Unsupervised learning methods for the analysis of microarray data

Jörg Rahnenführer

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

(March 2004)

Overview

- Classification tasks for microarrays
- Similarity measures
- Cluster algorithms
- Graph-theoretic algorithms and Bi-Clustering
- Other exploratory methods
- Assessment of cluster quality

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.

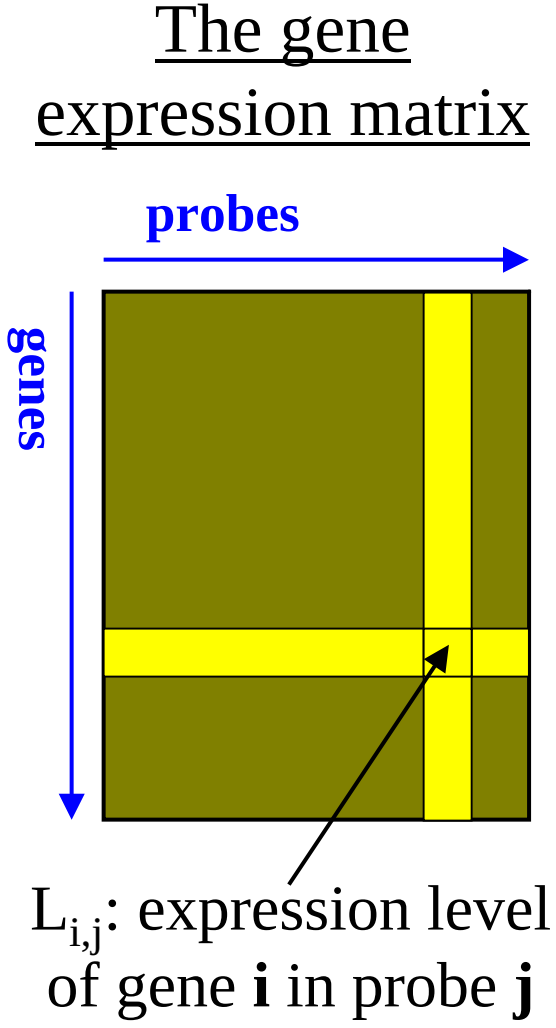
- **Classification of GENES**

  (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.

Cluster analysis

- **Clustering columns:**
  Grouping similar samples
- **Clustering rows:**
  Grouping genes with similar trajectories
- **Bi-Clustering:**
  Grouping genes that have similar partial trajectories in a subset of the samples



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.

1. **Distance measure**

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

2. **Cluster algorithm**

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

Distance measures

- **Euclidean distance:** $d(x,y) = \sqrt{\sum (x_i - y_i)^2}$
  - Two genes are close if they have similar expression values for all samples.
- **Manhattan distance:** $d(x,y) = \sum |x_i - y_i|$
  - Less sensitive to outliers than Euclidean distance.
- **Correlation distance:** $d(x,y) = 1 - \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$
  - Measures joint trend.
  - 1 and -1 correspond to perfect pos./neg. linear correlation.
  - Two genes with different means/variances can have small distance, if they jointly diverge from their averages for the same samples.

Distance measures - standardization

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 and $\hat{\sigma}$ is an estimator of the variation (e.g. standard deviation).

- After standardization, Euclidean distance and Correlation distance are equivalent: $d_E(x_1, x_2)^2 = 2n \cdot d_C(x_1, x_2)$

- Standardization makes sense if you are more interested in the direction of the effects rather than their magnitude. Can be misleading for noisy data.

Cluster algorithms

Popular cluster algorithms:

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

- K-means and SOM's take original data as vectors in a metric vector space.
- Hierarchical cluster algorithms and PAM work on the dissimilarity matrix **d** (assigns to each pair of objects $\mathbf{x}_i$ and $\mathbf{x}_j$ a distance $d(\mathbf{x}_i,\mathbf{x}_j)$).

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 similarity matrix is updated. This process is repeated until only a single cluster remains.
- Calculation of distance between 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

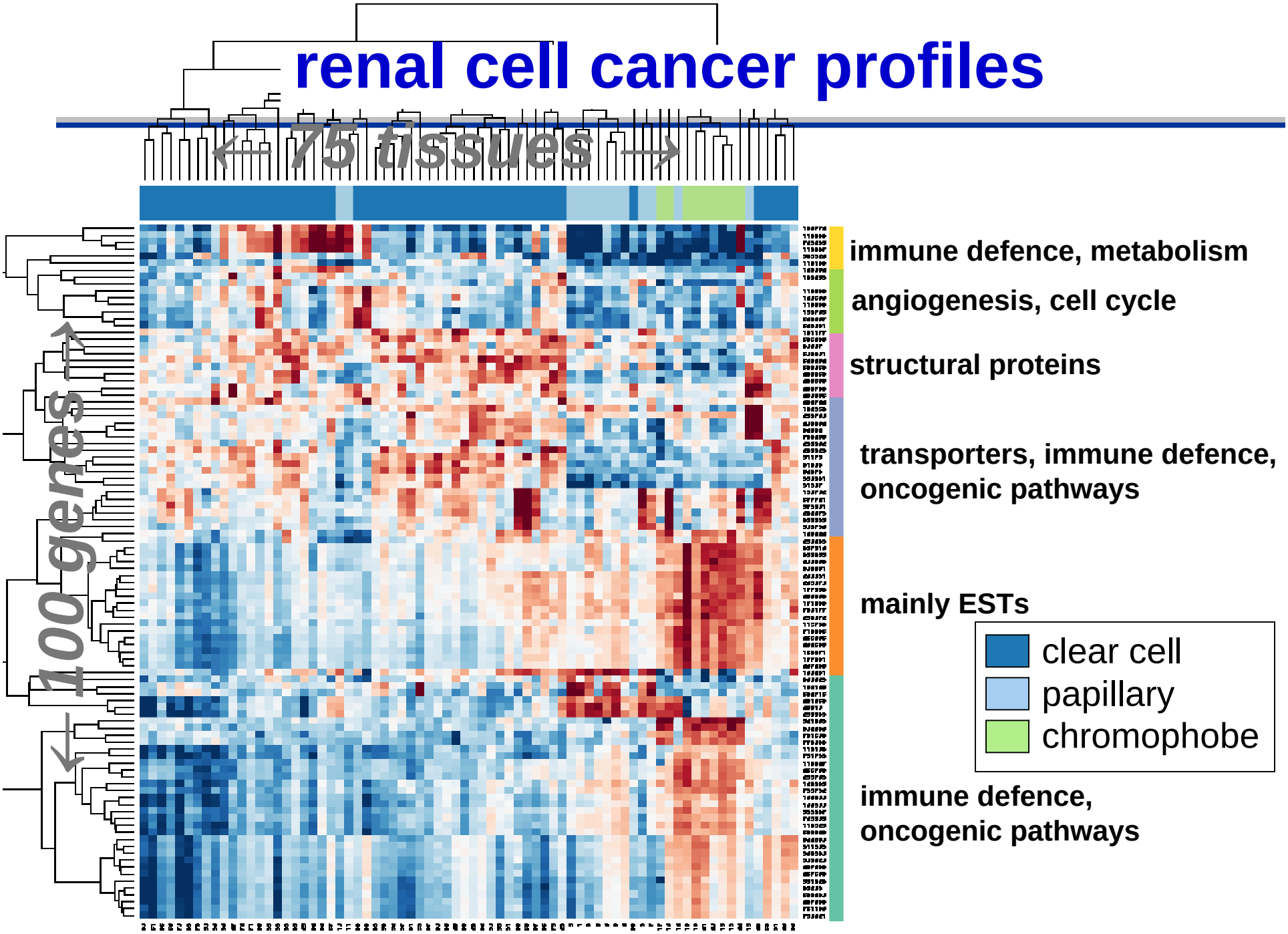
- **Visualization** of hierarchical clustering through **dendrogram**:
  - Clusters that are joined are combined by a line.
  - Height of line is average distance between clusters.
  - Cluster with smaller variation is plotted on left side.

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

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

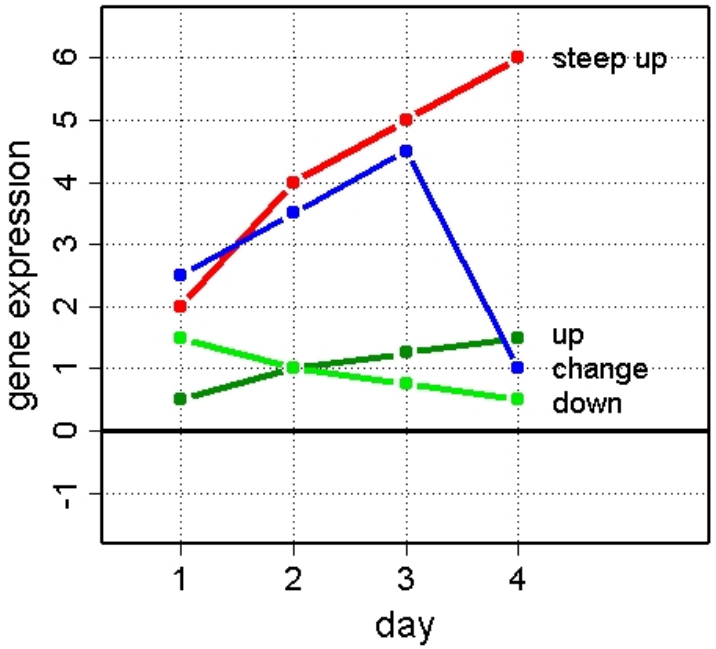
  ⟶ Hierarchical clustering does not show the full picture!

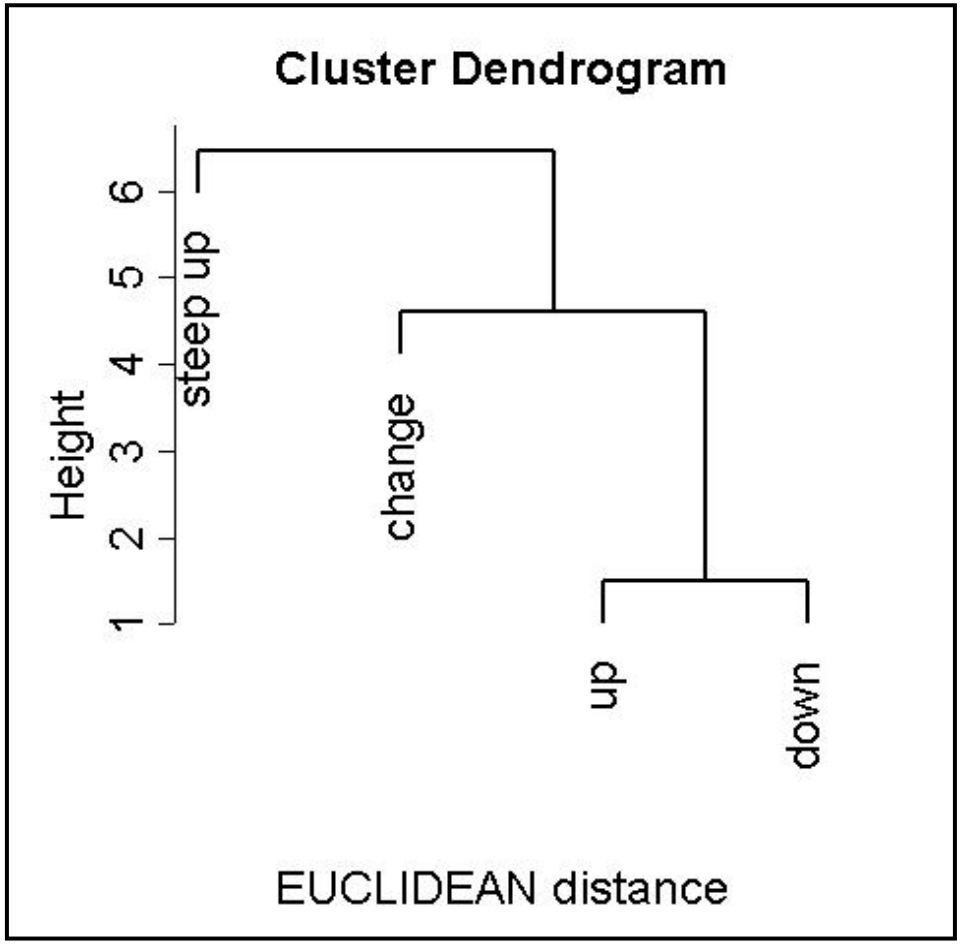
renal cell cancer profiles



Time series example

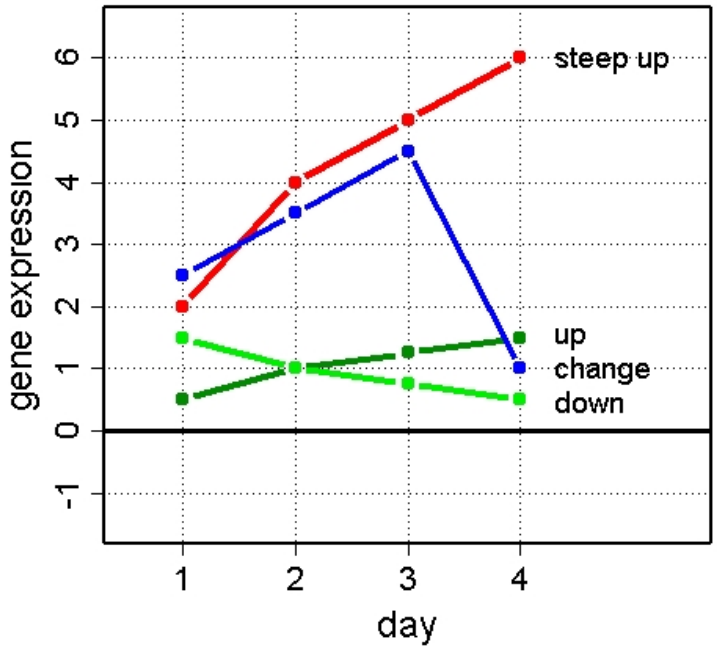
- **Euclidean distance**
  Similar values are clustered together

