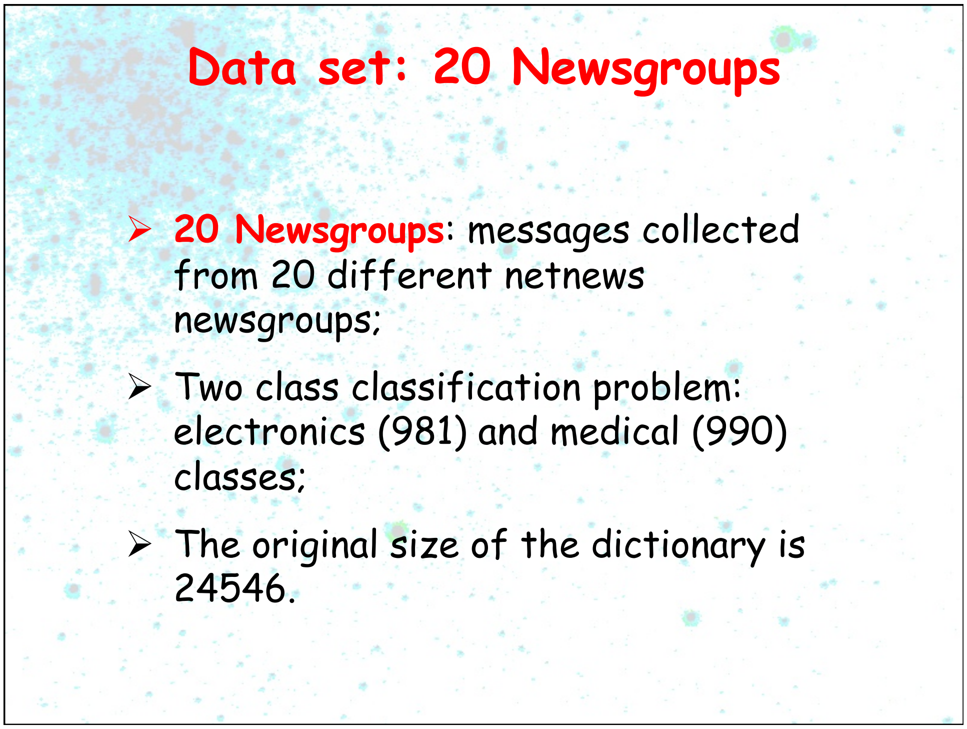
Cluster	w_1	w_2
C0	0.92	0.08
C1	0.44	0.56
C2	0.94	0.06

