

Data analysis for gene expression, Fall 2004

Clustering and information visualization

Samuel Kaski

University of Helsinki
Department of Computer Science

<http://www.cs.helsinki.fi/>

Material

A.K. Jain, M.N. Murty and P.J. Flynn. Data Clustering: A Review. *ACM Computing Surveys*, 31(3):264–323, 1999. (A good review.)

V. Estivill-Castro. Why so many clustering algorithms—A position paper. *SIGKDD Explorations*, 4(1):65-75. (I do not agree with everything but describes many of the problems in defining clusters.)

These papers contain some of the case studies discussed in the lectures:

A. Bhattacharjee, W. G. Richards, and J. S. et al. Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. *PNAS*, 98:13790–13795, 2001.

T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.

+ the same old books

Contents and aims

- Introduction with the help of lung cancers (Bhattacharjee et al.)
- Philosophy about goals of clustering and definition of a cluster
- Some clustering algorithms
 - Aim is to understand the basics of a few basic types of methods, and their pros and cons
 - Many details must be skipped; can be found in the books
 - Focus is on metric multivariate data
- Distance measures
- Number of clusters
- Cluster validation

Q: Why clustering? A: Exploratory (descriptive) data analysis

Goal: To make sense of unknown, large data sets by “looking at the data” through

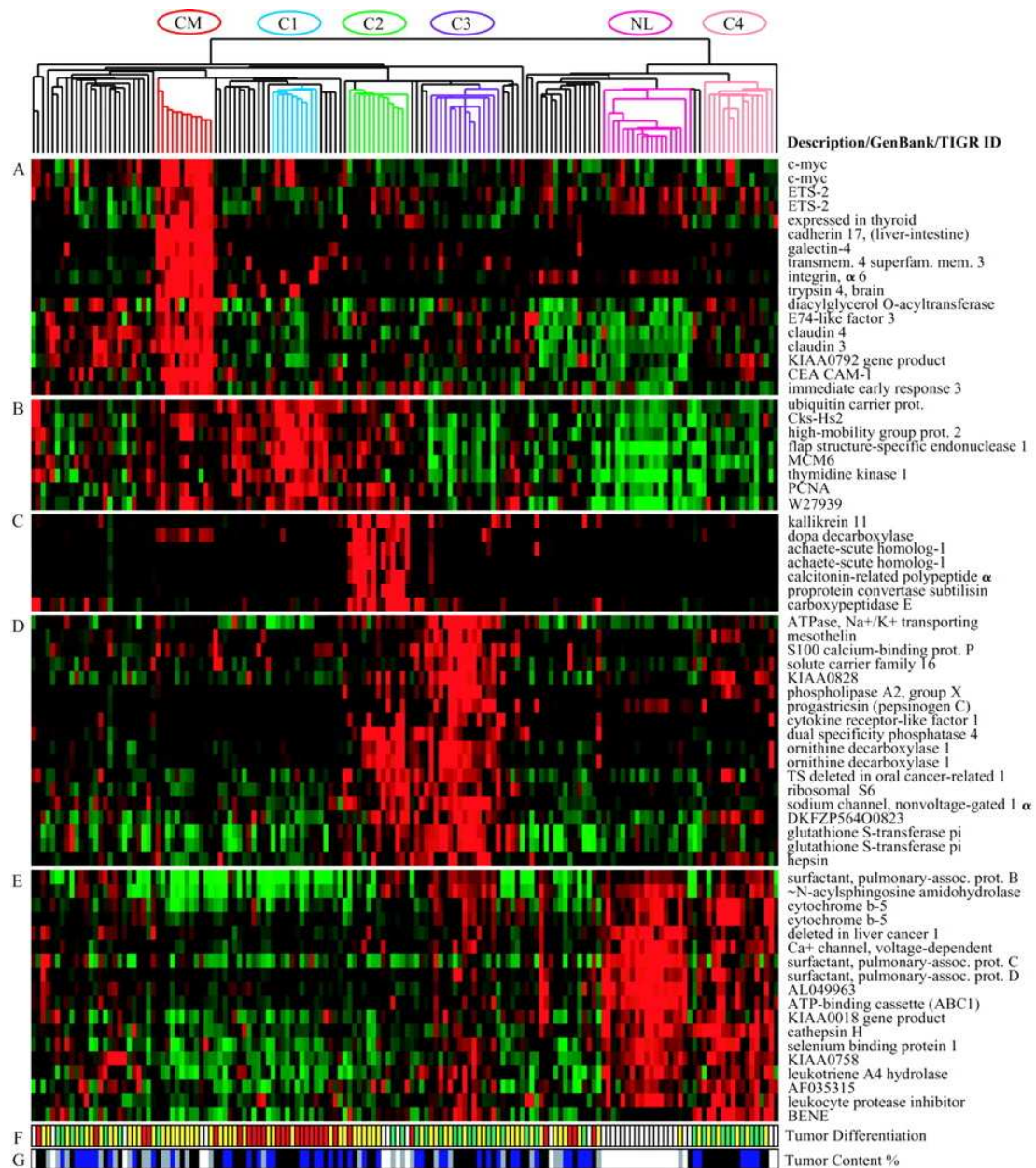
- statistical descriptions
- visualizations

Often additionally: Hunt for discoveries to **generate hypotheses** for further confirmatory analyses.

