S. Dray, D. Chessel, and J. Thioulouse. Co-inertia analysis and the linking of ecological tables. *Ecology*, 84:3078–3089, 2003.

R. Gittins. *Canonical analysis, a review with applications in ecology. Vol.12 of Biomathematics.* Springer- Verlag, Berlin, 1985.

T.R. Golub, D.K. Slonim, P. Tamayo *et al.* Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.

M. Gouy, C. Gautier, M. Attimonelli *et al.* ACNUC–a portable retrieval system for nucleic acid sequence databases: logical and physical designs and usage. *Comput Appl Biosci*, 1(3):167–72, 1985.

S. Holmes. Multivariate analysis: The french way. In D. Nolan and T. Speed, editors, *Festschrift for David Freedman*. IMS, Beachwood, OH, 2006.

I. B. Jeffery, S. F. Madden, P. A. McGettigan *et al.* Integrating transcription factor binding site information with gene expression datasets. *Bioinformatics*, 23(3):298–305, 2007.

J. Khan, J.S. Wei, M. Ringner *et al.* Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nat Med*, 7(6):673–9, 2001.

B. Le Roux and H. Rouanet. *Geometric Data Analysis.* Kluwer Academic Publishers, Dordrecht, 2004.

J. R. Lobry and D. Chessel. Internal correspondence analysis of codon and amino-acid usage in thermophilic bacteria. *J Appl Genet*, 44(2):235–61, 2003.

D.T. Ross, U. Scherf, M.B. Eisen *et al.* Systematic variation in gene expression patterns in human cancer cell lines. *Nat Genet*, 24(3):227–35, 2000.

J. E. Staunton, D.K. Slonim, H.A. Coller *et al.* Chemosensitivity prediction by transcriptional profiling. *Proc Natl Acad Sci USA*, 98(19):10787–92, 2001.

C. Ter Braak. Canonical correspondence analysis: a new eigenvector technique for multivariate direct gradient analysis. *Ecology*, 69:1167–1179, 1986.

J. Thioulouse and J. Lobry. Co-inertia analysis of amino-acid physico-chemical properties and protein composition with the ade package. *Comput Appl Biosci*, 11(3):321–9, 1995.

J. Thioulouse, D. Chessel, S. Dolèdec *et al.* ADE-4: a multivariate analysis and graphical display software. *Statistics and Computing*, 7(1):75–83, 1997.

Aedín C. Culhane
Department of Biostatistics and Computational Biology, Dana-Farber Cancer Institute & Department of Biostatistics, Harvard School of Public Health, Boston, MA, USA.
aedin@jimmy.harvard.edu

Jean Thioulouse
Biométrie et Biologie Evolutive,
CNRS & Université Lyon 1, France.
jthioulouse@biomserv.univ-lyon1.fr

Using amap and ctc Packages for Huge Clustering

by Antoine Lucas and Sylvain Jasson

Introduction

Huge clustering is often required in the field of DNA microarray (DeRisi et al., 1997) analysis. A new use of clustering results appears with presentation and exploration software like *TreeView* (Eisen et al., 1998).

DNA microarray is the most appropriate method for high throughput gene studies, allowing expression evaluation of vast gene numbers in different cells types or conditions. From a technical point of view, microarray analysis first needs image processing (for example *Imagene* (http://www.biodiscovery.com), *BZScan* (Lopez et al., 2004) or *ScanAlyze* (Eisen et al., 1998)) that gives large tables of data, followed by statistical processing including data normalization.

A main goal of microarray analysis is to detect coregulated genes presenting similar expression profiles, which can be achieved by various classification techniques. In this area, hierarchical clustering is of special interest as it allows multi-scale cluster visualization.

Some R extensions provide efficient clustering tools (mainly: **stats** and **cluster**; Struyf et al., 1997). The packages **amap** and **ctc** aim to complete the set of clustering tools for R with:

- Additional features to standard clustering functions.

- A novel PCA method, robust to extreme values.

- Fast and optimized and parallelized algorithms, which drastically reduce time and memory requirements so that any computer is able to cluster large data sets.

