Clustering

Tim Beißbarth

Deutsches Krebsforschungszentrum

Molekulare Genomanalyse

Slide mainly from

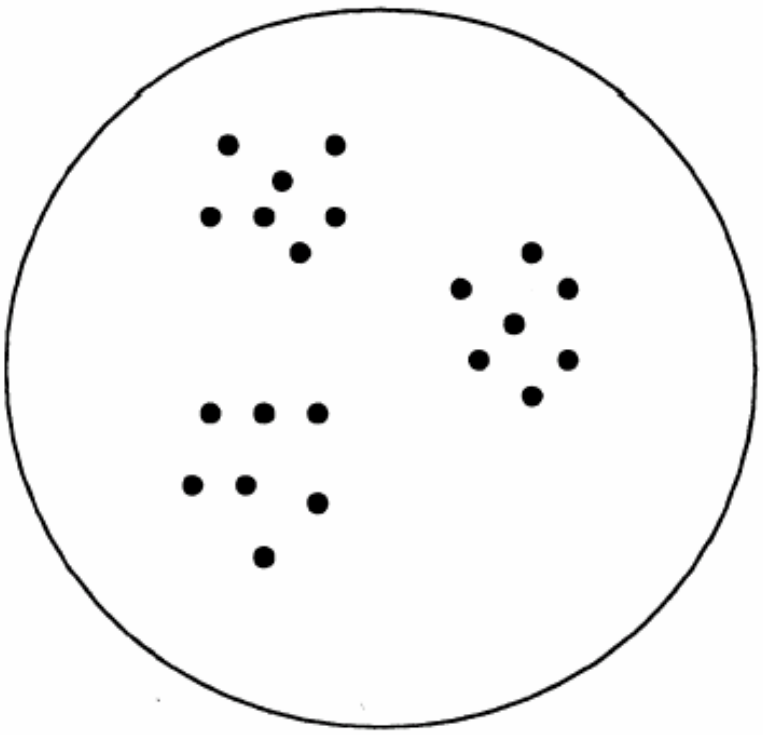
Prof. Dr. Jörg Rahnenführer

Universität Dortmund, Fachbereich Statistik

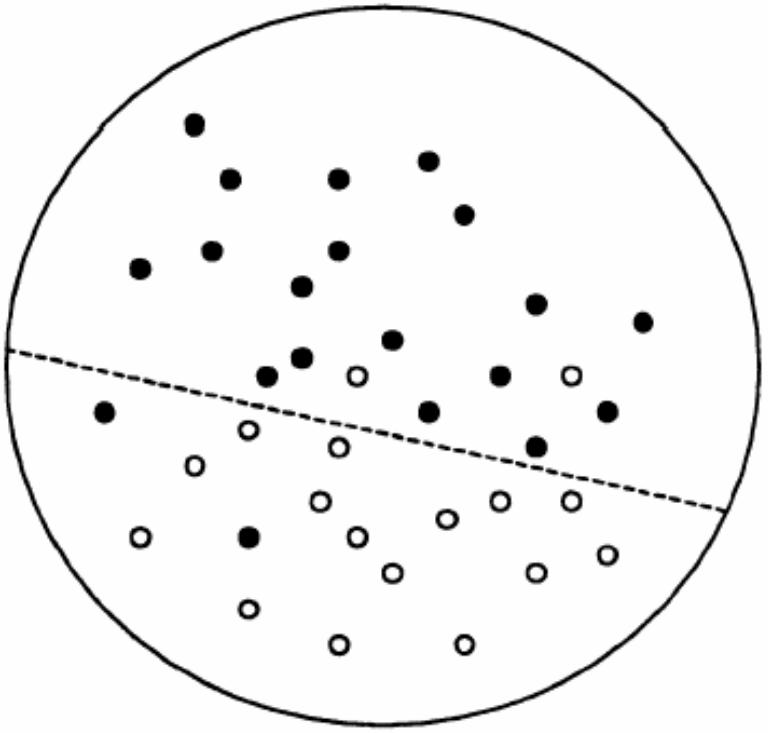
Overview

- Classification tasks for microarrays
- Cluster analysis
  - Time series example
  - Distance measures
  - Cluster algorithms
- Comparisons and recommendations
  - Estimating the number of clusters
  - Assessment of cluster validity
  - Comparative study for tumor classification
  - Gene selection

Prediction of labels: Supervised vs. Unsupervised



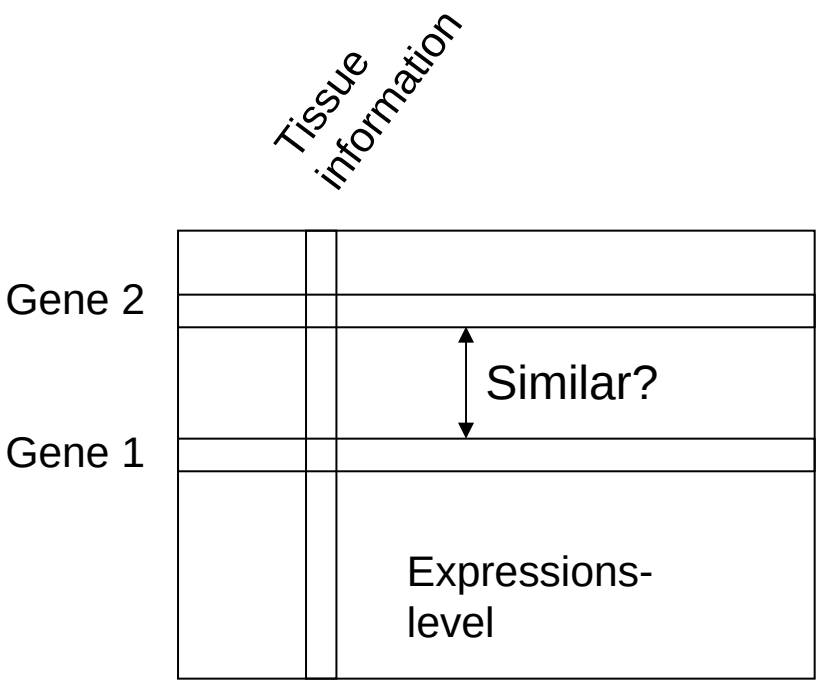
Unsupervised



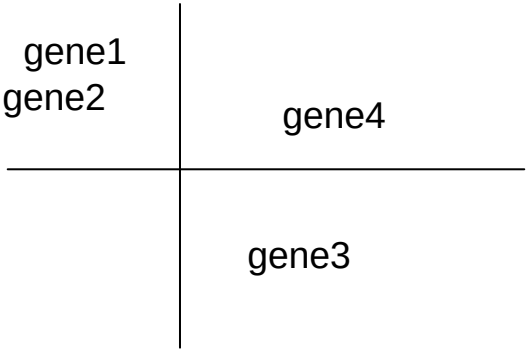
Supervised

Distances in Tables of Expression-data

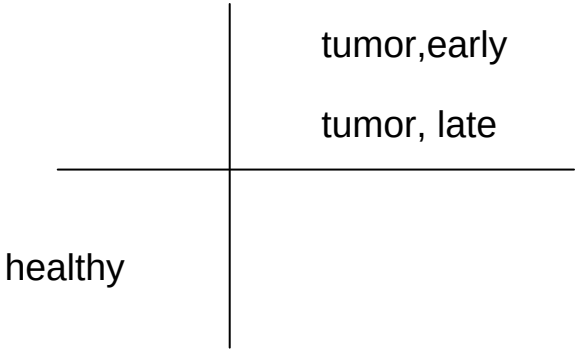
Tabelle von Expressionsleveln:



Clustering of Genes:



Clustering of Experiments:



Classification Tasks for Microarrays

- Classification of SAMPLES

  Generate gene expression profiles that can

  (i) discriminate between different known cell types or conditions, e.g. between tumor and normal tissue,

  (ii) identify different and previously unknown cell types or conditions, e.g. new subclasses of an existing class of tumors.

- Classification of GENES

  (i) Assign an unknown cDNA sequence to one of a set of known gene classes.

  (ii) Partition a set of genes into new (unknown) functional classes on the basis of their expression patterns across a number of samples.

Cancer classification	Class discovery	Class prediction
Machine learning	Unsupervised learning	Supervised learning
Statistics	Cluster analysis	Discriminant analysis

Classification Tasks for Microarrays

- Difference between discriminant analysis (supervised learning) and cluster analysis (unsupervised learning) is important:
- If the class labels are known, many different supervised learning methods are available. They can be used for prediction of the outcome of future objects.
- If the class labels are unknown, then unsupervised learning methods have to be used. For those, it is ***difficult to ascertain the validity of inferences*** drawn from the output.

