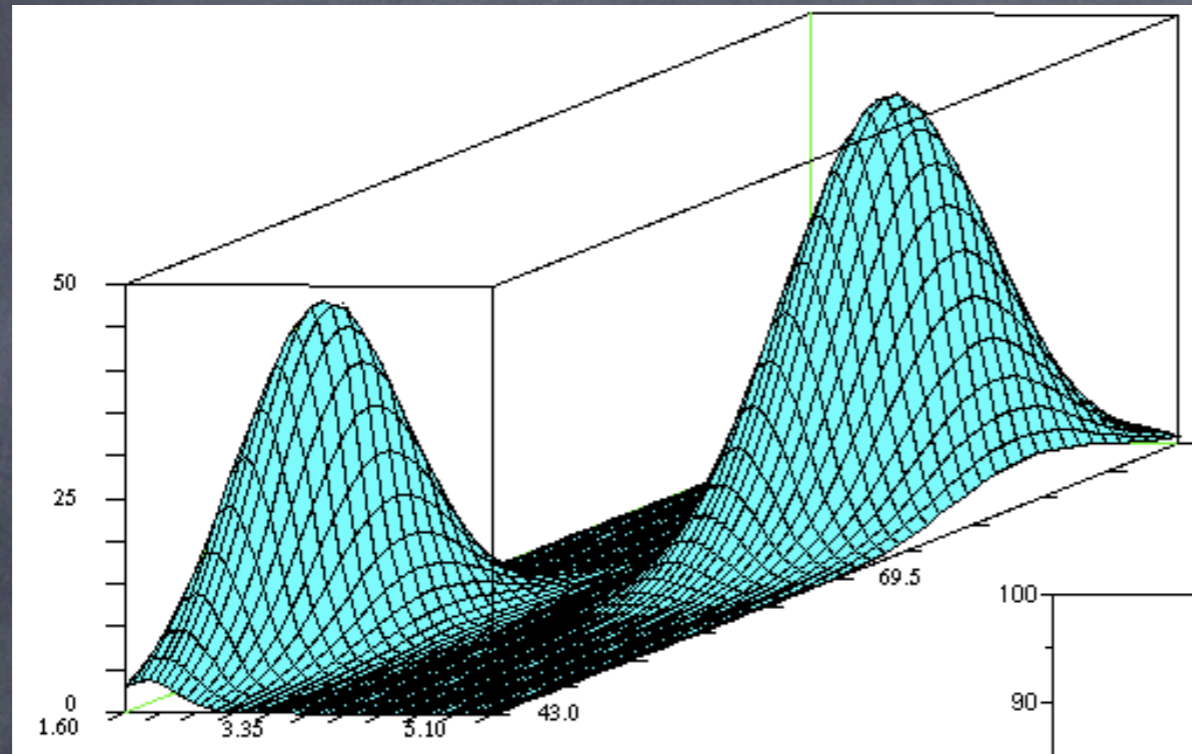


Unsupervised Learning and Cluster Analysis

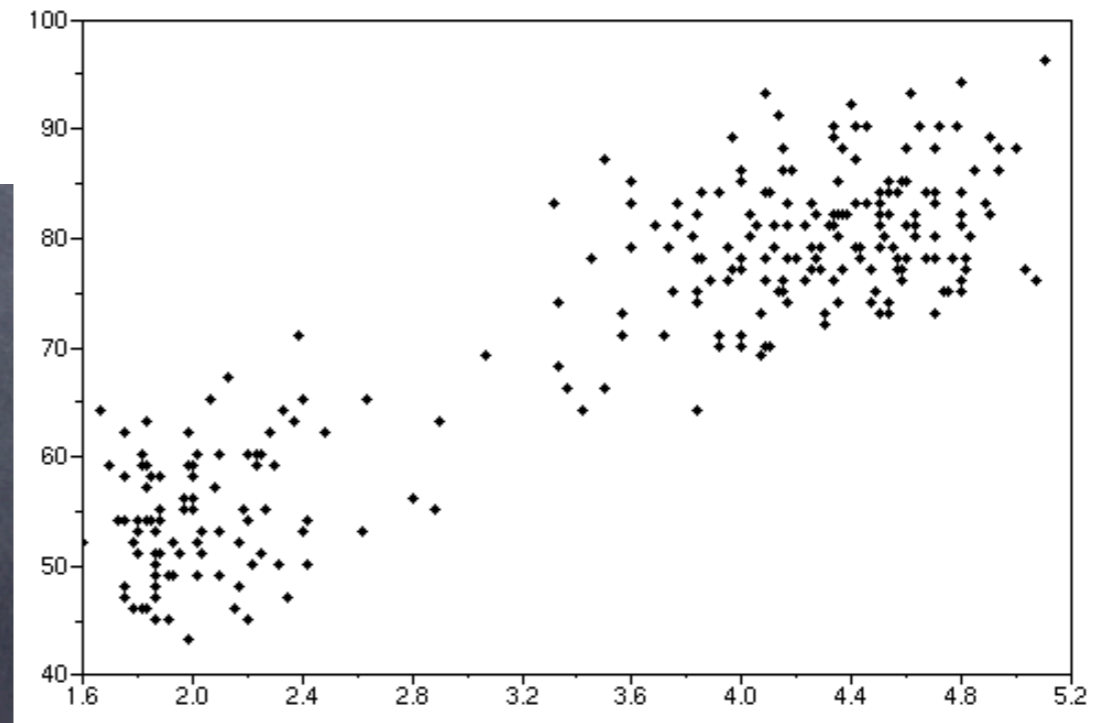
- Problem
- Roadmap to finding a solution
- Example: K-means
- Choices
- Overcoming arbitrariness