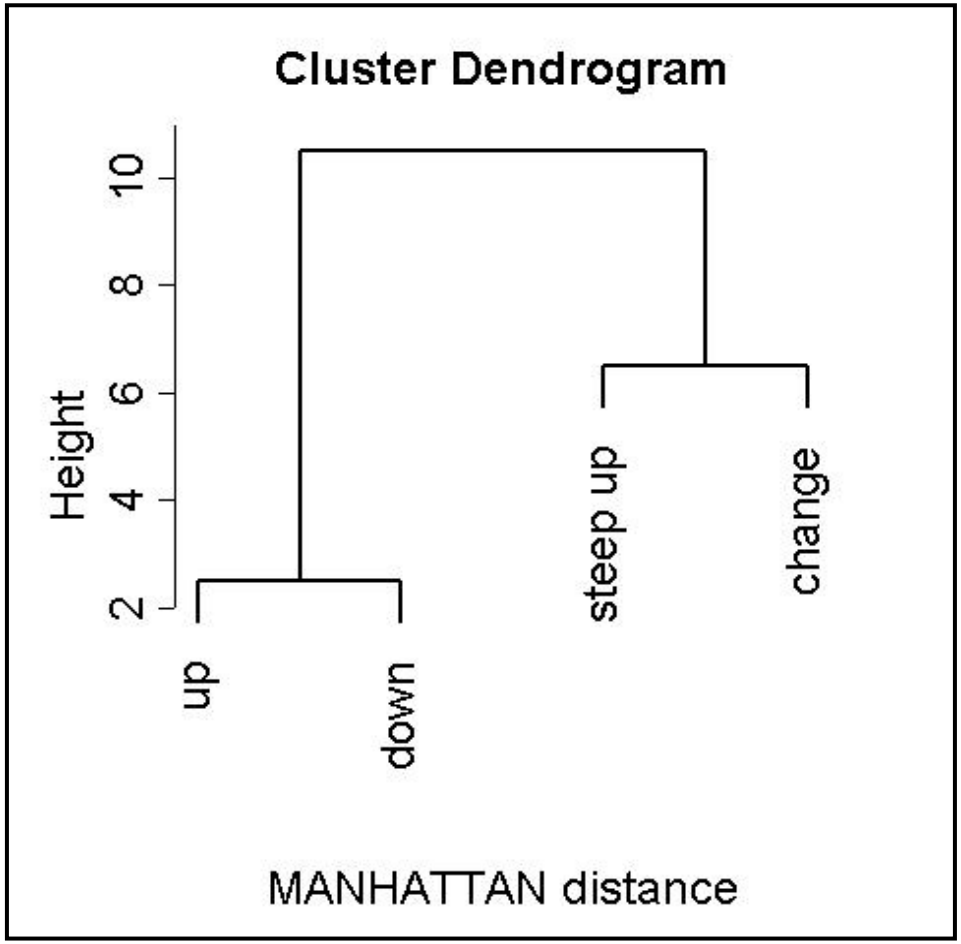




Time series example

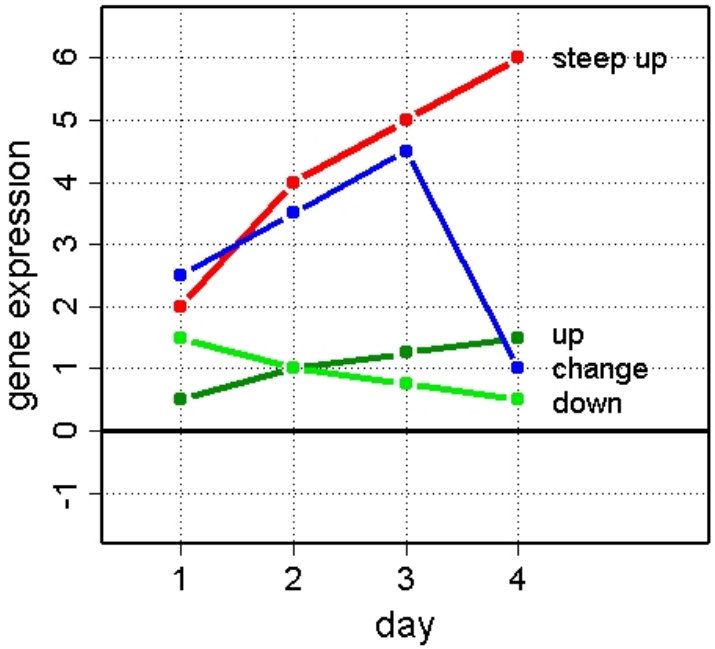
- **Manhattan distance**
  Similar values are clustered together (robust)

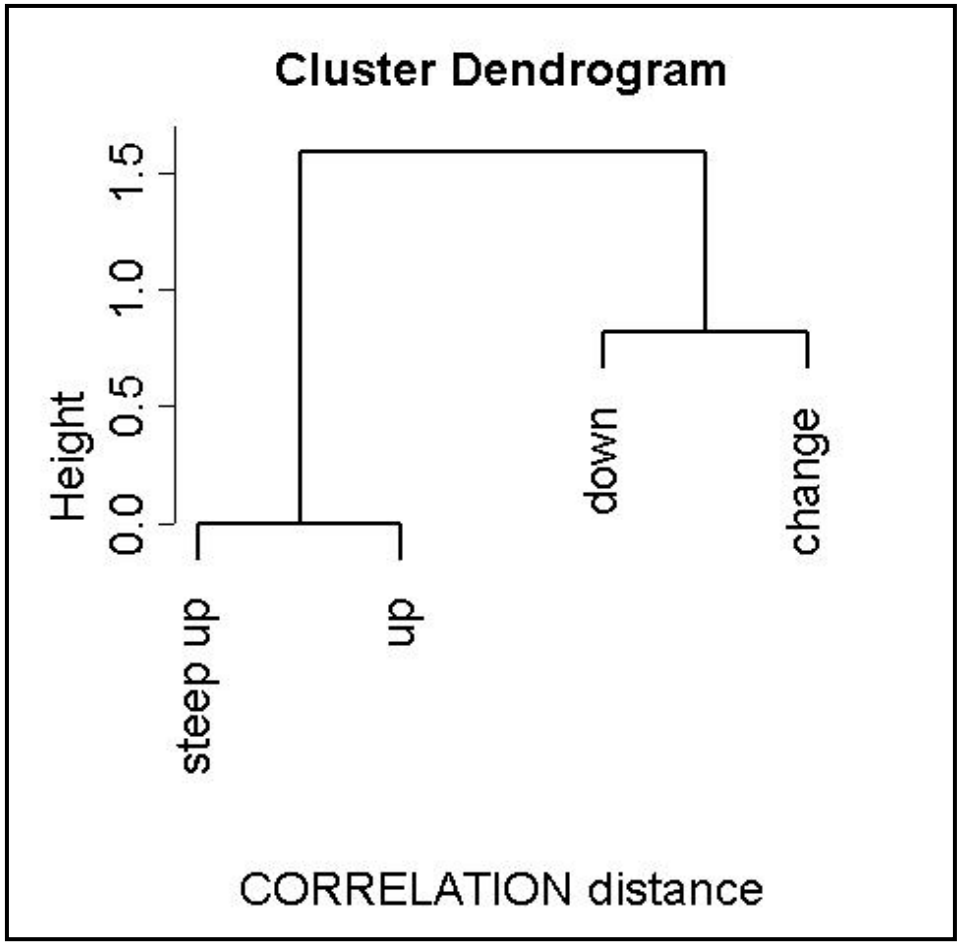




Time series example

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



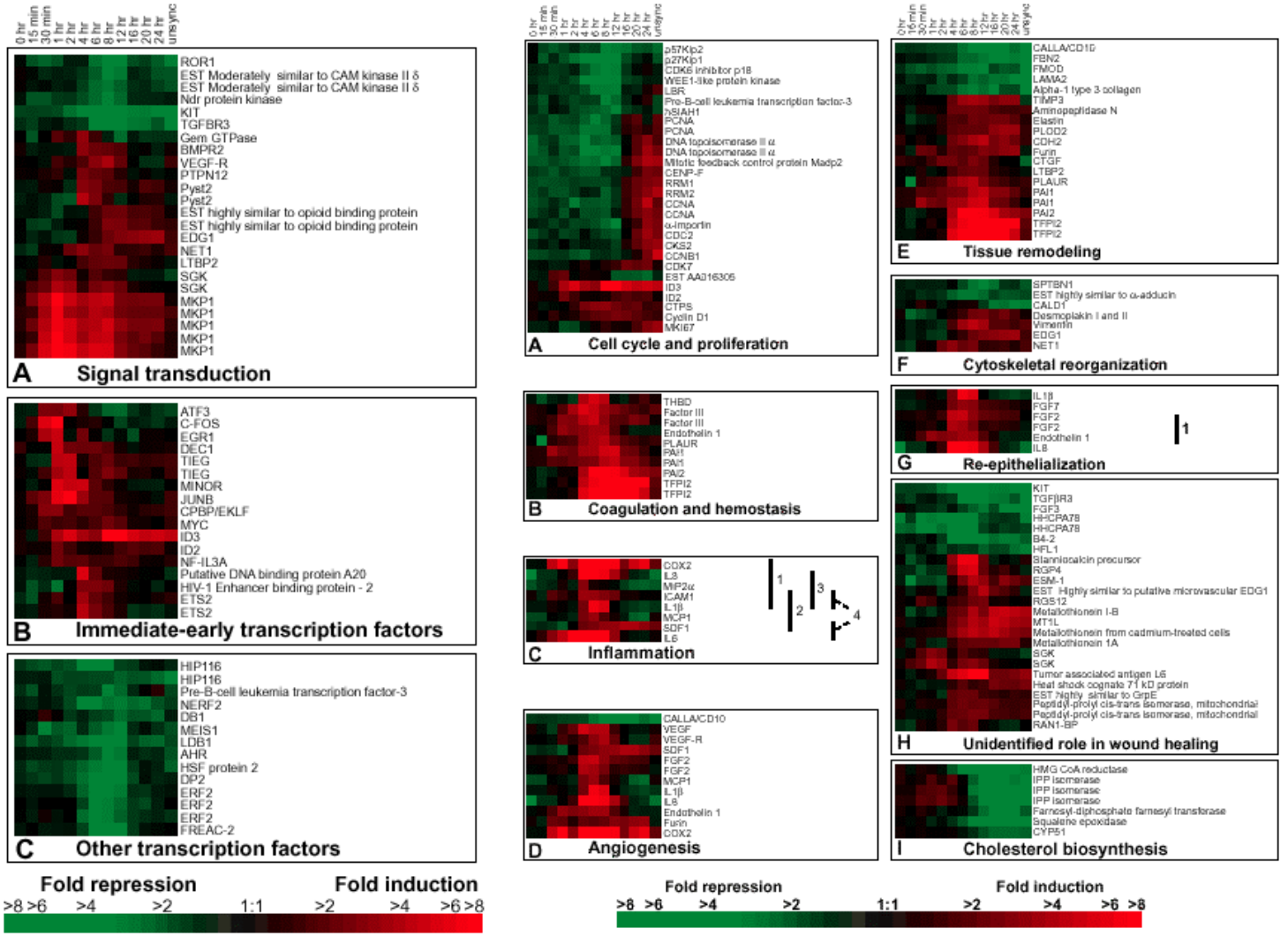


Clustering time series - literature example

Iyer et al., Science, Jan 1999:

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

Careful interpretation neccessary!