Classification

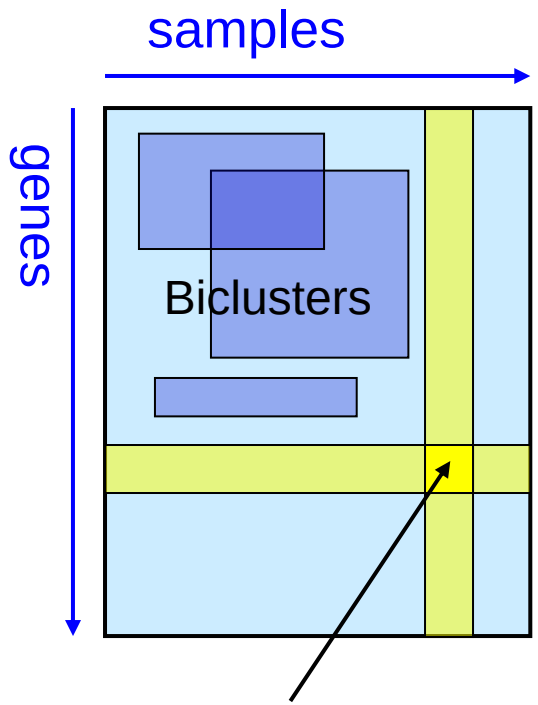
MESSAGE 1

Discriminant analysis: CLASSES KNOWN

Cluster analysis: CLASSES NOT KNOWN

Cluster Analysis

The gene expression matrix



$L_{i,j}$: expression level of gene **i** in sample **j**

- Clustering columns

  Grouping similar samples

- Clustering rows

  Grouping genes with similar trajectories across samples

- Bi-Clustering

  Grouping genes that have similar partial trajectories in a subset of the samples

  - Tanay A, Sharan R, and Shamir R (2002): **Discovering Statistically Significant Biclusters in Gene Expression Data.** *Bioinformatics* 18, Suppl.1, 136-144.
  - Genes and samples both represented as nodes of a bipartite graph and connected with weights according to expression of the respective gene and sample.
  - Then the heaviest subgraph is determined with an algorithm that runs in polynomial time.

Cluster Analysis – Distance Measures

- Goal in cluster analysis

  Grouping a collection of objects into subsets or “clusters”, such that those within each cluster are more closely related to one another than objects assigned to different clusters.

- Distance measure

  A notion of distance or similarity of two objects: When are two objects close to each other?

- Cluster algorithm

  A procedure to minimize distances of objects within groups and/or maximize distances between groups.

- Euclidean distance:

$$d(x,y) = \sqrt{\sum \left(x_i - y_i\right)^2}$$

- Manhattan distance:

$$d(x,y) = \sum \left|x_i - y_i\right|$$

- Correlation distance:

$$d(x,y) = 1 - \frac{\sum \left(x_i - \overline{x}\right)\left(y_i - \overline{y}\right)}{\sqrt{\sum \left(x_i - \overline{x}\right)^2 \sum \left(y_i - \overline{y}\right)^2}}$$

Time series example

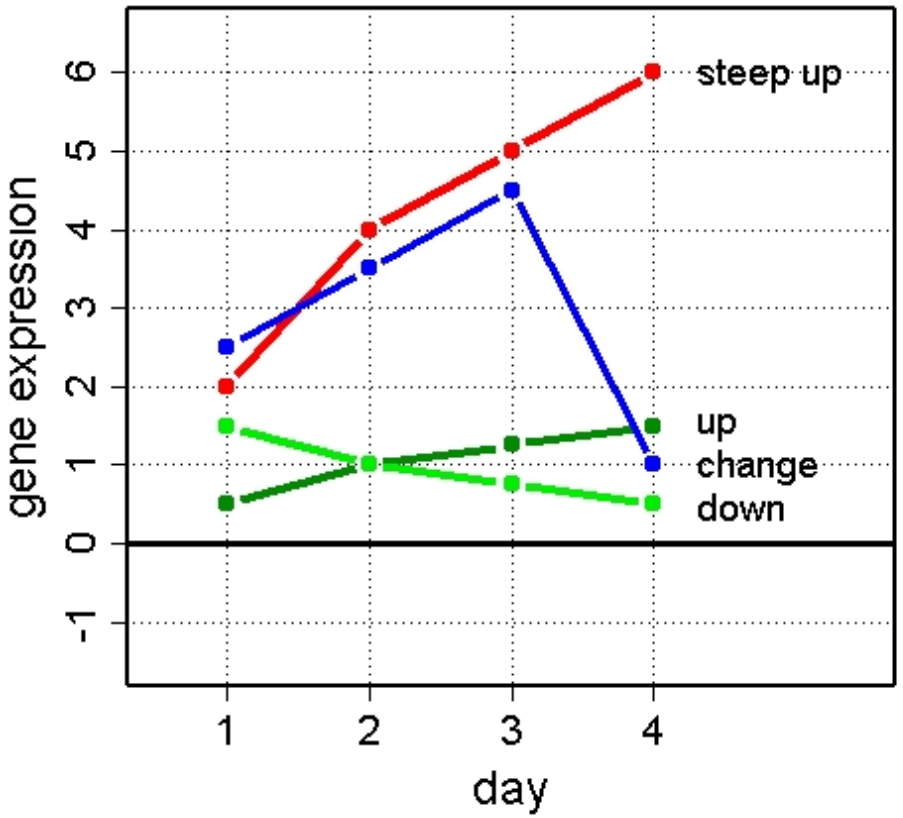
Biology

Measurements of gene expression on 4 (consecutive) days.

Statistics

Every gene is coded by a vector of length 4.

- steep up: $x_1 = (2, 4, 5, 6)$
- up: $x_2 = (2/4, 4/4, 5/4, 6/4)$
- down: $x_3 = (6/4, 4/4, 3/4, 2/4)$
- change: $x_4 = (2.5, 3.5, 4.5, 1)$



Distance Measures - Time Series Example

Euclidean distance

The distance between two vectors is the square root of the sum of the squared differences over all coordinates.

$$d_E(x_1, x_2) = \sqrt{(2-2/4)^2 + (4-4/4)^2 + (5-5/4)^2 + (6-6/4)^2} = 3\sqrt{3/4} \approx 2.598$$

- steep up: $x_1 = (2, 4, 5, 6)$
- up: $x_2 = (2/4, 4/4, 5/4, 6/4)$
- down: $x_3 = (6/4, 4/4, 3/4, 2/4)$
- change: $x_4 = (2.5, 3.5, 4.5, 1)$

0	**2.60**	**2.75**	**2.25**
2.60	**0**	**1.23**	**2.14**
2.75	**1.23**	**0**	**2.15**
2.25	**2.14**	**2.15**	**0**

Matrix of pairwise distances

Distance Measures - Time Series Example

Manhattan distance

The distance between two vectors is sum of the absolute (unsquared) differences over all coordinates.

$$d_M(x_1, x_2) = |2-2/4| + |4-4/4| + |5-5/4| + |6-6/4| = 51/4 = 12.75$$

- steep up: $x_1 = (2, 4, 5, 6)$
- up: $x_2 = (2/4, 4/4, 5/4, 6/4)$
- down: $x_3 = (6/4, 4/4, 3/4, 2/4)$
- change: $x_4 = (2.5, 3.5, 4.5, 1)$

0	**12.75**	**13.25**	**6.50**
12.75	**0**	**2.50**	**8.25**
13.25	**2.50**	**0**	**7.75**
6.50	**8.25**	**7.75**	**0**

Matrix of pairwise distances

Distance Measures - Time Series Example

Correlation distance

Distance between two vectors is 1-ρ (or 1-|ρ|), where ρ is the Pearson correlation of the two vectors.

$$d_C(x_1, x_2) = 1 - \frac{(2-\frac{17}{4})(\frac{2}{4}-\frac{17}{16})+(4-\frac{17}{4})(\frac{4}{4}-\frac{17}{16})+(5-\frac{17}{4})(\frac{5}{4}-\frac{17}{16})+(6-\frac{17}{4})(\frac{6}{4}-\frac{17}{16})}{\sqrt{(2-\frac{17}{4})^2+(4-\frac{17}{4})^2+(5-\frac{17}{4})^2+(6-\frac{17}{4})^2}\sqrt{(\frac{2}{4}-\frac{17}{16})^2+(\frac{4}{4}-\frac{17}{16})^2+(\frac{5}{4}-\frac{17}{16})^2+(\frac{6}{4}-\frac{17}{16})^2}}$$

- steep up: $x_1 = (2, 4, 5, 6)$
- up: $x_2 = (2/4, 4/4, 5/4, 6/4)$
- down: $x_3 = (6/4, 4/4, 3/4, 2/4)$
- change: $x_4 = (2.5, 3.5, 4.5, 1)$

0	**0**	**2**	**1.18**
0	**0**	**2**	**1.18**
2	**2**	**0**	**0.82**
1.18	**1.18**	**0.82**	**0**

Matrix of pairwise distances

Distance Measures - Time Series Example

Summary

- Euclidean distance measures average difference across coordinates.
- Manhattan distance measures average difference across coordinates, in a robust way.
- Correlation distance measures difference with respect to trends.

Standardization

- Data points (e.g. genes) are normalized with respect to mean and variance:

  Apply transformation $x \mapsto \frac{x-\hat{\mu}}{\hat{\sigma}}$, where $\hat{\mu}$ is an estimator of the mean (usually average across coordinates) and $\hat{\sigma}$ is an estimator of the variation (usually empirical standard deviation).

- After standardization, Euclidean distance and Correlation distance are equivalent(!): $d_E(x_1, x_2)^2 = 2nd_C(x_1, x_2)$

- Standardization makes sense, if one is not interested in the magnitude of the effects, but in the effect itself. Results can be misleading for noisy data.

Distance measures

MESSAGE 2

Appropriate choice of distance measure depends on your intention!

