Exploratory data analysis for microarrays

Jörg Rahnenführer

Computational Biology and Applied Algorithmics
Max Planck Institute for Informatics
D-66123 Saarbrücken
Germany

NGFN - Courses in Practical DNA Microarray Analysis

Heidelberg, October 8, 2003

Overview

- **Exploratory data analysis:** Unsupervised learning
- **Example:** Time series
- **Distance measures:** Object (dis-)similarities
- **Cluster algorithms:** Grouping of data
- **Clustering microarray data:** Comparisons and hints
- **Other exploratory methods** for microarray data

Interactive exploratory data analysis

YOU ARE ALL GENES…

Classification tasks for microarrays

MAX-PLANCK-GESELLSCHAFT

- **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.

Classification

- **Paper of Golub et al. (1999):**

  Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, Science 286, p. 531-537.

- **Machine learning:**

  Supervised learning vs. unsupervised learning.

- **Statistics:**

  Discriminant analysis vs. cluster analysis.

Classification

MESSAGE 1:

Discriminant analysis: CLASSES KNOWN

Cluster analysis: CLASSES NOT KNOWN

Classification

- **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, unsupervised learning methods have to be used. For those, it is *difficult to ascertain the validity of inferences* drawn from the output.**

Cluster analysis

- **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.

- **<u>Questions:</u>**

  1. **What does “closely related” mean?**
  2. **How do we find such subsets or clusters?**

Cluster analysis

- **Two "ingredients" are needed to group objects into subsets:**

1. **<u>Distance measure:</u>**

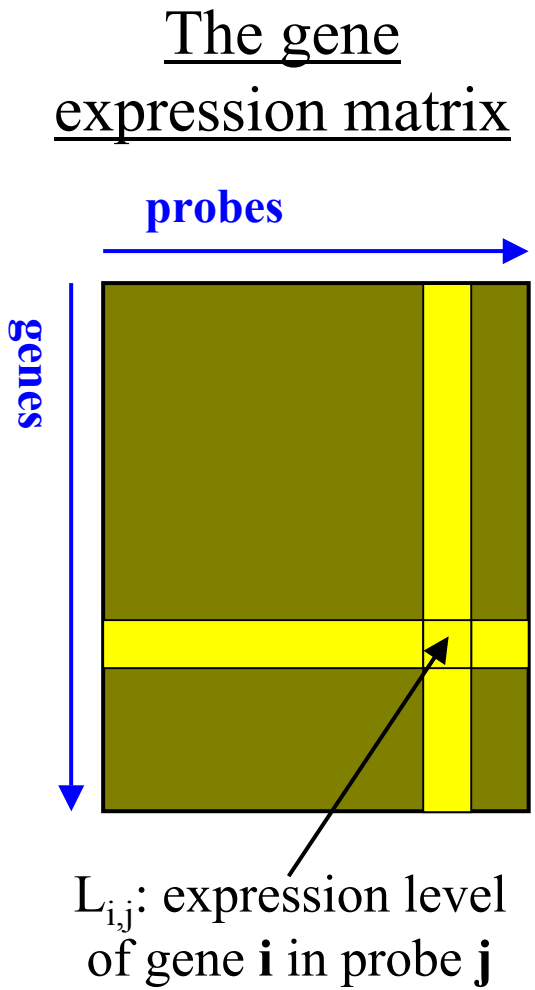
   A notion of distance or similarity of two objects: When are two objects close to each other?

2. **<u>Cluster algorithm:</u>**

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

Cluster analysis

- Clustering columns: grouping similar samples
- Clustering rows: Grouping genes with similar trajectories



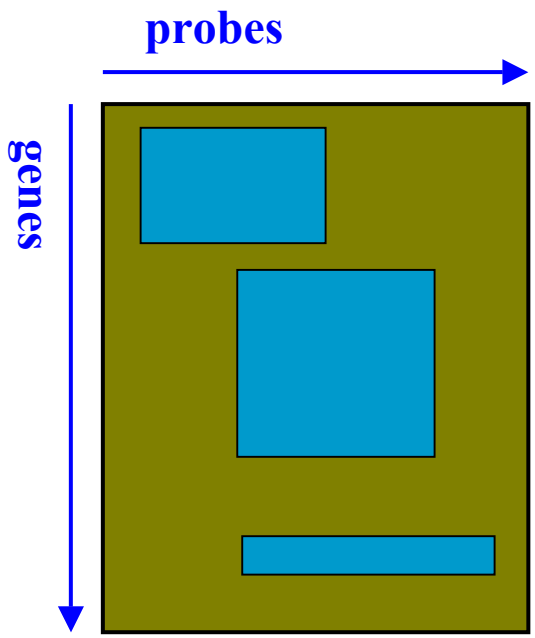
Cluster analysis: Bi-Clustering

- Clustering columns: grouping similar samples
- Clustering rows: Grouping genes with similar trajectories
- Biclustering: Group genes that have similar partial trajectories in a subset of the samples

Literature:

Tanay, A., Sharan, R., and Shamir, R. (2002): **Discovering Statistically Signifnicant Biclusters in Gene Expression Data,** *Bioinformatics* 18, Suppl.1, 136-144.

The gene expression matrix



Time series example

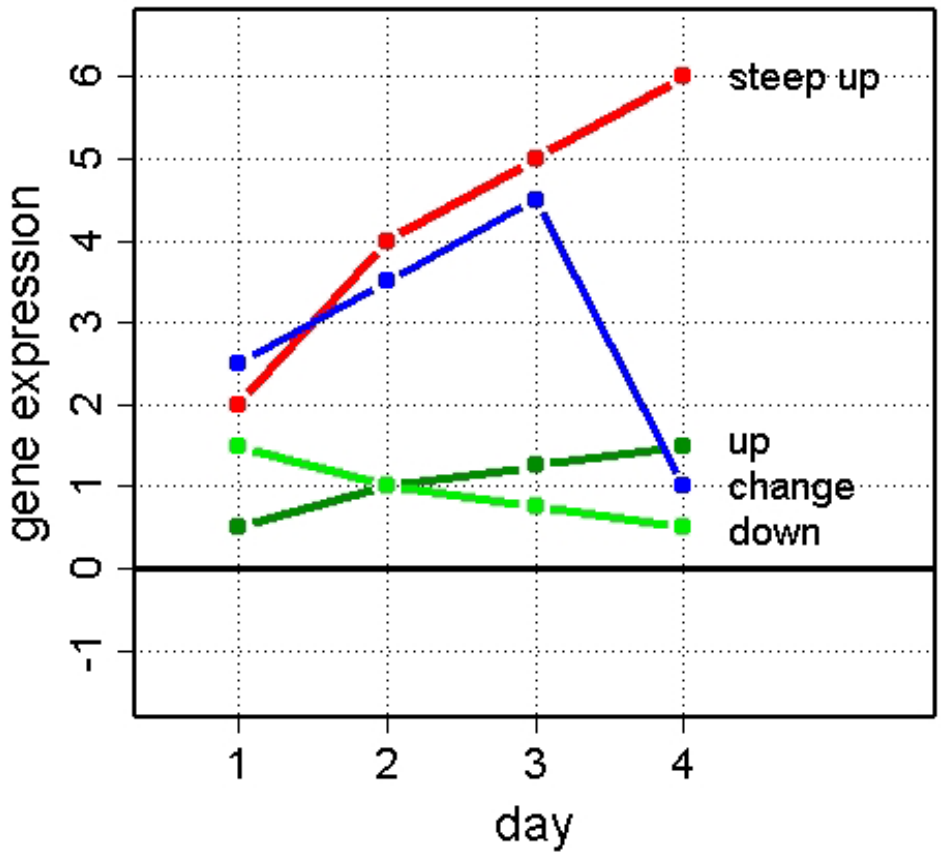
Biology:

