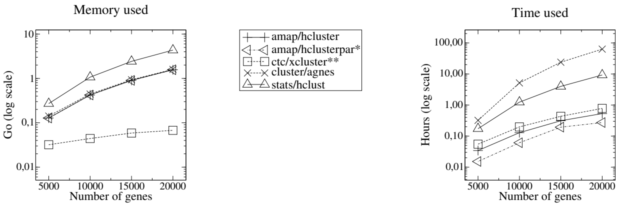

Figure 3: Benchmark of R package. Sample: simulated data, 5000 to 20000 genes under 200 conditions, using average link and Euclidean distance for clustering. Computer: dual Xeon processor server, with 4 GB RAM and 20 GB swap. System command "time" provides time and memory usage. R version 2.0.1
* `hclusterpar` 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) fluorescent dyes. After hybridization, the arrays are scanned at the corresponding wavelengths separately to obtain the images corresponding to the two channels. The fluorescence measurements are used to determine the relative abundance of mRNA or DNA in the samples.

The quantification of the amount of fluorescence from the hybridized sample can be affected by a variety of defects that occur during both the manufacturing and processing of the arrays, such as perturbations of spot positions, irregular spot shapes, holes in spots, unequal distribution of DNA probe within spots, variable background, and artifacts such as dust and precipitates. Ideally these events should be automatically recognized in the image analysis, and the estimated intensities adjusted to take account of them.

Li et al. (2005) proposed a method for segmenting microarray image blocks, along with a robust model-based method for estimating foreground and background intensities. Peaks and valleys are first located in the image signal with a sliding window to automatically separate the microarray blocks into regions containing the individual spots. Model-based clustering (McLachlan and Peel 2000, Fraley and Raftery 2002) is then applied to the (univariate) sum of the intensities of the two channels measuring the red and green signals for each spot region to provide an initial segmentation. Models are fit for up to three groups (background, foreground, uncertain), and the number of groups present is then determined via the Bayesian Information Criterion (BIC). Whenever there is more than one group, the segmentation is postprocessed in order to remove artifacts. This is done by reclassifying connected components in the brightest group that are below a certain threshold in size as unknown. The procedure is described in Figure 1.

1. Automatic gridding.
2. Model-based clustering for ≤ 3 groups.
3. Foreground / background determination:
   - If there is more than one group, threshold connected components. The foreground is taken to be the group of highest mean intensity and the background the group of lowest mean intensity.
   - If there is only one group, it is assumed that no foreground signal is detected.

Figure 1: Basic Procedure for Model-based Segmentation of Microarray Blocks.

This approach combines the strengths of histogram-based and spatial methods. It deals effectively with inner holes and artifacts. It also provides a formal inferential basis for deciding when no foreground signal is present in a spot. The method has been shown to compare favorably with other methods on experimental microarray data with replicates (Li et al. 2005).

The method is implemented in the Bioconductor package `spotSegmentation`, which consists of two basic functions:

`spotgrid`: determines spot locations in blocks within microarray slides

`spotseg`: determines foreground and background signals within individual spots

The `spotseg` function uses the R package `mclust` (Fraley and Raftery, 1999, 2003, 2006) for model-based clustering. Other life-sciences applications of model-based clustering include grouping coexpressed genes (Yeung et al. 2001), *in vivo* MRI of patients with brain tumors (Wehrens et al. 2002), and contrast-enhanced MRI for breast tumors (Forbes et al. 2004).

The `spotSegmentation` functions will be illustrated on the first block from the first microarray slide image from van't Wout et al. (2003), available as a dataset in Bioconductor under the name `HIVcDNAvantWout03`. The encoded image data from the two channels for this block are provided as datasets `hiv1raw` and `hiv2raw`, and can be obtained via the `data` command.

```
> data(hiv1raw)
> data(hiv2raw)
```

The data come from a supplementary website `http://ubik.microbiol.washington.edu/HIV/array1/supplemental.htm`, where they are encoded for compact storage. We have chosen to provide these data as given there, so that the following transformation is needed in order to extract the intensities:

```
> dataTrans <- function(x,A=4.7154240E-05)
  matrix((256*256-1-x)^2*A,nrow=450,ncol=1000)
> hiv1 <- dataTrans(hiv1raw)
> hiv2 <- dataTrans(hiv2raw)
```

Note that this transformation is specific to this data; in general stored image data must be converted as needed to image intensities. Figure 2 shows the image data for the two channels in reverse gray scale. These plots can be obtained with the `spotSegmentation` package using the following commands:

```
> plotBlockImage(sqrt(hiv1))
> plotBlockImage(sqrt(hiv2))
```