Density Estimation



Process, described by
probability distribution,
generates data

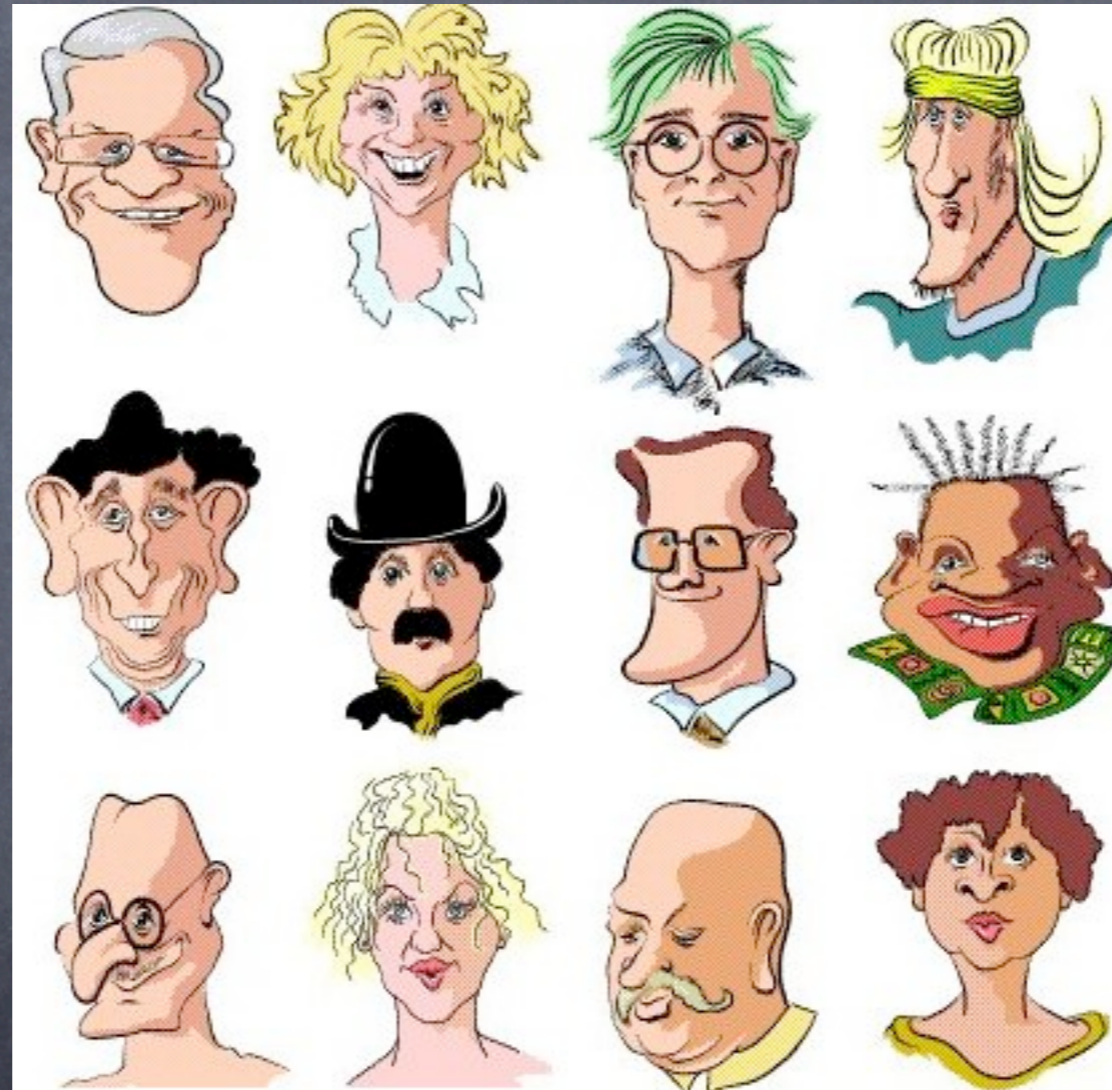
Inference task:
Infer the underlying
distribution from the data



Solving a simpler problem

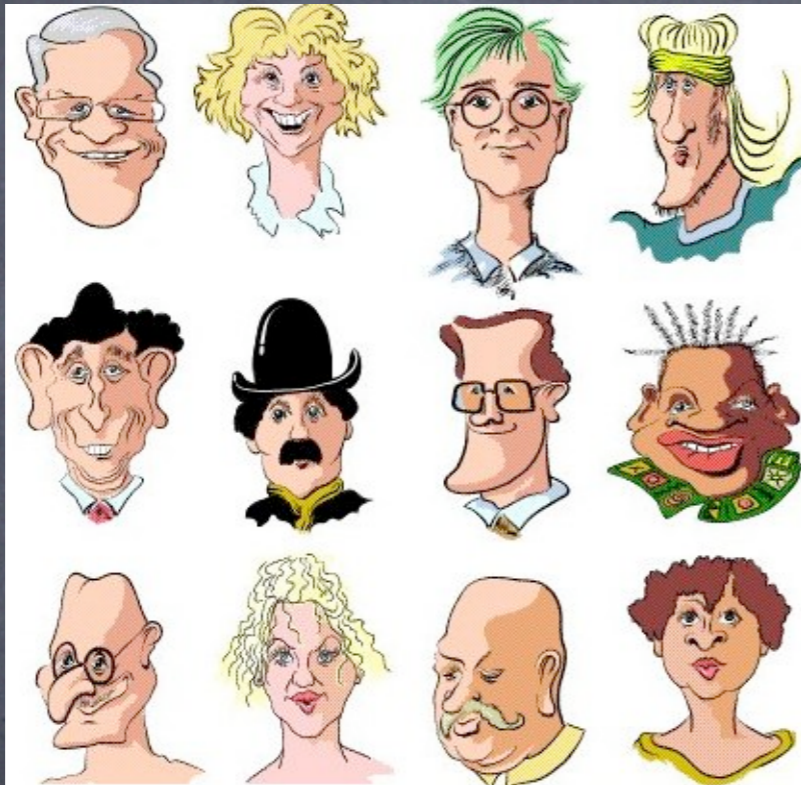
- Recall: Supervised Learning – Find decision boundary.
- Now: Find high density regions of support.
- Find a partition of the input.
- “Group similar objects together, different objects into different groups”

Raw data



How do we describe the data?