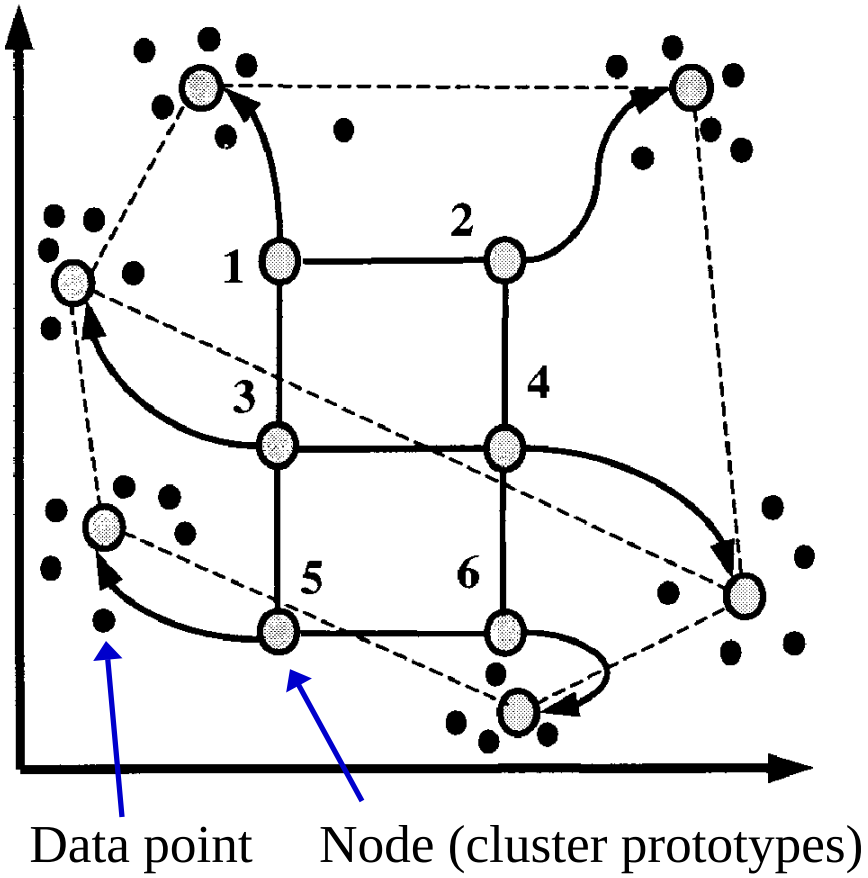


Cluster algorithms – k-means

- **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.

Graph-theoretic methods and Bi-Clustering

- **CAST (Cluster Affinity Search Technique)**

  Ben-Dor A, Shamir R, Yakhini Z (1999): **Clustering gene expression patterns.** *J. Comput Biology* 6: 281-97.

  - Input: Similarity matrix and a threshold parameter.
  - Iteratively, clusters are generated one at a time. Genes are added to an open cluster, as long as their average similarity (affinity) exceeds the threshold. Then a new cluster is started.
  - After termination of the assignment process, objects can still be added or removed from clusters.
    This improves standard hierarchical clustering.

Graph-theoretic methods and Bi-Clustering

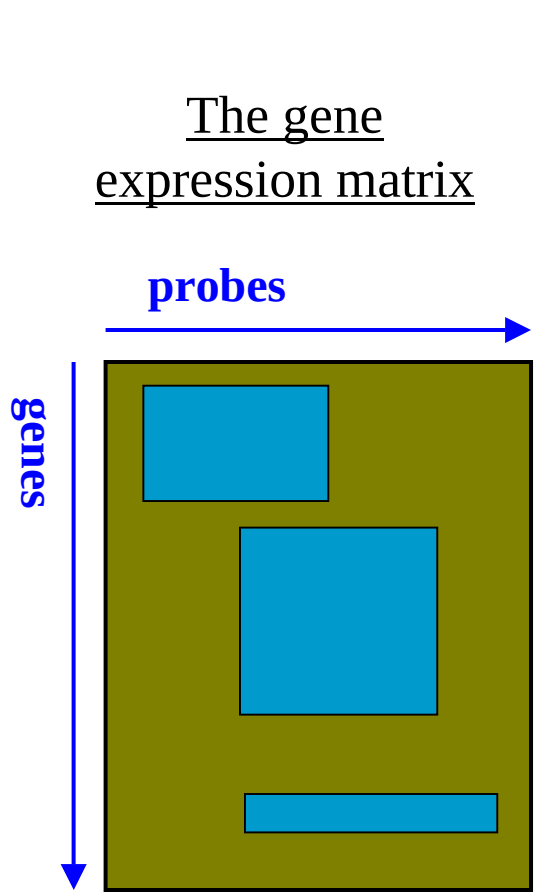
- **Bi-Clustering**

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

- Graph-theoretic algorithm coupled with statistical modeling.

- Genes and samples are both represented as nodes of a bipartite graph and are connected with weights according to the gene expression of the respective gene and sample.

- Then the heaviest subgraph is determined with an algorithm that runs in polynomial time.



Other exploratory methods

- **PCA: Principal Component Analysis**
  - Projection of data on lower dimensional space.
  - Iteratively, the direction with largest variance is selected as principal component (orthogonality constraint).
  - Can be used as preprocessing step, but low interpretability.
- **“Gene shaving” (Hastie et al. 2000)**
  - Goal: Find several small and possibly overlapping groups of genes with small between-gene variance and large between-sample variance.
  - Clusters are generated iteratively based on the principal components.
  - Repeatedly, a fraction of genes having lowest (absolute) inner product with the principal component is discarded from the potential clusters.

Cluster validity – external indices

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

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):**

  Prediction-based resampling method *Clest* to estimate the number of clusters in a dataset.

- **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

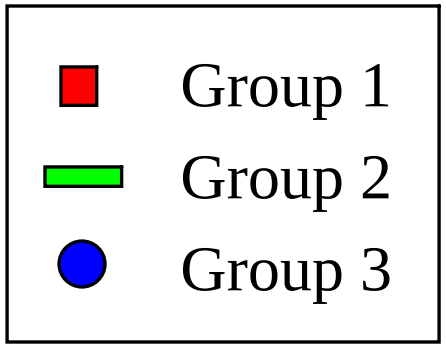
- **Data sets**:
  - Golub et al: Leukemia dataset, *http://www.genome.wi.mit.edu/MPR*, 3 cancer classes: 25 acute myeloid leukemia (AML) and 47 acute lymphoblastic leukemia (ALL) (9 T-cell and 38 B-cell), Affymetrix.
  - Ross et al.: NCI60 cancer dataset, *http://genome-www.stanford.edu/nci60*, 9 cancer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

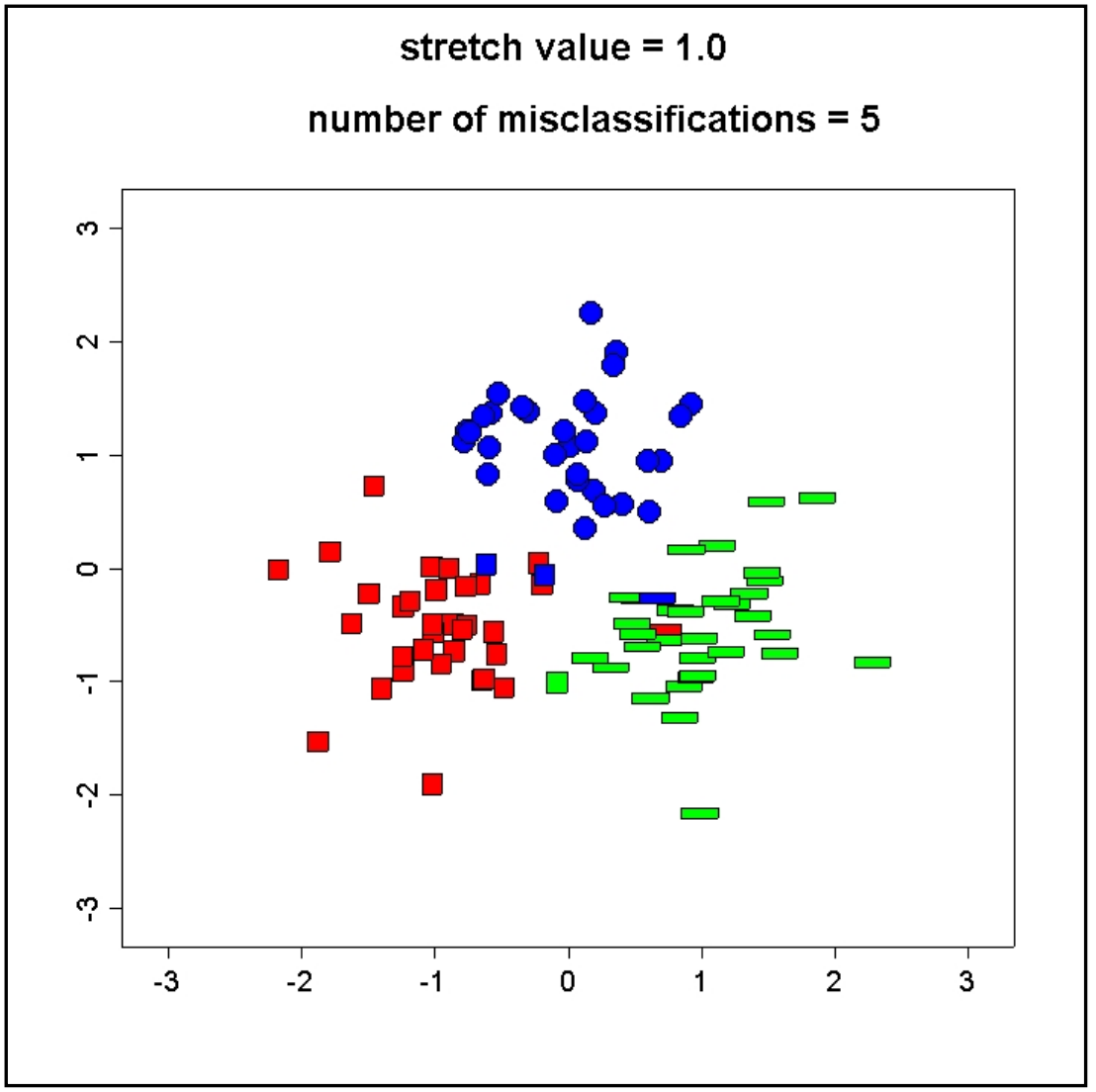
- Rahnenführer (2002): **Efficient clustering methods for tumor classification with gene expression arrays,** *Proceedings of '26th Annual Conference of the Gesellschaft für Klassifikation'*, Mannheim, July 2002.