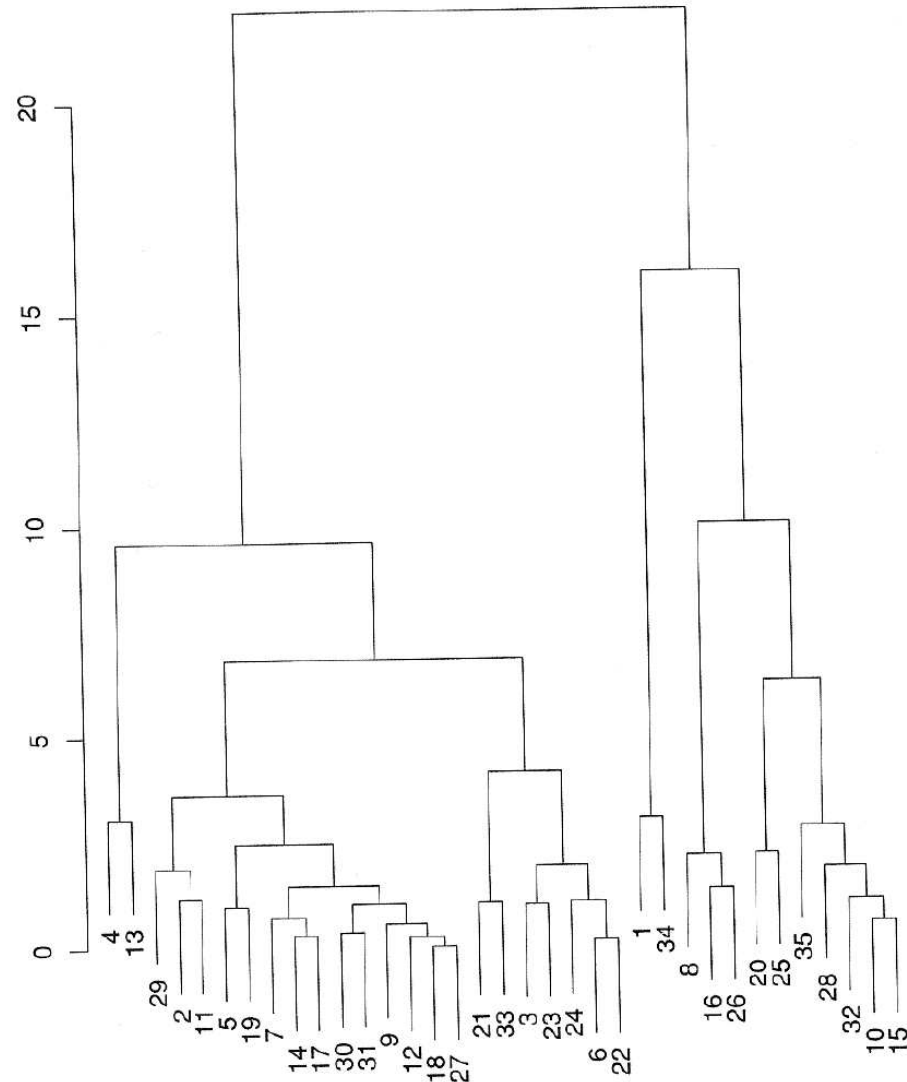
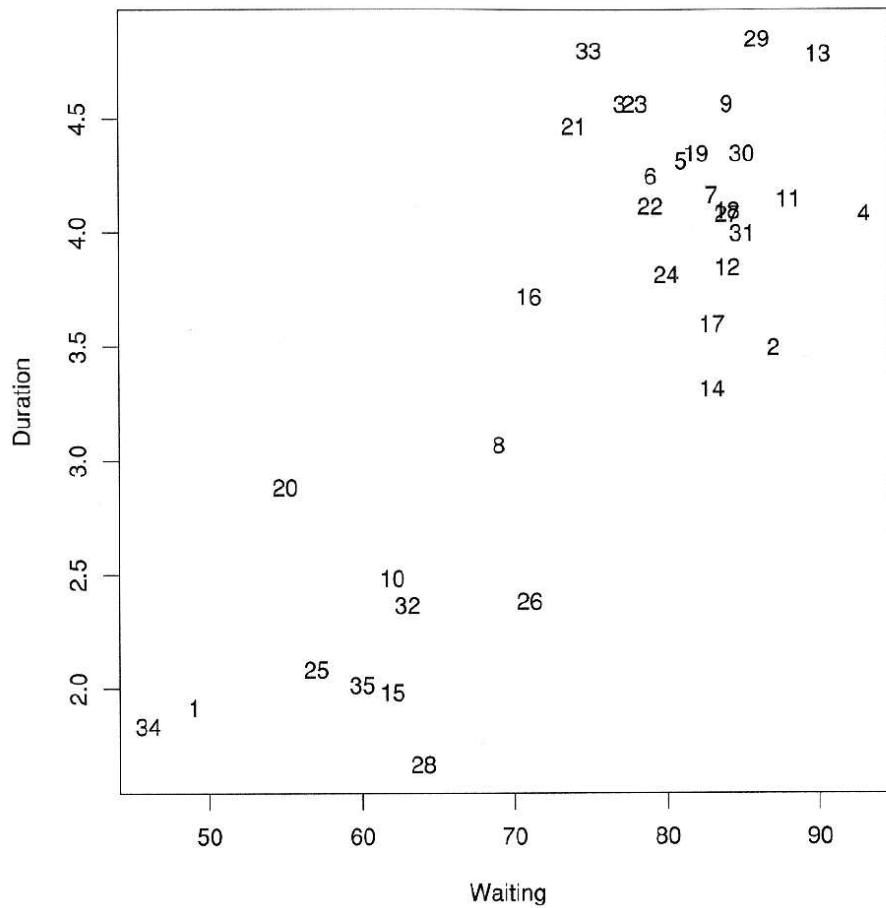
This means flexible model families with additional constraints set by the discovery task, computational and modeling resources, and interpretability.