Categorization and Keyword Identification of Unlabeled Documents



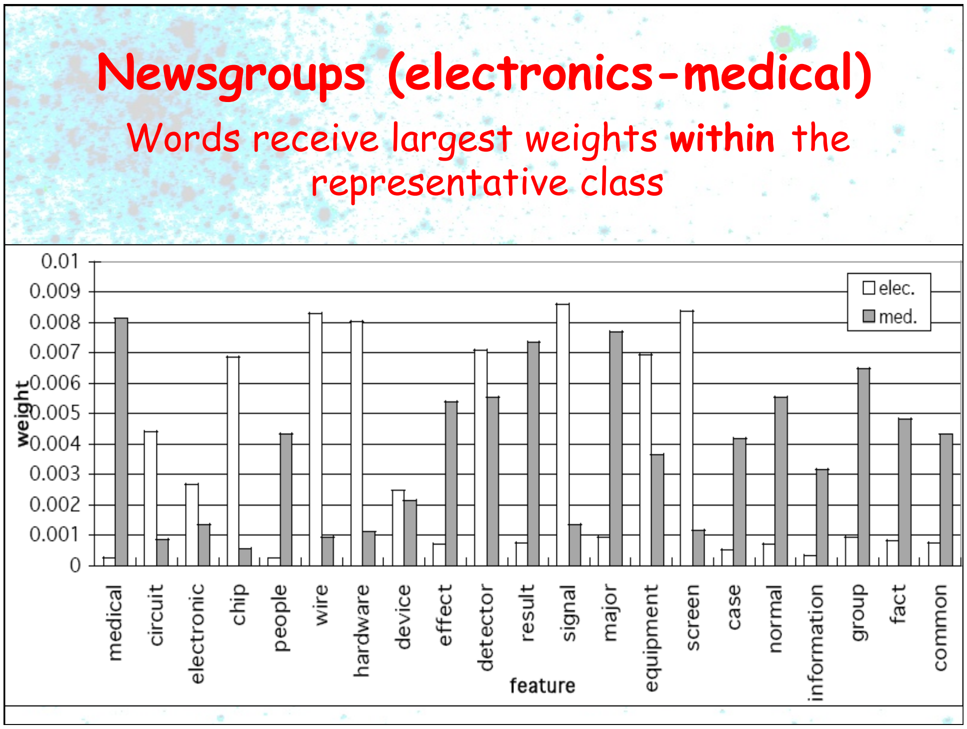
The Overall Idea

- The result of LAC is twofold:
  - It achieves a ***clustering*** of the documents;
  - It achieves the identification of ***cluster-dependent keywords*** via a continuous term-weighting mechanism.



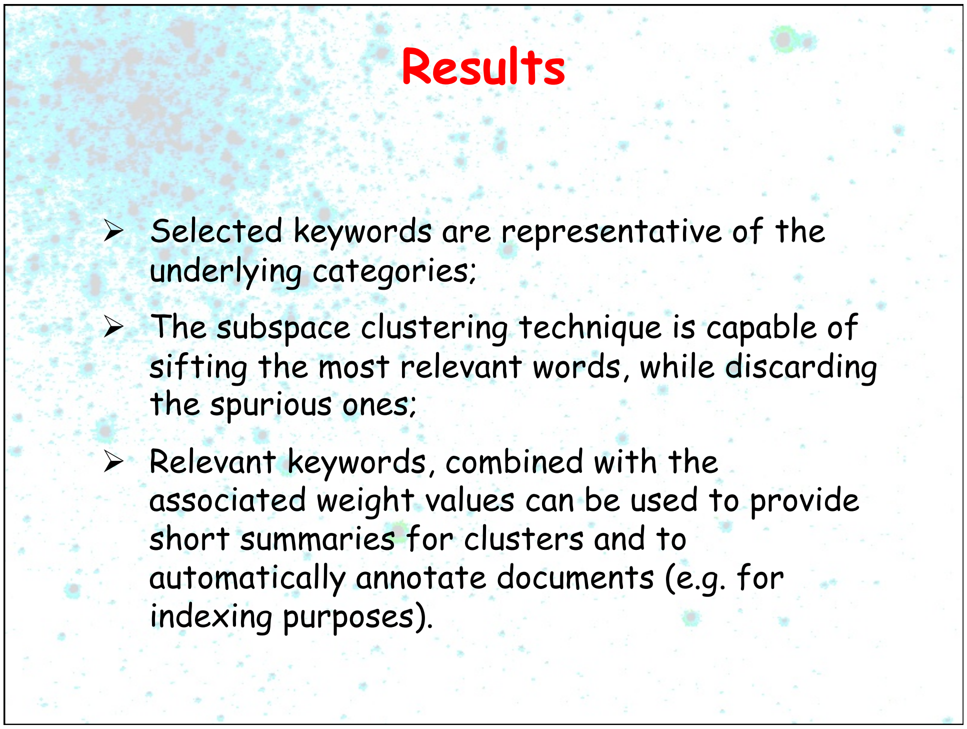
Data set: 20 Newsgroups

- **20 Newsgroups**: messages collected from 20 different netnews newsgroups;
- Two class classification problem: electronics (981) and medical (990) classes;
- The original size of the dictionary is 24546.