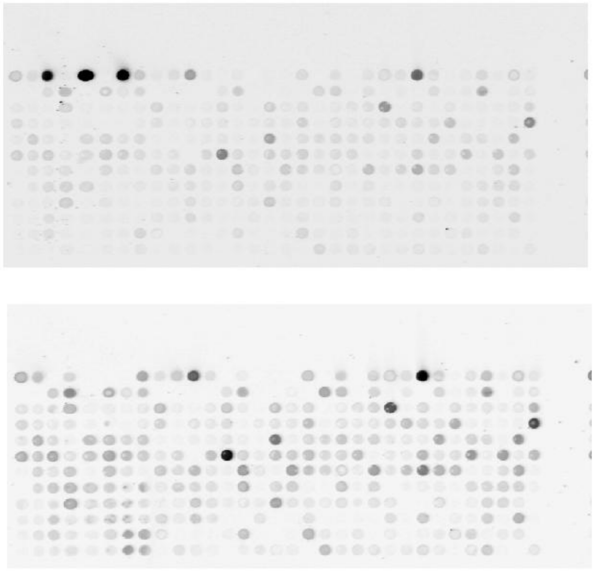

Figure 2: Reverse gray-scale plots of image intensities from channel 1 (Cy3 green) and channel 2 (Cy5 red) of the first block from the first slide of HIV data from the Bioconductor dataset HIVcDNAvantWout03.

The function spotgrid can be used to divide the microarray image block into a grid separating the individual spots.

```
> hivGrid <- spotgrid( hiv1, hiv2,
          rows = 12, cols = 32, show = TRUE)

> hivGrid
$rowcut


 [1] 105 138 163 189 219 244 271 297 326 354
     379 407 438

$colcut
 [1]  11  41  66  94 126 163 192 222 250 279
     307 338 364 392 419 445 474 501 531 558
     587 614 641 671 697 727 754 782 808 836
     862 889 922
```

Here we have used the knowlege that there are 12 rows and 32 columns in a block of the microarray image. The show option allows display of the image, shown in the top pannel of Figure 3.
The individual spots can now be segmented using the function spotseg. The following segments all spots in the block:

```
hivSeg <- spotseg( hiv1, hiv2,
           hivGrid$rowcut, hivGrid$colcut)

plot(hivSeg)
```

The corresponding plot is shown in the bottom pannel in Figure 3.

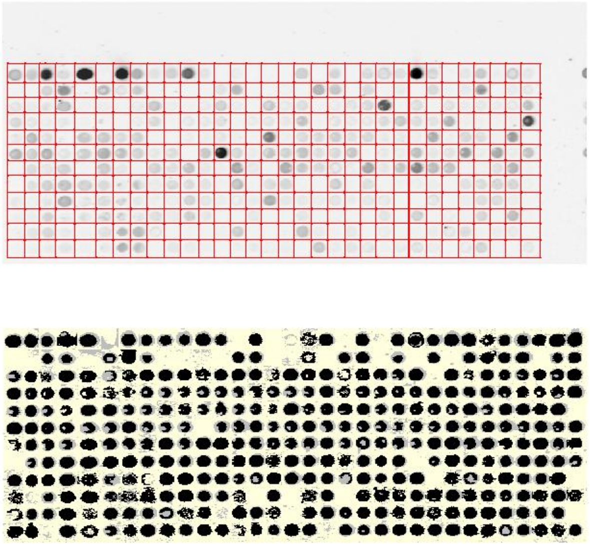

Figure 3: Above: The grid delimiting microarray spots determined by function spotgrid, superimposed on the sum of the intensities for the two channels for the Bioconductor dataset HIVcDNAvantWout03. Below: The segmented spots produced by spotseg. The color scheme is as follows: *black* denotes the spots, *yellow* denotes background, *gray* denotes pixels of uncertain classification.