Measured Features



case	sex	glasses	moustache	smile	hat
1	m	y	n	y	n
2	f	n	n	y	n
3	m	y	n	n	n
4	m	n	n	n	n
5	m	n	n	y?	n
6	m	n	y	n	y
7	m	y	n	y	n
8	m	n	n	y	n
9	m	y	y	y	n
10	f	n	n	n	n
11	m	n	y	n	n
12	f	n	n	n	n

<http://149.170.199.144/multivar/ca.htm>

What does it mean for two faces to be similar?

Roadmap

- Similarity measure
- Clustering criterion (objective function)
- Algorithm (finds optimal K partition)
- Number of clusters K

Need to specify these criteria.

Problem: Solution depends on choices.

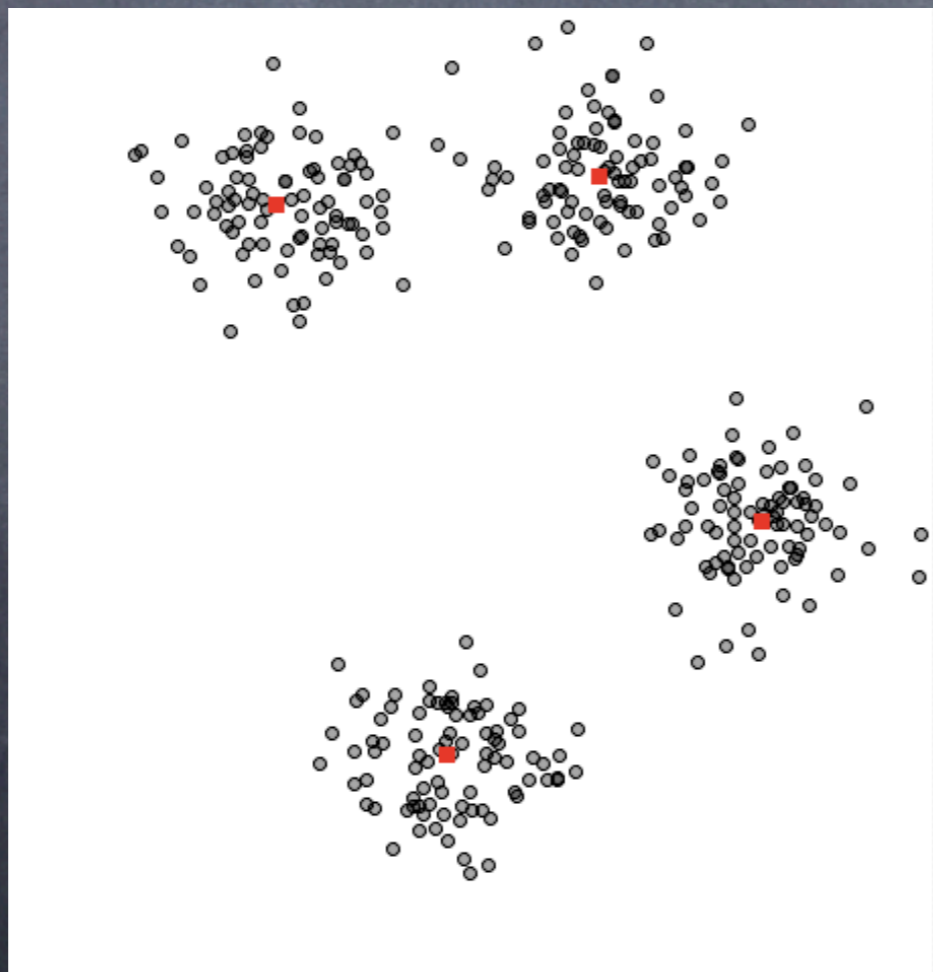
Theoretical guidance?

Underlying principles?

Simple Example: K-means

(MacQueen, 1967)

The data, visualized in 2-dim input space:

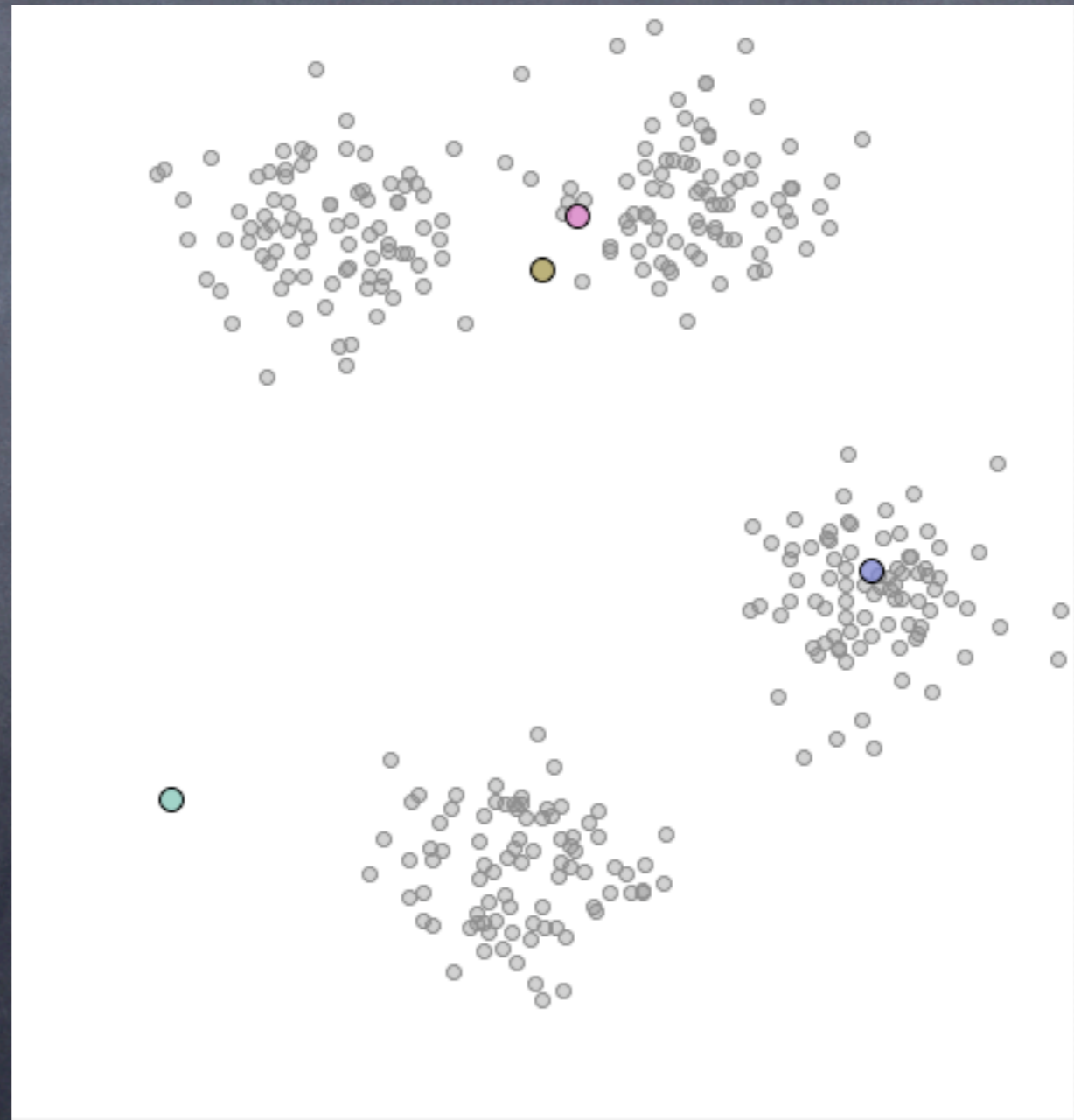


The red squares indicate the "true" cluster centers.

Algorithm doesn't know those.