Newsgroups (electronics-medical)

Words receive largest weights **within** the representative class



Results

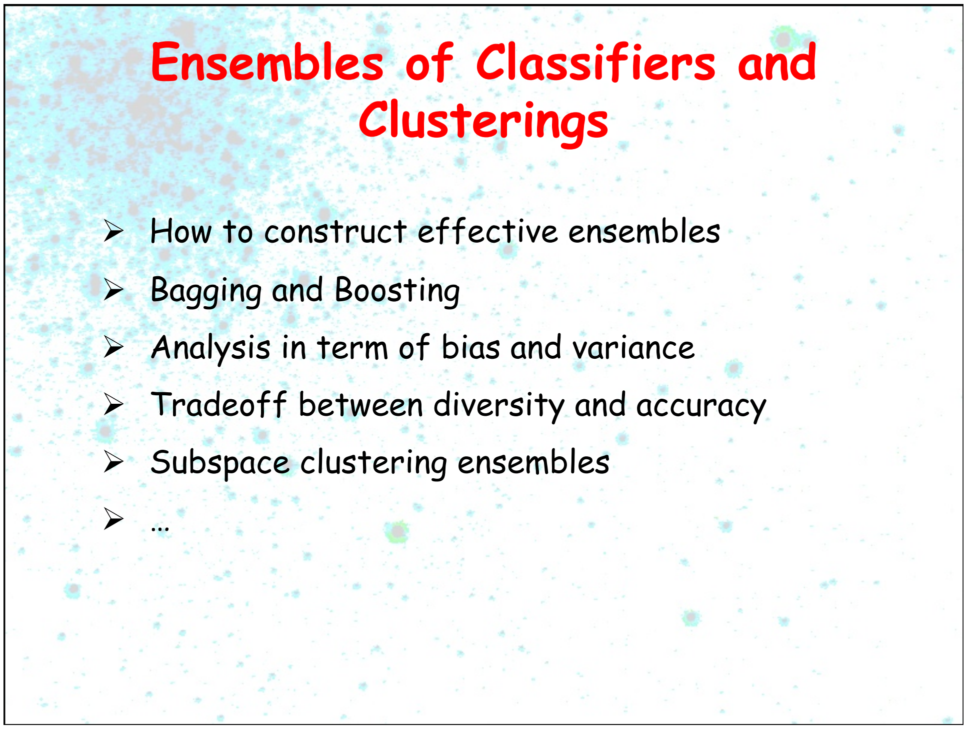
- Selected keywords are representative of the underlying categories;
- The subspace clustering technique is capable of sifting the most relevant words, while discarding the spurious ones;
- Relevant keywords, combined with the associated weight values can be used to provide short summaries for clusters and to automatically annotate documents (e.g. for indexing purposes).

Clustering: An ill-posed Problem

- Document clustering: Based on content? Based on style? Based on authorship?
- Given a data set, different clustering algorithms are likely to produce different results.
- Given a data set, the same algorithm with different parameter settings is likely to produce different results. E.g.: k-means with different random initialization.
- What do we do?

