Introduction to the Bioconductor marray package : Input component

Yee Hwa Yang[1] and Sandrine Dudoit[2]

October 3, 2006

1. Department of Medicine, University of California, San Francisco,
jean@biostat.berkeley.edu
2. Division of Biostatistics, University of California, Berkeley.

Contents

1 Overview

This document provides a tutorial for the **data input** component of the `marray` package. This is similar to the previous `marrayInput` package which has now been combined with the suite of other four packages for diagnostic plots and normalization of cDNA microarray data. This package relies on object–oriented class/method mechanism, provided by the R `methods` package, to allow efficient and systematic representation and manipulation of microarray data.

This vignette describeds functionality for reading microarray data into R, such as intensity data from image processing output files (e.g. `.spot` and `.gpr` files for the `Spot` and `GenePix` packages, respectively) and textual information on probes and targets (e.g. from gal files and god lists). A `tcltk` widget is supplied to facilitate and automate data input and the creation of microarray specific R objects for storing these data.

2 Getting started

To load the `marray` package in your R session, type `library(marray)`. We demonstrate the functionality of this R packages using gene expression data from the Swirl zebrafish experiment which is included as part of the package. To load the swirl dataset, use `data(swirl)`, and to view a description of the experiments and data, type `? swirl`.

3 Case study: Swirl zebrafish microarray experiment

We demonstrate the functionality of this collection of R packages using gene expression data from the Swirl zebrafish experiment. These data were provided by Katrin Wuennenberg–Stapleton from the Ngai Lab at UC Berkeley. (The swirl embryos for this experiment were provided by David Kimelman and David Raible at the University of Washington.) This experiment was carried out using zebrafish as a model organism to study early development in vertebrates. Swirl is a point mutant in the BMP2 gene that affects the dorsal/ventral body axis. Ventral fates such as blood are reduced, whereas dorsal structures such as somites and notochord are expanded. A goal of the Swirl experiment is to identify genes with altered expression in the swirl mutant compared to wild–type zebrafish. Two sets of dye–swap experiments were performed, for a total of four replicate hybridizations. For each of these hybridizations, target cDNA from the swirl mutant was labeled using one of the Cy3 or Cy5 dyes and the target cDNA wild–type mutant was labeled using the other dye. Target cDNA was hybridized to microarrays containing 8,448 cDNA probes, including 768 controls spots (e.g. negative, positive, and normalization controls spots). Microarrays were printed using 4×4 print–tips and are thus partitioned into a 4×4 grid matrix. Each grid consists of a 22×24 spot matrix that was printed with a single print–tip. Here, spot row and plate coordinates should coincide, as each row of spots corresponds to probe sequences from the same 384 well–plate.

Each of the four hybridizations produced a pair of 16–bit images, which were processed using the image analysis software package `Spot` (Buckley, 2000; Yang et al., 2002). The dataset includes four output files `swirl.1.spot`, `swirl.2.spot`, `swirl.3.spot`, and `swirl.4.spot` from the `Spot` package. Each of these files contains 8,448 rows and 30 columns; rows correspond to spots and columns to different statistics from the `Spot` image analysis output. The file `fish.gal` is a gal file generated by the `GenePix` program; it contains information on individual probe sequences, such as gene names, spot ID, spot coordinates. Hybridization information for the mutant and wild–type target samples is stored in `SwirlSample.txt`. All fluorescence intensity data from processed images are also included in this package (see Section 4 for greater details).

To load the swirl dataset, use `data(swirl)`, and to view a description of the experiments and data, type `? swirl`. Below, we give step–by–step instructions for reading the swirl data into R. For convenience, we have also stored the resuls in the object `swirl` of class `marrayRaw`.

```
> library(marray)
> data(swirl)
```

4 Package marrayInput – Reading microarray data into R

We begin our analysis of microarray data with the fluorescence intensities produced by image processing of the microarray scanned images. These data are typically stored in tables whose rows correspond to the spotted probe sequences and columns to different spot statistics: e.g. grid row and column coordinates, spot row and column coordinates, red and green background and foreground intensities for different segmentation and background adjustment methods, spot morphology statistics, etc. For the `GenePix` image processing software, these are the `.gpr` files, and for `Spot`, these are the `.spot` files. We also consider probe and target textual information stored, for example, in `.gal` and `.gdl` (god list) files. The main functions in the `marrayInput` package are `read.marrayLayout`, `read.marrayInfo`, and `read.marrayRaw`, which create objects of classes `marrayLayout`, `marrayInfo`, and `marrayRaw`, respectively. Widgets are provided for each of these functions to facilitate data entry.