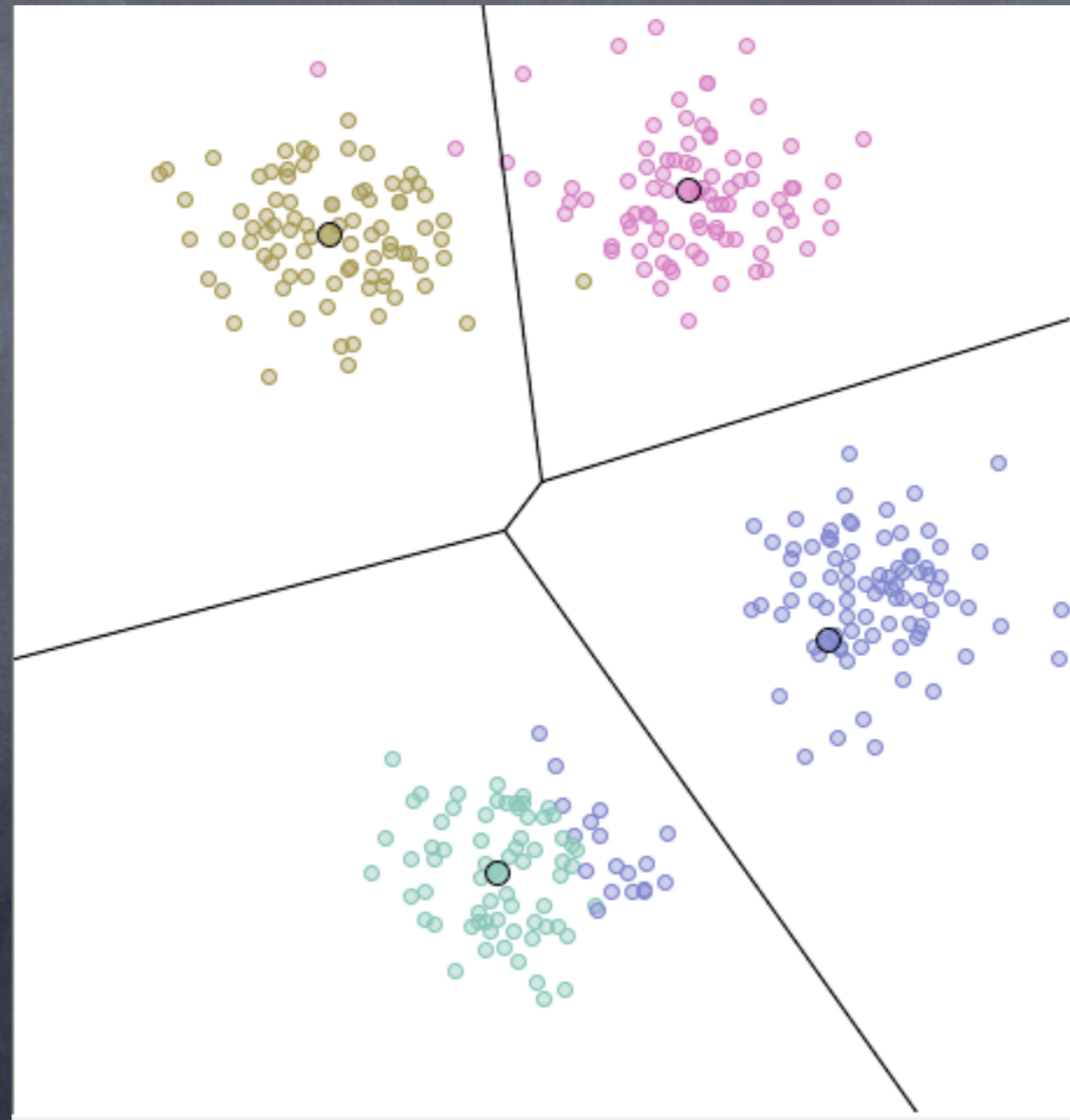
The algorithm

1. Initialize K cluster centers. Here $K=4$.

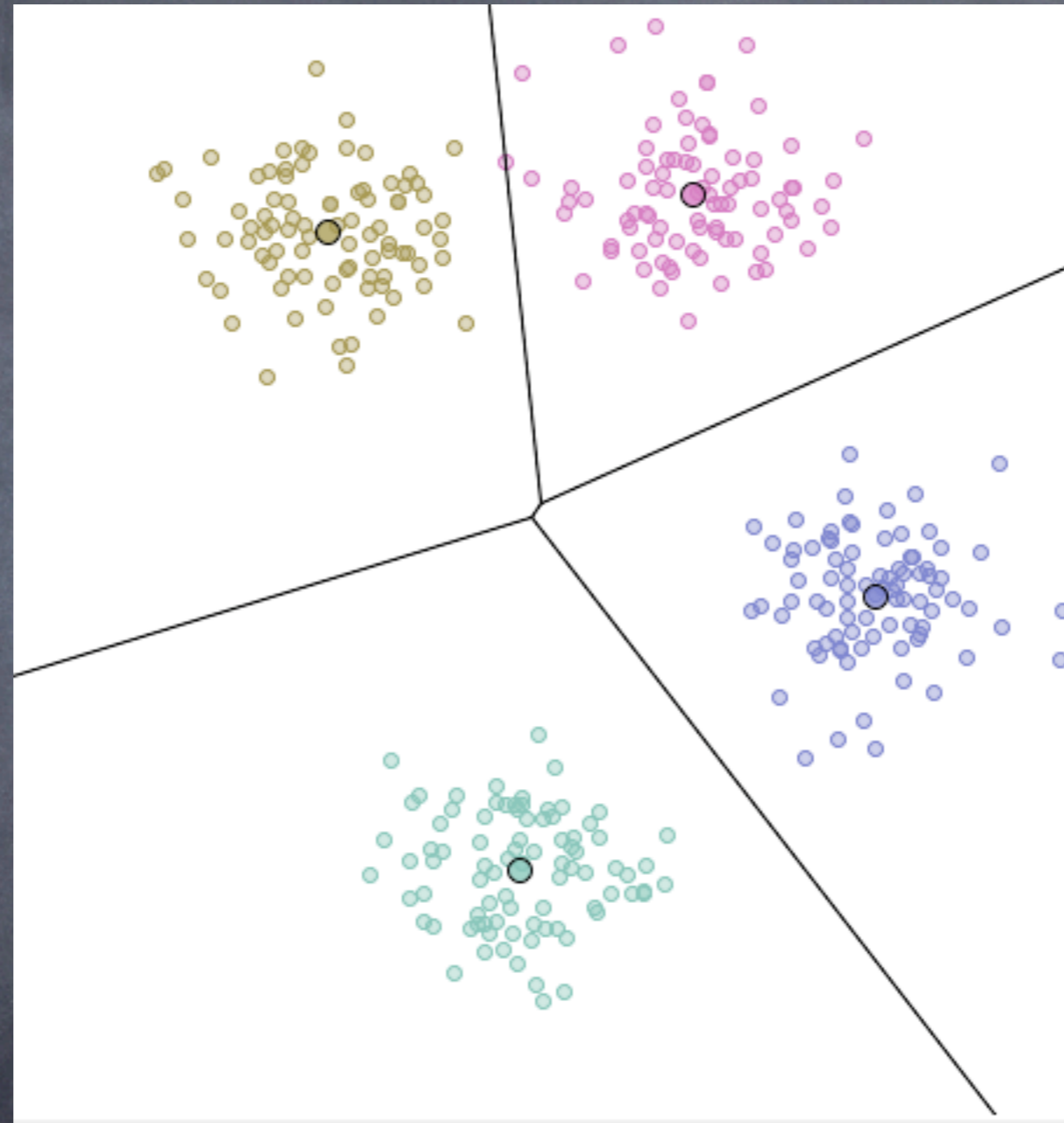


2. Assign each point to nearest cluster center

3. Re-compute cluster centers as the means of the points assigned to each cluster



- Repeat until nothing changes anymore



- Fast convergence! (here after only 2 steps)