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

MAX-PLANCK-GESELLSCHAFT

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: $x_2 = (2/4, 4/4, 5/4, 6/4)$

Distance measures - Time series example

MAX-PLANCK-GESELLSCHAFT

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

MAX-PLANCK-GESELLSCHAFT

Manhattan distance:

The distance between two vectors is the 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: $x_2 = (2/4, 4/4, 5/4, 6/4)$

Distance measures - Time series example

MAX-PLANCK-GESELLSCHAFT

Manhattan distance:

The distance between two vectors is the 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

MAX-PLANCK-GESELLSCHAFT

Correlation distance:

Distance between two vectors is 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: $x_2 = (2/4, 4/4, 5/4, 6/4)$

Distance measures - Time series example

MAX-PLANCK-GESELLSCHAFT

Correlation distance:

Distance between two vectors is 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

MAX-PLANCK-GESELLSCHAFT

Euclidean

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

Manhattan

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

Correlation

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

Comparison: All distances are normalized to the interval [0,10] and then rounded.

	steep up	up	down	change
steep up	0 0 0	9 9 0	10 10 10	8 4 5
up	9 9 0	0 0 0	4 1 10	7 6 5
down	10 10 10	4 1 10	0 0 0	7 5 4
change	8 4 5	7 6 5	7 5 4	0 0 0

Distance measures - Time series example

MAX-PLANCK-GESELLSCHAFT

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.**

Distance measures - standardization

MAX-PLANCK-GESELLSCHAFT

Standardization:

- Data points 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 you are 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

Most popular cluster algorithms:

- **Hierarchical clustering algorithms**
- **K-means**
- **PAM (Partitioning around medoids)**
- **SOM's (Self-Organizing Maps)**

- K-means and SOM's take original data directly as input.
- Hierarchical cluster algorithms and PAM allow the choice of a dissimilarity matrix **d**, that assigns to each pair of objects $\mathbf{x}_i$ and $\mathbf{x}_j$ a value $d(\mathbf{x}_i,\mathbf{x}_j)$ as their distance.

Hierarchical cluster algorithms

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

- First, each object is assigned to its own cluster. Then, **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. Then, the similarity matrix is updated with this new node replacing the two joined clusters. This process is repeated until only one single cluster remains.

Hierarchical cluster algorithms

- Calculation of distance between two clusters is based on object dissimilarity between the objects from the two clusters:
  - Average linkage: Average distance
  - Single linkage: Smallest distance
  - Complete linkage: Largest distance
- Instead of agglomerative clustering, sometimes divisive clustering is used:

  Iteratively, best possible splits are calculated.

Hierarchical cluster algorithms

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

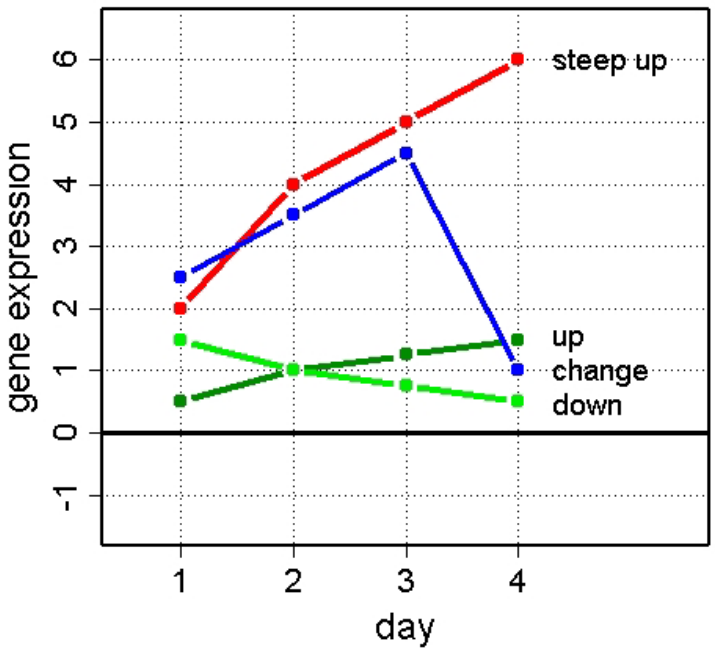
- The procedure provides a **hierarchy of clusterings**, with the number of clusters ranging from 1 to the number of objects.

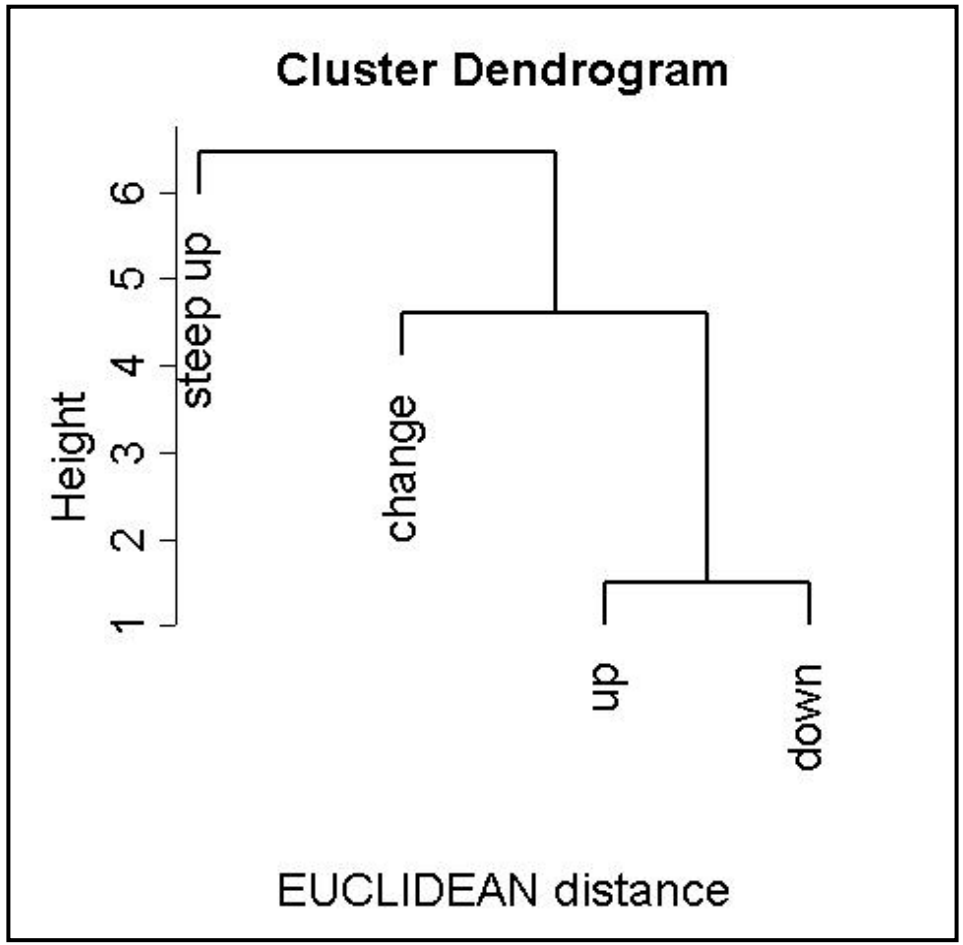
- **BUT**:
  - Parameters for distance matrix: **n(n-1)/2**
  - Parameters for dendrogram: **n-1**.