For the Swirl zebrafish experiment, textual information and fluorescence intensity data from processed images were included as part of the package and can be accessed as follows, where `datadir` is the name of the R package sub–directory containing the data files.

```
> datadir <- system.file("swirldata", package = "marray")
> dir(datadir)
```

```
[1] "fish.gal"        "swirl.1.spot"    "swirl.2.spot"    "swirl.3.spot"
[5] "swirl.4.spot"    "SwirlSample.txt"
```

In general, microarray data consist of three distinct components; probes (genes) information, target(samples) information and measured gene expression levels information. Analyzing expression intensities alone with no corresponding probes or target information is meaningless. Therefore a data structure `marrayRaw` is created to store and link these information together in one R object.

4.1 Reading target information

We refer to *target file* as a file that lists the microarrays hybridization and describes which RNA samples were hybridized to each array. A target file is typically a tab-delimited text file which include at least the complete and **exact** name of each image processing file you would like to include in the data analysis and the corresponding names for the Cy3 and Cy5 labeled sample information. It is also informative to include other variables of interest that are useful for downstream analysis or for quality assessment. Examples include subject identification number, gender, age, date of hybridization, scanning conditions amongst others.

The main functions in the `marray` package for this purpose is `read.marrayInfo`, which will create an R object of class `marrayInfo`. Objects of class `marrayInfo` may be used to store information on probe sequences and target samples. For example, reading in the target information for the swirl experiment can done with

```
> swirlTargets <- read.marrayInfo(file.path(datadir, "SwirlSample.txt"))
> summary(swirlTargets)
```

```
Object of class marrayInfo.

      maLabels        Names slide number experiment Cy3 experiment Cy5
1 swirl.1.spot swirl.1.spot           81          swirl      wild type
2 swirl.2.spot swirl.2.spot           82      wild type          swirl
3 swirl.3.spot swirl.3.spot           93          swirl      wild type
4 swirl.4.spot swirl.4.spot           94      wild type          swirl
       date comments
1 2001/9/20       NA
2 2001/9/20       NA
3 2001/11/8       NA
4 2001/11/8       NA

Number of labels:  4
Dimensions of maInfo matrix:  4  rows by  6  columns

Notes:
 D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/SwirlSample.txt
```

4.2 Reading probes related information

Probes related information refers to descriptions of the spotted probe sequences (e.g. matrix of gene names, annotation, notes on printing conditions). Printing conditions or array fabrication information include the dimensions of the spot and grid matrices, and, for each probe on the array, its grid matrix and spot matrix coordinates. In addition, we also include plate origin of the probes, and information on the spotted control sequences (e.g. negative controls, housekeeping genes, spiked in-control probes and many others). These information are store separately using two objects: an object of class `marrayInfo` on the probes annotation information and an object of class `marrayLayout` to store arrays fabrication information.
There are two ways to read probes related information: the **first** method is to use the function `read.Galfile` as follow:

```
> galinfo <- read.Galfile("fish.gal", path = datadir)
> names(galinfo)

[1] "gnames"   "layout"   "neworder"
```

Users can modify the arguments `info.id` and `layout.id` to specify which column names or index represent probe annotation and printer layout (array fabrication) information respectively. For example the following code reads in the galfile `fish2.gal` where probe information are stored under the columns `Gene ID` and `Gene description` and the printeralayout information is stored under the columns `Grid`, `Row` and `Column`.

```
> fish2Gal <- read.Galfile(galfile="fish2.txt",
                           info.id = c("Gene ID", "Gene description"),
                           layout.id = c(Block="Grid", Row="Row",
                 Column="Column"), labels="Gene ID")
```

This function returns a list of 3 components. The first `gnames` is an `marrayInfo` object storing probe annotation information; the second `layout` is an `marrayLayout` object storing array fabrication (printing) information and lastly a numerical vector `neworder` which provides a resorting of data. The probes are assumed to be ordered and numbered consecutively starting from the top left grid and the top left spot within each grid. For most standard array layout, we typically recommend using this method.