It is possible to process a subset of the regions in the grid using the arguments R for grid (as opposed to pixel) row location of the spot and C for grid column location. The show option in spotseg can be used to display details for each spot as it is classified. When more than one spot is processed, the graphics command par(ask = TRUE) should be set so that the displays can be stepped through. The following is an example of the segmenting and display of an individual splot.

```
hivSeg <- spotseg( hiv1, hiv2,
           hivGrid$rowcut, hivGrid$colcut
           R = 1, C = 4, show = TRUE)
```

The resulting display is shown in Figure 4.
Mean and median pixel intensities for the foreground and background for each channel and each spot can be recovered through the summary function applied to the output of spotseg. For example, the following extracts the summary intensities for the spot shown in Figure 4.

```
> hivSumry <- summary(hivSeg)

> hivSumry$channel1$foreground$mean[1,4]
[1] 1475.053

> hivSumry$channel2$background$median[1,4]
[1] 249.0123
```

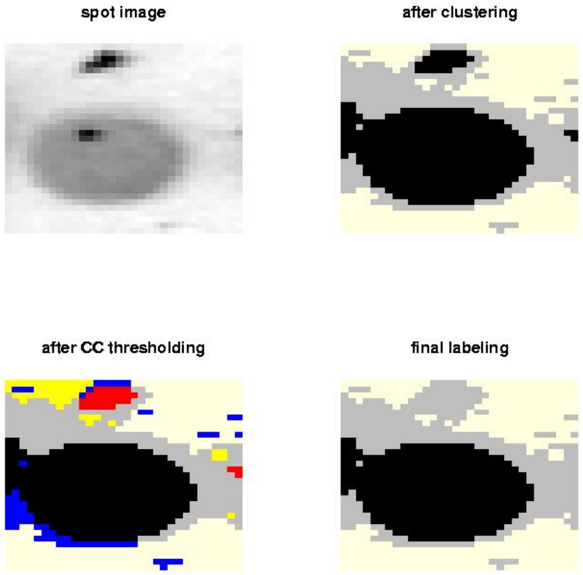

Figure 4: `spotseg` processing of the 1,4 section of the gridded HIV image data. Clockwise from top left: gray-scale image, labeled image after model-based clustering (light yellow: lowest intensity; black: highest intensity), clustered image with connected components less than threshold in size labeled (bright yellow, blue, red denote components below threshold in size for the light yellow, gray, and black groups, respectively), final labeling.

Summary

The Bioconductor package `spotSegmentation` provides functionality for automatic gridding of microarray blocks given the number of rows and columns of spots. It also provides functionality for determining foreground and background of spots in microarray images via model-based clustering. This approach deals effectively with inner holes and artifacts, as well as providing a formal inferential basis for deciding when no foreground signal is present in a spot.

Bibliography

F. Forbes, N. Peyrard, C. Fraley, D. Georgian-Smith, D. Goldhaber, and A. Raftery. Model-based region-of-interest selection in dynamic breast MRI. *Journal of Computer Assisted Tomography*, 30:675–687, 2006.

C. Fraley and A. E. Raftery. MCLUST: Software for model-based cluster analysis. *Journal of Classification*, 16:297–306, 1999.

C. Fraley and A. E. Raftery. Model-based clustering, discriminant analysis and density estimation. *Journal of the American Statistical Association*, 97:611–631, 2002.

C. Fraley and A. E. Raftery. Enhanced software for model-based clustering, density estimation, and discriminant analysis: MCLUST. *Journal of Classification*, 20:263–286, 2003.

Q. Li, C. Fraley, R. E. Bumgarner, K. Y. Yeung, and A. E. Raftery. Donuts, scratches, and blanks: Robust model-based segmentation of microarray images. *Bioinformatics*, 21:2875–2882, 2005.

G. J. McLachlan and D. Peel. *Finite Mixture Models*. Wiley, 2000.

R. Wehrens, A. W. Simonetti and L. M. .C. Buydens, Mixture modeling of medical magnetic resonance data. *Journal of Chemometrics*, 16:274-282, 2002.

A. B. van't Wout, G. K. Lehrman, S. A. Mikeeva, G. C. O'Keefe, M. G. Katze, R. E. Bumgarner, G. K. Geiss, and J. I. Mullins. Cellular gene expression upon human immunodeficiency type 1 infection of CD4(+)-T-cell lines. *Journal of Virology*, 77(2):1392–1402, January 2003.

K. Y. Yeung, C. Fraley, A. Murua, A. E. Raftery, and W. L. Ruzzo. Model-based clustering and data transformation for gene expression data. *Bioinformatics 17*, 977–987, 2001.

C. Fraley and A. E. Raftery. MCLUST version 3 for R: Normal mixture modeling and model-based clustering. Technical Report No. 504, University of Washington, September, 2006.

Chris Fraley, Adrian Raftery
Department of Statistics, Box 354322
University of Washington
Seattle, WA 98195-4322 USA
`fraley@stat.washington.edu`
`raftery@stat.washington.edu`