  ⟶ Hierarchical clustering does not show the full picture!

Time series example

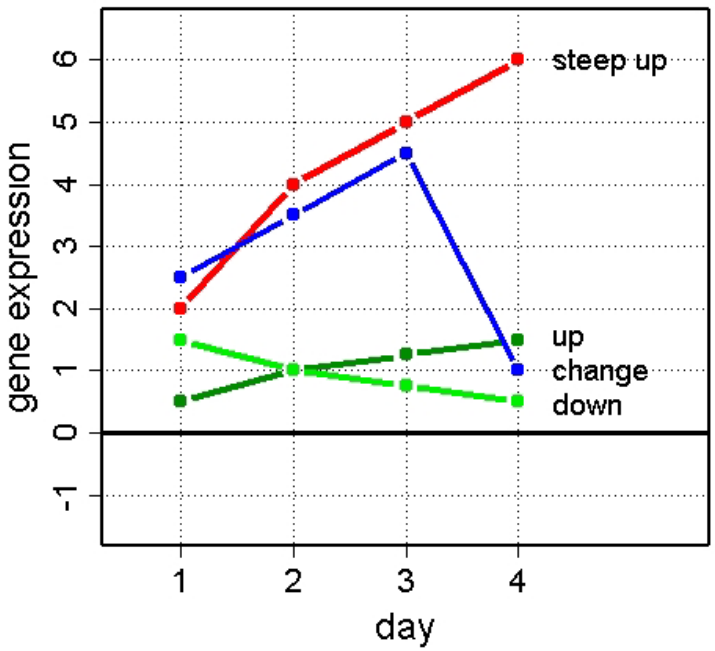
- **Euclidean distance:**
  Similar values are clustered together

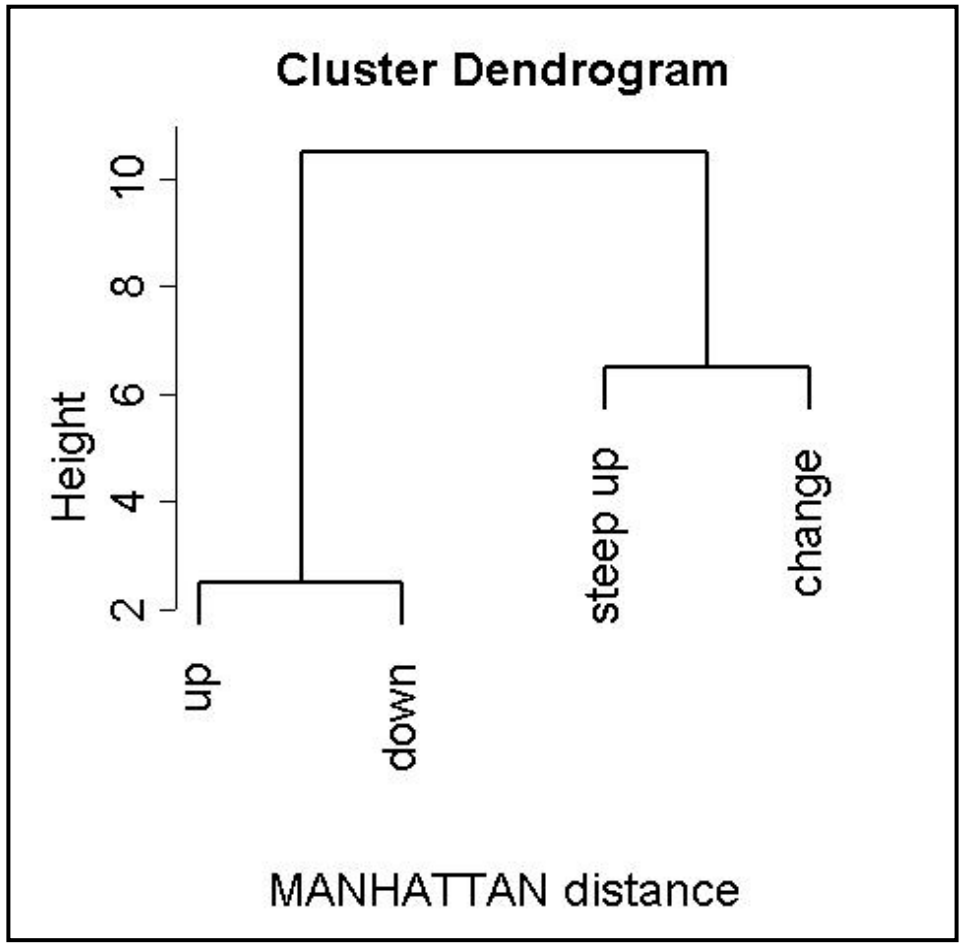




Time series example

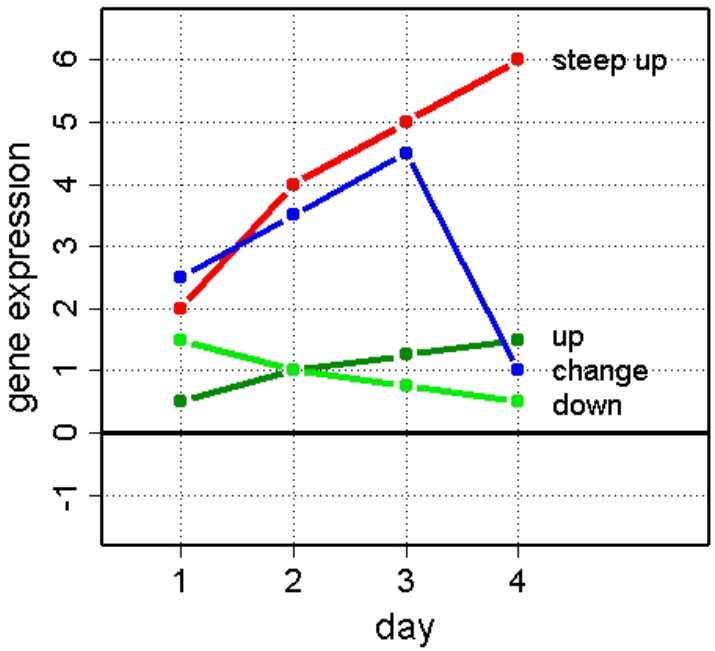
- **Manhattan distance:**
  Similar values are clustered together (robust)