Note: The slot `maSub` is included to allow importing data from non-complete arrays. `maSub` is a "logical" vector indicating which spots are currently being stored in the slots containing Cy3 and Cy5 background and foreground fluorescence intensities.
The **second** method uses both functions `read.marrayLayout` and `read.marrayInfo` to read and store information on array fabrication and probe annotation information respectively. This is usually done for more complex array structures. For example, reading in the probe annotation information for the swirl experiment can done with:

```
> swirl.gnames <- read.marrayInfo(file.path(datadir, "fish.gal"),
+     info.id = 4:5, labels = 5, skip = 21)
> summary(swirl.gnames)

Object of class marrayInfo.

   maLabels    "ID" "Name"
1     geno1 control  geno1
2     geno2 control  geno2
3     geno3 control  geno3
4     3XSSC control  3XSSC
5     3XSSC control  3XSSC
6      EST1 control   EST1
7     geno1 control  geno1
8     geno2 control  geno2
9     geno3 control  geno3
10    3XSSC control  3XSSC
...

Number of labels:  8448
Dimensions of maInfo matrix:  8448  rows by  2  columns

Notes:
 D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/fish.gal
```

The following command stores such layout information in the object `swirl.layout` of class `marrayLayout`. The location of the control spots is extracted from the fourth (`ctl.col=4`) column of the file `fish.gal`.

```
> swirl.layout <- read.marrayLayout(fname = file.path(datadir,
+     "fish.gal"), ngr = 4, ngc = 4, nsr = 22, nsc = 24, skip = 21,
+     ctl.col = 4)
```

```
> ctl <- rep("Control", maNspots(swirl.layout))
> ctl[maControls(swirl.layout) != "control"] <- "probes"
> maControls(swirl.layout) <- factor(ctl)
> summary(swirl.layout)

Array layout:         Object of class marrayLayout.

Total number of spots:                        8448
Dimensions of grid matrix:                4 rows by 4 cols
Dimensions of spot matrices:                22 rows by 24 cols

Currently working with a subset of 8448spots.

Control spots:
There are   2 types of controls :

Control  probes
    768    7680


Notes on layout:
 D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/fish.gal
```

4.3 Reading gene-expression data

Microarray image processing results are stored in ASCII files and by default, assumed to be tab-delimited. These can be loaded into R using `read.marrayRaw` or customized functions like `read.Spot`, `read.Agilent` and `read.GenePix` for `Spot`, `Agilent` and `GenePix` output, respectively. The customized functions are simply "wrapper" functions around `read.marrayRaw` which extract relevant spot statistics for different image processing packages. In addition, these functions will also setup the probe annotation and array layout information. The following command illustrate illustrate how to read in the raw expression data for the swirl data.

```
> mraw <- read.Spot(path = datadir, layout = galinfo$layout, gnames = galinfo$gnames,
+     target = swirlTargets)

Reading ...  D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.1.spot
Reading ...  D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.2.spot
Reading ...  D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.3.spot
Reading ...  D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.4.spot

> summary(mraw)

Pre-normalization intensity data:         Object of class marrayRaw.

Number of arrays:         4 arrays.
```

```
A) Layout of spots on the array:
Array layout:         Object of class marrayLayout.

Total number of spots:                        8448
Dimensions of grid matrix:                4 rows by 4 cols
Dimensions of spot matrices:                22 rows by 24 cols

Currently working with a subset of 8448spots.

Control spots:
There are   1 types of controls :
probes
  8448


Notes on layout:


B) Samples hybridized to the array:
Object of class marrayInfo.

      maLabels        Names slide number experiment Cy3 experiment Cy5
1 swirl.1.spot swirl.1.spot           81          swirl      wild type
2 swirl.2.spot swirl.2.spot           82      wild type          swirl
3 swirl.3.spot swirl.3.spot           93          swirl      wild type
4 swirl.4.spot swirl.4.spot           94      wild type          swirl
       date comments
1 2001/9/20       NA
2 2001/9/20       NA
3 2001/11/8       NA
4 2001/11/8       NA

Number of labels:  4
Dimensions of maInfo matrix:  4  rows by  6  columns

Notes:
 D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/SwirlSample.txt

C) Summary statistics for log-ratio distribution:
                                                                        Min.
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.1.spot -2.73
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.2.spot -2.72
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.3.spot -2.29
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.4.spot -3.21
                                                                     1st Qu.
```

```
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.1.spot   -0.79
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.2.spot   -0.15
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.3.spot   -0.75
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.4.spot   -0.46
                                                                    Median
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.1.spot  -0.58
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.2.spot   0.03
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.3.spot  -0.46
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.4.spot  -0.26
                                                                     Mean
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.1.spot -0.48
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.2.spot  0.03
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.3.spot -0.42
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.4.spot -0.27
                                                                    3rd Qu.
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.1.spot   -0.29
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.2.spot    0.21
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.3.spot   -0.12
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.4.spot   -0.06
                                                                    Max.
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.1.spot 4.42
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.2.spot 2.35
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.3.spot 2.65
D:/biocbld/1.9d/tmpdir/Rinst443230300/marray/swirldata/swirl.4.spot 2.90

D) Notes on intensity data:
Spot Data
```