Comparative study - method

Color → Group

Shape → Cluster

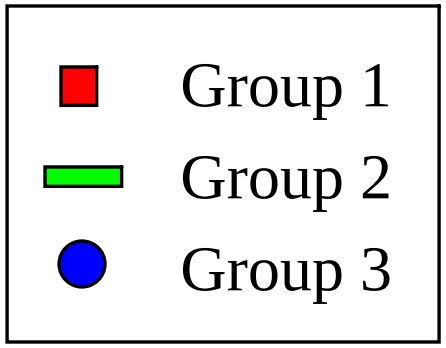


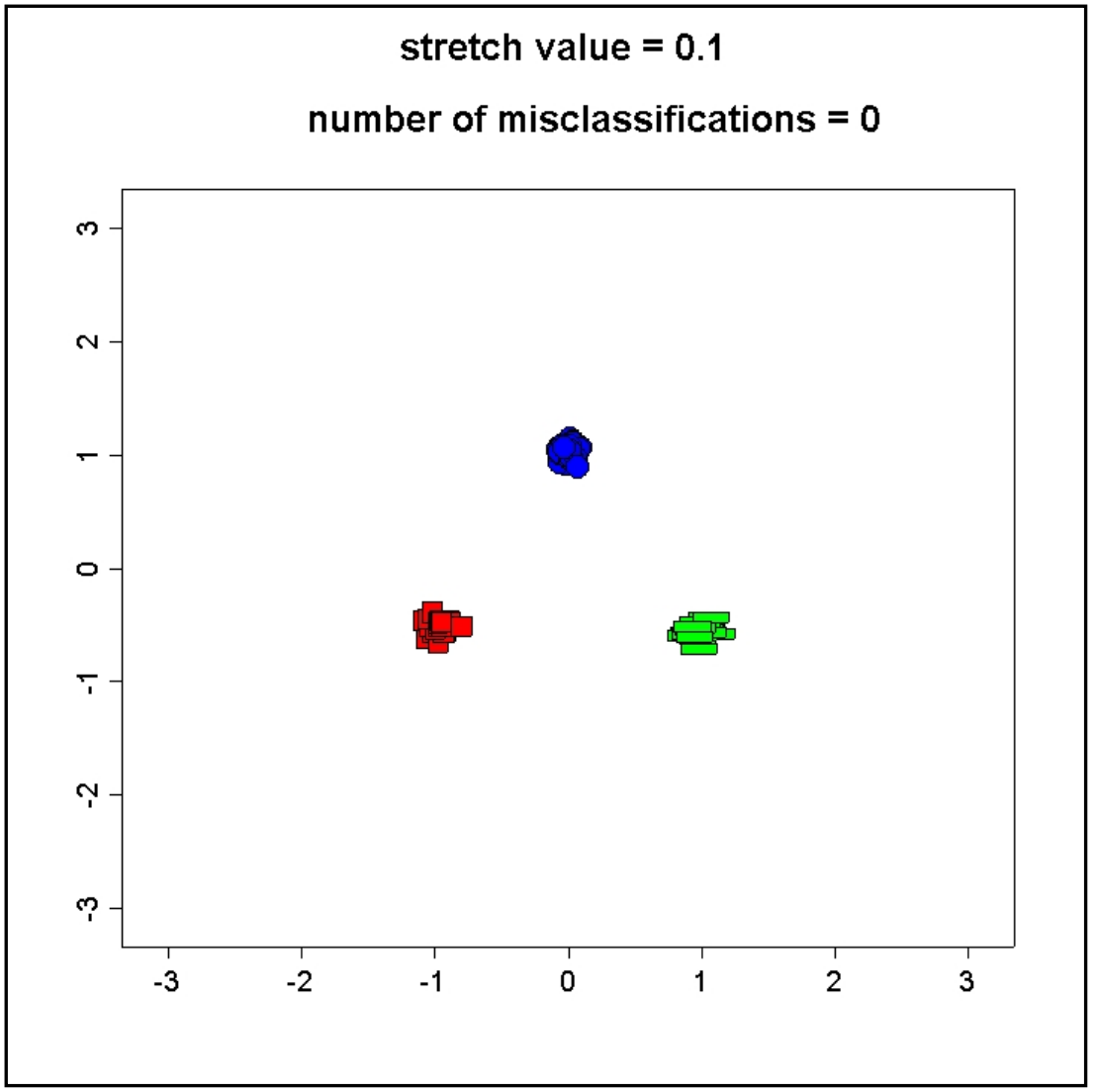


Comparative study - method

Color → Group

Shape → Cluster

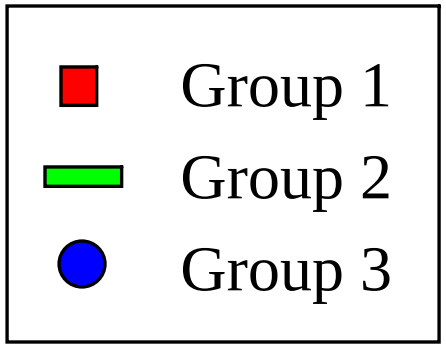


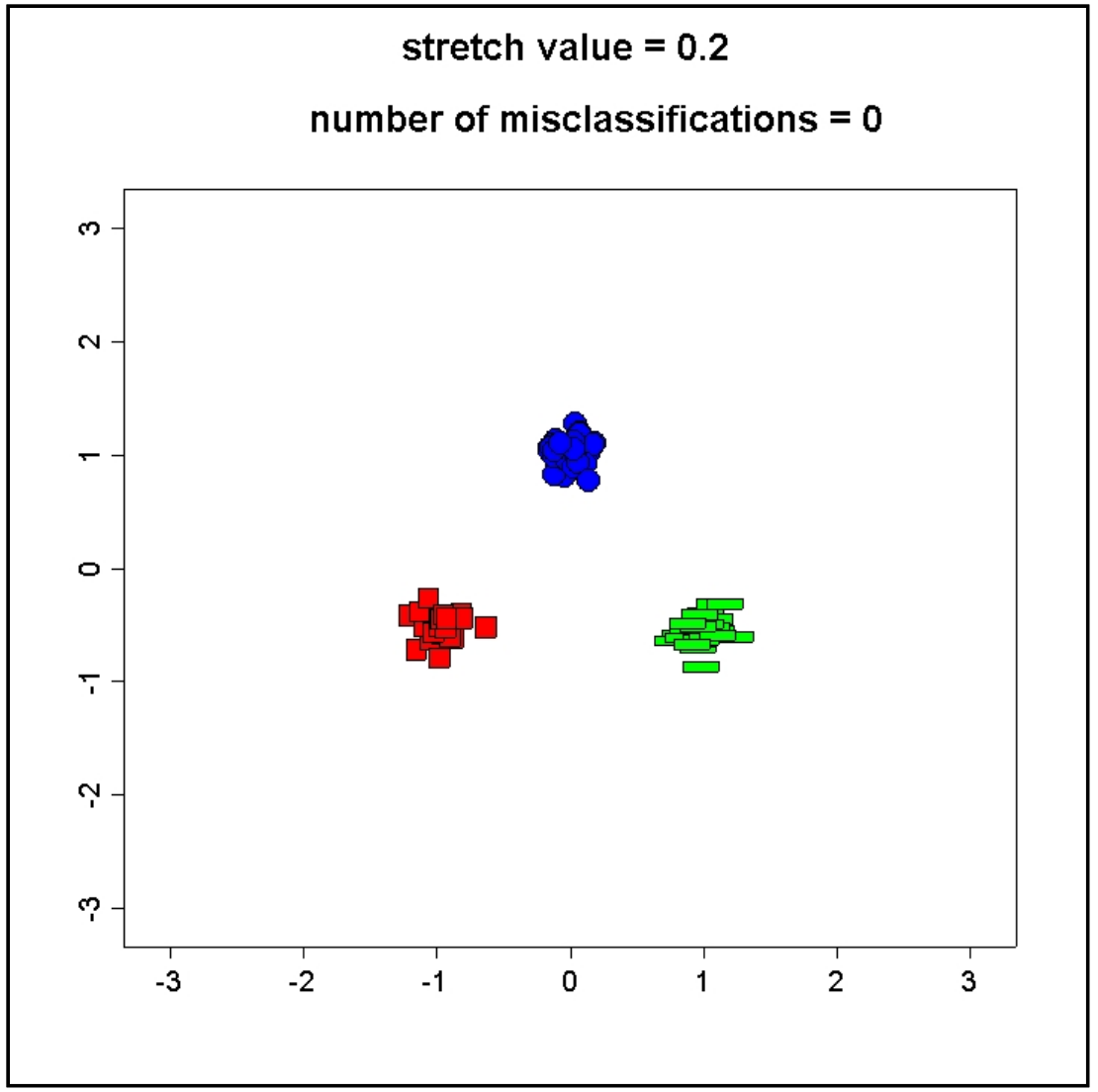


Comparative study - method

Color → Group

Shape → Cluster

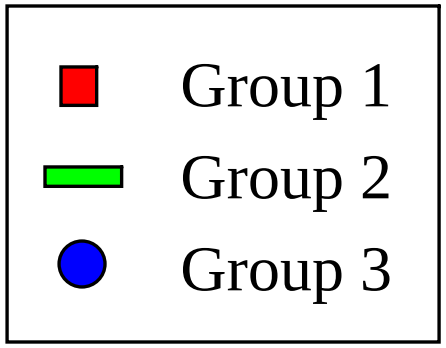


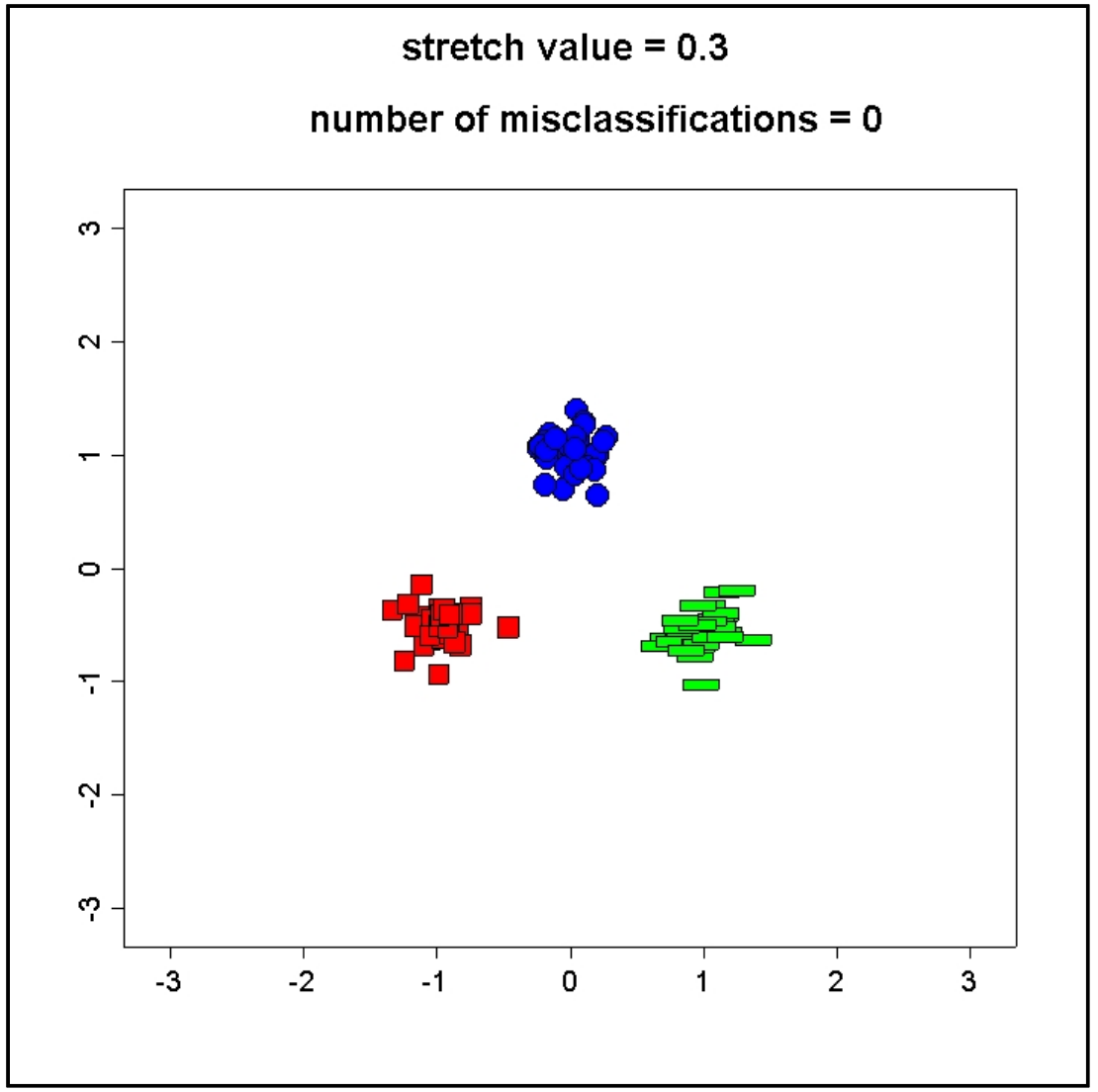


Comparative study - method

Color → Group

Shape → Cluster

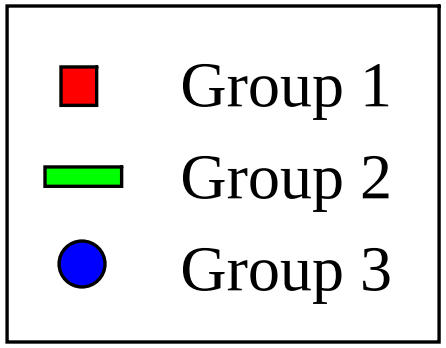


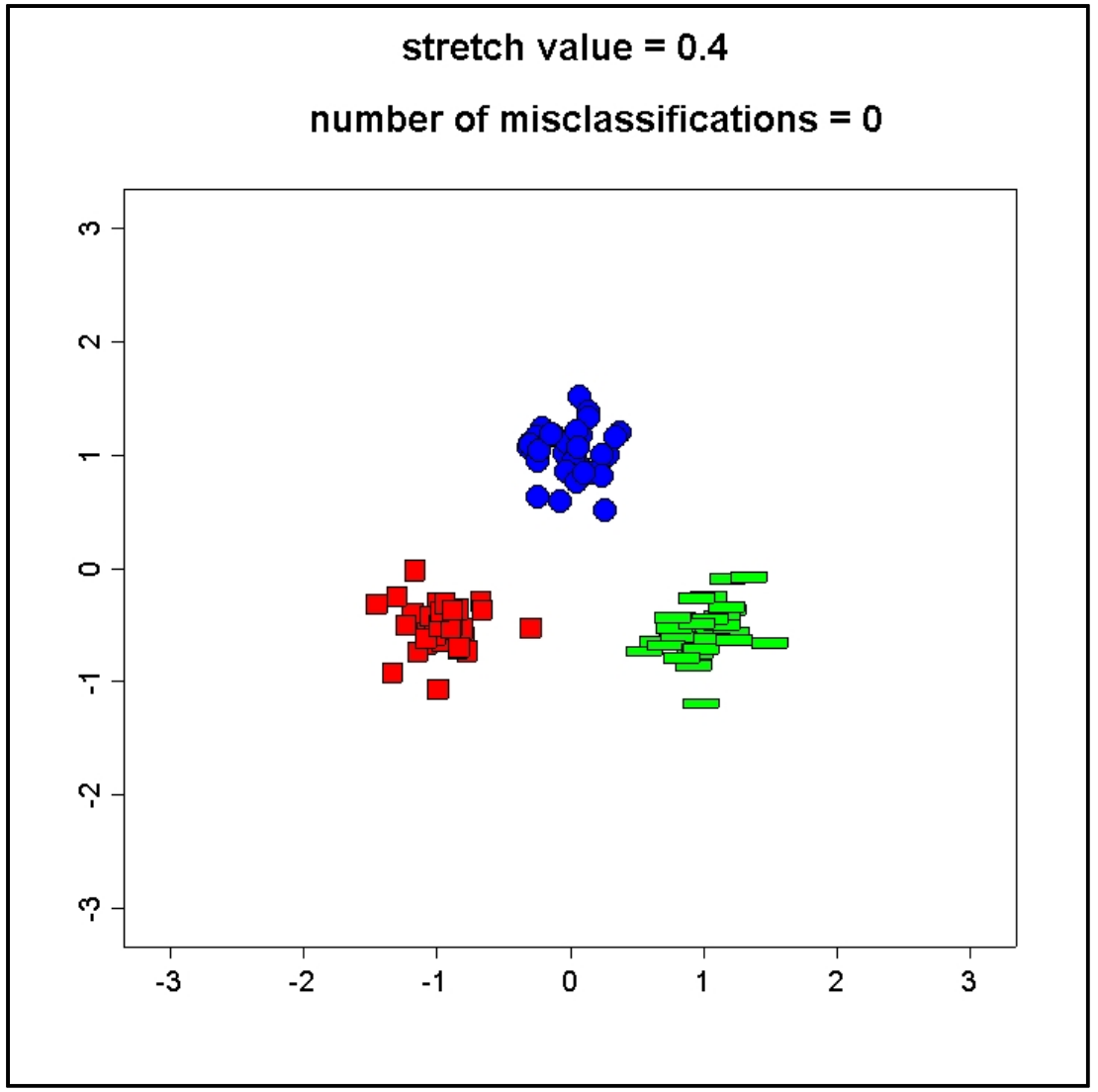


Comparative study - method

Color → Group

Shape → Cluster

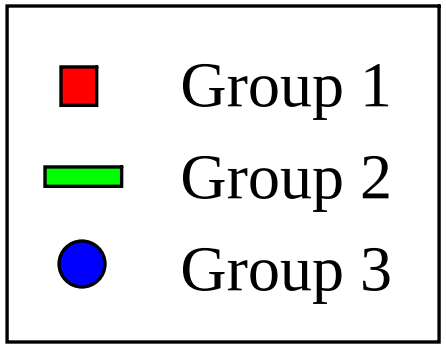