Cluster Algorithms

- Types of clustering algorithms:
  Combinatorial algorithms, mixture modeling and mode seeking
- Popular algorithms for clustering microarray data:
  - Hierarchical clustering
  - K-means
  - PAM (Partitioning around medoids)
  - SOMs (Self-Organizing Maps)
- K-means and SOMs take original data directly as input:
  Attributes are assumed to live in Euclidean space.
- Hierarchical cluster algorithms and PAM allow the choice of a dissimilarity matrix d, that assigns to each pair of objects x_i and x_j a value $d(x_i,x_j)$ as their distance.

Hierarchical Clustering

- Hierarchical clustering was the first algorithm used in microarray research to cluster genes (Eisen et al. (1998)).

1. First, each object is assigned to its own cluster.
2. Iteratively:
   - the two most similar clusters are joined, representing a new node of the clustering tree. The node is computed as average of all objects of the joined clusters,
   - the similarity matrix is updated with this new node replacing the two joined clusters.
3. Step 2 is repeated until only one single cluster remains.

Hierarchical Clustering

- Calculation of distance *d(**G**,**H**)* between clusters ***G*** and ***H*** is based on object dissimilarity between the objects from the two clusters:
  - Single linkage uses the smallest distance: $d_S(G,H) = \min_{i \in G, j \in H} d_{ij}$
  - Complete linkage uses the largest distance: $d_C(G,H) = \max_{i \in G, j \in H} d_{ij}$
  - Average linkage uses the average distance: $d_A(G,H) = \frac{1}{N_G N_H} \sum_{i \in G} \sum_{j \in H} d_{ij}$
- Alternative to agglomerative clustering: Divisive clustering: Iteratively, best possible splits are calculated.

Hierarchical Clustering

- Visualization of hierarchical clustering with dendrogram:
  - Clusters that are joined are combined by a line.
  - Height of line is average distance between clusters.
  - Cluster with smaller variation typically plotted on left side.

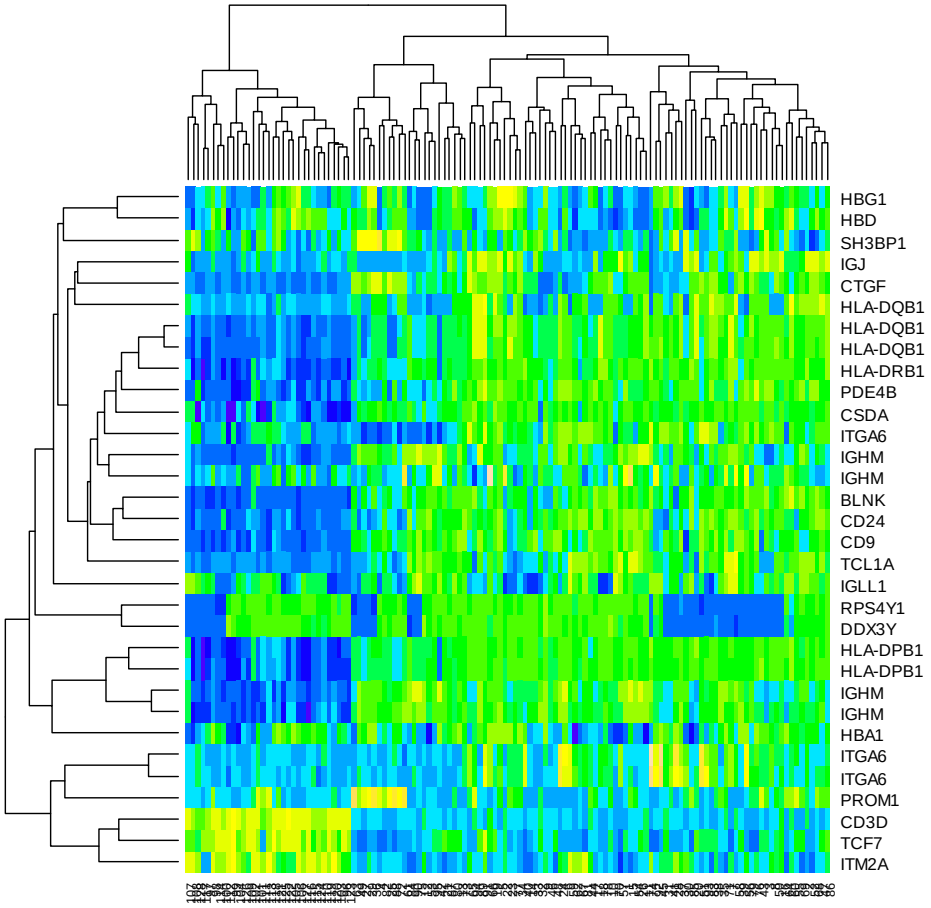
- Procedure provides a hierarchy of clusterings, with the number of clusters ranging from 1 to the number of objects.

- **BUT**:
  - Parameters for distance matrix: n(n-1)/2
  - Parameters for dendrogram: n-1.

  → Hierarchical clustering does not show the full picture!

Hierarchical Clustering

- Visualization with heat map and dendrogram
- Leukemia dataset:
  Chiaretti et al. (2004): Gene expression profile of adult T-cell acute lymphocytic leukemia identities distinct subsets of patients with different response to therapy and survival. Blood 103(7):2771-8.

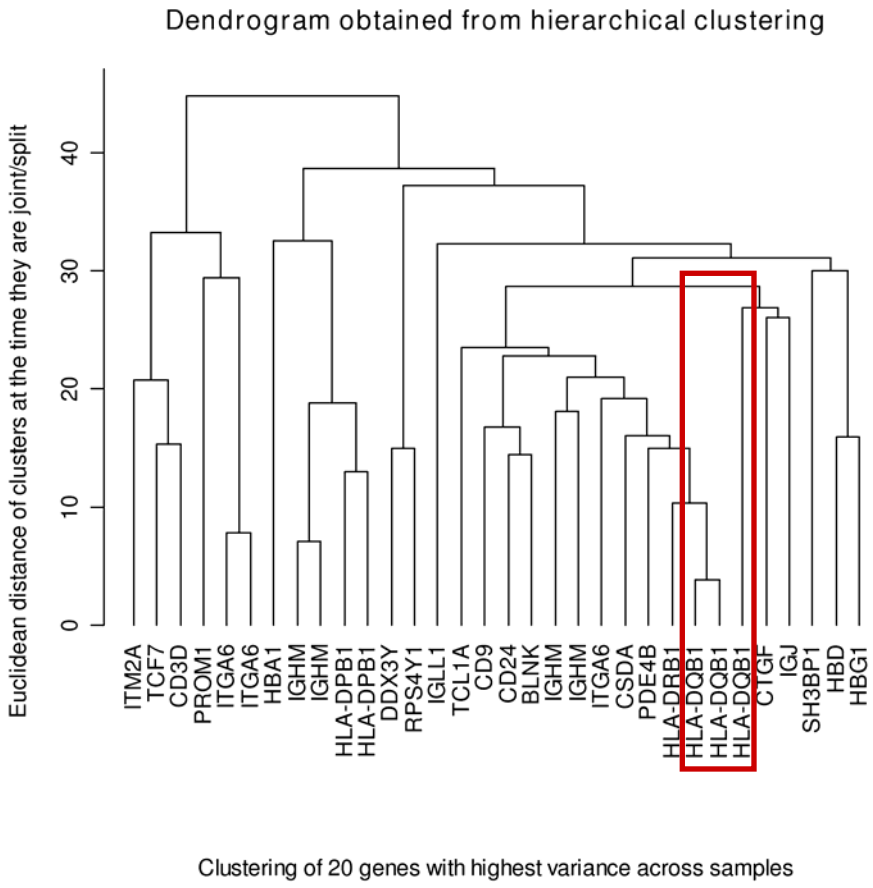


Hierarchical Clustering

- Visualization with heat map and dendrogram
- Leukemia dataset:
  Chiaretti et al. (2004): Gene expression profile of adult T-cell acute lymphocytic leukemia identities distinct subsets of patients with different response to therapy and survival. Blood 103(7):2771-8.
- Interest in specific genes:

  If you search for genes that are co-regulated with a specific gene of your choice, **DO SO!**

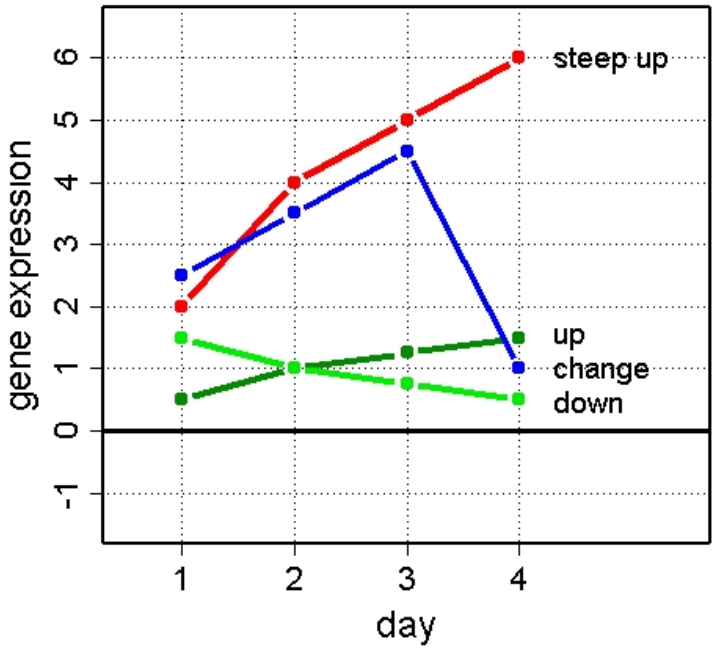
  Don't do clustering, but generate a list of genes close to your gene with respect to a distance of your choice.

