Modifying existing CDF environments to make alternative CDF environments

Laurent Gautier
October 17, 2016

Introduction

First we need to load the package:

```r
> library(altcdfsenvs)
```

The *Plasmodium / Anopheles* is taken as an example:

```r
> library(plasmodiumanophelescdf)
```

One will adapt easily the code below for other chips.

How to build a CdfEnvAffy object from the cdfenv package

The first step is to wrap the naked environment in the package *plasmodiumanophelescdf* in an object:

```r
> planocdf <- wrapCdfEnvAffy(plasmodiumanophelescdf, 712, 712, "plasmodiumanophelescdf")
> print(planocdf)
```

Instance of class CdfEnvAffy:

- **name**: *plasmodiumanophelescdf*
- **chip-type**: *plasmodiumanophelescdf*
- **size**: 712 x 712
- 22769 probe set(s) defined.

The numbers 712 and 712 correspond to the dimension of the array. If you do not know these numbers for your chip, the easiest (for the moment) is to read CEL data in an *AffyBatch* and call the function *print* on this object. Hopefully, the cdf packages offered on the bioconductor website will be modified, which will make this step (and the complication to know the dimension of the chip) unnecessary.
How to subset an environment using probe set ids
(see the vignette ‘n-genomes chips’)

How to work with given index / XY coordinates

Getting index
The method indexProbes is implemented for objects of class AltCdfEnvs

> #indexProbes(planocdf, "pm", "")

One can directly work on the CDF data, without having to load CEL data.

Removing probe sets
The function removeIndex let one remove probe sets given their index.

Multiple use of index
When crafting an AltCdfEnv, it can happen that probe indexes are used by several probe sets.
The unique.CdfEnvAffy is designed to help one to deal with the issue.

How to use this environment
(see the vignette ‘n-genomes chips’)

2