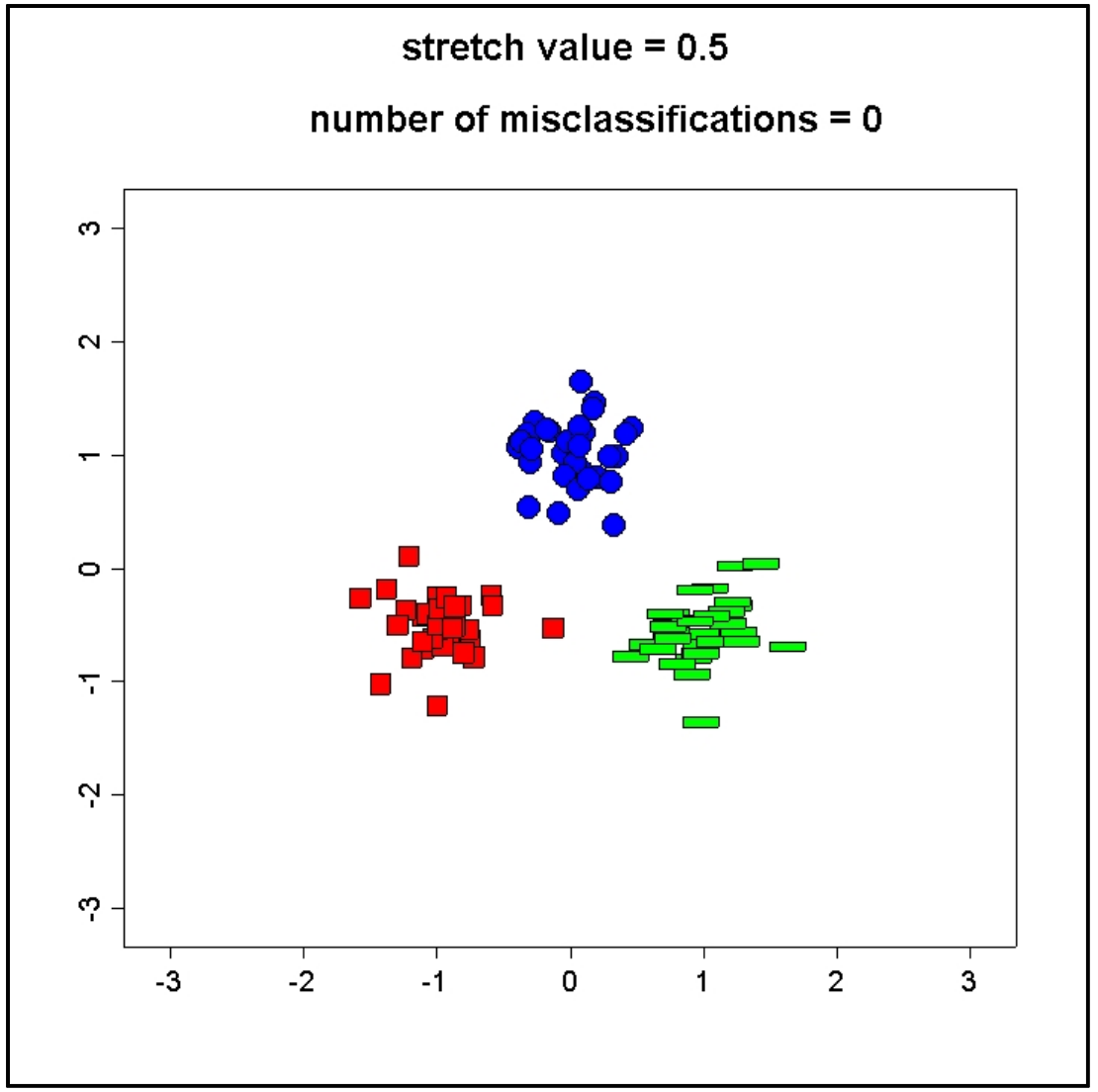


Comparative study - method

Color → Group

Shape → Cluster

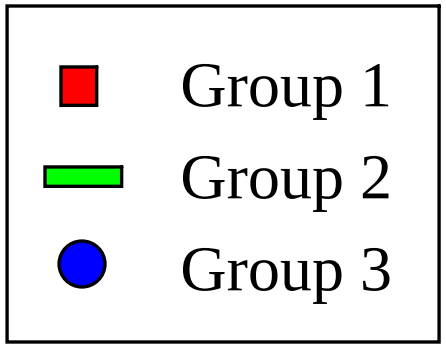


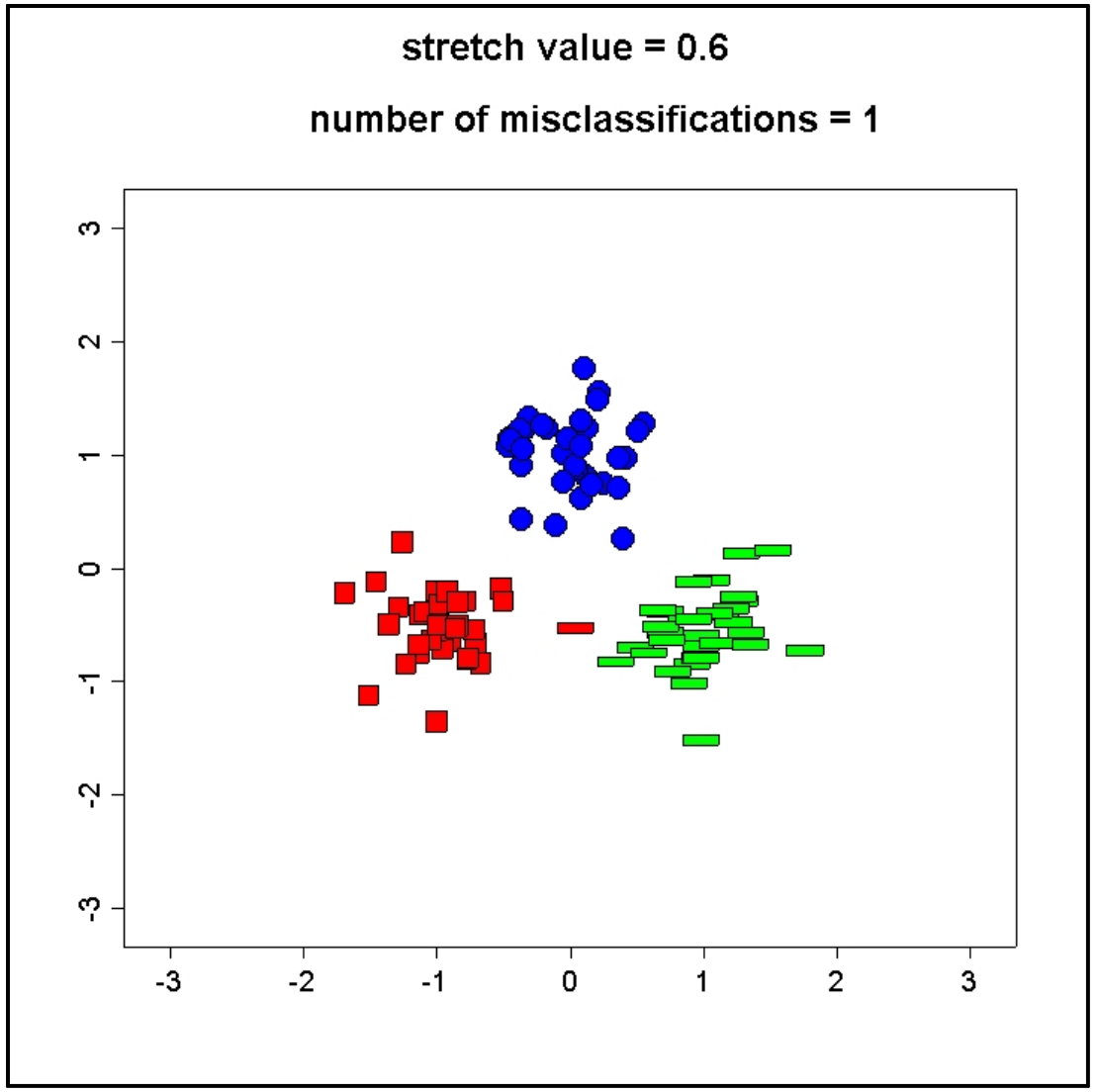


Comparative study - method

Color → Group

Shape → Cluster

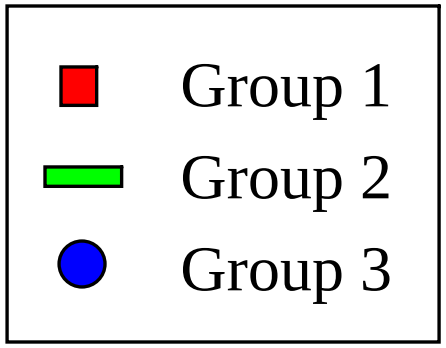


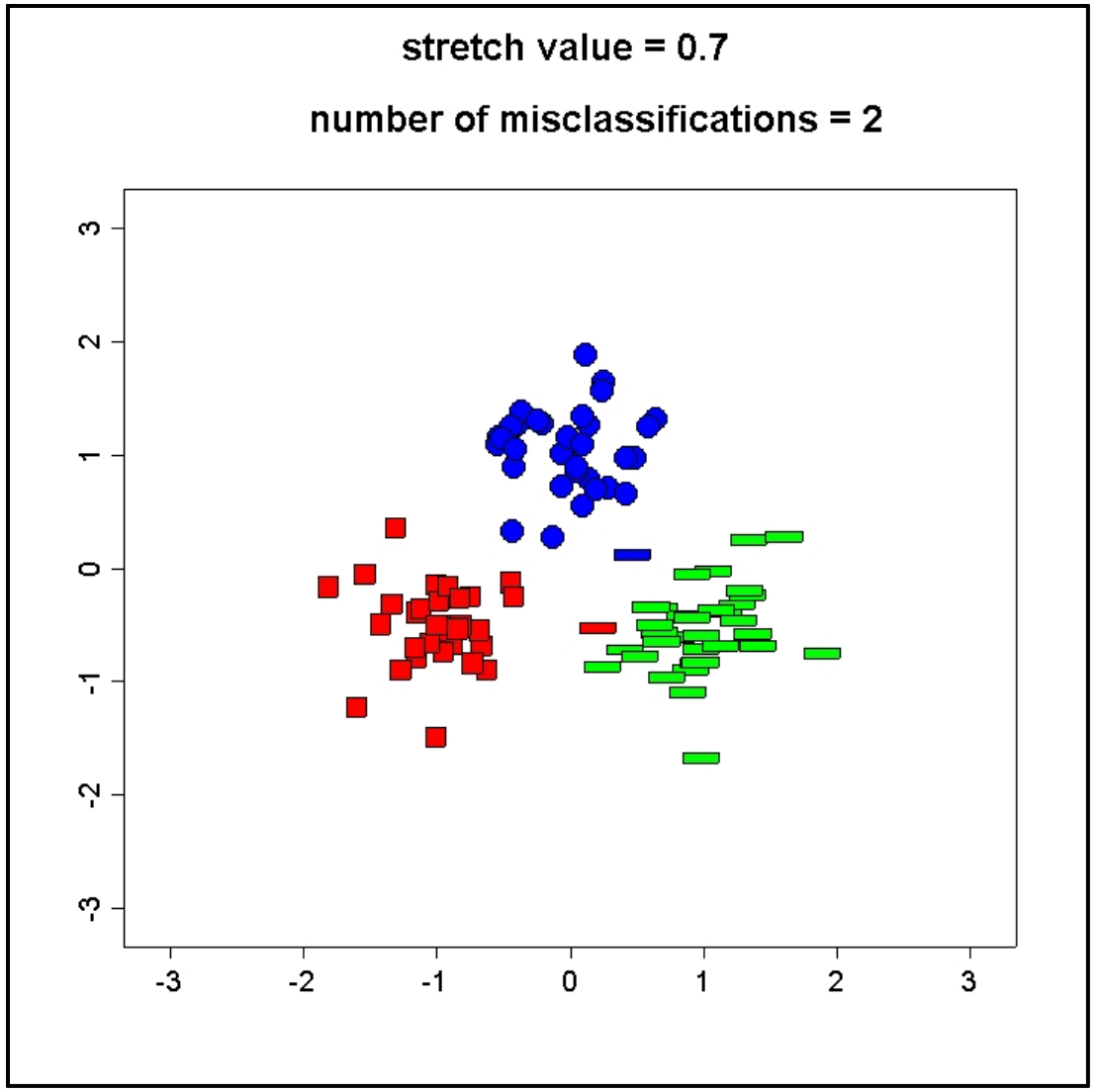


Comparative study - method

Color → Group

Shape → Cluster

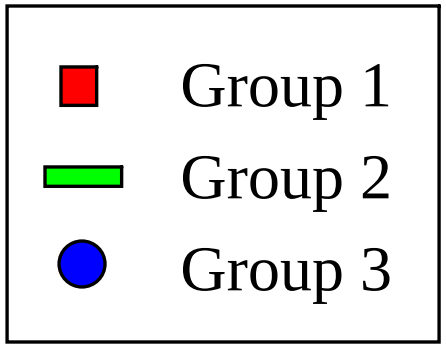


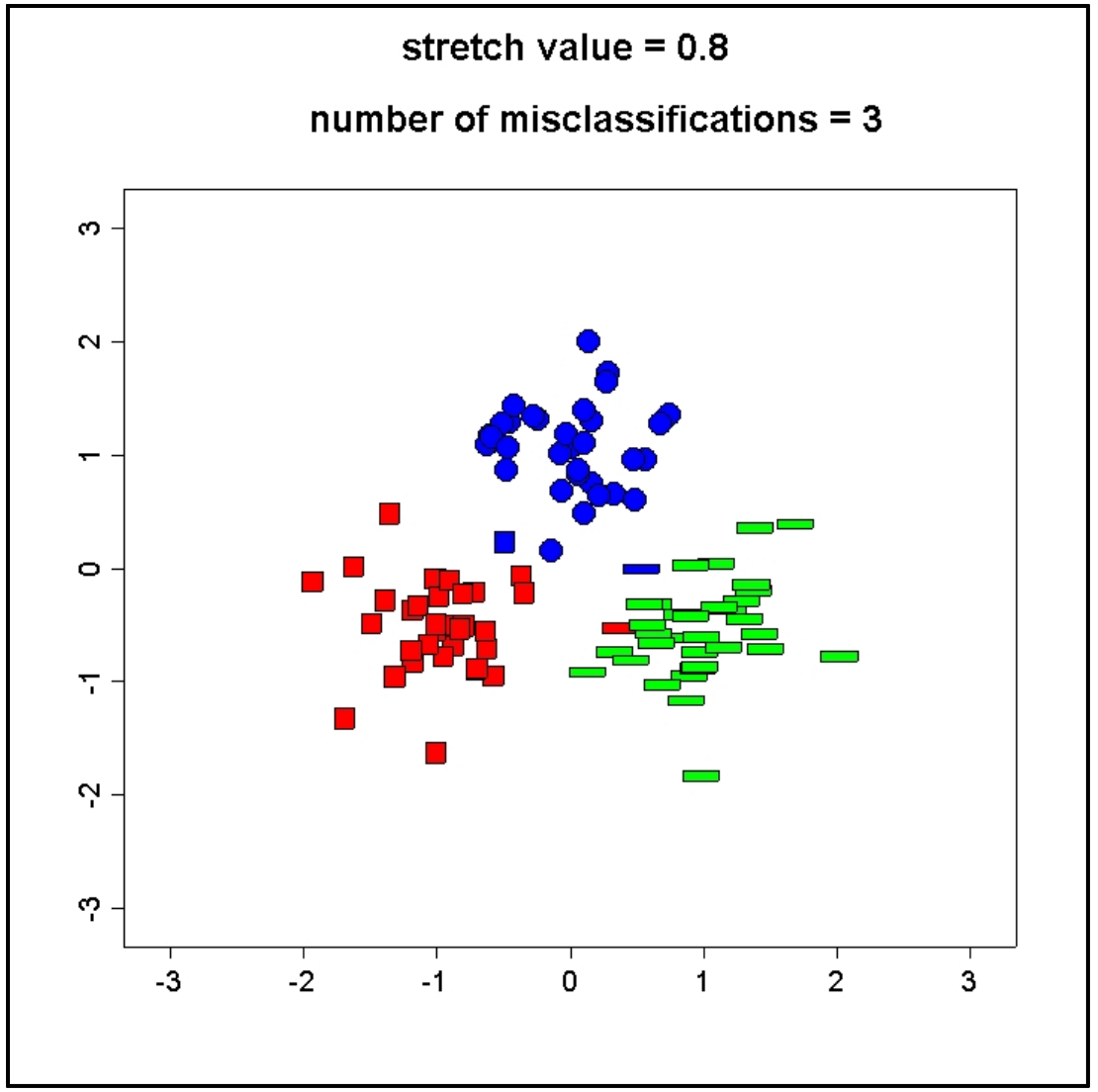


Comparative study - method

Color → Group

Shape → Cluster

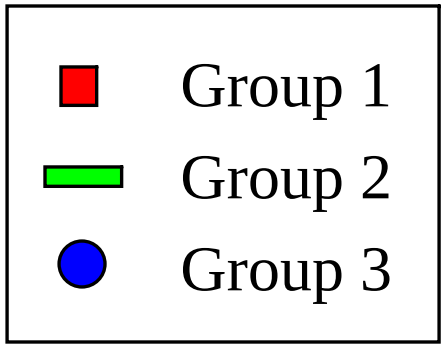


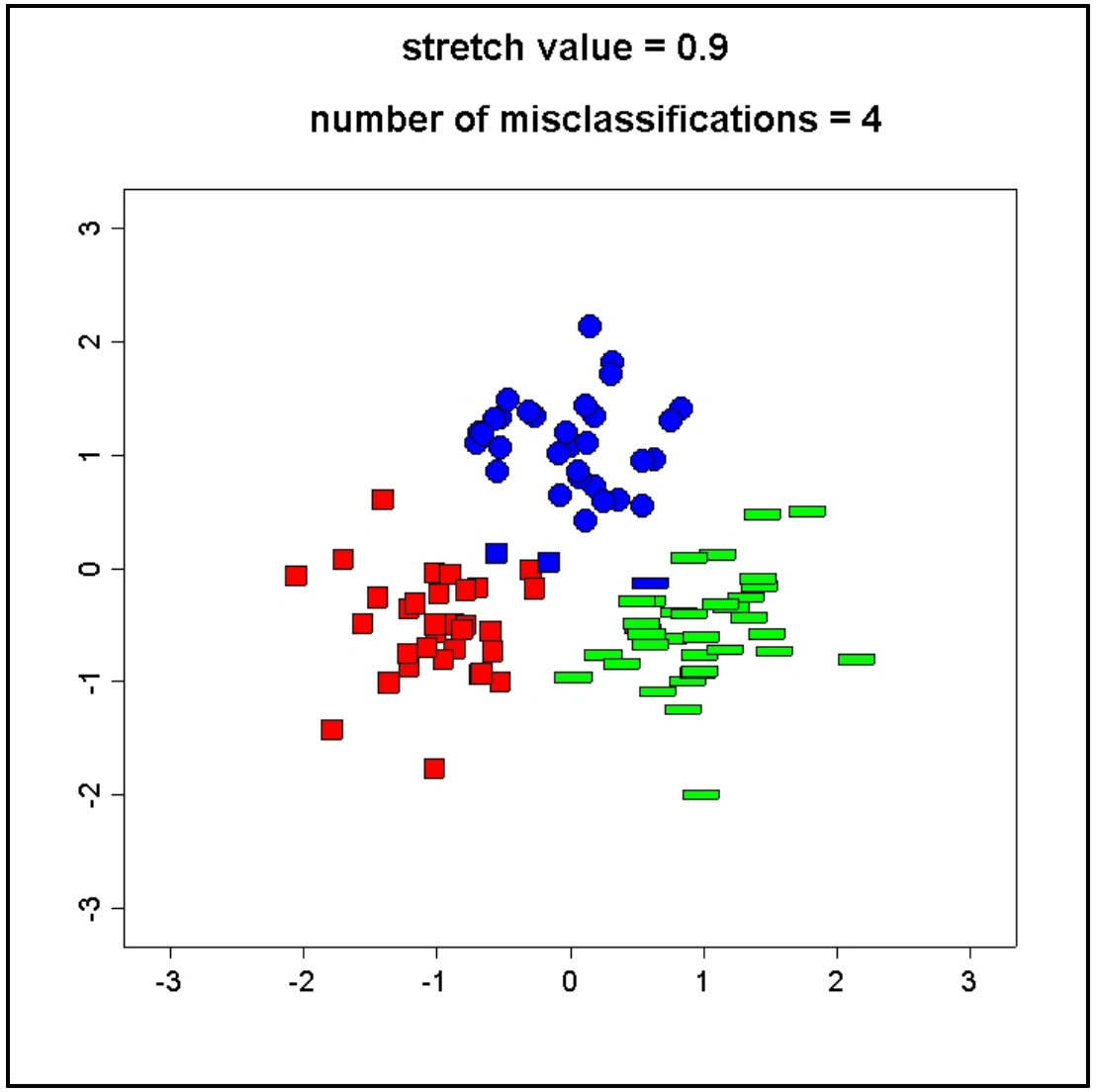


Comparative study - method

