

Overview: Visualization of Microarray Data

Robert Gentleman

October 14, 2015

1 Overview

In this document we present a brief overview of the visualization methods that are available in Bioconductor project. To make use of these tools you will need the packages: *Biobase*, *annotate*, and *geneplotter*. These must be installed in your version of R and when you start R you must load them with the `library` command.

A quick word of warning regarding the interpretation of these plots. We can only plot where the gene is supposed to be. If there are translocations or amplifications these will not be detected by microarray analyses.

```
> library(geneplotter)
```

2 Whole Genome Plots

The functions `cPlot` and `cColor` allow the user to associate microarray expression data with chromosomal location. The plots can include any subset (by default all chromosomes are shown) of chromosomes for the organism being considered.

To make these plots we use the complete reference set of genes for the organism being studied. We must then obtain the chromosomal location (in bases) and orientation (which strand) the gene is on. Chromosomes are represented by straight lines parallel to the x -axis. Genes are represented by short perpendicular lines. All genes for the experiment (i.e. for an Affymetrix U95A analysis we show all genes on the chips).

The user can then change the color of different sets of the genes according to their needs.

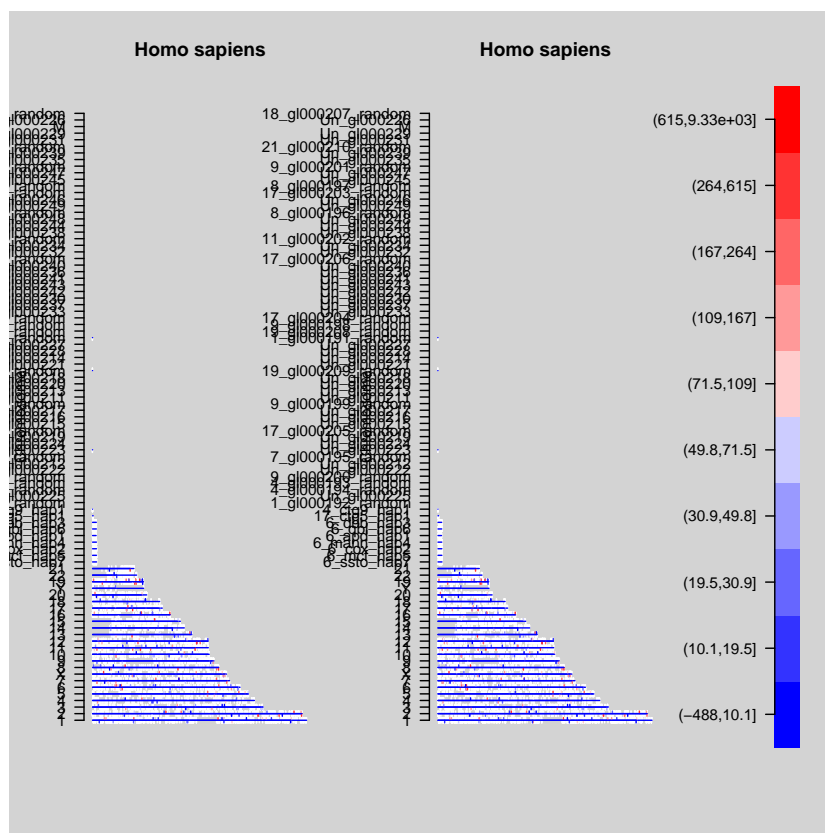
The original setup is done using `cPlot`. The subsequent coloring is done using `cColor`.

We will use the example data in `sample.ExpressionSet` to show how this function might be used.

```
> data(sample.ExpressionSet)
> eset = sample.ExpressionSet # legacy naming
> mytt <- function(y, cov2) {
+   ys <- split( y, cov2 )
+   t.test( ys[[1]], ys[[2]] )
+ }
> ttout <- esApply(eset, 1, mytt, eset$type)
> s1means <- sapply(ttout, function(x) x$estimate[1])
> s2means <- sapply(ttout, function(x) x$estimate[2])
> deciles <- quantile(c(s1means, s2means), probs=seq(0,1,.1))
> s1class <- cut(s1means, deciles)
> names(s1class) <- names(s1means)
> s2class <- cut(s2means, deciles)
> names(s2class) <- names(s2means)
```

Next we need to set up the graphics output. We do this in a rather complicated way. In the plot below we can compare the mean expression levels for genes in Group 1 with those in Group 2. The Group 1 values are in the left-hand plot and the Group 2 values are in the right-hand plot.

```
cols <- dChip.colors(10)
nf <- layout(matrix(1:3,nr=1), widths=c(5,5,2))
chrObj <- buildChromLocation("hgu95av2")
cPlot(chrObj)
cColor(featureNames(eset), cols[s1class], chrObj)
cPlot(chrObj)
cColor(featureNames(eset), cols[s2class], chrObj)
image(1,1:10,matrix(1:10,nc=10),col=cols, axes=FALSE,
      xlab="", ylab="")
axis(2, at=(1:10), labels=levels(s1class), las=1)
```



3 Single Chromosome Plots

A different view of the variation in expression level can be obtained by plotting characteristics of expression levels over contiguous regions of a chromosome. For these plots cumulative expression or per-gene expressions can be plotted. There are some issues of interpretation here (as in most places) – expression is not likely to be controlled too much by chromosomal locality. However these plots may be helpful in detecting deletions (of both chromatids) or amplifications, or other interesting features of the genome.

In this section we will show how one can explore a particular chromosome for an amplicon. The data arise from a study of breast cancer in the Iglehart Laboratory.

<environment: 0x10aac7ab0>

Figure 1: Cumulative expression levels of representative genes. The y-axis is 'Cumulative expression levels' (0 to 8000). The x-axis is 'Representative Genes' with 40 labels. A horizontal cyan line is at y=0. Above the line, there are 40 '+' signs, each aligned with a gene label. The '+' signs are at various heights, with some reaching up to 8000.

Representative Genes	Cumulative Expression Level (approx.)
31540 at	8000
31647 at	8000
31617 at	8000
31478 at	8000
31598 s at	0
31598 s at	0
31457 at	8000
31625 at	8000
31625 at	8000
31625 at	8000
31625 at	8000
31625 at	8000
31569 at	8000
31569 at	8000
31717 at	8000
31714 at	8000
31624 at	8000
31533 s at	8000
31508 at	8000
31662 at	8000
31662 at	8000
31662 at	8000
31662 at	8000
31662 at	8000
31603 at	8000
31499 s at	0
31499 s at	0
31499 s at	0
31499 s at	0
31642 at	8000
31642 at	8000
31591 s at	8000
31502 at	8000
31470 at	8000
31737 at	8000
31342 at	8000
31342 at	8000
31342 at	8000