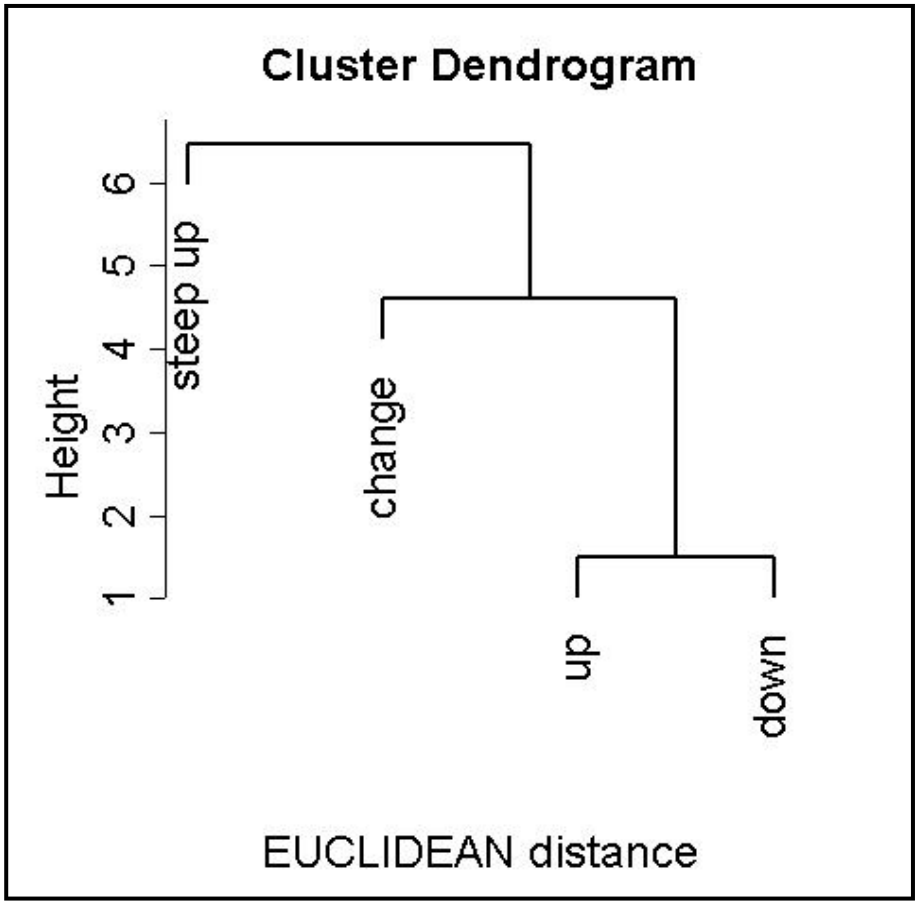


Time Series Example

- **Euclidean distance**

  Similar values are clustered together

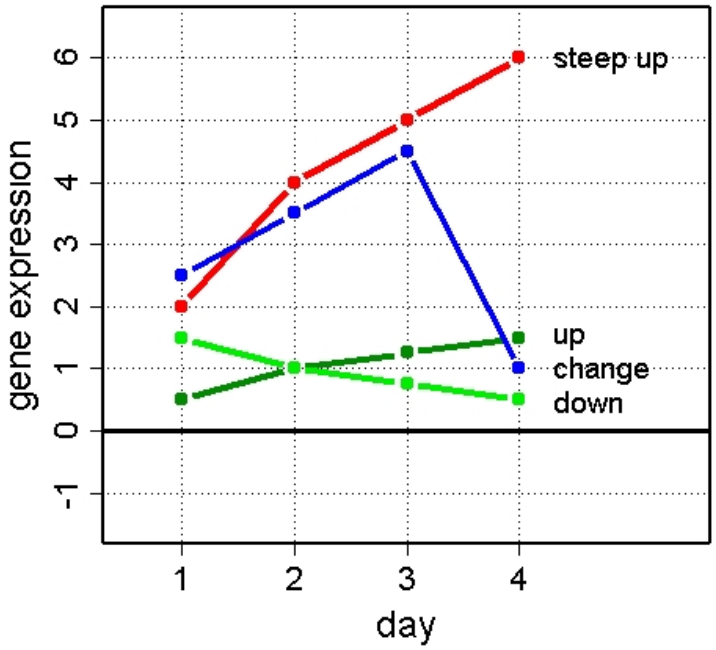


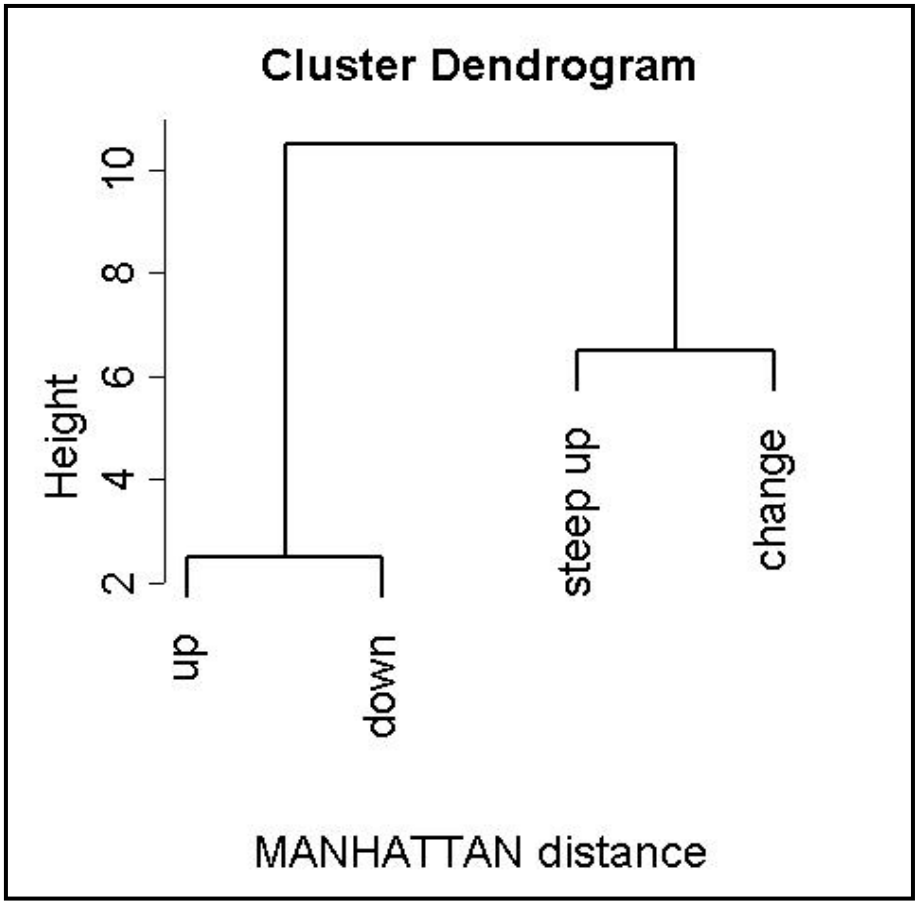


Time Series Example

- **Manhattan distance**

  Similar values are clustered together (robust)