Color → Group

Shape → Cluster

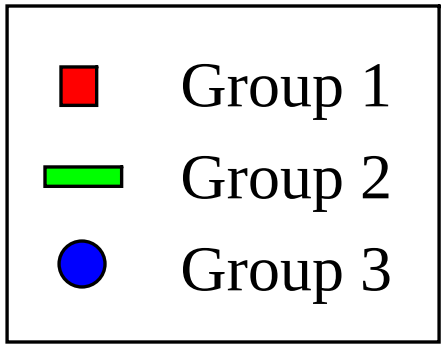


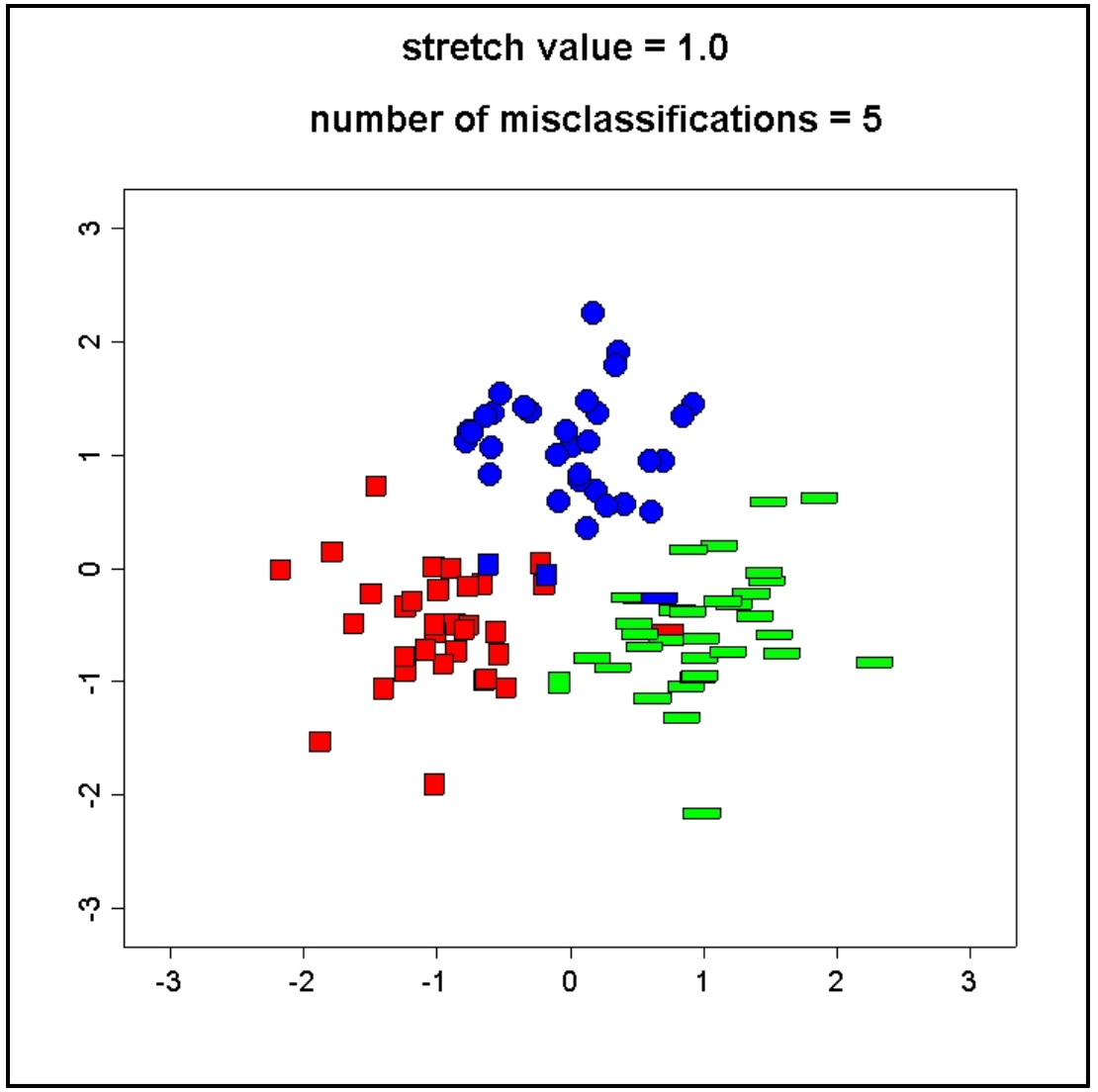


Comparative study - method

Color → Group

Shape → Cluster

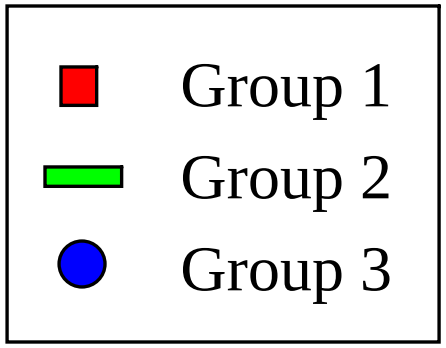


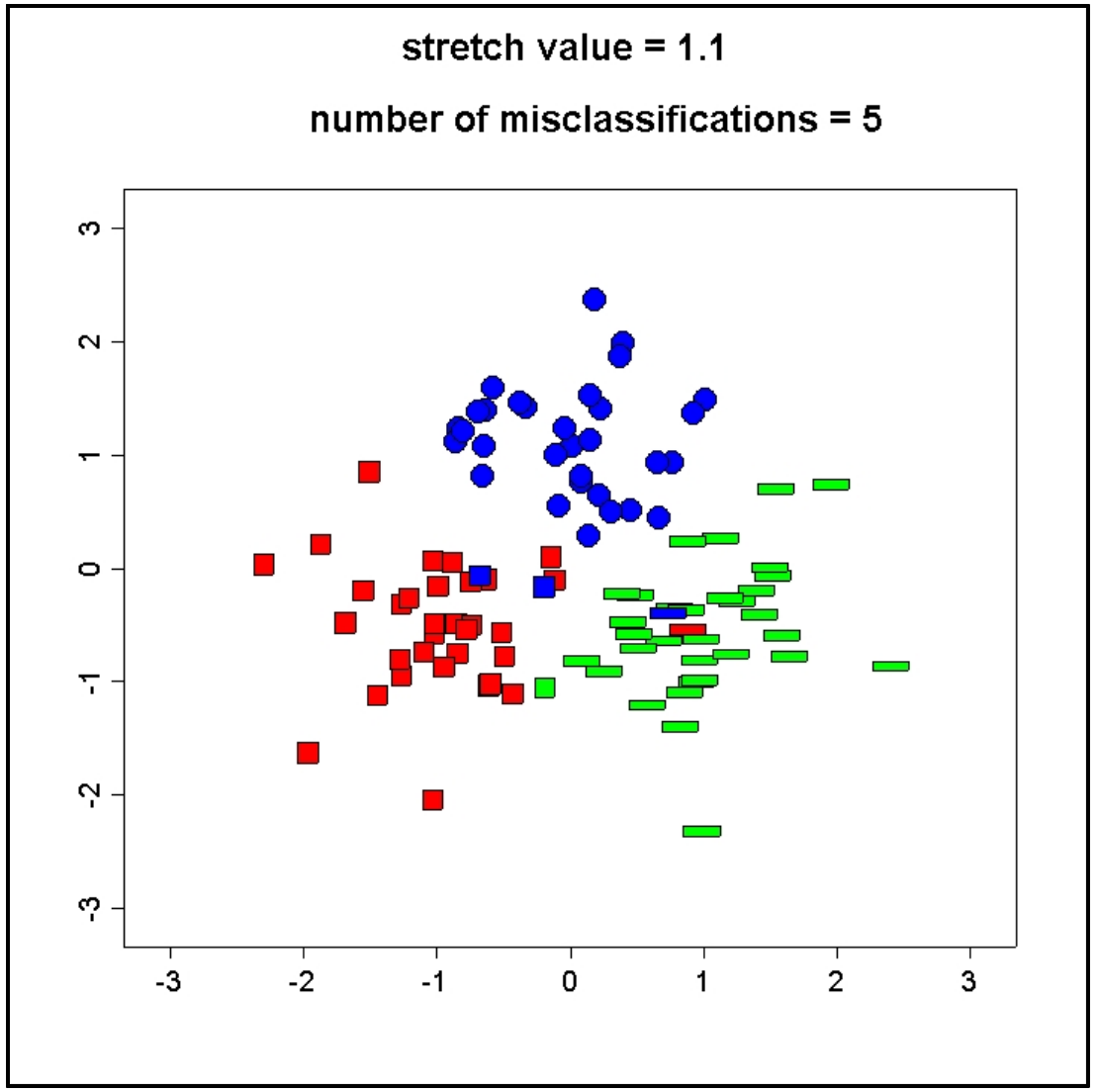


Comparative study - method

Color → Group

Shape → Cluster

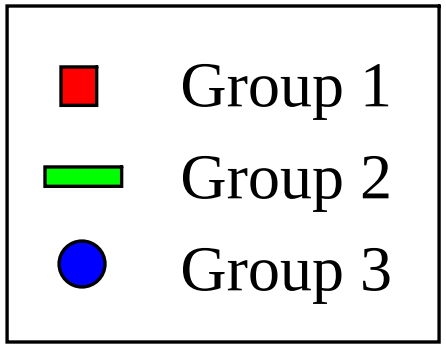


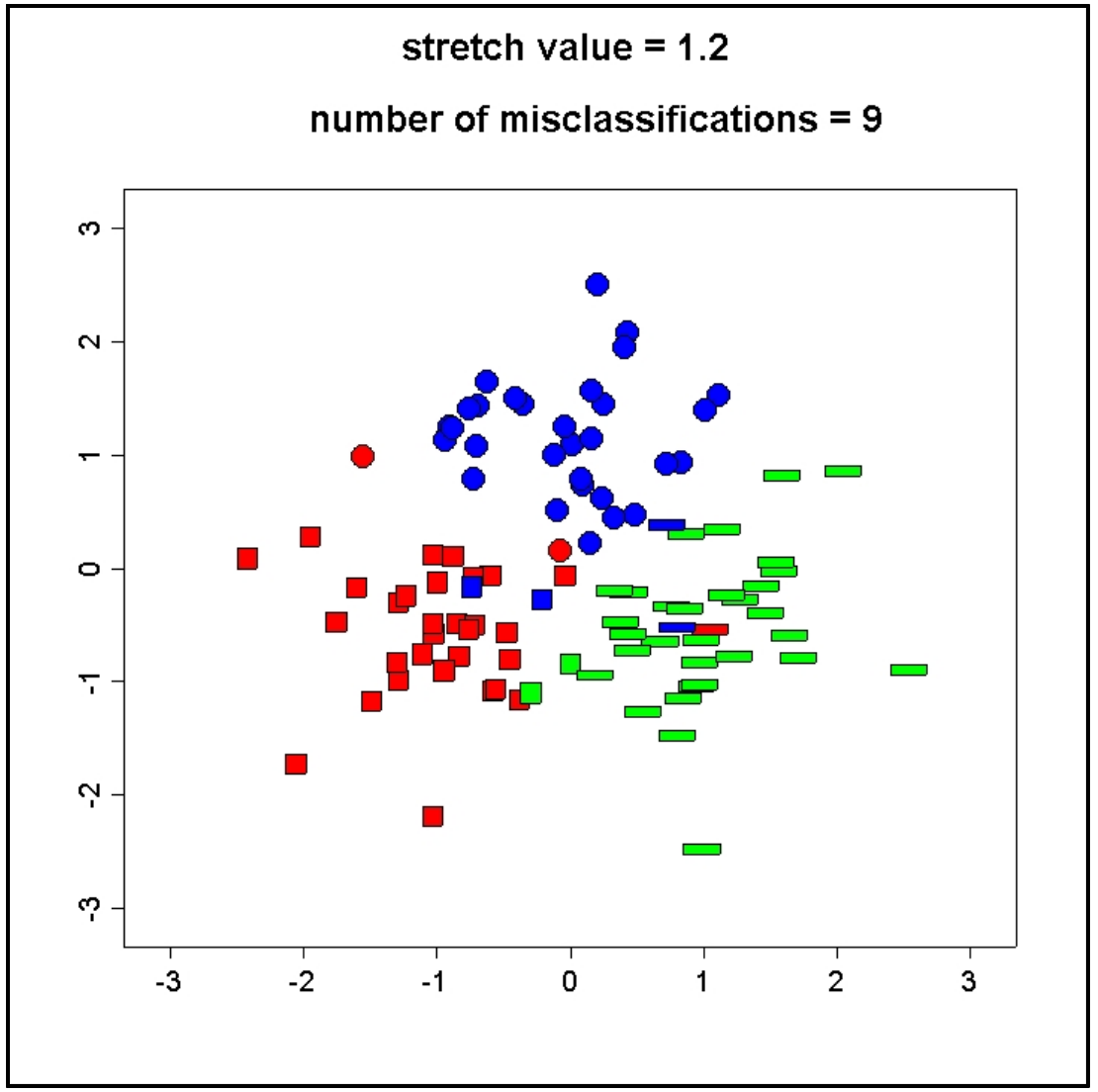


Comparative study - method

Color → Group

Shape → Cluster

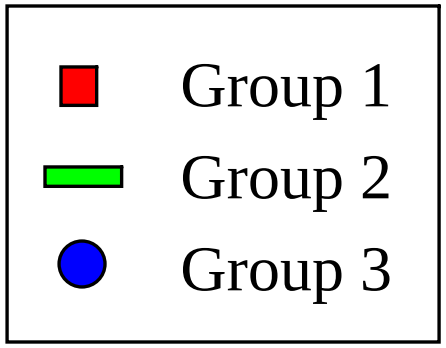


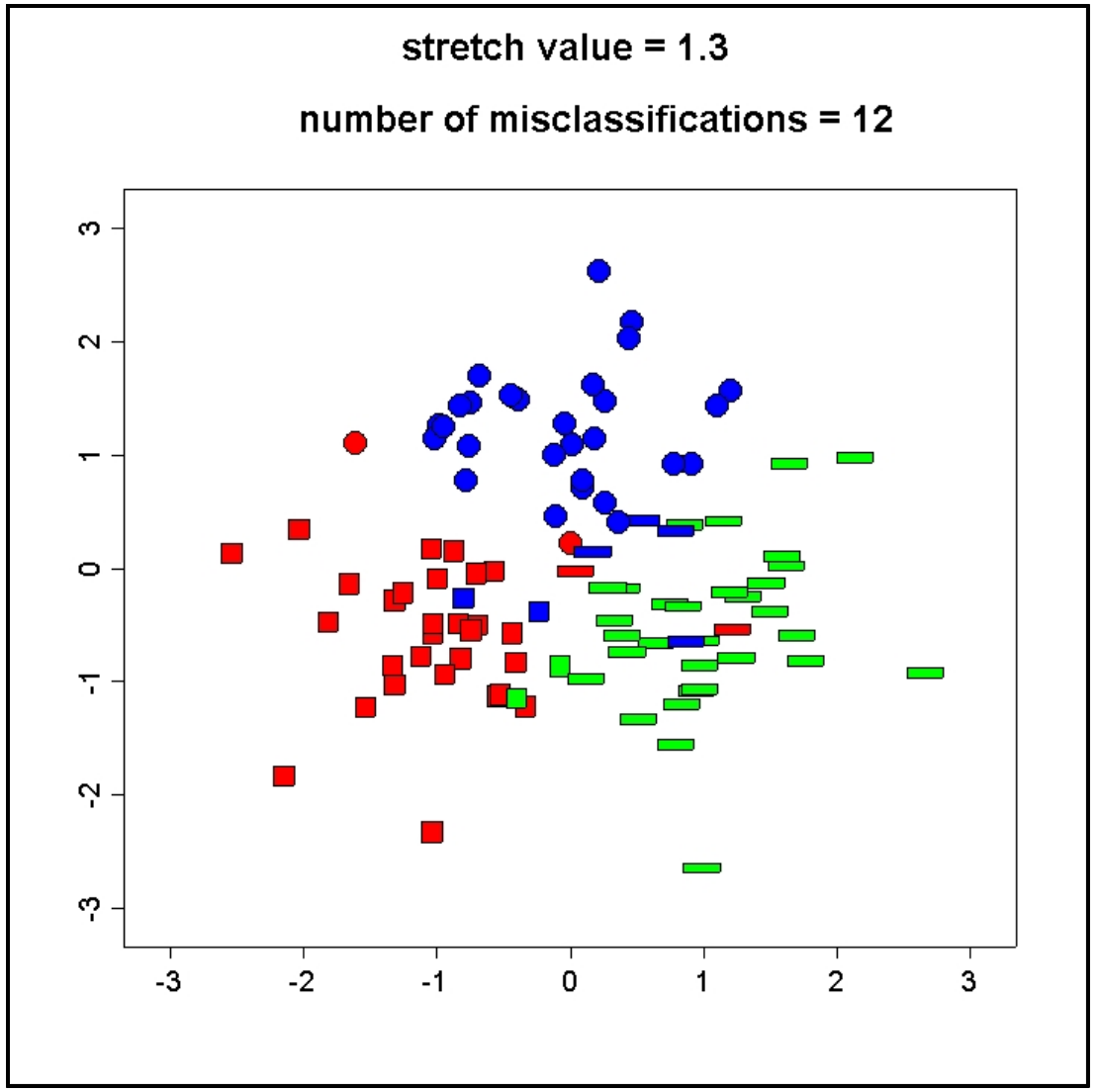


Comparative study - method

Color → Group

Shape → Cluster

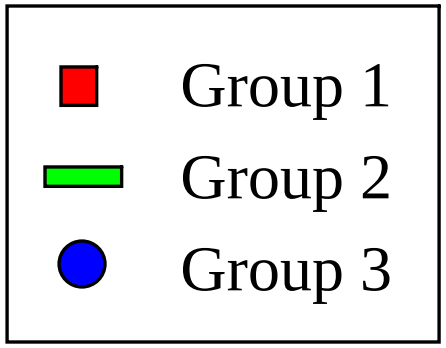