For any arbitrary image analysis output file, we can use the function `read.marrayRaw`. The function takes as its main argument a list of names for files containing the intensity data (e.g. `GenePix` output files `.gpr`). It also takes as arguments the names of already created layout, probe, and target description objects, e.g., `swirl.layout`, `swirl.gnames`, and `swirlTargets` for the Swirl experiment. The following commands read in all the `Spot` files residing in the `datadir` directory. The arguments further specify that the red and green foreground intensities are stored under the headings `Rmean` and `Gmean`, and that the red and green background intensities are store under the headings `morphR` and `morphG`, respectively.

```
> fnames <- as.vector(swirlTargets@maInfo[,1])
> swirl.raw <- read.marrayRaw(fnames, path = datadir,
                              name.Gf = "Gmean", name.Gb = "morphG",
                              name.Rf = "Rmean", name.Rb = "morphR",
                              layout = swirl.layout,
                              gnames = swirl.gnames,
                              targets = swirlTargets
                              )
```

5 Other input functions

Widget input functions

To facilitate the creation of microarray data objects, each of these three input functions has a corresponding `tcltk` widget: `widget.marrayLayout`, `widget.marrayInfo`, and `widget.marrayRaw`. A screen–shot of the `marrayRaw` widget is shown in Figure 1; the command to launch the widget is as follows (here, `ext` specifies the image output file extension):

```
>  widget.marrayRaw(path=datadir, ext="spot")
```

Wrapper input functions
As mentioned before, for users who prefer command line input for a specific class of image processing output files, we have defined three additional functions. The functions `read.Spot`, `read.GenePix`, `read.Agilent` and `read.SMD` automate the creation of `marrayRaw` objects from `Spot`, `GenePix` and `Agilent` image analysis files, and from the Stanford Microarray Database (SMD) raw data files (`.xls`). The main arguments to these functions are a list of files and the directory path of the files. The following commands read two specific files from the `datadir` directory.

```
> fnames <- dir(path=datadir,pattern=paste("*", "spot", sep="\."))[1:2]
> swirl <- read.Spot(fnames, path=datadir,
          layout = swirl.layout,
          gnames = swirl.gnames,
          targets = swirl.samples)
```

Alternatively, without specifying any arguments, the functions `read.spot` and `read.GenePix` by default will read in all `Spot` or `GenePix` files within a current working directory. One has the option of setting the layout, probe, and target information manually at a later stage.

```
> swirl <- read.Spot()
> test.raw <- read.GenePix()
```

Note: Sweave. This document was generated using the `Sweave` function from the R *tools* package. The source file is in the `/inst/doc` directory of the package *marray*.

References

M. J. Buckley. *The Spot user's guide*. CSIRO Mathematical and Information Sciences, August 2000. `http://www.cmis.csiro.au/IAP/Spot/spotmanual.htm`.

Y. H. Yang, M. J. Buckley, S. Dudoit, and T. P. Speed. Comparison of methods for image analysis on cDNA microarray data. *Journal of Computational and Graphical Statistic*, 11(1), 2002.

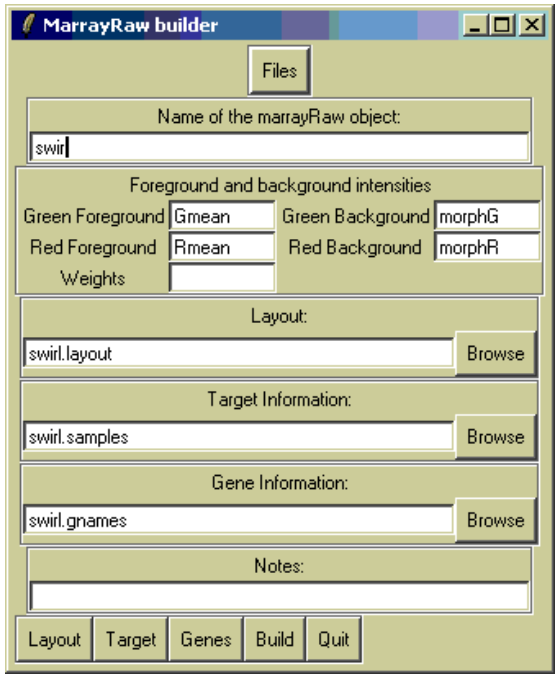

Figure 1: Screenshot of the widget for creating objects of class `marray` from image processing output files.