Example: Hierarchical clustering of gene expression data

- Data: Expression (activity) of a set of genes measured by DNA chips in tissue samples
- The samples are adenocarcinomas from humans
- The goal is to find sets of mutually similar tissue samples. Maybe subcategories will be found that respond differentially to treatments.



How was the clustering carried out?



Variants

Agglomerative vs. divisive clustering

Different criteria for agglomeration and division:

- single linkage
- complete linkage
- average linkage
- Ward etc.

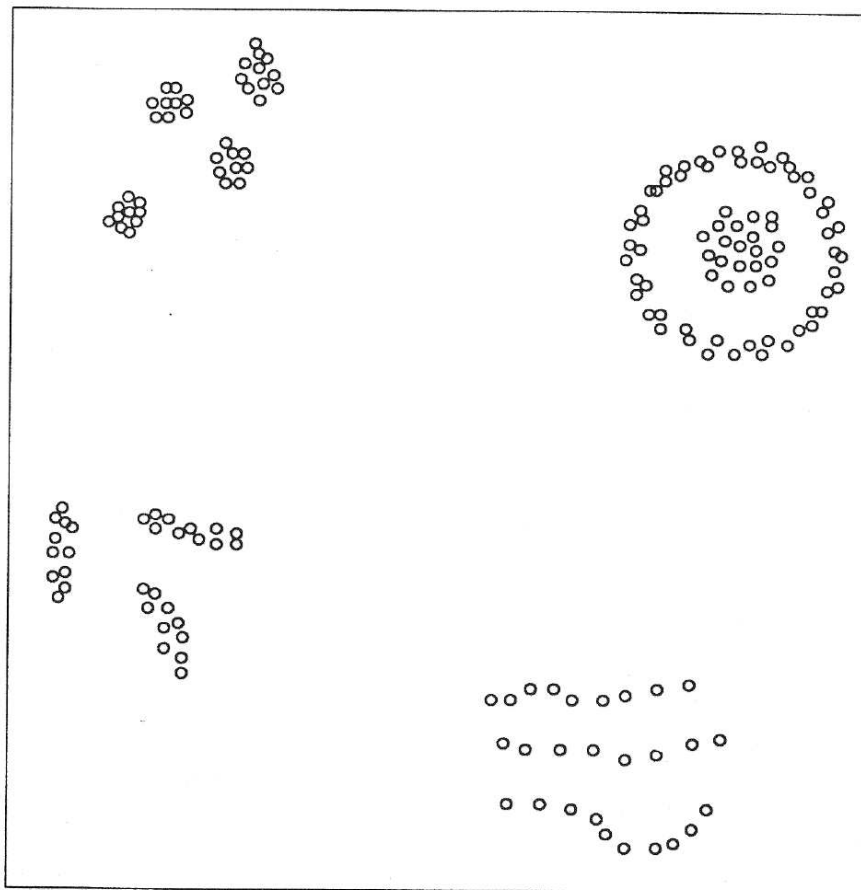
Pros and cons of hierarchical clustering

- + The result is intuitive and easily interpretable.
- + The dendrogram can be used for both (i) displaying similarity relationships between clusters and (ii) partitioning by cutting at different heights.
- + Possibly tedious to interpret for large data sets
- Sensitivity to noise
- Clustering has been defined by an algorithm. Can the result be described as such? Is there a goodness criterion?

What is clustering (segmentation) really?

What is a cluster?

Which are clusters?



Goals of clustering

1. **Compression.** Because it is easy to define the cost function for compression, there is a natural goal and criterion for clustering as well:
As effective compression as possible.
2. **Discovery of “natural clusters” and description of the data.** There does not exist any single well-posed and generally accepted criterion.

Definition of a cluster

Typically either

1. A group of mutually similar samples, or
2. A mode of the distribution of the samples (more dense than the surroundings)

The definitions depend on the similarity measure or the metric of the data space.

Note:

Distinguish between the goal of clustering and the clustering algorithm.

The goal can be defined by

- a cost function to be optimized
- a (statistical) model
- characterizing somehow what a “good” cluster is like
- indirectly by introducing an algorithm

All are only partial solutions; so far nobody has proposed a globally satisfactory definition of a cluster!

A clustering algorithm describes how the clusters are found, given the goal.

Partitional clustering

Definition of a cluster:

Assume a distance measure $d(\mathbf{x}, \mathbf{y})$ and define a cluster based on it:

A cluster consists of a set of samples having small mutual distances, that is,

$$E_k = \sum_{w(\mathbf{x})=w(\mathbf{y})=k} d^2(\mathbf{x}, \mathbf{y})$$

is small. Here the cluster of sample \mathbf{x} has been indexed by $w(\mathbf{x})$.