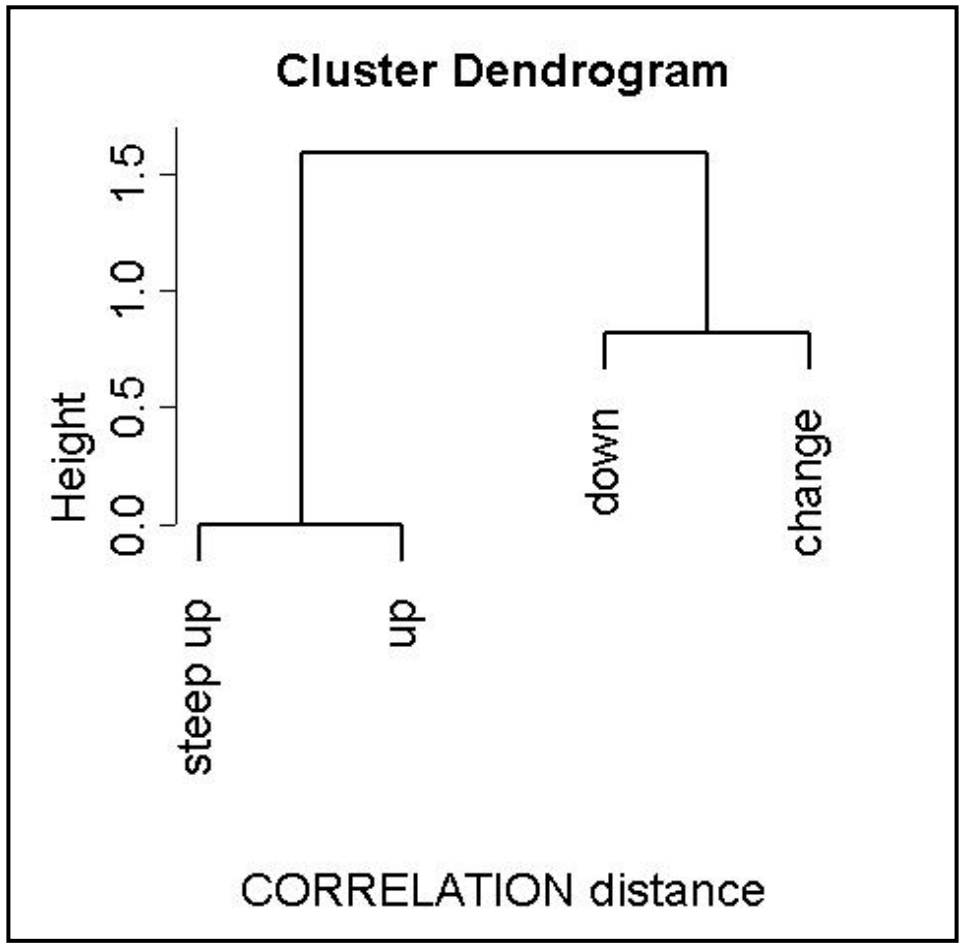




Time series example

- **Correlation distance:**
  Similar trends are clustered together

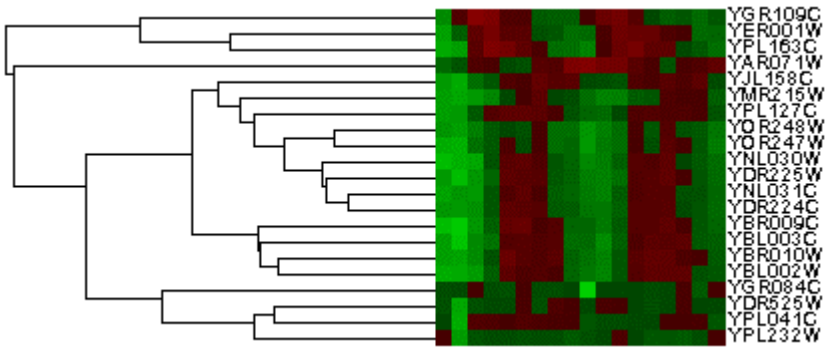




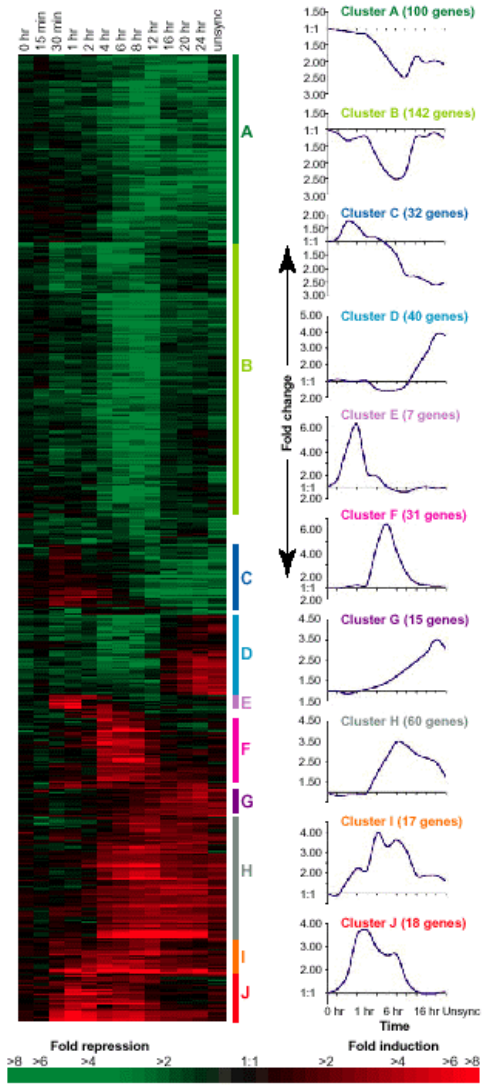
Clustering time series data – literature examples

Iyer et al., Science, Jan 1999:

Genes from functinal classes are clustered together.