Objective function

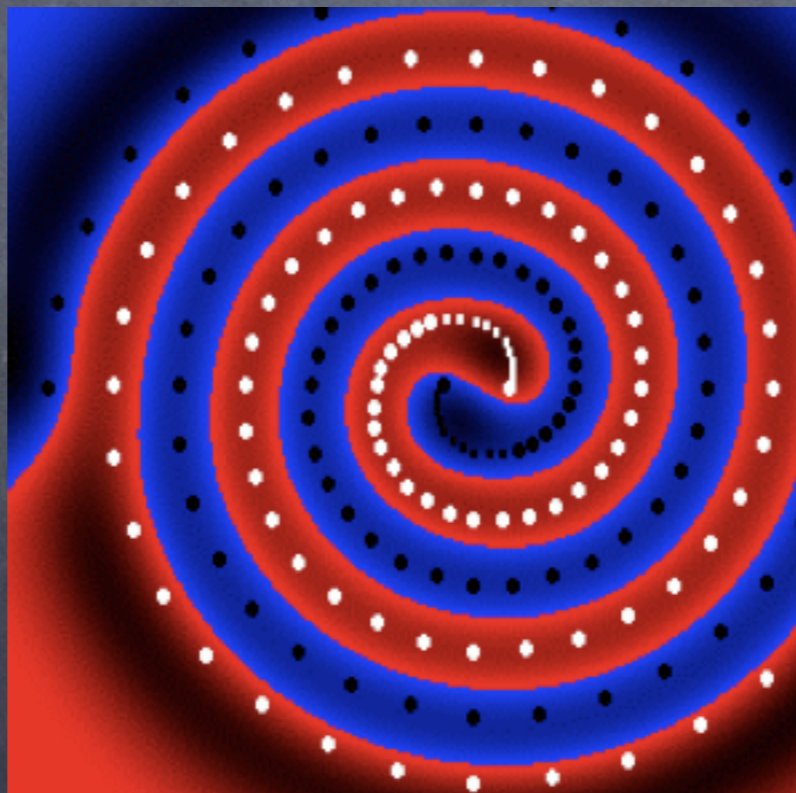
- Can show that K-means algorithm minimizes the squared error function ($c = 1, \dots, K$ cluster centers; x : N data points)

$$\sum_{j=1}^K \sum_{i=1}^N \|x_i^{(j)} - c_j\|^2$$

- Other objective functions could be used!
- It is not always obvious which one to choose.

Similarity measure

- Squared distance (euclid norm) $\|x_i^{(j)} - c_j\|^2$
- Not good for data distributions with non-euclidean structure. Example:



Similarity measures

- Metric, i.e. dis-similarity (j: dimension; l,m: object index)

- * Minkowski
$$d_{lm} = \left(\sum_{j=1}^D w_j^\lambda |x_l^j - x_m^j|^\lambda \right)^{\frac{1}{\lambda}}$$

- * City block
$$d_{lm} = \sum_{j=1}^D w_j |x_l^j - x_m^j|$$

- Similarity

- * Correlation coefficient; $\bar{x}_l = \sum_{j=1}^D x_l^j / D$

$$s_{lm} = \frac{\sum_{j=1}^D (x_l^j - \bar{x}_l)(x_m^j - \bar{x}_m)}{\left(\sum_{j=1}^D (x_l^j - \bar{x}_l)^2 \sum_{j=1}^D (x_m^j - \bar{x}_m)^2 \right)^{1/2}}$$

Measures cosine of angle between two vectors, originating at the mean of the data.

- and many more...