Partitional clustering algorithm

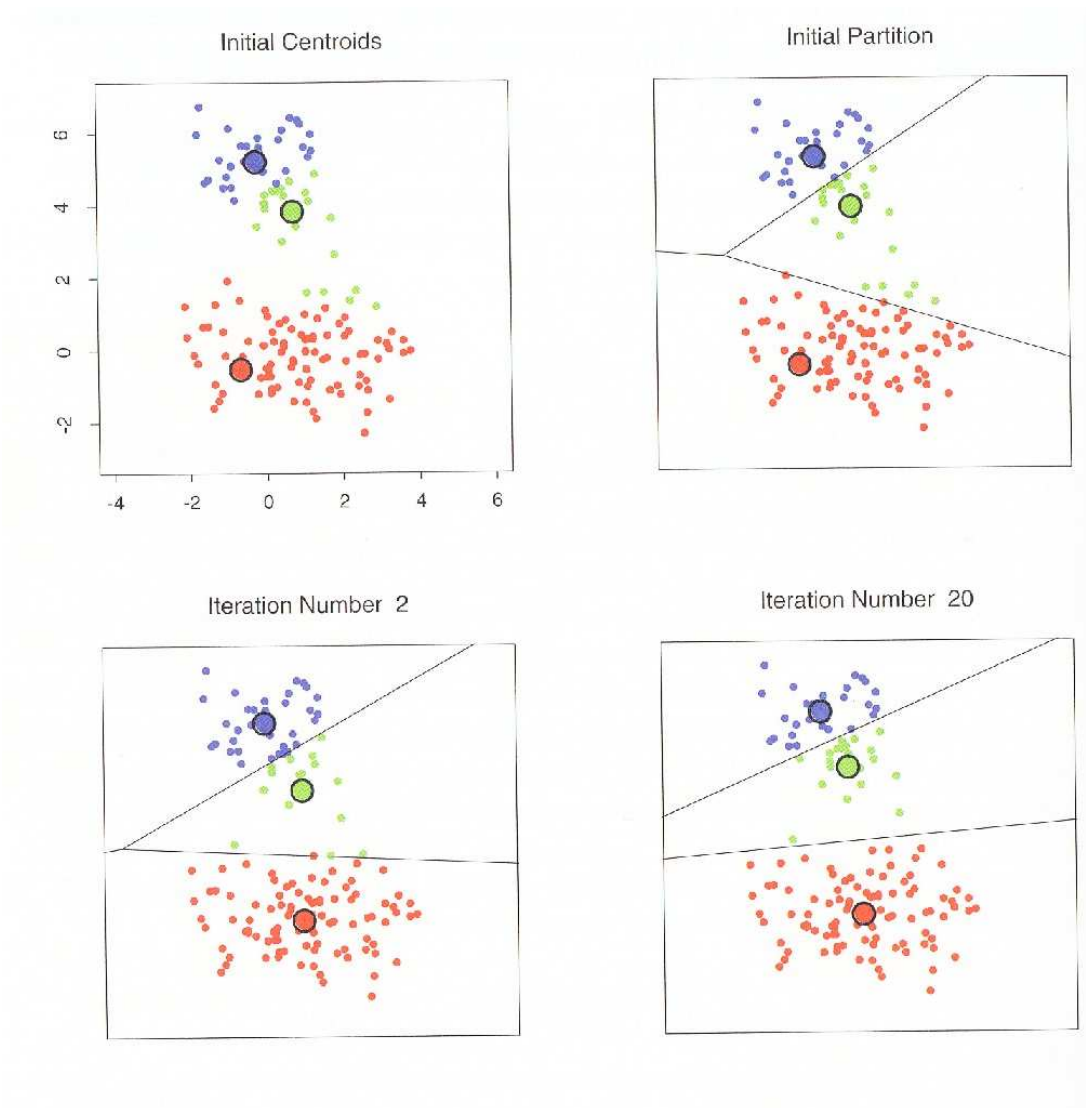
A partitional clustering algorithm tries to assign the samples to clusters such that mutual distances are small in *all clusters*.

In other words, the cost function

$$E = \sum_k E_k$$

is minimized.

In the **K-means algorithm** the distance measure is Euclidean, and the clusters are defined by a set of K *cluster prototypes*: Samples are assigned to the cluster with the closest prototype.



Pros and cons of partitional clustering

- + Fast (although not faster than hierarchical clustering)
- + The result is intuitive, although possibly tedious to interpret for large data sets
- The number of clusters K must be chosen, which may be difficult
- Tries to find “spherical” clusters in the sense of the given distance measure. (This may be the desired result, though.)

Model-based clustering: Mixture density model

Assume that each sample \mathbf{x} has been generated by one generator $k(\mathbf{x})$, but it is not known which one.

Assume that the generator k produces the probability distribution $p_k(\mathbf{x}; \theta_k)$, where θ_k contains the parameters of the density.

Assume further that the probability that generator k produces a sample is p_k .