Cluster dendrogram



Cluster average patterns

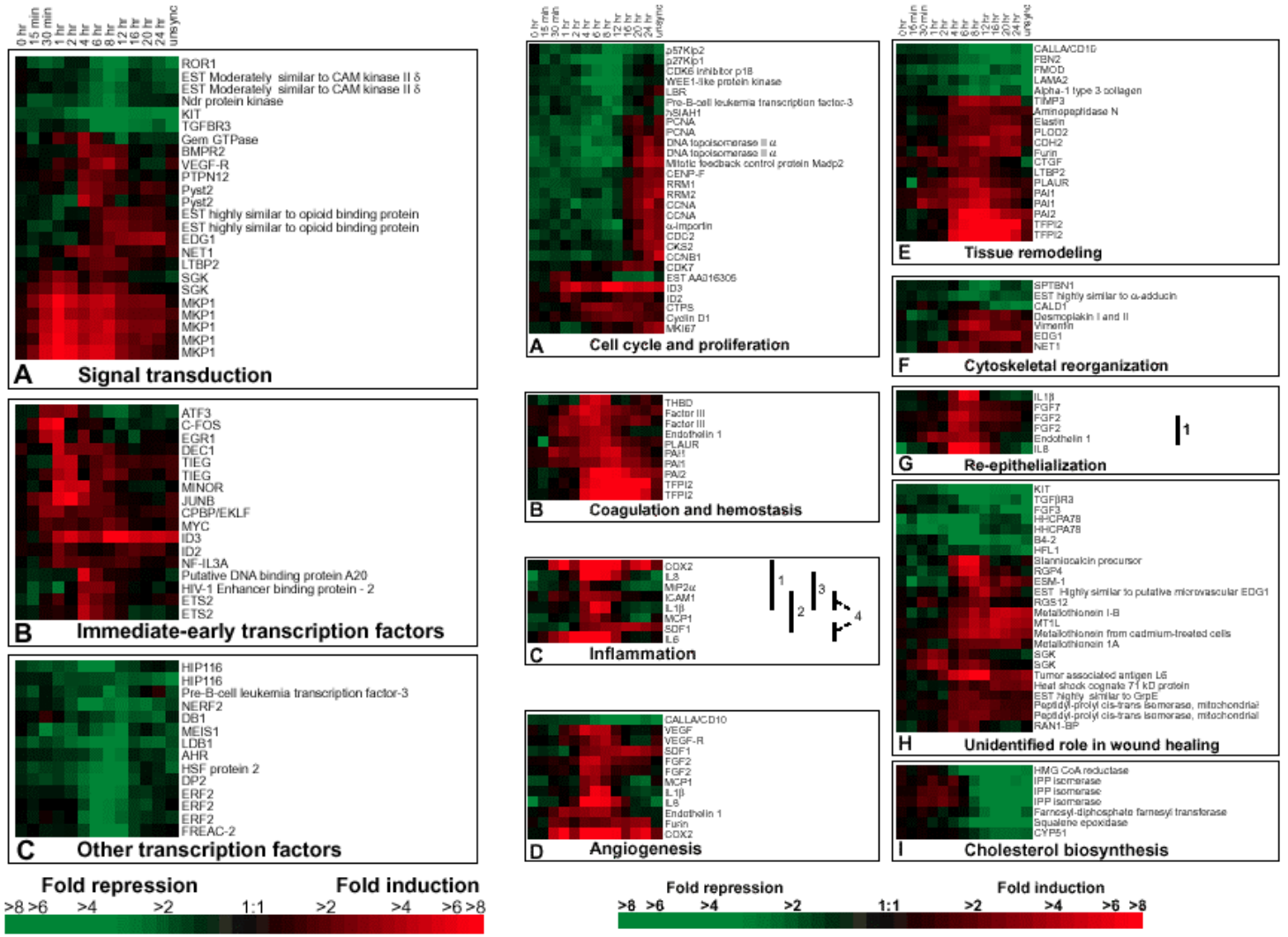
Clustering time series data – literature examples

MAX-PLANCK-GESELLSCHAFT

Iyer et al., Science, Jan 1999:

Genes from functinal classes are clustered together (sometimes!).

Careful interpretation neccessary!