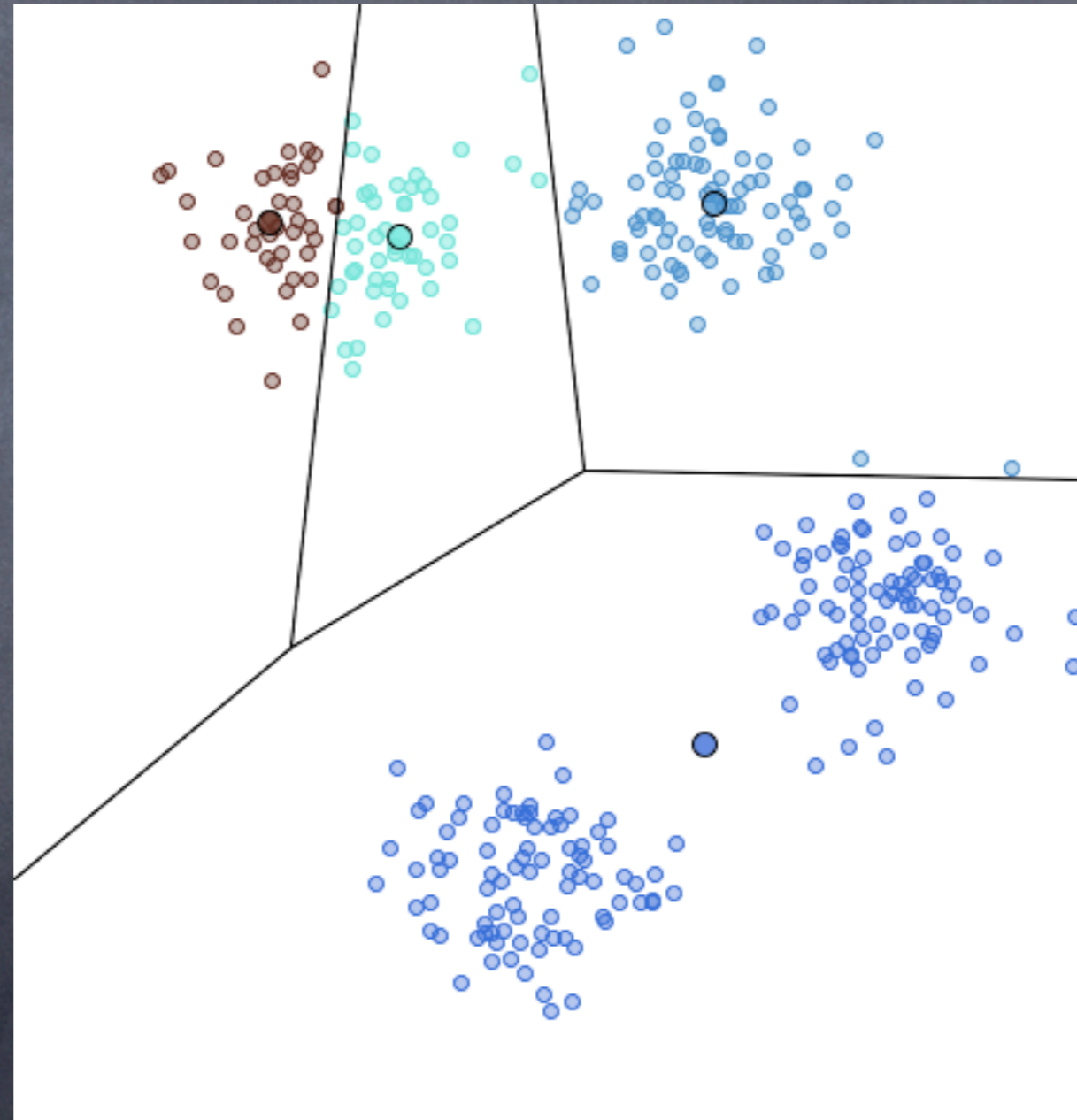
Objective functions

1. Measures of heterogeneity and isolation for each cluster:
 - **Averages** (sum of squares as before; average L1 norm $|x-c|$)
 - **Extrema** (Diameter: measures the dissimilarity between the most dissimilar points in a cluster; Split: smallest dissimilarity between an object in a class and an object outside the class)
2. Combine those by {sum, min, max} → Clustering Criterion (= objective function, OF)
3. Optimize objective function (min or max)
 - lots of freedom to choose; no principle behind choice, just "common sense" → hand craft OF for data set. Problem: Results of the analysis depend on OF.

Algorithm

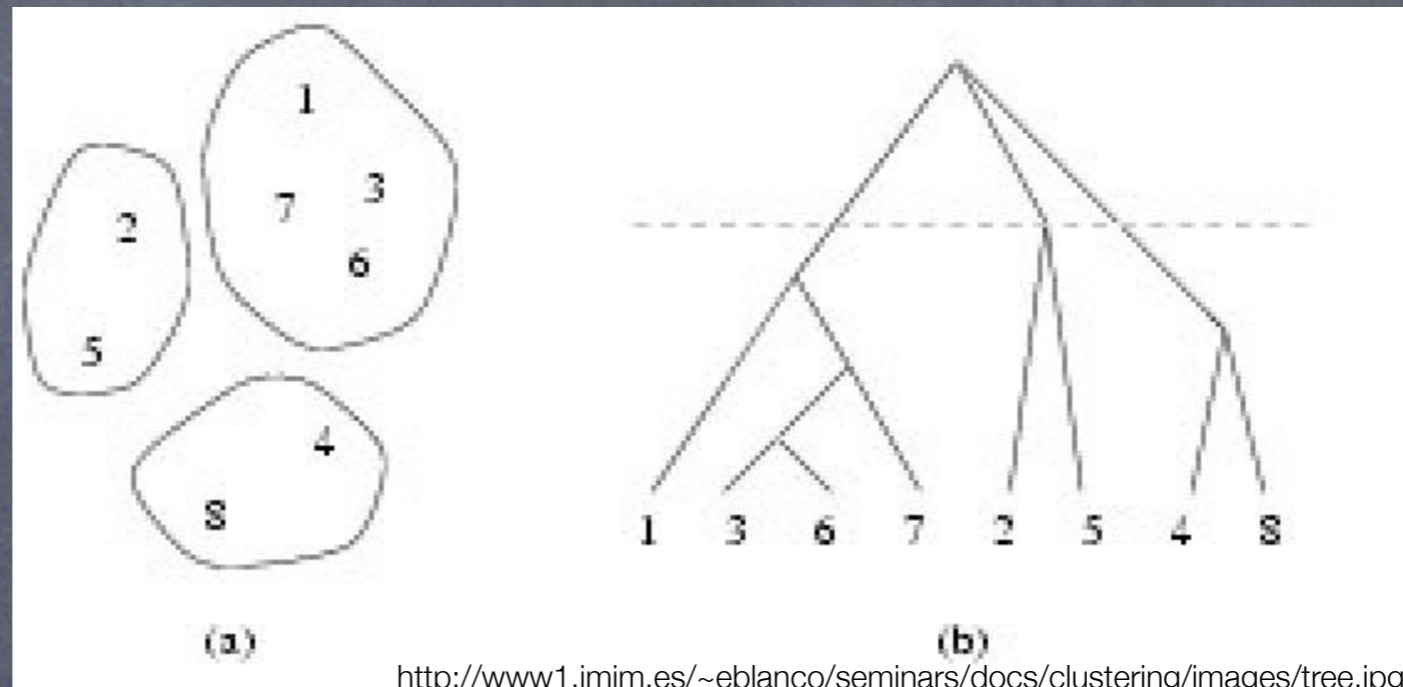
- Iterative reallocation
- Other option: hierarchical clustering algorithms
- Problem with K-means: Local minima

- A different initialization leads to a sub-optimal solution.



Hierarchical Clustering

- Build a hierarchy of similarities (i.e. a rooted tree structure)