The probability density generated by the mixture is

$$p(\mathbf{x}) = \sum_k p_k(\mathbf{x}; \theta_k) p_k$$

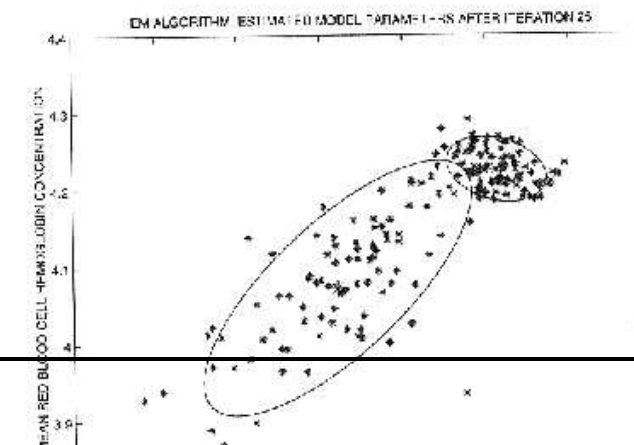
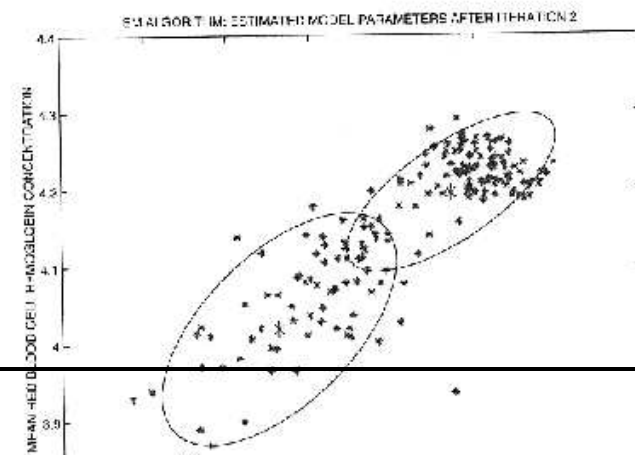
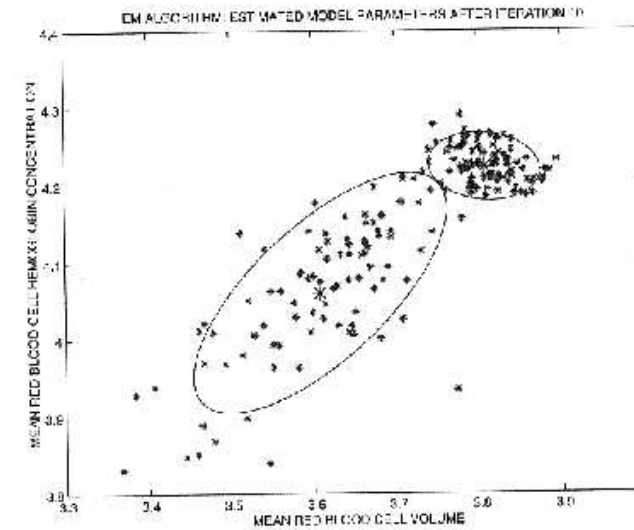
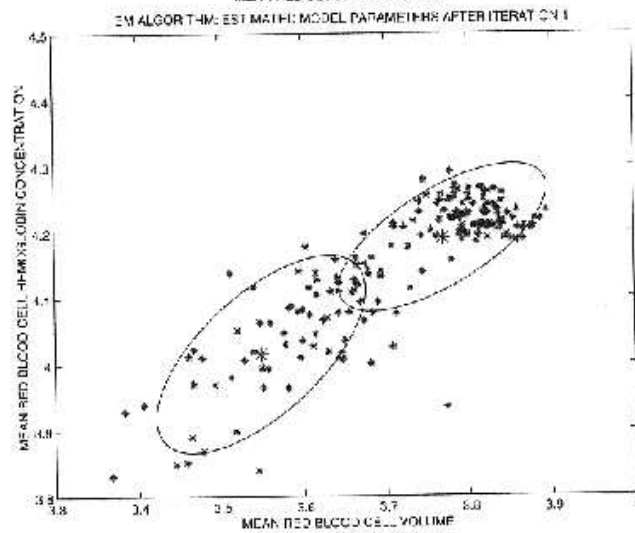
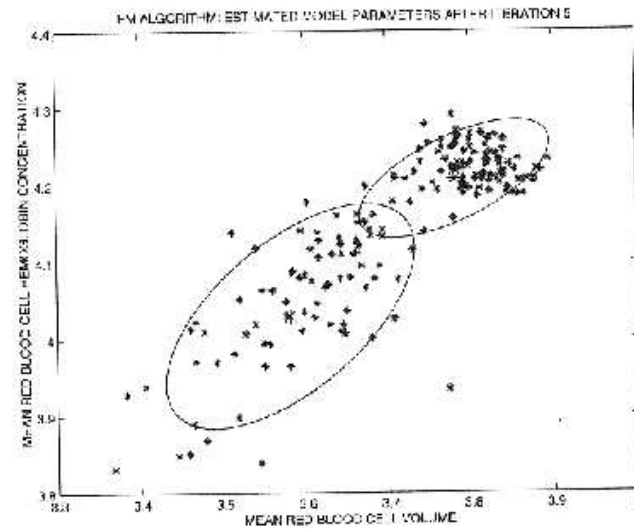
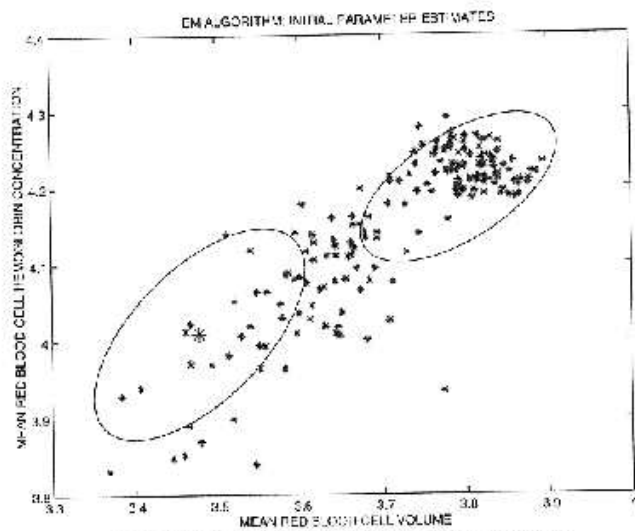
The model can be fitted to the data set with basic methods of statistical estimation:

- maximum likelihood
- maximum a posterior

Conveniently optimizable by EM-based algorithms.

Suitable model complexity (number of clusters) can be learned by Bayesian methods, approximated by BIC (or AIC, MDL, ...)

Note that K-means is obtained as the limit when generators of normal distributions sharpen.



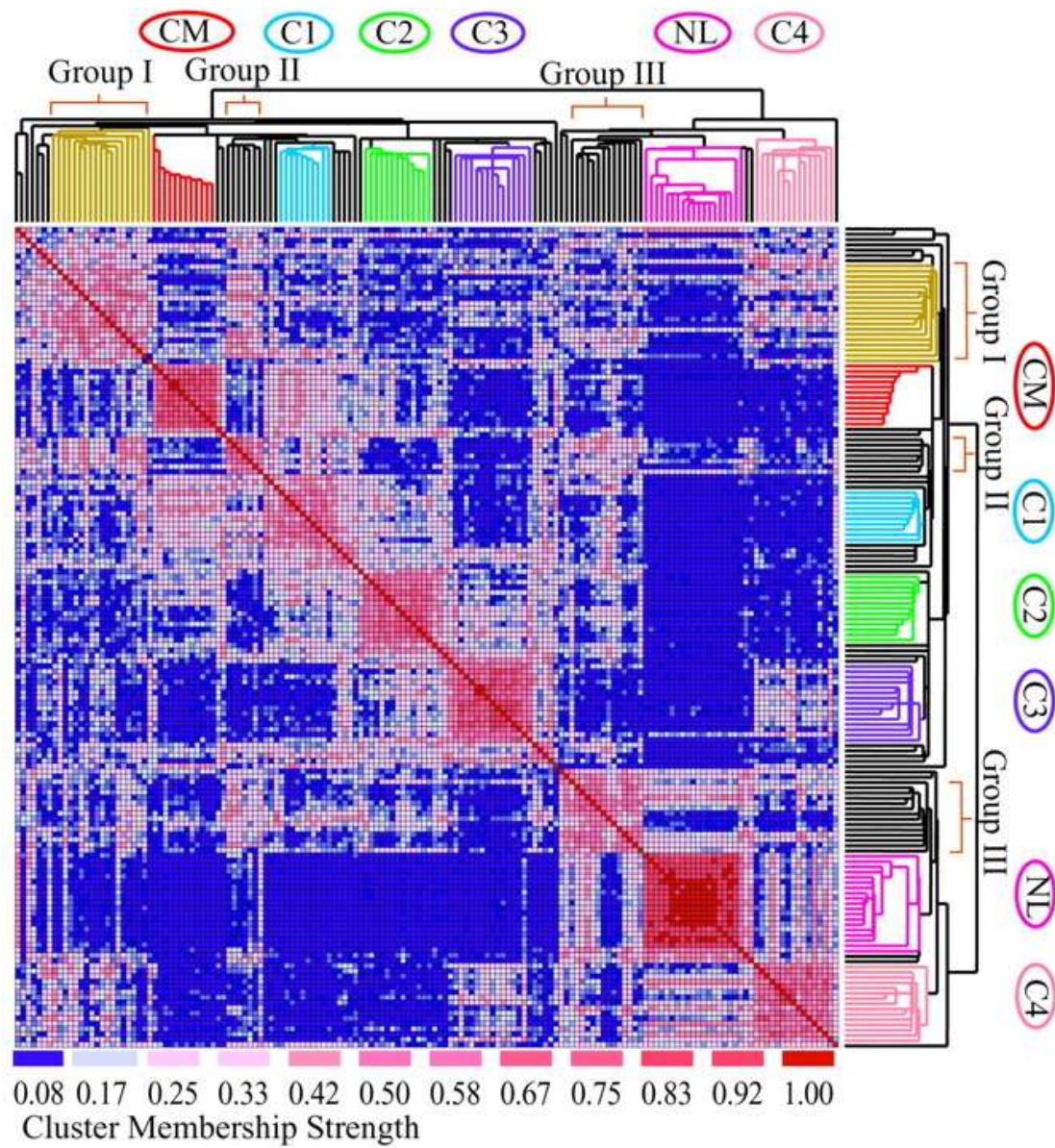
Pros and cons of clustering by mixture density models

- + The model is well-defined. It is based on explicit and clear assumptions on the uncertainty within the data
- + As a result, all tools of probabilistic inference are applicable:
 - + evaluation of the generalizability and quality of the result
 - + choosing the number of clusters
- Is the goal of clustering the same as the goal of density estimation? The probabilistic tools work properly only if the assumptions are correct!