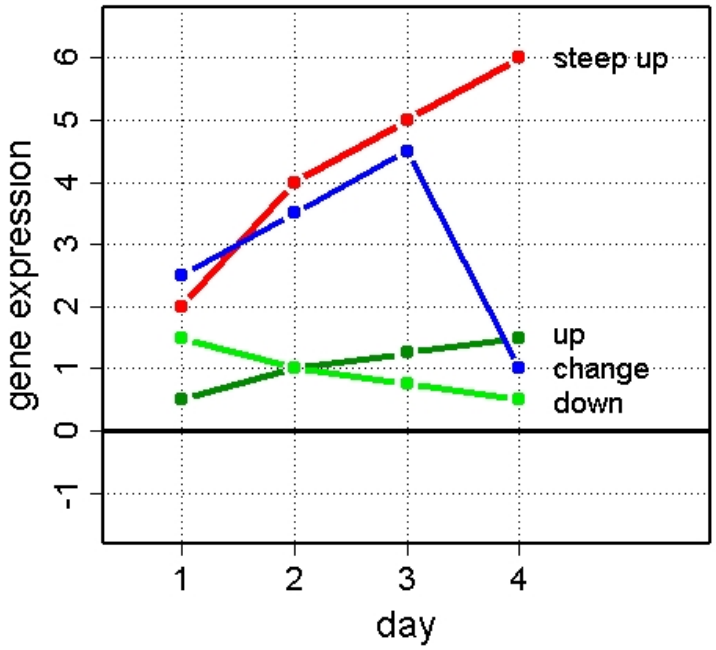


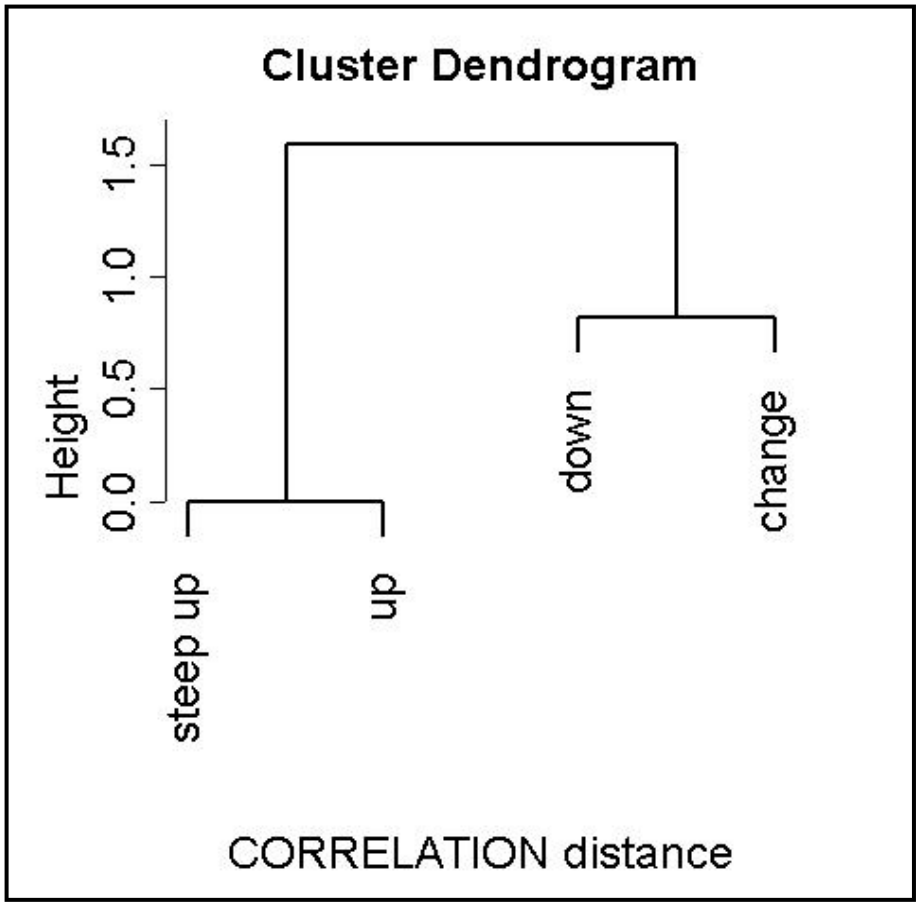


Time Series Example

- **Correlation distance**

  Similar trends are clustered together



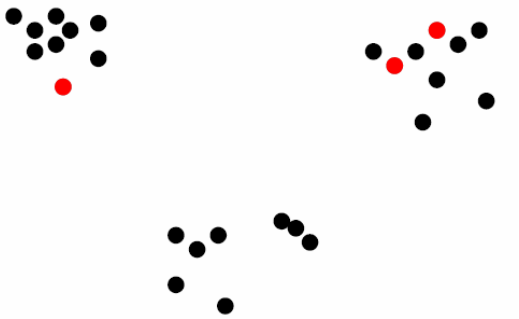


Cluster Algorithms – K-means

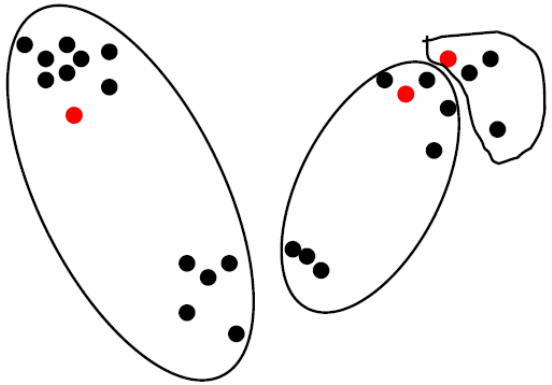
- **K-means** is a partitioning algorithm with a prefixed number *k* of clusters. It tries to minimize the sum of within-cluster-variances.
- The algorithm chooses a random sample of *k* different objects as initial cluster midpoints. Then it alternates between two steps until convergence:
    1. Assign each object to its closest of the *k* midpoints with respect to Euclidean distance.
    2. Calculate *k* new midpoints as the averages of all points assigned to the old midpoints, respectively.
- K-means is a randomized algorithm, two runs usually produce different results. Thus it has to be applied several times to the same data set and the result with minimal sum of within-cluster-variances should be chosen.

K-Means Clustering

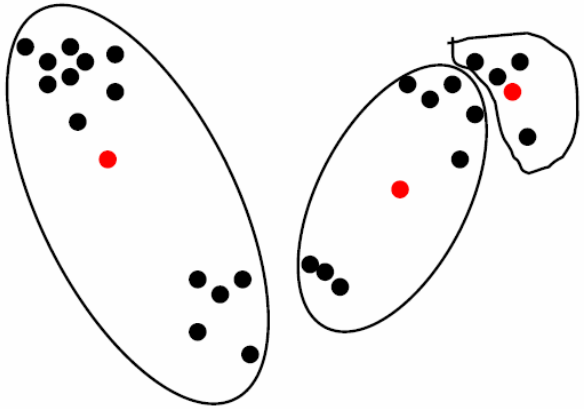
1. Choose Number of Clusters (3) and random Cluster-means.



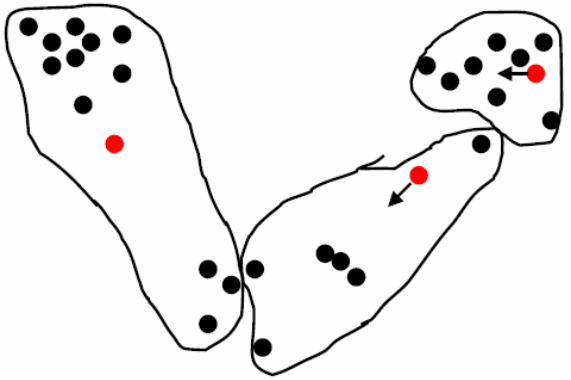
2. Assign Data-points to the respective cluster with the closest mean.



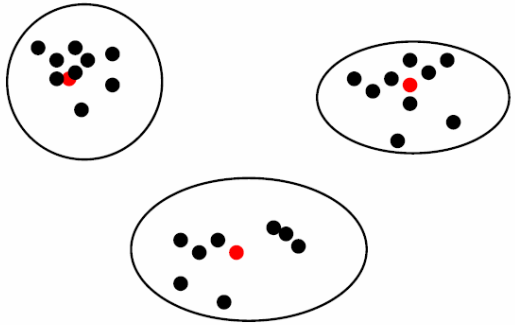
3. Compute new Cluster-means based on the new clusters..



4. Assign Data-points to the new cluster means..

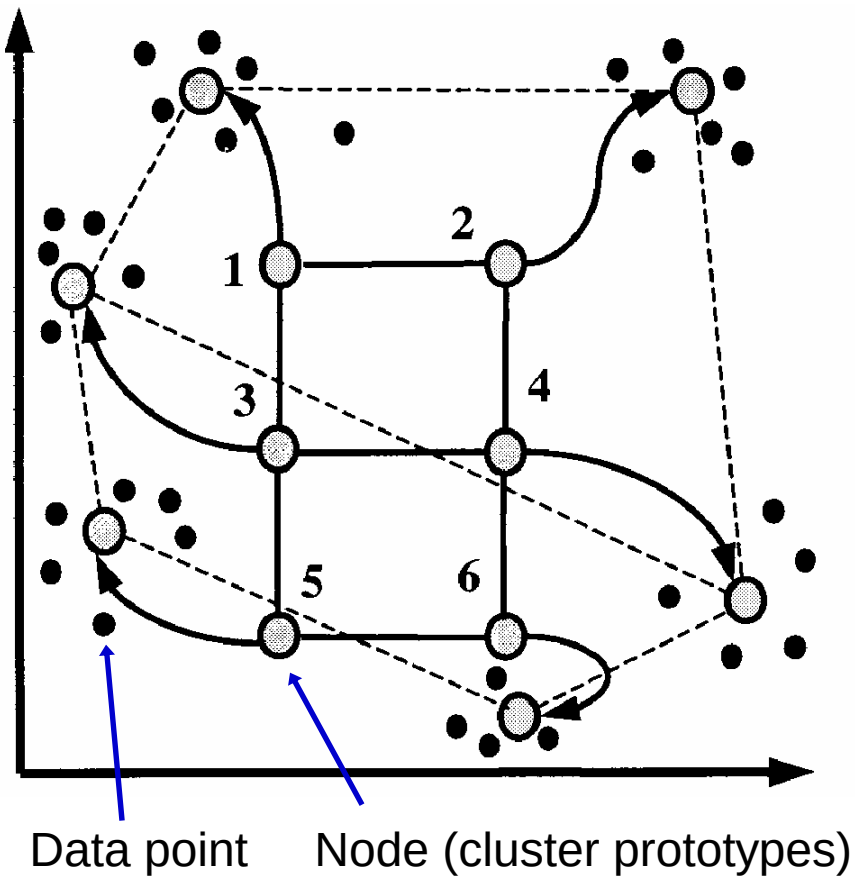


5. – n Iterate 3. and 4. until the clusters do not change any more.