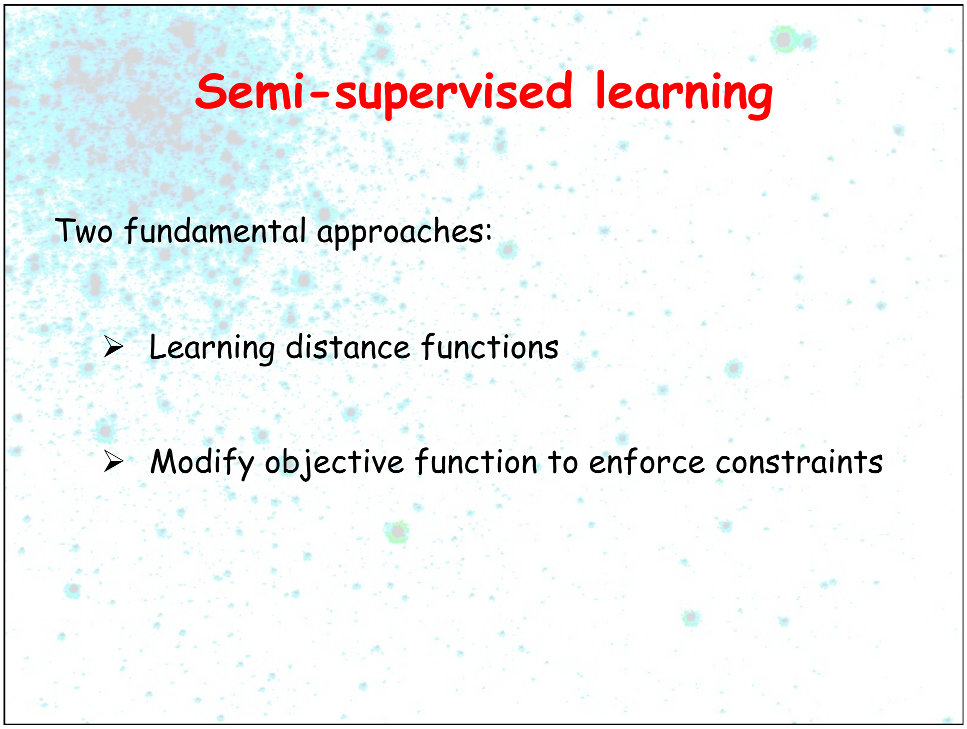
Clustering: An ill-posed Problem

- Solutions:
  - CLUSTERING ENSEMBLES
  - SEMI-SUPERVISED CLUSTERING



Ensembles of Classifiers and Clusterings

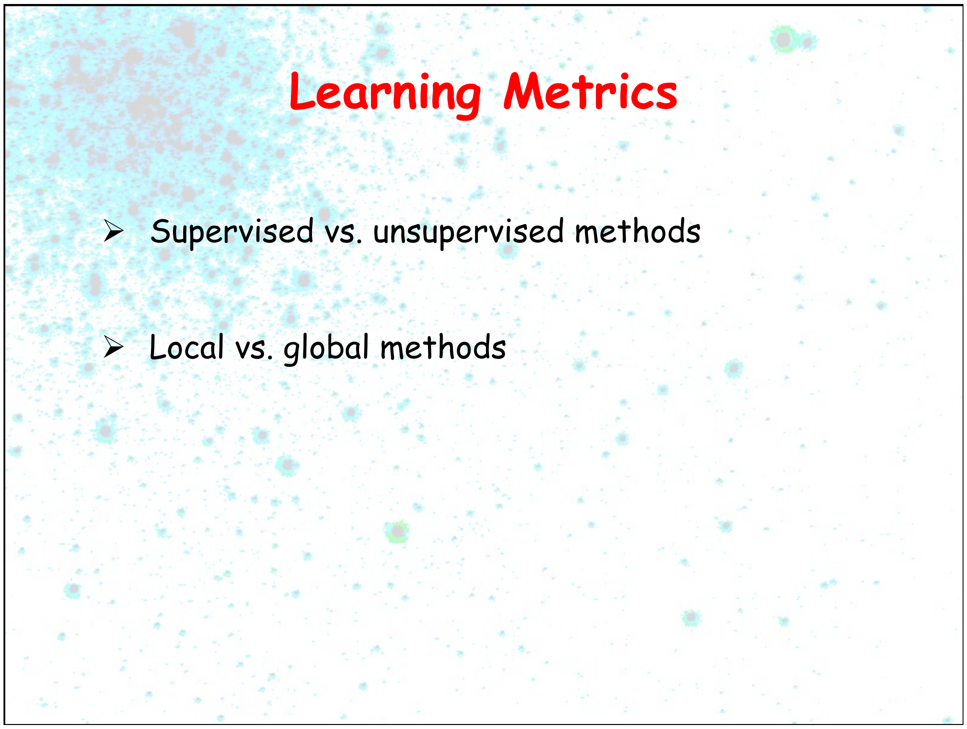
- How to construct effective ensembles
- Bagging and Boosting
- Analysis in term of bias and variance
- Tradeoff between diversity and accuracy
- Subspace clustering ensembles
- …



Semi-supervised learning

Two fundamental approaches:

- Learning distance functions
- Modify objective function to enforce constraints



Learning Metrics

- Supervised vs. unsupervised methods
- Local vs. global methods