Textual description of makecdfenv

Wolfgang Huber and Rafael Irizarry

April 30, 2008

Contents

1 Introduction

The *makecdfenv* package is part of the Bioconductor[1] project. It can be used to create cdf environments from Affymetrix chip description files to be used with the package *affy*.

For many *CDF* files a pre-assembled package is available from `http://www.bioconductor.org/`. The following notes explain what to do if you want to create your own.

2 Creating a CDF package

To install packages, you need certain tools besides R itself. They are generally available on most Unix platforms. Under MS-Windows, if you use the precompiled binary distribution of R from CRAN, you will need to install some extra tools, like the *source package installation files* and *Perl*. If you think this is too complicated, you can contribute the package to Bioconductor and we will create a windows binary for you. Alternatively, you may skip this section and proceed to section 3

Let's say your CDF file is called `eggplantgenome.cdf` and is in your working directory. To make a package, simply write

[1] `http://www.bioconductor.org/`

```
R> make.cdf.package("eggplantgenome.cdf", species = "Solanum_sp")
```

This will create a subdirectory `eggplantgenomecdf` in your working directory, which contains the package. Please consult the help page for `make.cdf.package` to find out about further options.

Now, open a terminal with an operating system shell, and write

```
> R CMD INSTALL eggplantgenomecdf
```

This will install the package into your R. You are now ready to use the affy package to process eggplant genechips.

3 Creating CDF environments

If you do not choose to use the package creation mechanism, you can still produce a data structure (in R lingo, it is called an *environment*) that can be used by the affy package. Let's say your CDF file is called `EggPlantGenome.cdf` and is in your working directory. Simply write

```
R> EggPlantGenome = make.cdf.env("EggPlantGenome.cdf")
```

Please consult the help page for `make.cdf.env` to find out about further options.

4 Naming of CDF packages and environments

What should you call your package or environment?

`make.cdf.package` chooses a default name by stripping all non-letters and non-numbers from the CDF file name, and converting everything to lower case. In almost all cases you should use this default name for a package.

This is important because the *affy* package contains functionality to automatically detect the correct CDF package for a given chip. If the package is not found, it will try to download from the Bioconductor servers. If the package is not found there, it will error out. At that point the only recourse is to manually change the cdfName slot of the `AffyBatch`, or re-create the CDF package with the correct name.

You can obtain the CDF name that is associated with a CEL file through

```
R> pname <- cleancdfname(whatcdf("mycelfile.cel"))
```

Then call the package making function this way:

```
R> make.cdf.package("eggplantgenome.cdf", packagename=pname)
```

Unfortunately, the naming convention for environments is slightly different. Instead of using the `cleancdfname`, the *affy* package will look for an environment with the same *exact name as the original CDF file, minus the suffix (the '.CDF' part). So for instance, if the CDF name was EggPlantGenome.CDF, you would want to create an environment this way:*

```
R> EggPlantGenome <- make.cdf.env("EggPlantGenome.CDF")
```

*There are certain characters that will not work in a variable name (such as +, -, /, *). If your CDF name contains one of these characters, the only recourse is to either create a package (which by default will not contain these characters), or to manually change the cdfName of your* **AffyBatch** *so the affy package will find your environment. Say your CDF name is an-unacceptable-name.CDF.*

```
R> mycdfenv <- make.cdf.env("an-unacceptable-name.CDF")
R> dat <- ReadAffy()
R> dat@cdfName <- "mycdfenv"
```

In instances of **AffyBatch**, *the* **cdfName** *slot gives the name of the appropriate CDF file for the arrays that are represented in the* **intensity** *slot. The functions* **read.celfile**, **read.affybatch**, *and* **ReadAffy** *extract the CDF filename from the CEL files that they read. The function* **cleancdfname** *converts Affymetrix' CDF name to the name that is used in Bioconductor. Here are two examples:*

```
> cat("HG_U95Av2 is", cleancdfname("HG_U95Av2"), "\n")

HG_U95Av2 is hgu95av2cdf

> cat("HG-133A is", cleancdfname("HG-133A"), "\n")

HG-133A is hg133acdf
```

The method **getCdfInfo** *takes as an argument* **AffyBatch** *and returns the appropriate environment. If* **x** *is an* **AffyBatch**, *this function will look for an environment with name* **x@cdfName**.

5 Location-ProbeSet Mapping

On Affymetrix GeneChip arrays, several probes are used to represent genes in the form of probe sets. From a CEL file we get for each physical location, or cel, (uniquely identified by its x and y coordinates) an intensity. The CEL file also contains the name of the CDF file needed for the location-probe-set mapping. The CDF files store the name of the probe set related to each location on the array. We store this mapping information in R environments, such as the ones produced by **make.cdf.env** *or contained in the packages made by* **make.cdf.package**.

In affy, the x and y coordinates are internally stored as one number i. The mapping between (x, y) and i is provided by the functions `i2xy(i)` *and* `xy2i` *that are contained in each CDF package. They are very simple:*

$$\begin{aligned} i &= y s_x + x + 1 \\ x &= (i-1) \,\%\%\, s_x \\ y &= (i-1) \,\%/\%\, s_x, \end{aligned}$$

where s_x is the side length of the chip (measured in number of probes) in x-direction.