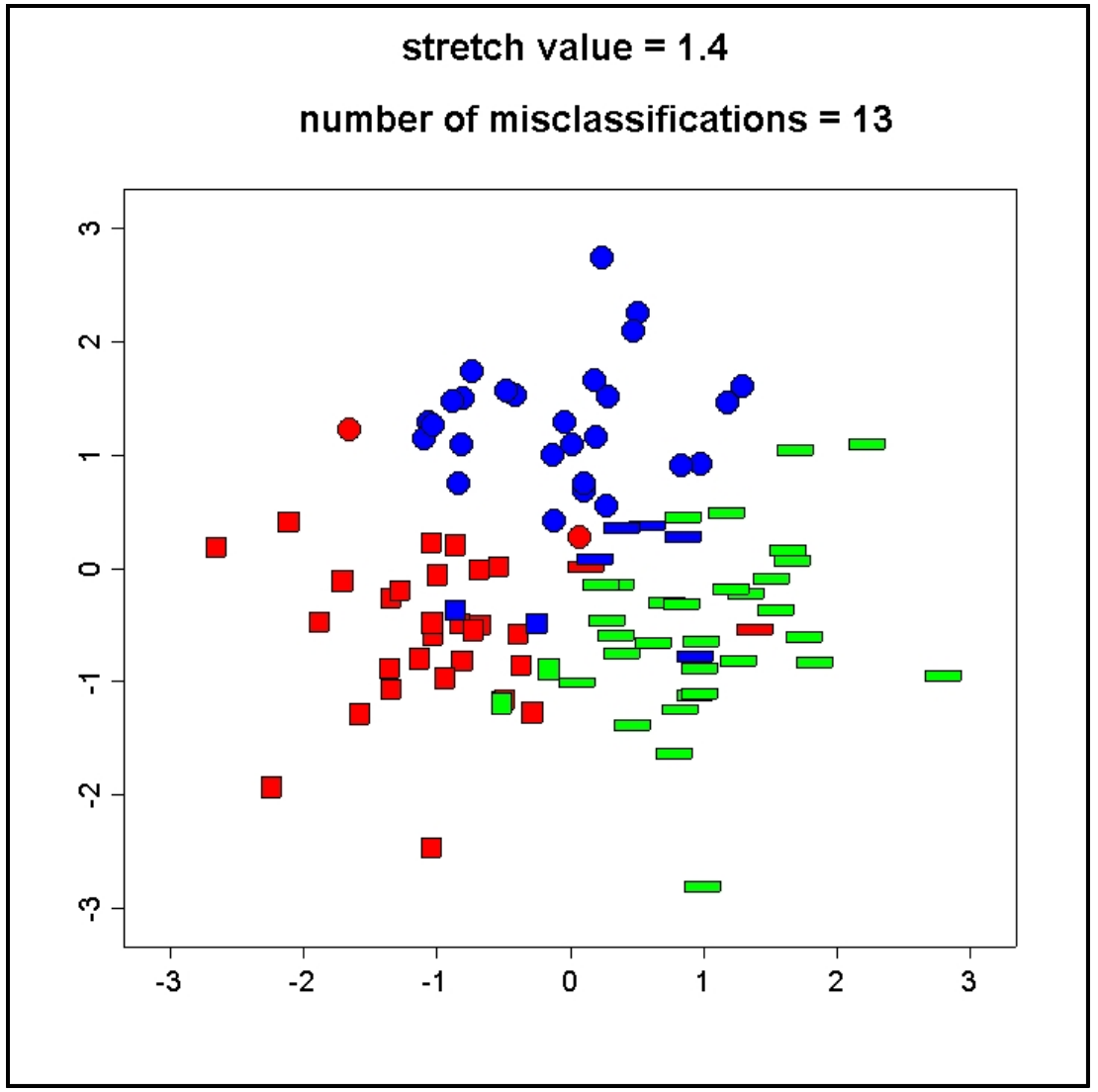


Comparative study - method

Color → Group

Shape → Cluster

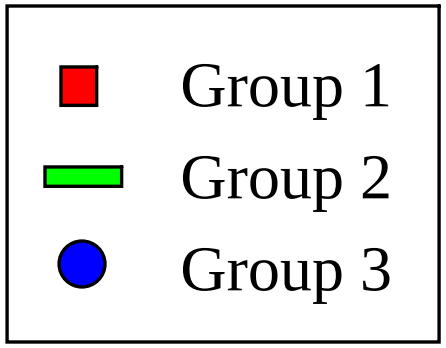


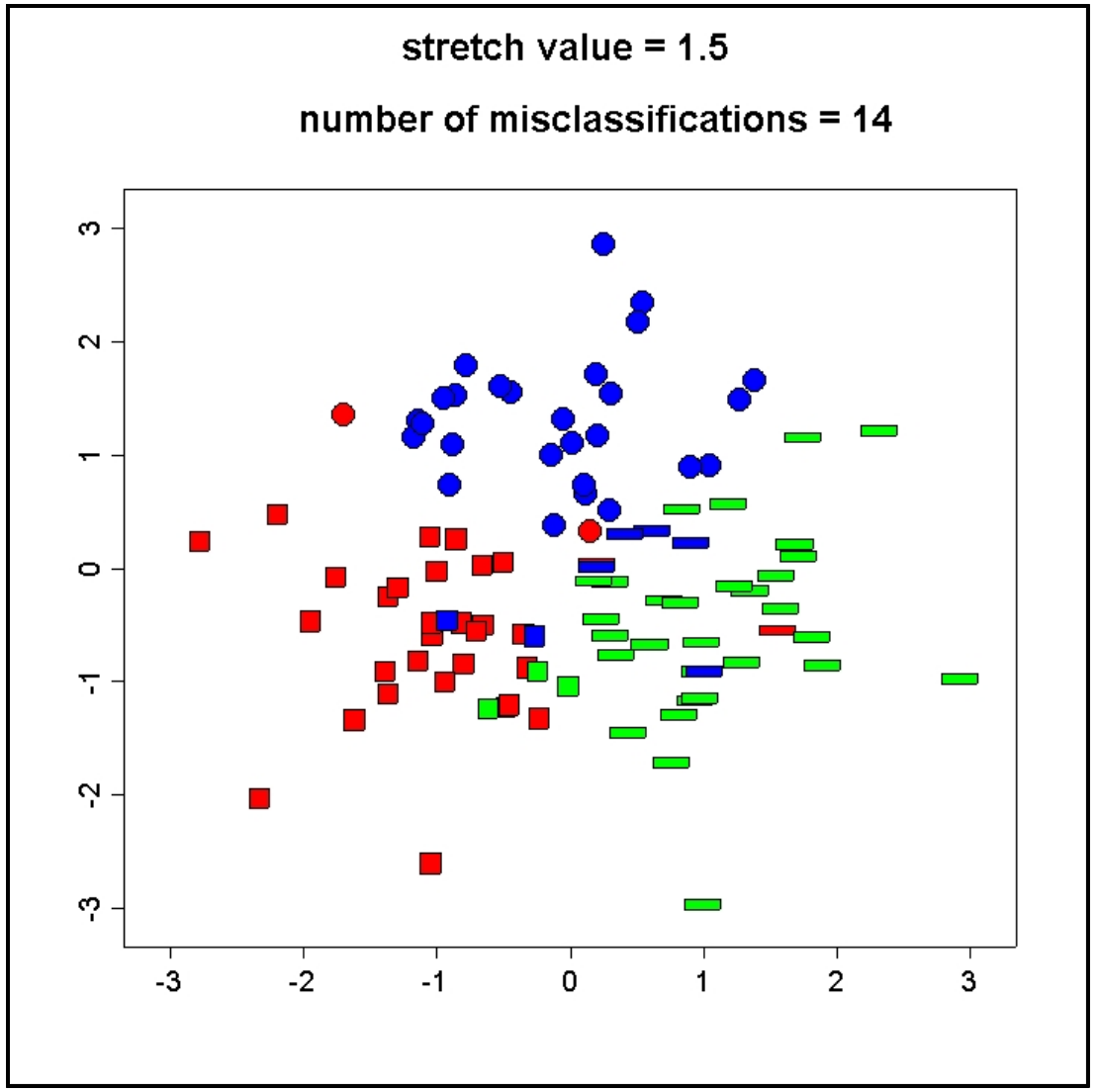


Comparative study - method

Color → Group

Shape → Cluster





Comparative study - results

- **Superiority of k-means with repeated runs**

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

- **Superiority of PAM with Manhattan distance** for noisy data

- Differences depend on the specific dataset

- **Preselection of genes**

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

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

Recommendations

- **Interest in specific genes:**

  If you search for genes that are co-regulated 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! 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!

Recommendations

- **Estimation of number of clusters**

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

  Heuristic approach

  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 $b(i) := min_C\, d(i,C)$.

  Silhouette width: $s(i) := (b(i)-a(i)) / max(a(i),b(i))$.

  Minimal **average silhouette width** over all observations can be used to select the number of clusters.

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.

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…