BIOINFORMATICS APPLICATIONS NOTE Vol. 23 no. 9 2007, pages 1168–1169
doi:10.1093/bioinformatics/btm072

Gene expression

Codelink: an R package for analysis of GE healthcare gene expression bioarrays

Diego Diez[1,*,†], Rebeca Alvarez[2] and Ana Dopazo[2]

[1]Instituto de Investigaciones Biomédicas Alberto Sols, Consejo Superior de Investigaciones Cientificas-Universidad Autonoma de Madrid, Arturo Duperier 4, 28029 Madrid, Spain and [2]Genomics Unit, Centro Nacional de Investigaciones Cardiovasculares (CNIC), Melchor Fernández Almagro 3, 28029 Madrid, Spain

Received on January 30, 2007; revised on February 21, 2007; accepted on February 22, 2007

Advance Access publication March 7, 2007

Associate Editor: Joaquin Dopazo

ABSTRACT

Motivation: Microarray-based expression profiles have become a standard methodology in any high-throughput analysis. Several commercial platforms are available, each with its strengths and weaknesses. The R platform for statistical analysis and graphics is a powerful environment for the analysis of microarray data, because it has many integrated statistical methods available as well as the specialized microarray analysis project Bioconductor. Many packages have been added in the last few years increasing the range of possible analysis. Here, we report the availability of a package for reading and analyzing data from GE Healthcare Gene Expression Bioarrays within the R environment.

Availability: The software is implemented in the R language, is open source and available for download free of charge through the Bioconductor (http://www.bioconductor.org) project.

Contact: diez@kuicr.kyoto-u.ac.jp

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

Besides the inherent problems in experiment reproducibility and platform comparability (Tan *et al*., 2003), microarray expression profiling has become a standard analysis in any high-throughput experiment. There are several popular commercial microarray platforms available with different array designs. The popularity of GE Healthcare Gene Expression Bioarrays (Codelink) is growing within the microarray community since the number of publications involving Codelink data is increasing, both in the academic world and industry. This is shown by the fact that more than half of the articles related to Codelink data have been published within the last year. To better understand the biological meaning of the results obtained with different microarray platforms, we need to be able to exploit the data using the detailed singularities of each platform. This is becoming more important as the number of studies using data from different platforms or reanalyzing already published data is increasing (Dupuy and Simon, 2007).

Codelink is a single-channel microarray platform that uses 30-bp oligonucleotide probes designed for three different organisms; human, mouse and rat. The arrays are presented in several sizes, ranging from 10k to whole genome arrays. In addition, focused arrays are available, such as the Inflammation 16 Assay or the ADME Rat Toxicology 16 Assay arrays. The Codelink software is typically used to analyze the images obtained from the scanner. This software assigns to each probe a signal-to-noise ratio (SNR)-based detection call. The SNR is computed as the spot mean divided by the product of the background median plus 1.5 times the background standard deviation. Spots are labeled present (G) if the SNR is greater than one or absent (L) otherwise. In addition, other flags are assigned based on different considerations, such as spot contamination (C), signal saturation (S), irregular shape (I), manufacturer-discarded spots (M) and user-discarded spots (X). Several control probes are spotted, for instance positive and negative controls, allowing array hybridization efficiency to be assessed. Finally, some probes are unique whereas others map to several genes and some genes are represented by more than one probe. All this information is important to know how reliable our results are and can help to select which genes are going to be the subject of further analysis.

Here, we report the availability of the *codelink* R Development Core Team, 2005 package, which is available through the Gentleman, R. C. *et al*. 2004 project, for the analysis of Codelink arrays. In addition, annotation packages for the different arrays are generated for each Bioconductor release and are available to the community.

2 DESCRIPTION

The Codelink software does not have a standard exported file format but instead has a custom format, decided at export time. It is, therefore, important to select all the information needed in order to use it with the *codelink* package. Raw intensities must be exported, including spot mean, background median and background standard deviation (needed in order to compute the SNR). Other useful information is the spot physical location, the quality flags and the probe type. Background subtracted and normalized intensities can also be exported if needed.

Different background correction methods have been developed to date, designed to deal with different background estimation situations. In this package, some of the more popular have been implemented or adapted, in order to allow different options to be used, including *subtract* (the default method applied by the Codelink software), *half* and *normexp*. The *normexp* method is a wrapper to the corresponding method found in the *limma* package (Smyth, 2005).

Normalization of intensities is an important step aiming to allow data comparison by removing some systematic bias.

*To whom correspondence should be addressed.
†Present address: Bioinformatics Center, Institute for Chemical Research, Kyoto University, Uji, Kyoto 611–0011, Japan.

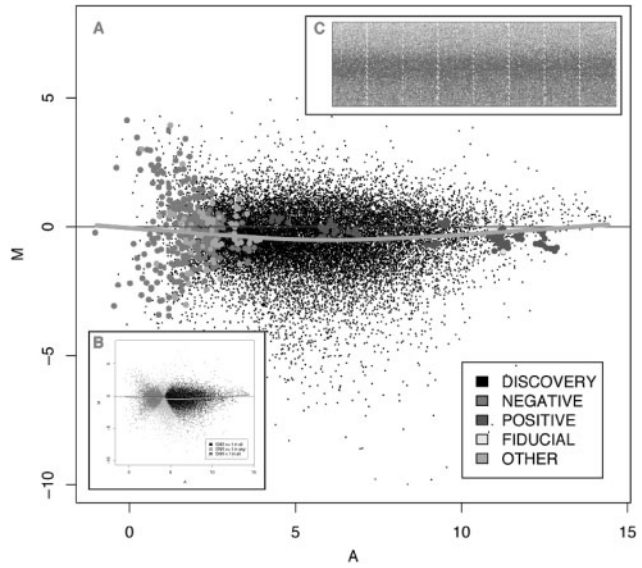

Fig. 1. (**A**) MA plot comparing two samples with different probe types is highlighted. (**B**) MA plot for the same samples but using the SNR information to label probes. If the SNR is greater than one in the two samples, the spots are black, orange if one of the SNR is less than one and red when both are less than one. (**C**) Array pseudo image showing the background intensities indicated as increasing levels of blue. A colour version of this figure is available as supplementary data.