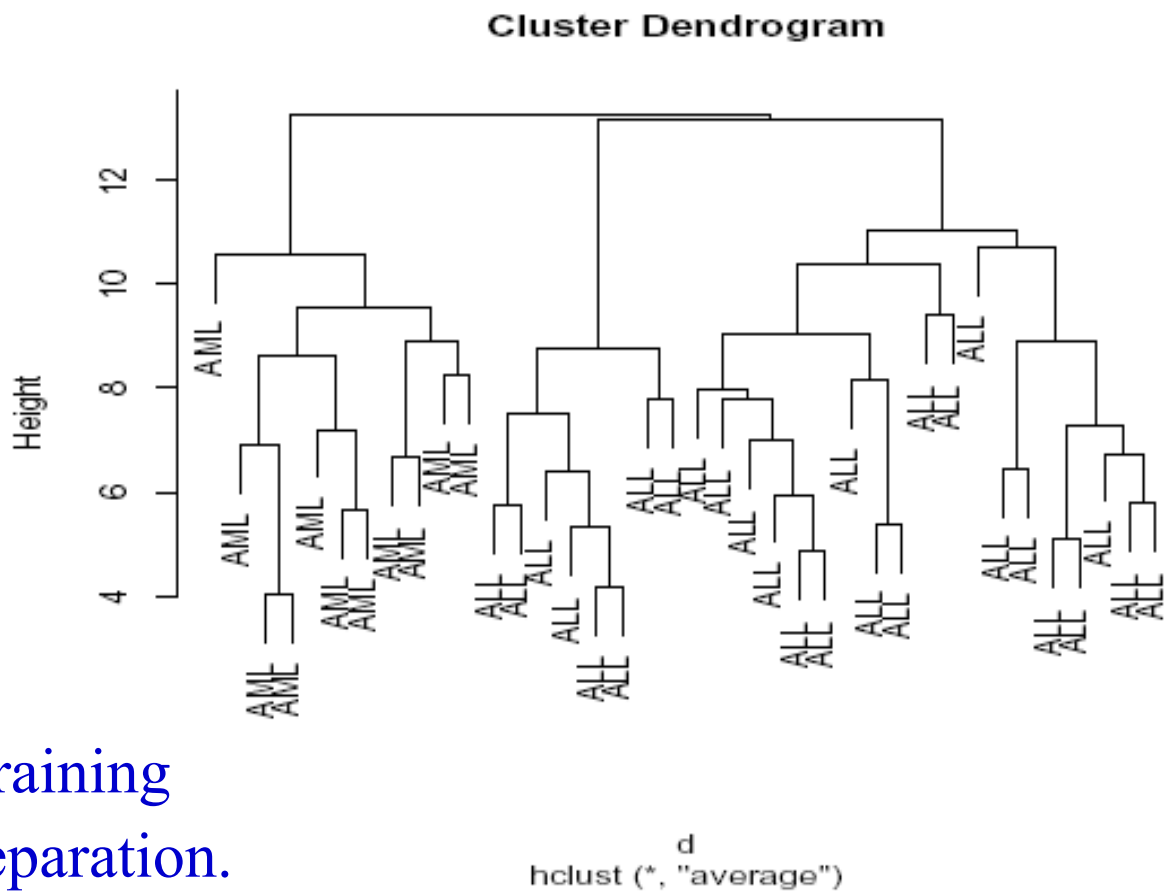


Clustering time series data – literature examples

MAX-PLANCK-GESELLSCHAFT

Golub et al.: Leukemia dataset, *http://www.genome.wi.mit.edu/MPR*

3 cancer classes:
25 acute myeloid leukemia (AML),
47 acute lympho-blastic leukemia (ALL), the latter
9 T-cell and 38 B-cell.



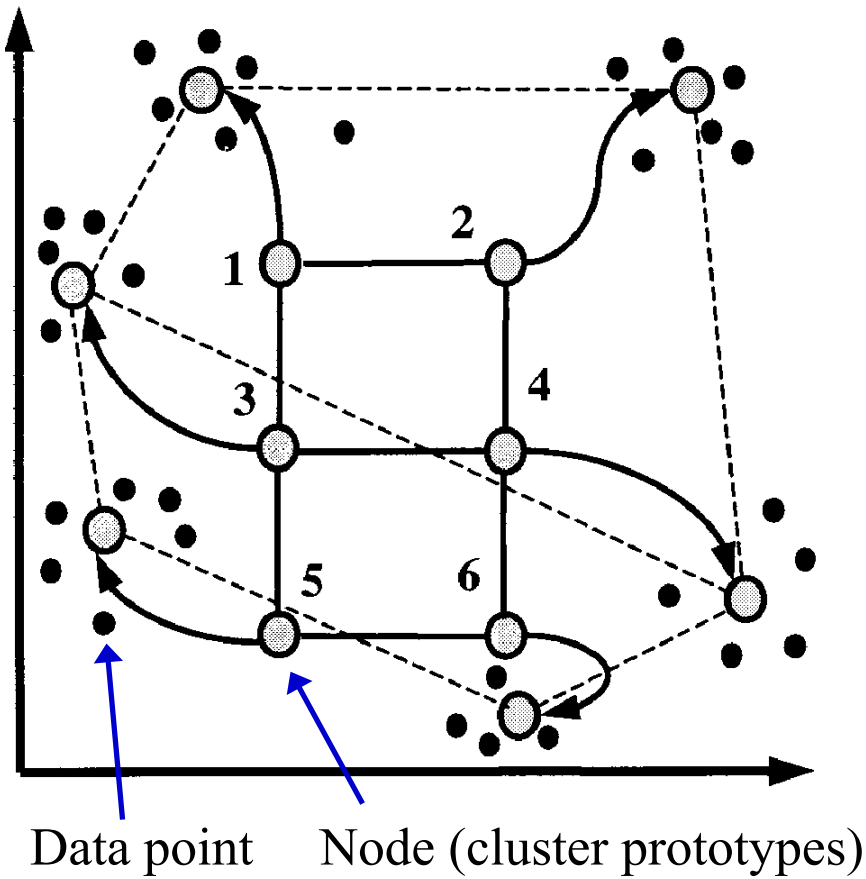
Dendrogram for 38 training data shows perfect separation.

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 a few times to the same data set and the result with minimal sum of within-cluster-variances should be chosen.

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 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, etc.)

- 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.

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), the latter 9 T-cell and 38 B-cell, Affymetrix high-density oligonucleotide chip, threshold and filtering preprocessing steps as in paper.
  - 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, log, imputation of missing values with k-nearest-neighbor method and correlation distance.

Comparative study – cluster validity

- If true class labels are known, the validity of the clustering can be verified by comparing true class labels and clustering label.

N… table of observations
n_{ij}… number of observations in class i and cluster j

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

Rand index:
Probability of randomly drawing 'consistent' pair of observations

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

Comparative study – method

Definition 5.1 *BOSC (Bootstrap-scaling):*
Let $n, p, k, B \in \mathbb{N}_{\geq 0}$. Let $(M_{ij})_{i=1..p, j=1..n}$ be a data (gene expression) matrix of dimension $p \times n$ (for p genes and n samples) and $(l_1, \ldots, l_n) \in (1, \ldots, k)^n$ an n-dimensional label vector that assigns every sample to one of k clusters (tumor types).

1. *For $b \in (1, \ldots, B)$:*
   Create replicate M^b_{ij} by randomly drawing from all values $M_{i_0 j_0}$ of the original data matrix that fulfill $i_0 = i$ and $j_0 \in \{j : l_j = l_{j_0}\}$.

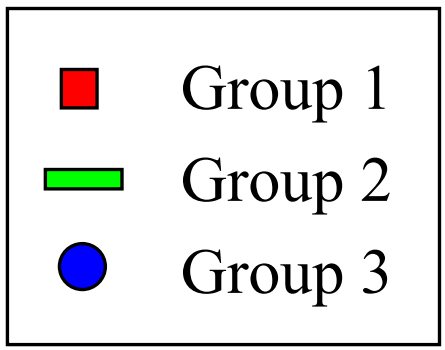
2. *For $s \in (s_1, \ldots, s_S)$ with $0 < s_1 < s_2 < \ldots < s_S < \infty$:*
   Define the modified (stretched) replicate $M^b_{ij}(s)$ as