- The possibility of an external visualization with software such as *Treeview* (Eisen et al., 1998) and *Freeview* (`http://magix.fri.uni-lj.si/freeview`), as shown in Figure 1. This visualization software provides convenient cluster exploration and browsing to find relevant information in large cluster trees.

Description

Amap and **ctc** packages are complementary. The former implementa all statistical algorithms and the latter is used for all interactions with other software such as the Eisen suite and makes it possible to launch *Xcluster* (`http://genetics.stanford.edu/~sherlock/cluster.html`) software within R.

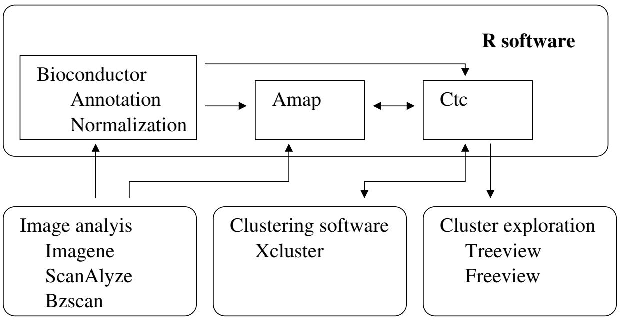
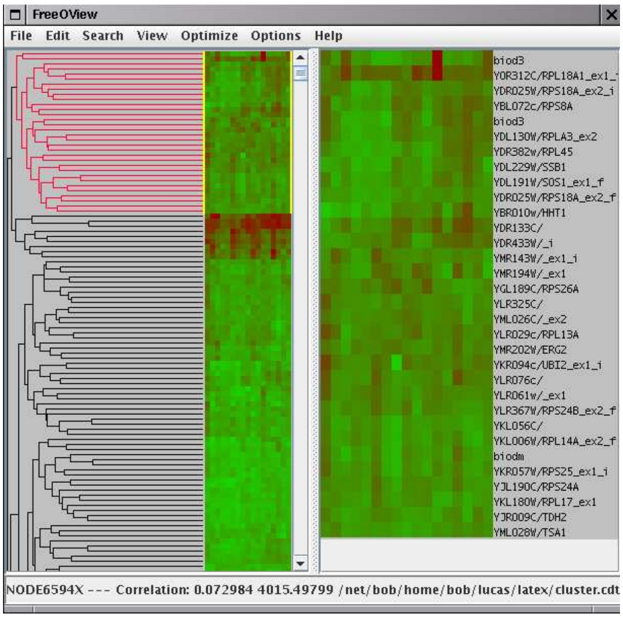


Figure 1: **Amap** and **ctc** usage for microarray analysis.

The **amap** package includes standard hierarchical clustering and k-means analysis. The novel features of **amap** are a larger selection of distances set like Pearson or Spearman (rank-based metric) adapted to microarray data and a better hierarchical clustering implementation (see the benchmark section).

Clustering can be pre-processed by a principal component analysis, as it projects data into an orthogonal vector space, which avoids counting correlated variables twice. With this analysis, a few extreme values may strongly affect the main components. As the high throughput implies a fully automated data acquisition and therefore outliers generation, we implement robust statistic tools including a principal component analysis. The main idea of such tools is to minimize the isolated points affected by lowering their relative weight. This is a recent method described by Caussinus et al. (2003).



```
NODE6594X --- Correlation: 0.072984 4015.49799 /net/bob/home/bob/lucas/latex/cluster.cdt
```

Figure 2: Results window with *Freeview* displaying a sub-cluster (red part of the tree).

The **ctc** package has tools to export cluster trees to visualization software that can explore trees at any scale to find the suitable magnification depending on the data specificity and the biological expertise as shown on Figure 2.

The **ctc** package also makes it convenient to use *Xcluster* software within R. *Xcluster* performs clustering with a very small memory allocation (it does not compute the whole distance matrix).

We propose the possibility to import results from Eisen *Cluster* software to perform post clustering processing with R. Other conversion functions are designed to dialog between R and Eisen software suite.

Benchmark

We compare time and memory use with other main implementations of standard hierarchical clustering: average link for agglomeration method and Euclidean distance (see Figure 3).