Bhattacharjee et al: Similarity of samples from a mixture model

Quantize the robustness of the clustering results to random variations in the observed data:

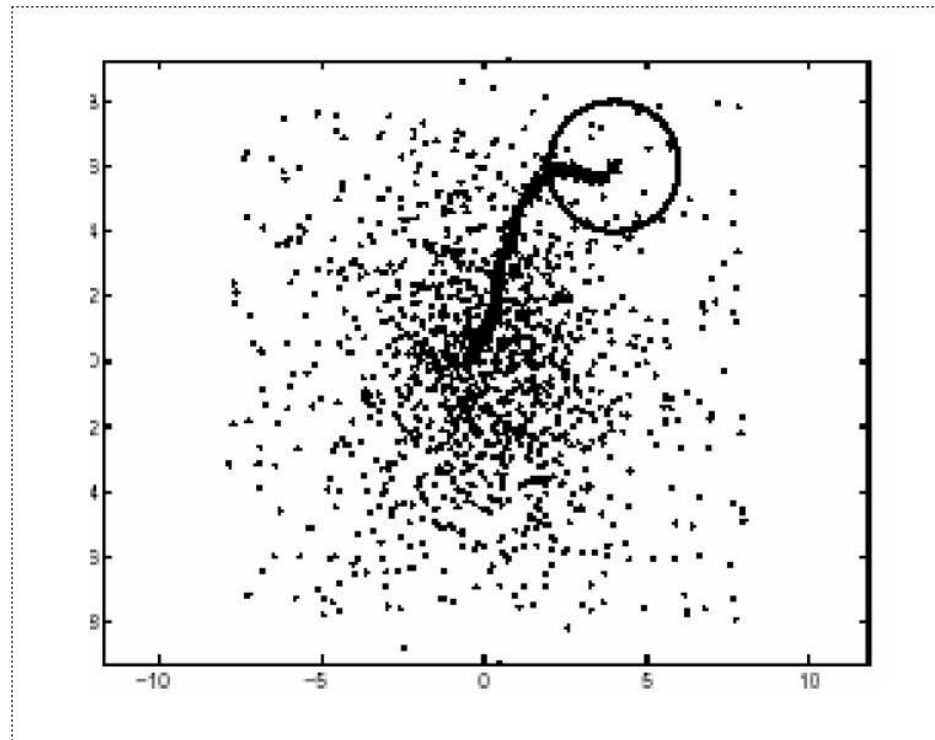
- Construct lots of (200) bootstrapped data sets by sampling with replacement from the original data
- Cluster each new set
- For each pair of samples (\mathbf{x}, \mathbf{y}) , compute the strength of association as the percentage of times they become clustered into the same cluster



Discussion

- Strengthens the faith to the hierarchical clustering
- Not a very illustrative visualization without the hierarchical clustering
- Would there exist a better clustering in the new similarity measure induced by the bootstrapping procedure?
- Is robustness to variation a good indication of clusteredness? The robust features may not be biologically interesting? (\Rightarrow external criteria might be better)

Mode seeking



Distance measures

Absolute magnitudes / Zero level	Reliable	Unreliable
Interesting	Euclidean metric	(Euclidean with mean subtracted)
Not interesting	Inner product	Correlation

According to some studies (including ours) the correlation may be best.

About metrics

Euclidean metric:

$$d_E^2(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^2 = (\mathbf{x} - \mathbf{y})^T \mathbf{I}(\mathbf{x} - \mathbf{y})$$

Becomes (essentially) inner products for normalized vectors,

$$\|\mathbf{x}\| = \|\mathbf{y}\| = 1:$$

$$d_E^2(\mathbf{x}, \mathbf{y}) = \|\mathbf{x}\|^2 + \|\mathbf{y}\|^2 - 2\mathbf{x}^T \mathbf{y} = 2(1 - \mathbf{x}^T \mathbf{y})$$

Correlation (with vector components interpreted as samples of the same random variable, and σ_x being standard deviation of \mathbf{x})

$$\rho(\mathbf{x}, \mathbf{y}) = \frac{(\mathbf{x} - \bar{\mathbf{x}})^T (\mathbf{y} - \bar{\mathbf{y}})}{\sigma_x \sigma_y}$$

becomes inner products by Z-score normalization, $z = (\mathbf{x} - \bar{\mathbf{x}})/\sigma_x$.



