Extending *oligo* with *SNPchip*

Robert Scharpf

February 20, 2008

Introduction

This vignette describes how to plot objects created from the package *oligo*. The *oligo* vignette creates an instance of `SnpCallSetPlus`, `crlmmOut`, from the call to the function `crlmm`. For purposes of illustration, I subset the object to only include SNPs on chromosome 1. I also took the liberty of adding chromosome and physical position to the featureData slot. This object can be loaded by

```
> library(SNPchip)
> data(crlmmOut)
> class(crlmmOut)

[1] "SnpCallSetPlus"
attr(,"package")
[1] "oligoClasses"
```

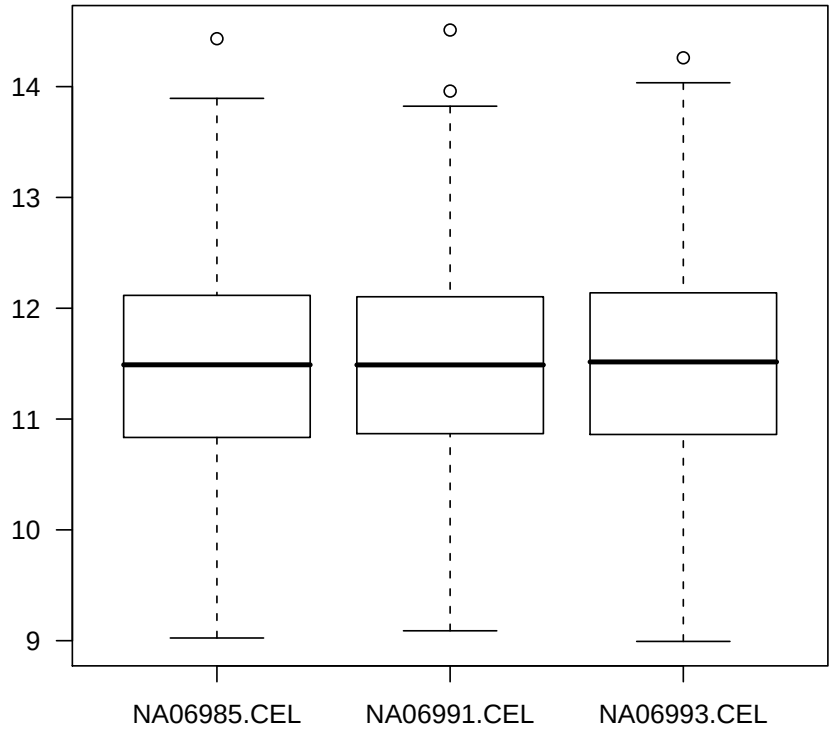
1 Creating an instance of `oligoSnpSet`

The elements in the assayData for instances of `SnpCallSetPlus` is dependent on the Affymetrix platform.

```
> annotation(crlmmOut)

[1] "pd.mapping50k.xba240"

> ls(assayData(crlmmOut))

[1] "antisenseThetaA" "antisenseThetaB" "calls"
[4] "callsConfidence" "senseThetaA"     "senseThetaB"
```

The method `calculateCopyNumber` averages the A and B allele intensities for objects of class `SnpCallSetPlus`. We assume that the averaged intensities are proportional to the copy number. Because the accessors for the A and B allele intensities are defined in the package *oligo*, we require this package here.

```
> if (require(oligo)) {
+     avgAB <- calculateCopyNumber(crlmmOut)
+     par(las = 1)
+     boxplot(as.data.frame(avgAB))
+ } else {
+     plot(1, 1, main = "oligo required for this plot")
+ }
```



Assuming that the average autosomal copy number in a sample is two, we can mean center the intensities using the autosomes. Because only chromosome 1 is present in this object, we assume that these subjects' had an average of two copies for each chromosome 1 SNP:

```
> colmn <- colMeans(avgAB)
> centerAB <- sweep(avgAB, 2, colmn) + log2(2)
```

One may use SNPchip to plot the copy number estimates versus physical position (as described in the SNPchip vignette) by coercing the `SnpCallSet` object to an instance of `oligoSnpSet`. This coercion assigns the uncentered average of the A and B intensities to the copyNumber element in assayData.

```
> snpset <- as(crlmmOut, "oligoSnpSet")
```