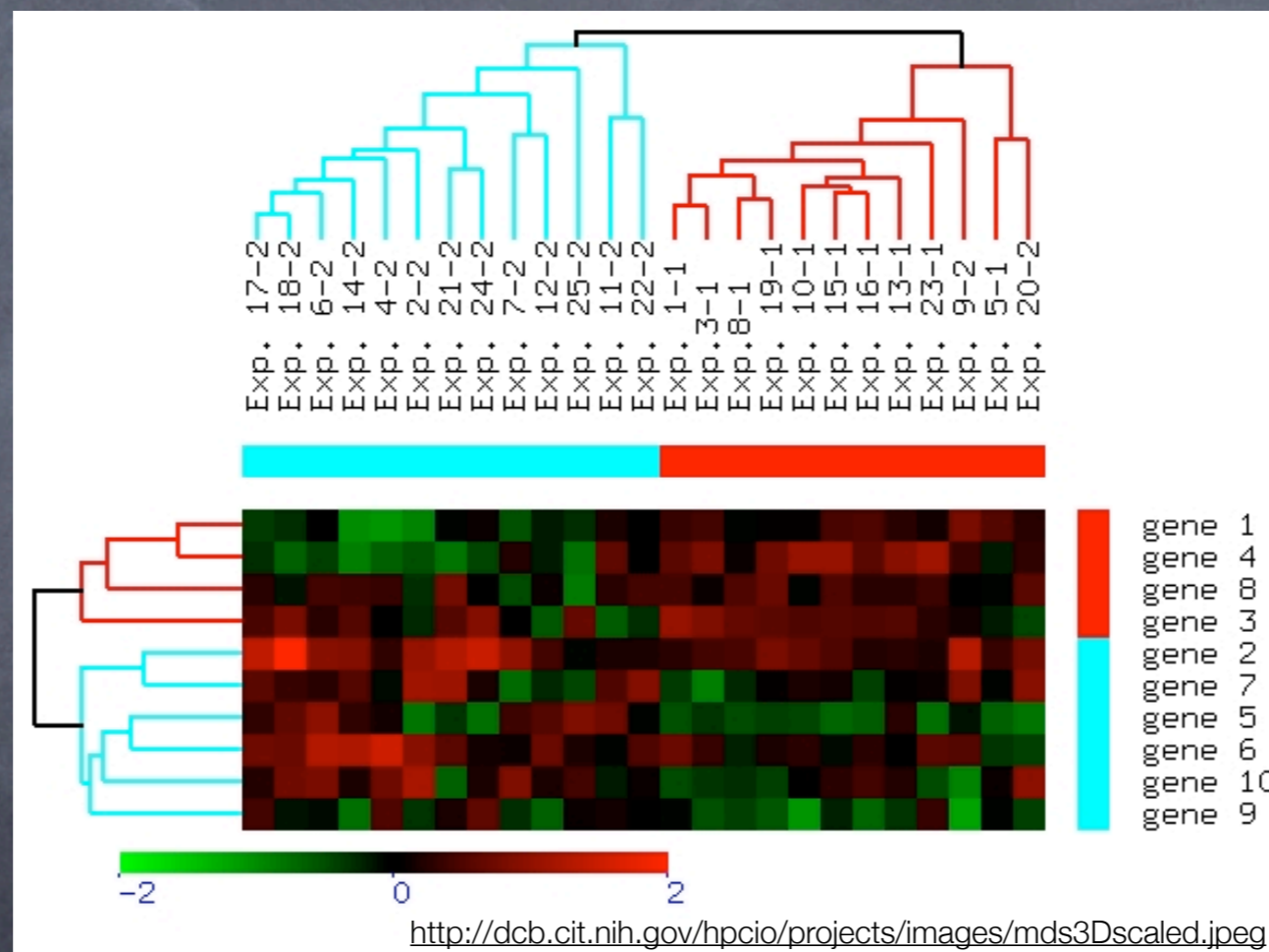


- Algorithms:

- Direct optimization, transform similarity into height on the tree (can be NP-hard).
- Agglomerative algorithms, greedy amalgamation.

<http://www1.imim.es/~eblanco/seminars/docs/clustering/images/tree.jpg>

- Hierarchical Clustering makes sense when we **expect a hierarchy in the data** (e.g. taxonomy)
- Careful when we can not assume a priori that a hierarchical structure is present in the data.



- Instead: use iterative reallocation algorithm for different numbers of classes, and observe if a hierarchy **emerges...**

Number of clusters

- Has to be set a priori in K-means
- The solution (optimal partition) depends on K!
- In general, we don't know K.
- How do we determine K?
- We can always do "better" (in terms of minimizing average distances) with more K
- However: in the extreme $K = N$, we are not summarizing the data anymore.
- What level of detail is appropriate?
- **Model complexity control!**

Complexity Control

- Many attempts in the statistical literature to find a “goodness” criterion for the best K -cluster fit. Use this criterion to determine K .
- Examples:
- Compare within-cluster sum of squared distances if cluster is split or not (Duda, Hart 1973).
- GAP-statistic (Tibshirani et, 2000): compare change in within cluster dispersion to that expected under a uniform null distribution.
- Problem: As before, arbitrary ad hoc measures.

Can we overcome the arbitrariness?

- Problems:

- Similarity measure
- Clustering Criterion
- Complexity Control

- Different approaches:

- Statistical modeling (use bayesian estimation); makes assumptions about distributions
- Stability arguments
- Information theoretic approach

Mixture models

- Assume that the underlying probability is a mixture of K specified probability functions.
- Those functions may be parameterized
- Most often Gaussians are used. Parameters: Number of gaussians (K); means; and covariances
- Find the most likely parameters (Bayesian estimation)
- Nice: the number of clusters becomes a parameter \rightarrow no need to introduce extra statistic
- Problem: the mixture model may not fit the data well (or: did we choose the right hypothesis class?)

EM algorithm

- Expectation-maximization. Iterative algorithm:
- Initialize the parameters
 1. (E-step): Compute expectation values for the membership variables of each data point, given the current parameters
 2. (M-step): Recompute the parameters from the membership
- Repeat until convergence

Stability

- Idea: Robustness of the partition is important!
- Any reasonable data clustering has to be stable under sample fluctuations.
- See for example (on reading list) Buhmann et al. (2000) Ben David (2004, 2005), within an information theoretic context: Still and Bialek (2004).

Information theoretic clustering

- Clustering = lossy compression
- Make minimal assumptions and no arbitrary ad hoc definitions of various statistics.
- Have to specify what is relevant to the analysis (with respect to what do we want to compress)
- Everything else follows from this decision and information theoretic principles: Objective; similarity measure; algorithm.
- Complexity control via stability arguments (also principled)

Homework

- Implement and/or test K-means.
- Make artificial data, e.g. drawn from m Gaussians.
- Analyze the performance. Some measures: % correct (global optimum); # iterations till convergence.
- Play with changing the input data. Make the task more or less difficult. Example: increasing class overlap \rightarrow more difficult.