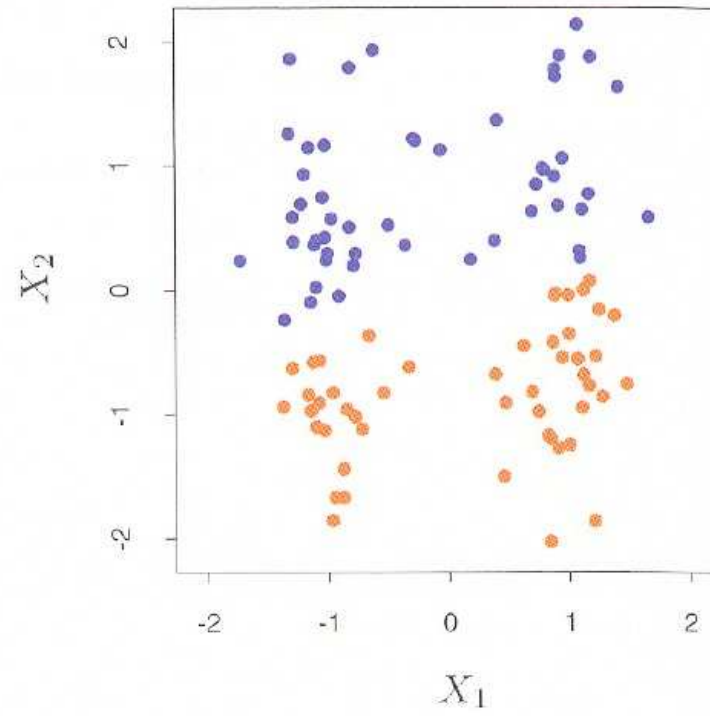
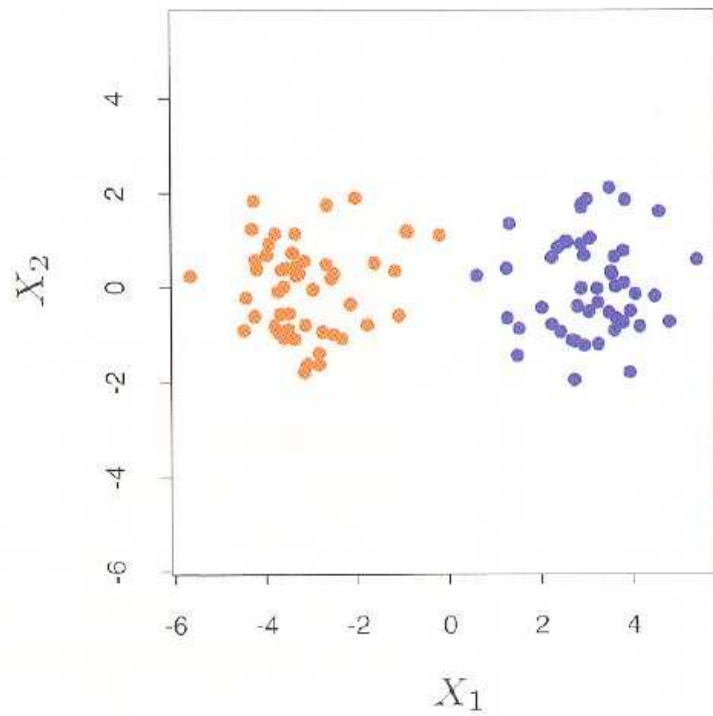
Global metric for $\mathbf{A} = \mathbf{S}^T \mathbf{S}$ is

$$d_{\mathbf{A}}^2(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^T \mathbf{A}(\mathbf{x} - \mathbf{y}) = \|\mathbf{S}\mathbf{x} - \mathbf{S}\mathbf{y}\|^2$$

Local (Riemannian) metric for $\mathbf{y} = \mathbf{x} + d\mathbf{x}$ is

$$d_{\mathbf{A}(\mathbf{x})}^2(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^T \mathbf{A}(\mathbf{x})(\mathbf{x} - \mathbf{y})$$

Clusteredness depends on scaling



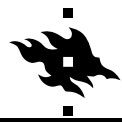
GIGO Principle

Supervised learning:

Garbage in \Rightarrow weaker results out

Unsupervised learning:

Garbage in \Rightarrow garbage out



(Successful) unsupervised learning is always implicitly supervised

by

- feature extraction
- variable selection
- model selection

Number of clusters?

In principle: Use the normal model complexity selection methods.

Lots of more or less heuristic solutions exist.

One possible solution: Visualization

Cluster validation

(Selecting the number of clusters is a sub-problem of this.)

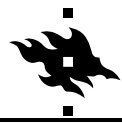
Since the data exploration process necessarily is partly subjective, the results must be validated: Are the clusters/other findings real?

Fundamentally boils down to generalizability to new data (which can be assessed by measuring more data!)

Bayesian averaging over models is hard because of

- label switching
- the end result will be discovery or “understanding of data.” Since we do not know how humans do that, it is hard to assign proper priors (=choose model families) for the analysis.

A temporary solution is to use cross-validation or bootstrap.



Note that most if not all of so-called “internal validation indices”, criteria computed from data itself, may overfit the data as well. Why not optimize those criteria if they are good?

External validation:

- Does a method find known clusters? Problem: Depends on the type of known clusters...
- Does the clustering correspond to known classes? Problem 1: Classes need not be clusters. Problem 2: If this is optimal, why not use supervised clustering?

Conclusions

Ill-defined problem with lost of proposed solutions.

Words of advice:

- The reason is that there actually are lots of different clustering tasks with different goals and not enough prior knowledge to define the problem exactly.
- This does not imply that sloppy application of clustering methods would be acceptable!
- In contrast, you have to understand the principles and key ideas, in order to use **your prior knowledge** to choose suitable methods to **your specific task**.
- Check the validity of the results somehow.