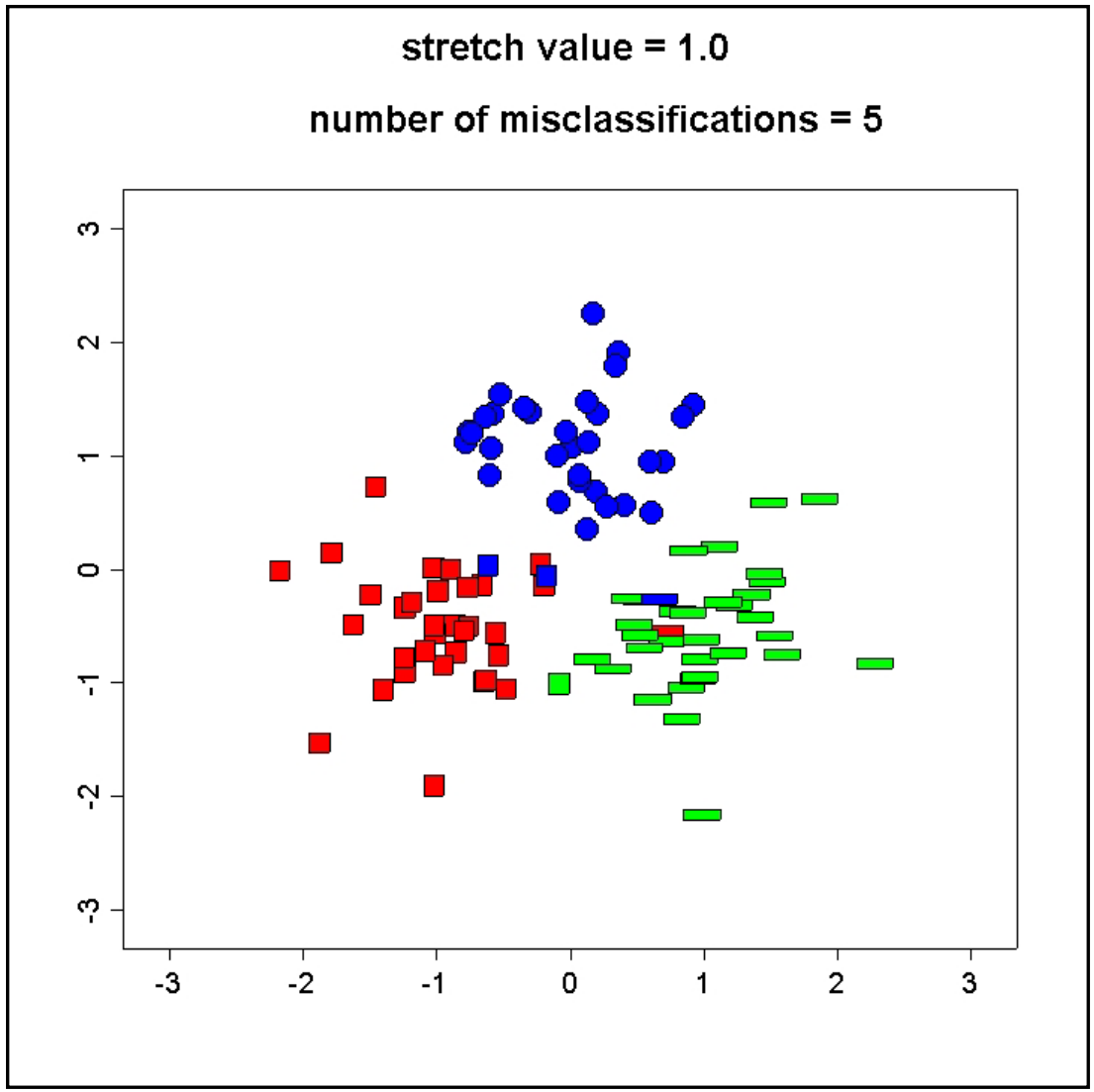
$$M^b_{ij}(s) := (1 - s) \left(\frac{1}{\#\{j_0 : l_j = l_{j_0}\}} \sum_{j_0 : l_j = l_{j_0}} M_{ij_0} \right) + s\, M^b_{ij}. \qquad (5.1)$$