Although there is not yet consensus about which normalization method is the most appropriate (Shippy *et al*., 2006), it has been noted that in general, methods that take into account the non-linear dependence on signal intensity perform better (Workman *et al*., 2002).

In particular, the default method available on the Codelink software, *median* normalization, has the disadvantage that it does not take this behavior into account. For this reason, several normalization methods have been included in the *codelink* package allowing easy comparisons.

In addition to *median* normalization, *quantile* and *CyclicLoess* are available. In several studies (Shi *et al*., 2006; Wu *et al*., 2005) comparing normalization methods in the Codelink platform, better performance has been found using *quantile* and *CyclicLoess* compared to *median* normalization. On the other hand, *CyclicLoess* performed slightly better compared to *quantile* normalization (Wu *et al*., 2005).

Several diagnostic plots are available, exploiting the singular characteristics of the Codelink platform. Density plots can be useful for assessing pre-processing steps, such as background correction and normalization. MA plots have been widely used to assess normalization performance in two-channel microarrays, but are also very valuable in one-channel microarrays. It is possible to highlight spots based on the spot type (e.g. discovery probes, positive probes, negative probes, etc.) or SNR (Fig. 1A and B). Other useful plots are the array pseudo images representing probe intensities, background or SNR into the chip location (Fig. 1C). These can be used to detect spot artifacts and to assess the effect of background correction and normalization.

In addition to the *codelink* package, annotation packages for the different arrays are provided. These packages, built using *AnnBuilder* (Zhang, 2006), contain information about the array probes with links to several biological databases like GeneOntology, KEGG, Uniprot, etc. The annotation packages are updated during every Bioconductor release, allowing an up-to-date description of the gene products corresponding to each probe. In addition, there are utility functions to help write report files from the list of differentially expressed genes, including the corresponding information from the annotation packages.

As with every package in the Bioconductor project, the *codelink* package comes with documentation in the form of a vignette file. For further details about the package usage, users are encouraged to refer to the vignette.

3 CONCLUSIONS

The Codelink platform is a valuable tool for performing high-throughput expression analysis, with accurate and consistent results. This platform has been used successfully in several experiments (Peng *et al*., 2003; Stanwood *et al*., 2006) and has been included recently in a survey comparing different microarray platforms (Shi *et al*., 2006). The R platform for statistical analysis and graphics is a powerful environment for the analysis of microarray data, but although it comes with an extensive and abundant documentation, the learning curve can be challenging. In order to manage the overwhelming amount of information needed for running an analysis, packages that allow an easy interface of the powerful methods available and the specific data are needed. The *codelink* package provides users with an easy to use interface for the analysis of Codelink data on the R platform.

ACKNOWLEDGEMENTS

The authors thank Nelson Hayes for reviewing the manuscript, and Juan Bernal and Beatriz Morte for their support. This publication is based upon work supported by the Programa I3P de la Red de Bioinformática del CSIC (pre-doctoral fellowship), Comunidad de Madrid 08.5/0042/2003 and GR/SAL/0382/2004, MEC BFI2002-00489 y Red de Centros RCMN (C03/08).

Conflict of Interest: none declared.

REFERENCES

Dupuy,A. and Simon,R.M. (2007) Critical review of published microarray studies for cancer outcome and guidelines on statistical analysis and reporting. *J. Natl. Cancer Inst*., **99**, 147–157.

Gentleman,R.C. *et al*. (2004) Bioconductor: open software development for computational biology and bioinformatics. *Genome Biol*., **5**, R80.

Peng,Y. *et al*. (2003) Transcriptional characterization of bone morphogenetic proteins (BMPs)-mediated osteogenic signaling. *J. Cell Biochem*., **90**, 1149–1165.

Shi,L. *et al*. (2006) The MicroArray Quality Control (MAQC) project shows inter- and intraplatform reproducibility of gene expression measurements. *Nat. Biotechnol*., **24**, 1151–1161.

Shippy,R. *et al*. (2006) Using RNA sample titrations to assess microarray platform performance and normalization techniques. *Nat. Biotechnol*., **24**, 1123–1131.

Smyth,G.K. (2005) Limma: linear models for microarray data. In Gentleman,R.V.C. *et al*. (eds.) *Bioinformatics and Computational Biology Solutions using R and Bioconductor*. Springer, New York, pp. 397–420.

Stanwood,G.D. *et al*. (2006) Genetic or pharmacological inactivation of the dopamine D1 receptor differentially alters the expression of regulator of G-protein signalling (Rgs) transcripts. *Eur. J. Neurosci*., **24**, 806–818.

Tan,P.K. *et al*. (2003) Evaluation of gene expression measurements from commercial microarray platforms. *Nucleic Acids Res*., **31**, 5676–5684.

Team,R.D.C. (2005) R: a language and environment for statistical computing.

Workman,C. *et al*. (2002) A new non-linear normalization method for reducing variability in DNA microarray experiments. *Genome Biol*., **3**, research0048.

Wu,W. *et al*. (2005) Comparison of normalization methods for CodeLink Bioarray data. *BMC Bioinformatics*, **6**, 309.

Zhang,J. (2006) AnnBuilder: bioconductor annotation data package builder.