Cluster Algorithms – Self-Organizing Maps

- **SOM's** are similar to k-means, but with additional **constraints**.
- Mapping from input space onto one or two-dimensional array of *k* total nodes.
- Iteration steps (*20000-50000*):
  - Pick data point *P* at random
  - Move all nodes in direction of *P*, the closest node in network topology most, the further a node is in network topology, the less
  - Decrease amount of movement with iteration steps



Tamayo et al. (1999): First use of SOM's for gene clustering from microarrays

Cluster Algorithms - PAM

- **PAM** (Partitioning around medoids, Kaufman and Rousseeuw (1990)) is a partitioning algorithm, a generalization of k-means.

- For an arbitrary dissimilarity matrix *d* it tries to minimize the sum (over all objects) of distances to the closest of *k* prototypes.

- Objective function: $\sum_{i=1}^{n} \min_{j=1,\ldots,k} d(i, m_j)$ (*d*: Manhattan, Correlation, ...)

- BUILD phase: Initial 'medoids'.

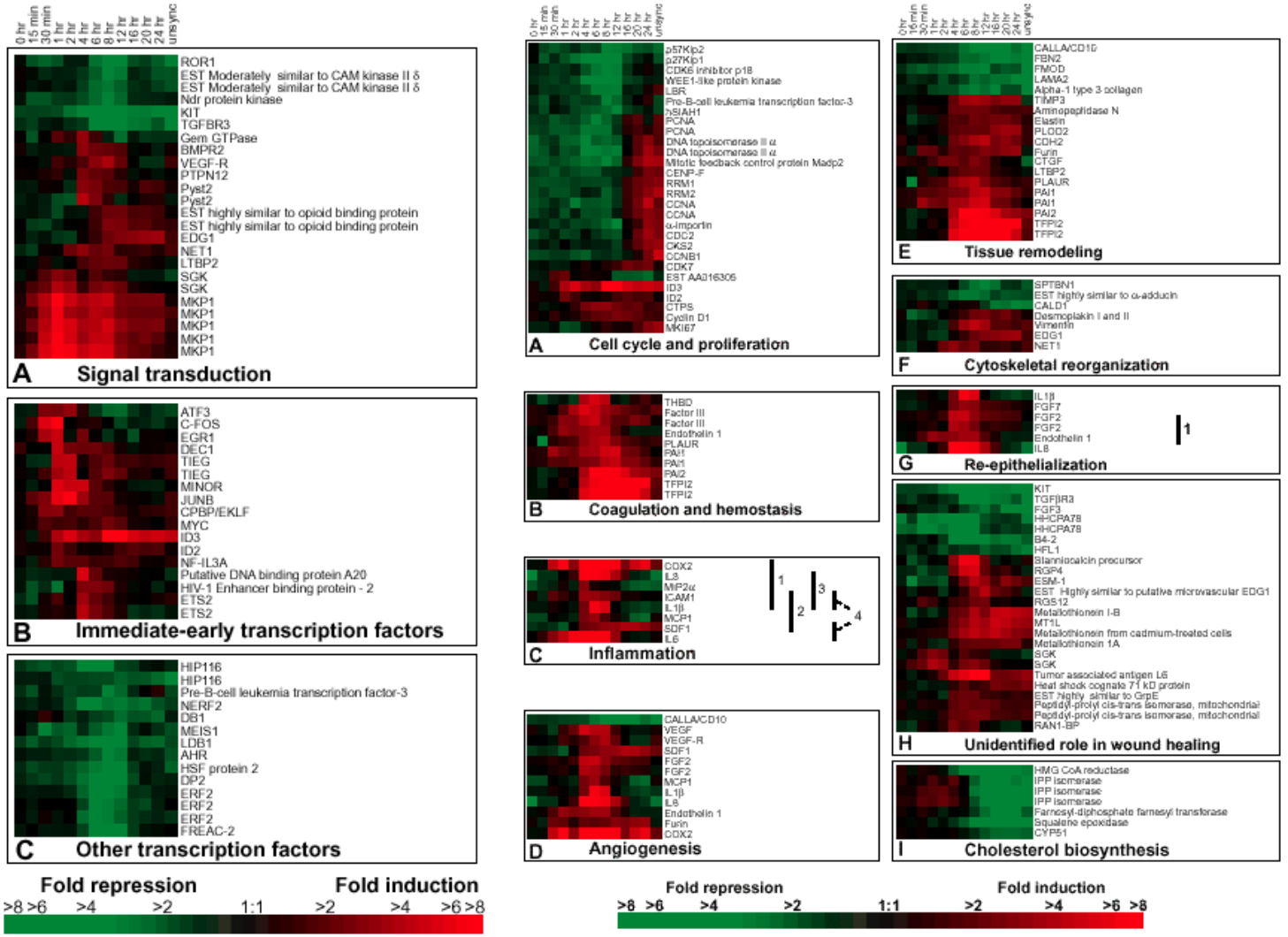
  SWAP phase: Repeat until convergence:
  - Consider all pairs of objects (i,j), where i is a medoid and j not, and make the i ↔ j swap (if any) which decreases the objective function most.

Clustering Time Series - Literature Example

Iyer et al., Science, Jan 1999:

Genes from functional classes are clustered together (sometimes!)

Careful interpretation neccessary!



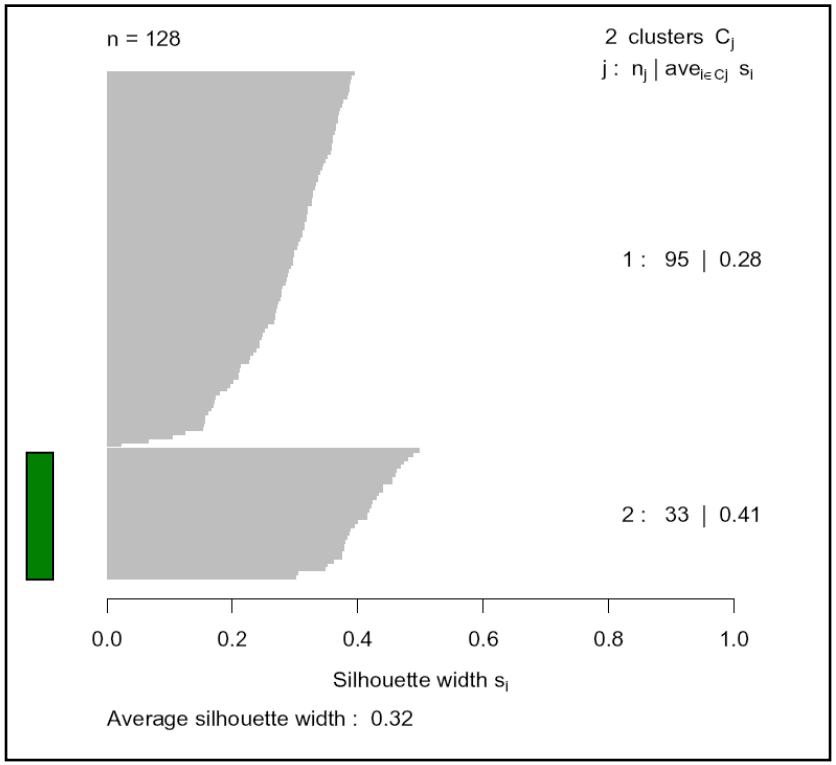
Estimating the Number of Clusters

- Internal indices
  - Statistics based on within- and between-clusters matrices of sums-of-squares and on cross-products (Milligan & Cooper (1985): exhaustive comparison of 30 indices)
  - Estimate is number of clusters *K* that minimizes/maximizes an internal index
- Model-based methods
  - EM algorithm for Gaussian mixtures, Fraley & Raftery (1998, 2000) and McLachlan et al. (2001)
- Gap statistic
  - Resampling method, for each *K* compare an observed internal index to its expected value under a reference distribution and look for *K* which maximizes the difference (Tibshirani et al., 2001)
    Caution: Does not work in high dimensions (e.g. large numbers of genes)
- Average silhouette width (Kaufman & Rousseeuw, 1990)

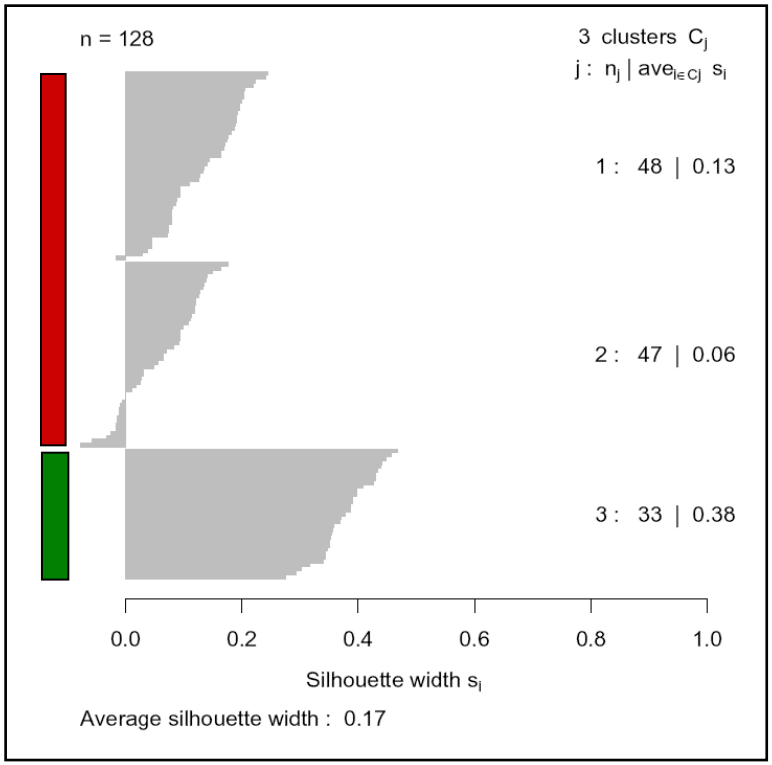
Estimating the Number of Clusters

- Silhouette plots for clustering Leukemia patients (Chiaretti et al., 2004)

K=2 clusters



K=3 clusters



Green: Well separated cluster

Red: No clear cluster structure

Estimating the Number of Clusters

Heuristic approach: Average silhouette width

- For each observation *i*, define *silhouette width s(i)* as follows:
  - *a(i)* := average dissimilarity between i and all other points of its cluster.
  - For all *other* clusters *C*, let *d(i,C)* := average dissimilarity of *i* to all observations of *C*. Define $b(i) := min_C\ d(i,C)$.
  - Define silhouette width: $s(i) := (b(i)-a(i)) / max(a(i),b(i))$.
- Maximal **average silhouette width** over all observations can be used to select the number of clusters.
- Observations with *s(i)* close to *1* can be considered well-clustered, observations with $s(i)<0$ are misclassified.
- The optimal number of clusters cannot be determined in general, as the quality of a clustering result depends on the concept of a cluster.