Comparative study - method

Color → Group

Shape → Cluster

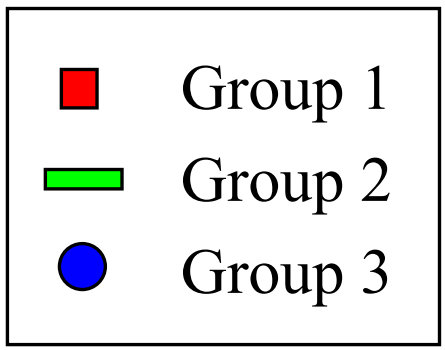


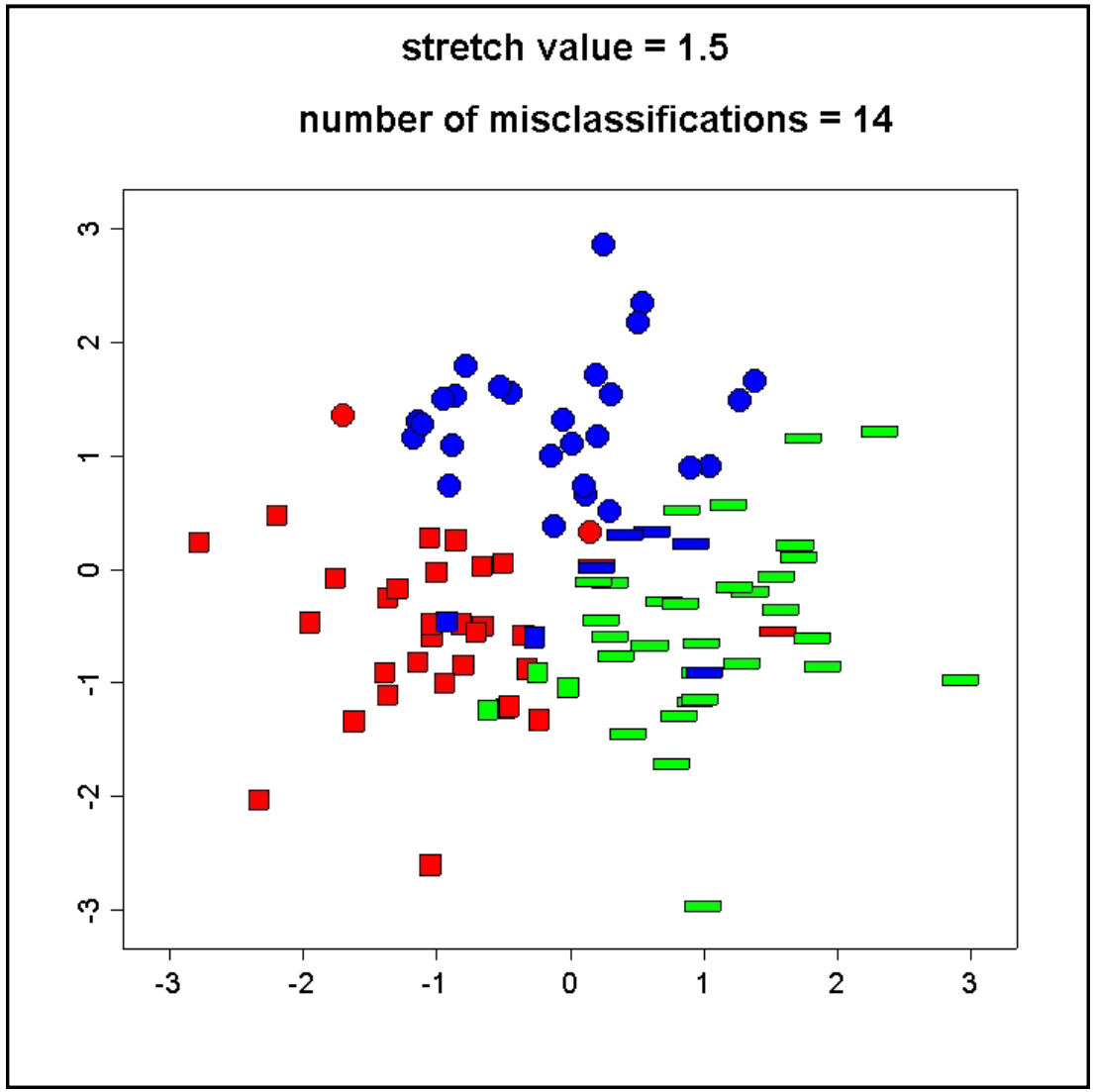


Comparative study - method

Color → Group

Shape → Cluster

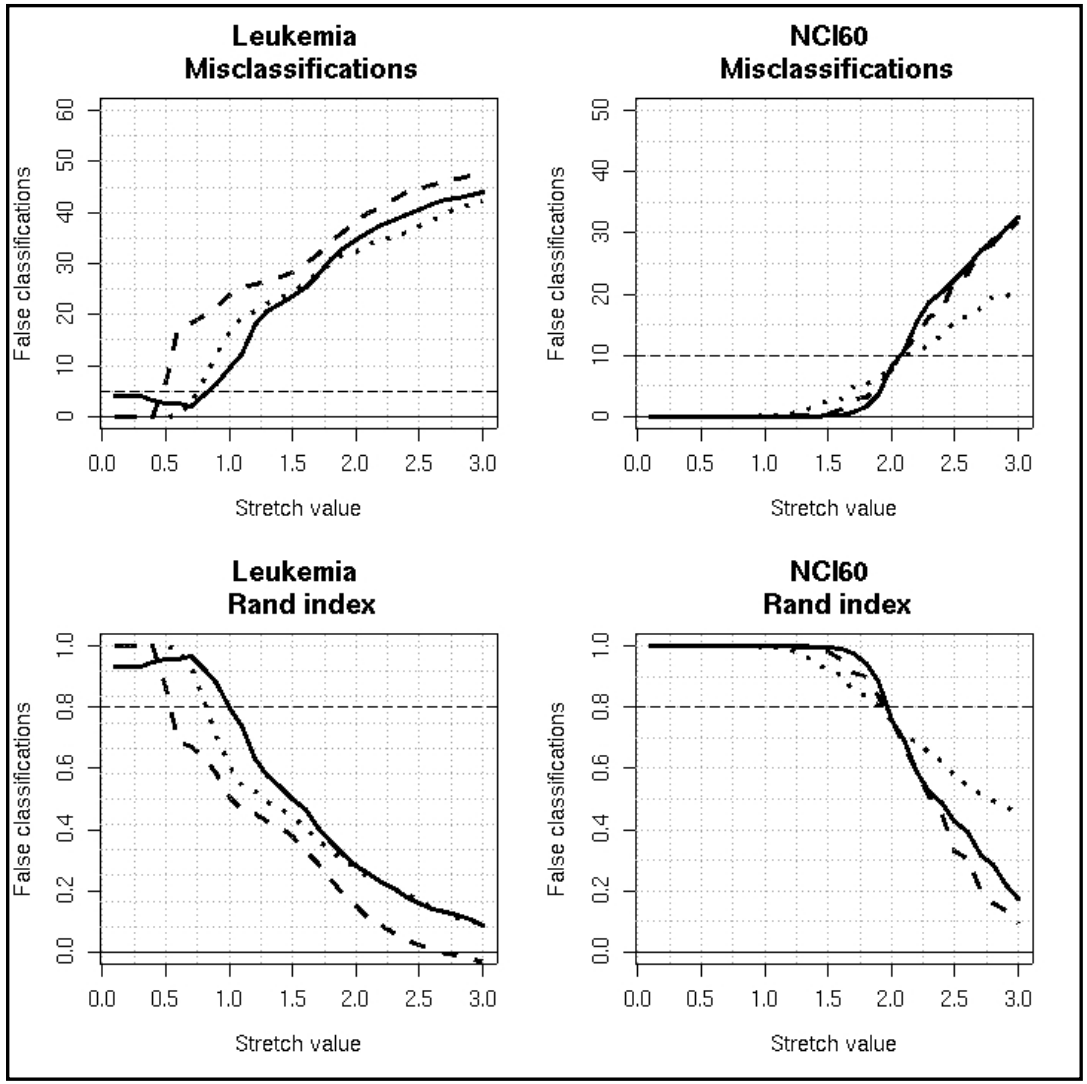




Comparative study - winners

Winners:

- —— K-means
- – – – Hierarchical, Correlation
- ..... PAM, Manhattan



Comparative study - conclusions

- **Superiority of k-means with repeated runs**
  Similar experience for discriminant analysis: FLDA is best (Dudoit et al., 2001)
- **Superiority of PAM with Manhattan distance,**
  especially for noisy data
- **BUT:** Differences are not very significant and depend on the specific dataset!
- **Preselection of genes important**

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)

Recommendations

- **Preselection of genes:**
  For clustering of samples, choose **top 100-200 genes with respect to variance across samples**.
  This decreases noise and computation time.

- **Number of clusters:**
  There is no general rule how to select the ‘correct’ number of clusters. **Try different numbers** and choose a cutoff, for which the performance of the clustering algorithm breaks down.

Recommendations

- **Interest in specific genes:**
  If you search for genes that are coregulated with a specific gene of you choice, **DO SO!**
  Don't do clustering, but create a list of genes close to your gene with respect to a distance of your choice.

- **Clustering after feature selection?**
  NO! Don't select genes first 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!

Other exploratory methods

- **PCA: Principal Component Analysis**
  Data are projected on lower dimensional space.
  Iteratively, the direction with largest variance is selected as i-th principal component (orthogonality constraint).
  Can be used as preprocessing step, but low interpretability.

- **Correspondence Analysis**
  Genes and samples are projected into two-dimensional plane to show associations between them.

- **ISIS: A class discovery method**
  Search for class distinctions that are characterized by differential expression of just a small set of genes, not by global similarity of the gene expression profile.

R commands and libraries

- **library(mva)**
  - Hierarchical clustering: ***hclust()***
  - Kmeans: ***kmeans()***
  - Principal components: ***princomp()***
- **library(cluster)**
  - PAM: ***pam()***
- **ISIS package:** http://www.molgen.mpg.de/~heydebre

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!

Much more interesting microarray analysis…

Contact: rahnenfj@mpi-sb.mpg.de

Jörg Rahnenführer
Computational Biology and Applied Algorithmics
Max Planck Institute for Informatics
D-66123 Saarbrücken, Germany
Phone: (+49) 681-9325 320

Visit us in Saarbrücken!

Saarvoir vivre…