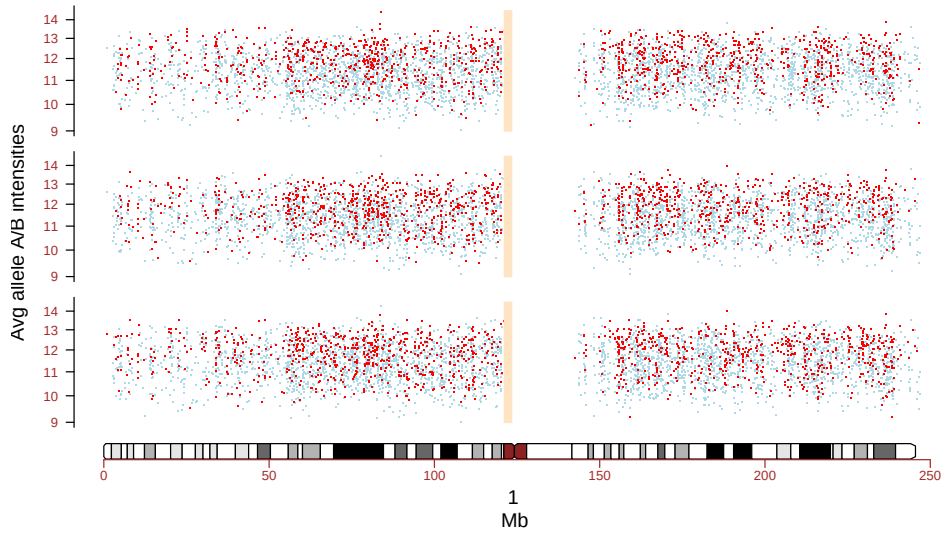
A plot of chromosome 1:

```
> gp <- new("ParSnpSet")
> gp <- getPar(gp, snpset)
```

```
[1] "one.ylim is FALSE. Calculating ylim based on the percentiles of the copy number distribution"
```

```
> gp$ylab <- "Avg allele A/B intensities"
> plotSnp(gp, snpset)
```

```
NULL
```



A hidden Markov model can be used to identify chromosomal alterations using genotype and copy number estimates as described in the *VanillaICE* vignette.

2 Combining objects that use different annotation packages

Here we illustrate how one may combine two objects of class `SnpCallSetPlus` that use different annotation packages: e.g., *pd.mapping50k.hind240* and *pd.mapping50k.xba240*. Following the *oligo* vignette, I created `hind` and `xba` instances of `SnpCallSetPlus`. The following code is not evaluated due to time constraints.

```
> library("oligo")
> library("hapmap100kxba")
> pathCelFiles <- system.file("celFiles", package = "hapmap100kxba")
> fullFilenames <- list.celfiles(path = pathCelFiles,
+     full.names = TRUE)
> aboutSamples <- data.frame(gender = c("female",
+     "female", "male"))
> rownames(aboutSamples) <- basename(fullFilenames)
> aboutVars <- data.frame(labelDescription = "male/female")
> rownames(aboutVars) <- "gender"
> pd <- new("AnnotatedDataFrame", data = aboutSamples,
+     varMetadata = aboutVars)
> xba <- justCRLMM(fullFilenames, phenoData = pd,
+     verbose = FALSE)
> library("hapmap100khind")
> pathCelFiles <- system.file("celFiles", package = "hapmap100khind")
> fullFilenames <- list.celfiles(path = pathCelFiles,
+     full.names = TRUE)
> aboutSamples <- data.frame(gender = c("female",
+     "female", "male"))
```

```
> rownames(aboutSamples) <- basename(fullFilenames)
> aboutVars <- data.frame(labelDescription = "male/female")
> rownames(aboutVars) <- "gender"
> pd <- new("AnnotatedDataFrame", data = aboutSamples,
+     varMetadata = aboutVars)
> hind <- justCRLMM(fullFilenames, phenoData = pd,
+     verbose = FALSE)
```

To combine into one object, simply

```
> callset <- combine(xba, hind)
```

3 Session Information

- R version 2.7.0 Under development (unstable) (2008-01-28 r44219), `powerpc-apple-darwin8.11.0`
- Locale: `C`
- Base packages: base, datasets, grDevices, graphics, methods, splines, stats, tools, utils
- Other packages: AnnotationDbi 0.99.23, Biobase 1.17.13, DBI 0.2-3, RSQLite 0.6-4, SNPchip 1.3.12, affxparser 1.9.5, oligo 1.3.15, oligoClasses 1.1.12, preprocessCore 0.99.22