Cluster Validity

- If true class labels are known, the validity of the clustering can be verified by comparing true class labels and clustering labels with external cluster indices.

Number of misclassifications

n_{ij} = # objects in class i and cluster j

Iteratively match best fitting class and cluster, and sum up numbers of remaining observations.

$$\begin{array}{|c|c|}\hline N & \cdot \\ \hline \cdot & n_{..} \\ \hline \end{array} = \begin{array}{|cccc|c|}\hline n_{11} & n_{12} & \dots & n_{1l} & n_{1.} \\ n_{21} & n_{22} & \dots & n_{2l} & n_{2.} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ n_{k1} & n_{k2} & \dots & n_{kl} & n_{k.} \\ \hline n_{.1} & n_{.2} & \dots & n_{.l} & n_{..} \\ \hline \end{array}$$

Rand index

Probability of randomly drawing 'consistent' pair of observations.

$$Rand = \frac{\sum_{i,j}\binom{n_{ij}}{2} - \left[\sum_i \binom{n_{i.}}{2}\sum_j \binom{n_{.j}}{2}\right] \Big/ \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{n_{i.}}{2} + \sum_j \binom{n_{.j}}{2}\right] - \left[\sum_i \binom{n_{i.}}{2}\sum_j \binom{n_{.j}}{2}\right] \Big/ \binom{n}{2}}$$

Cluster Validity

- Yeung et al. (Bioinformatics, 2001)

  Framework for assessing the quality of algorithms for clustering genes.

  Apply algorithm to data from all but one condition (sample) and use the remaining condition to assess predictive power of the resulting clusters (leave-one-out scenario).

- Dudoit and Fridlyand (Genome Biology, 2002)

  Resampling method *Clest* to estimate the number of clusters in a dataset based on prediction accuracy

- Smolkin and Ghosh (BMC Bioinformatics, 2003)

  Cluster stability scores for microarray data in cancer studies based on subsampling techniques

Cluster Validity - Comparative Study

- Comparative study for tumor classification with microarrays:

  Comparison of hierarchical clustering, k-means, PAM and SOM's

- Data sets:

- Golub et al: Leukemia dataset, *http://www.genome.wi.mit.edu/MPR,* 3 cancer classes: 25 acute myeloid leukemia (AML) and 47 acute lymphoblastic leukemia (ALL) (9 T-cell and 38 B-cell), Affymetrix.
- Ross et al.: NCI60 cancer dataset, *http://genome-www.stanford.edu/nci60,* 9 cancer classes: 9 breast, 6 central nervous system, 7 colon, 8 leukemia, 8 melanoma, 9 lung, 6 ovarian, 2 prostate, 8 renal, cDNA microarray

- Superiority of k-means with repeated runs

  (Similar for discriminant analysis: FLDA best, Dudoit et al. 2001)

- Superiority of PAM with Manhattan distance especially for noisy data

- Differences depend on the specific dataset

- Rahnenführer (2002): **Efficient clustering methods for tumor classification with gene expression arrays,** *Proceedings of '26th Annual Conference of the Gesellschaft für Klassifikation'*, Mannheim, July 2002.

Classification

MESSAGE 3

Simple cluster algorithms work better in case of little model knowledge!

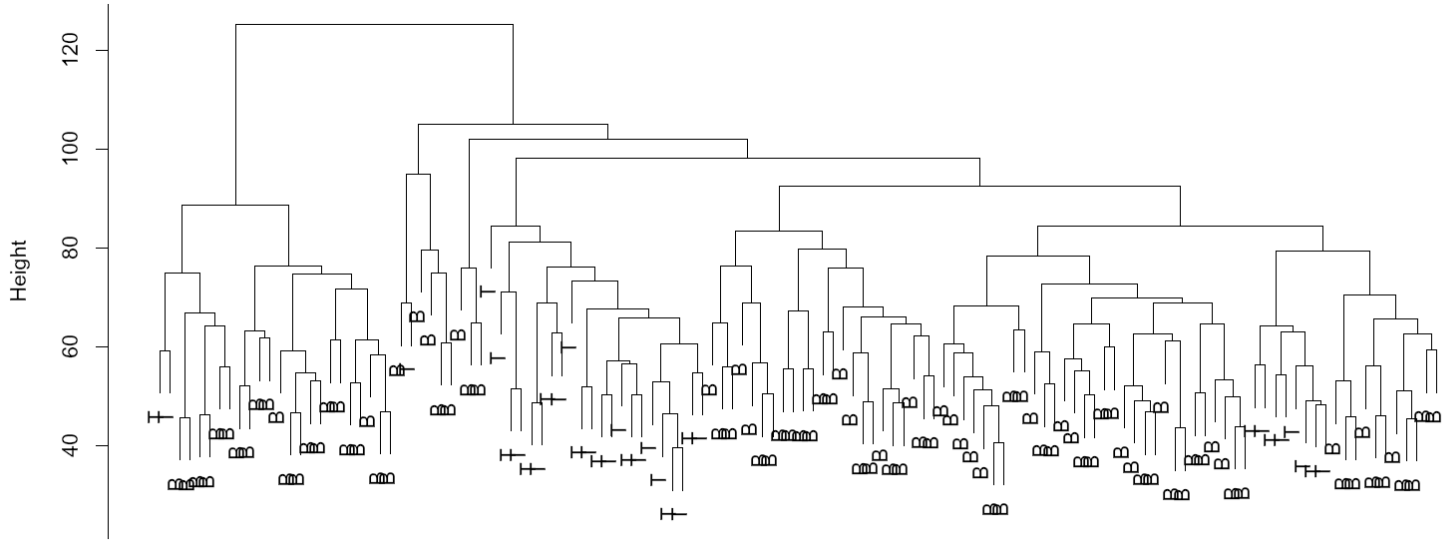
(But: More sophisticated methods might be more appropriate with more a priori knowledge)

Gene Selection

- Preselection of genes

  Various approaches for gene selection, especially in *supervised* learning.

  For clustering samples, a practical proceeding is to choose the top 100-200 genes with respect to variance (across samples). This decreases noise and computation time.



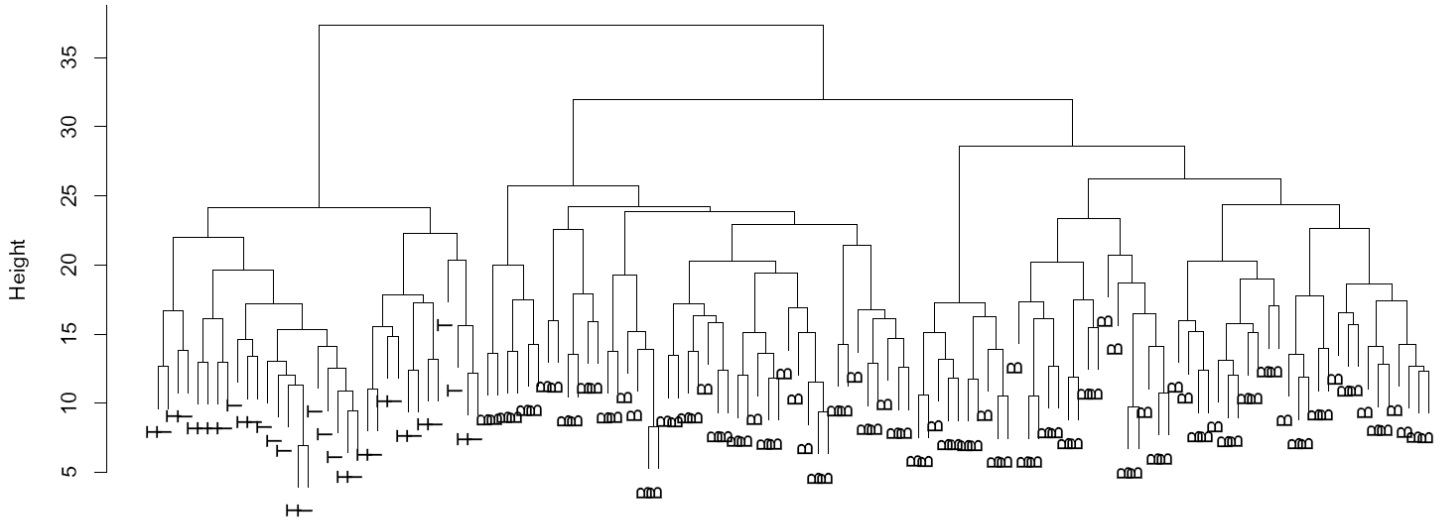
Dendrogram for clustering Leukemia patients (Chiaretti et al., 2004) without gene selection

Gene Selection

- Preselection of genes

  Various approaches for gene selection, especially in *supervised* learning.