It appears that *Xcluster* has less memory needs (less than 100 MB when others methods use more than 1.5 GB) since the algorithm used does not compute exhaustive distance while agglomerating clusters. When using the complete distance matrix, memory use is $O(n^2)$ Hierarchical clustering from **amap**, **stats** or **cluster** packages returns the same tree but **cluster** includes a post processing that reworks the tree display. The **amap** implementation of `hcluster` or `hclusterpar` functions are significantly faster and allow us to pass the limit of 15000 genes on a recent server in less than half an hour.

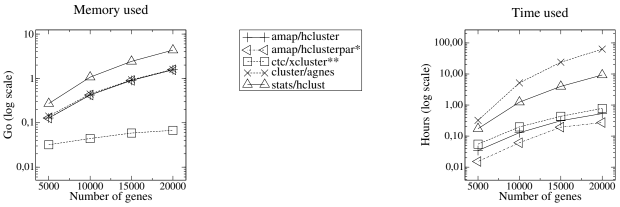


Figure 3: Benchmark of R package. Sample: simulated data, 5000 to 20000 genes under 200 conditions, using average link and Euclidean distance for clustering. Computer: dual Xeon processor server, with 4 GB RAM and 20 GB swap. System command "time" provides time and memory usage. R version 2.0.1
* `hclusterpar` is a parallelized version of `hcluster`, uses all CPU.
** `Xcluster` uses a slightly simplified algorithm.

Web application

As many end-users use graphical and intuitive interfaces, we propose a way to skip the R command line austerity while using a web interface. We provide files 'amap.php' and 'ctc.php' as part of the packages, which produce both form and CGI script, with any standard *apache* and *php* server.

A more sophisticated web application can be tested and downloaded at url: `http://bioinfo.genopole-toulouse.prd.fr/microarray`.

Methods and implementation

The **amap** core library is implemented in C. The package runs on Linux, Windows, and Mac OS X. Multi-threading and parallelization are disabled on Windows. Both **amap** and **ctc** use the free and open source license GPL.

The **amap** package is hosted on a sourceforge like project manager at `http://mulcyber.toulouse.inra.fr/projects/amap` by Inra that provides a cvs repository and a bug tracker.

The **amap** package is also available on CRAN, and the **ctc** package is available on Bioconductor.

Acknowledgments

We wish to thank Laurent Gautier and Steffen Möller for their contributions.

A. Lucas was supported by the *FNS Réseau des Génopôles* contract number A1457.

Bibliography

H. Caussinus, M. Fekri, S. Hakam, and A. Ruiz-Gazen. A monitoring display of multivariate outliers. *Computational Statistics & Data Analysis*, 44: 237–252, October 2003.

J. DeRisi, V. Iyer, and P. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278(5338):680–6, 1997.

M. Eisen, P. Spellman, P. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA*, 95: 14863–14868, 1998.

F. Lopez, J. Rougemont, B. Loriod *et al.* Feature extraction and signal processing for nylon DNA microarrays. *BMC Genomics*, 5(1):38, Jun 2004. Evaluation Studies.

A. Struyf, M. Hubert, and P. Rousseeuw. Integrating robust clustering techniques in S-PLUS. *Computational Statistics and Data Analysis*, 26:17–37, Nov 1997.

Antoine Lucas
`antoinelucas@gmail.com`

Sylvain Jasson
Unité de Biométrie et Intelligence Artificielle, INRA, Castanet Tolosan, France
`sylvain.jasson@toulouse.inra.fr`

Model-based Microarray Image Analysis

by Chris Fraley and Adrian E. Raftery

DNA microtechnology has enabled biologists to simultaneously monitor the expression levels of thousands of genes or portions of genes under multiple experimental conditions. Many microarray platforms exist; what they all have in common is that the gene expression data is obtained via image analysis of the array segments or spots corresponding to the individual experiments.

A common method for making DNA arrays consists of printing the single-stranded DNA representing the genes on a solid substrate using a robotic spotting device. The arrayed DNA spots are then mixed and hybridized with the cDNA extracted from the experimental and control samples. In the two-color array, these samples are treated before hybridization with both Cy3 (green) and Cy5 (red)