  For clustering samples, a practical proceeding is to choose the top 100-200 genes with respect to variance (across samples). This decreases noise and computation time.



Dendrogram for clustering Leukemia patients (Chiaretti et al., 2004)
with 100 top variance genes

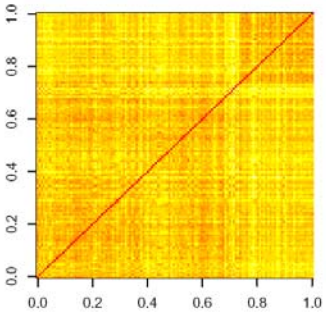
Gene Selection

- Preselection of genes

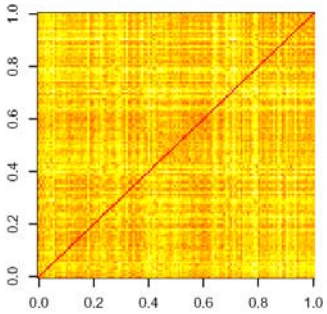
  Various approaches for gene selection, especially in *supervised* learning.

  For clustering samples, a practical proceeding is to choose the top 100-200 genes with respect to variance (across samples). This decreases noise and computation time.

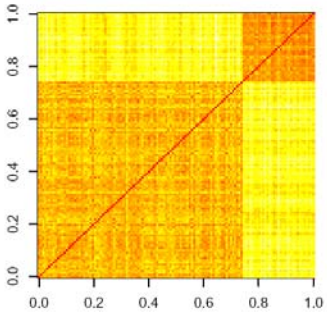
Distance matrices for clustering Leukemia patients (Chiaretti et al., 2004)



All genes

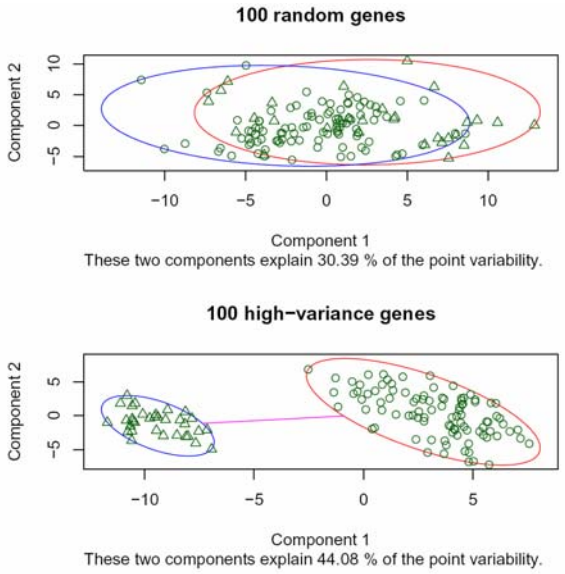


100 random genes



100 high-variance genes

Plot of sample types in first two principal components

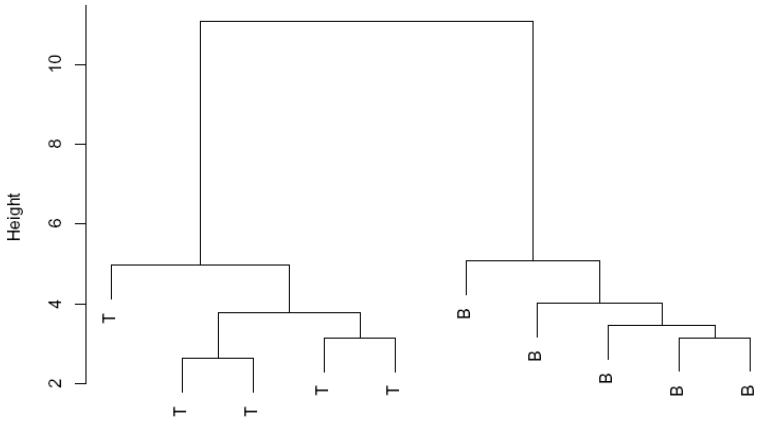


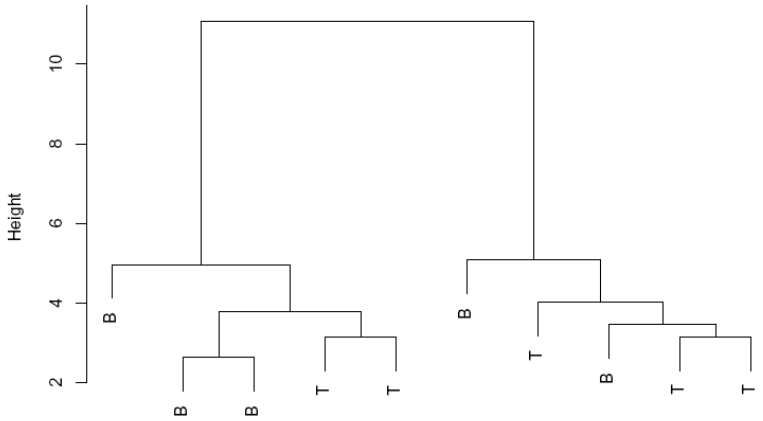
Gene Selection

- Clustering after supervised feature selection

  NO! Do not first select genes based on the outcome of some covariable (e.g. tumor type) and then look at the clustering.

  You will ALWAYS find difference w.r.t. your covariable, since this is how you selected the genes!





Left dendrogram obtained by

1. Random assignment of sample labels
2. Selection of best discriminating genes
3. Clustering with selected genes

Right plot shows original labels

R commands and libraries

- **library(mva)**
  - Hierarchical clustering: ***hclust()***
  - Kmeans: ***kmeans()***
  - Principal components: ***princomp()***
- **library(cluster)**
  - PAM: ***pam()***
  - Silhouette information: ***silhouette()***
- **library(cclust)**
- **library(mclust)**

SUMMARY

MESSAGE 1:

Discriminant analysis: CLASSES KNOWN

Cluster analysis: CLASSES NOT KNOWN

MESSAGE 2:

Appropriate choice of distance measure depends on your intention!

MESSAGE 3:

Simple cluster algorithms work better in case of little model knowledge!

Literature

1. Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, Mesirov JP, Coller H, Loh ML, Downing JR, Caligiuri MA, Bloomfield CD, Lander ES. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 1999; 286: 531-37.
2. Alizadeh AA, Eisen MB, Davis RE and 28 others. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature 2000; 403: 503-11.
3. Jain A, Dubes RC. Algorithms for Clustering Data. Englewood Cliffs, New Jersey: Prentice Hall; 1988.
4. Azuaje F. Clustering-based approaches to discovering and visualising microarray data patterns. Brief. Bioinformatics 2003; 4: 31-42.
5. Eisen MB, Spellman PT, Brown PO, Botstein D. Cluster analysis and display of genome-wide expression patterns. PNAS 1998; 95: 14863-68.
6. Tamayo P, Slonim D, Mesirov J, Zhu Q, Kitareewan S, Dmitrovsky E, Lander ES, Golub TR. Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. PNAS 1999; 96: 2907-12.
7. Kaufman L, Rousseeuw P. Finding Groups in Data. New York: John Wiley and Sons; 1990.
8. Ben-Dor A, Shamir R, Yakhini Z. Clustering gene expression patterns. J. Comput Biol. 1999; 6: 281-97.

Literature

9. Cheng Y, Church GM. Biclustering of expression data. Proc Int Conf Intell Syst Mol Biol. 2000; 8:93-103.
10. Tanay A, Sharan R, Shamir R. Discovering statistically significant biclusters in gene expression data. Bioinformatics 2002; Suppl 1: 136-44.
11. Hastie T, Tibshirani R, Eisen MB, Alizadeh A, Levy R, Staudt L, Chan WC, Botstein D, Brown P. 'Gene shaving' as a method for identifying distinct sets of genes with similar expression patterns. Genome Biol. 2000; 1(2): RESEARCH0003.
12. Yeung KY, Haynor DR, Ruzzo WL. Validating clustering for gene expression data. Bioinformatics 2001; 17: 309-18.
13. Rahnenführer J. Efficient clustering methods for tumor classification with microarrays. In: Between Data Science and Applied Data Analysis (Eds: M. Schader, W. Gaul, M. Vichi), Springer, Proc. 26th Ann. Conf. GfKl 2002; 670-679.
14. Dudoit S, Fridlyand J: A prediction-based resampling method to estimate the number of clusters in a dataset. Genome Biology 2002; 3:RESEARCH0036.
15. Smolkin, M, Ghosh, D. Cluster stability scores for microarray data in cancer studies. BMC Bioinformatics 2003, 4:36.
16. Rahnenführer, J. Clustering algorithms and other exploratory methods for microarray data analysis. Methods of Information in Medicine 2005; 44(3): 444-8.