

# Preprocessing & Genotyping Affymetrix Arrays for Copy Number Analysis

Rob Scharpf

March 31, 2011

## Abstract

This vignette describes the setup needed to analyze Affymetrix 6.0 (or 5.0) CEL files and the steps for preprocessing and genotyping. These steps must be completed prior to copy number analyses in `crlmm`. After completing these steps, users can refer to the `copynumber` vignette.

## 1 Set up

```
> library(crlmm)
> library(ff)
> library(cacheSweave)
```

This vignette analyzes HapMap samples assayed on the Affymetrix 6.0 platform. The annotation package for this platform is `genomewidesnp6Crlmm`. We assign the name of the annotation package without the `Crlmm` postfix to the name `cdfName`. We use the R package `cacheSweave` to cache long computations in this vignette. Users should refer to the `cacheSweave` package for additional details regarding caching.

```
> cdfName <- "genomewidesnp6"
```

The HapMap CEL files are stored in a local directory assigned to `pathToCels` in the following code. The genotyping step will create several files with `ff` extensions. We will store these files to the path indicated by `outdir`.

```
> pathToCels <- "/thumper/ctsa/snpmicroarray/hapmap/raw/affy/1m"
> if (getRversion() < "2.13.0") {
  rpath <- getRversion()
} else rpath <- "trunk"
> outdir <- paste("/thumper/ctsa/snpmicroarray/rs/ProcessedData/crlmm/",
  rpath, "/copynumber_vignette", sep = "")
> dir.create(outdir, recursive = TRUE, showWarnings = FALSE)
```

By providing the path in `outdir` as an argument to the R function `ldPath`, all of the `ff` files created during the genotyping step will be stored in `outdir`.

```
> ldPath(outdir)
```

The R functions `ocProbesets` and `ocSamples` manage the RAM required for our analysis. See the documentation for these functions and the `CopyNumberOverview` vignette for additional details.

```
> ocProbesets(1e+05)
> ocSamples(200)
```

Next we indicate the local directory that contains the CEL files. For the purposes of this vignette, we only analyze the CEPH ('C') and Yoruban ('Y') samples.

```

> celFiles <- list.celfiles(pathToCels, full.names = TRUE,
  pattern = ".CEL")
> celFiles <- celFiles[substr(basename(celFiles), 13, 13) %in%
  c("C", "Y")]

```

Finally, copy number analyses using `crmm` require specification of a batch variable that is used to indicate which samples were processed together. For example, if some of the samples were processed in April and another set of samples were processed in June, we could name the batches 'April' and 'June', respectively. A useful surrogate for batch is often the chemistry plate or the scan date of the array. For the HapMap CEL files analyzed in this vignette, the CEPH (C) and Yoruban (Y) samples were prepared on separate chemistry plates. In the following code chunk, we extract the population identifier from the CEL file names and assign these identifiers to the variable `plate`.

```

> plates <- substr(basename(celFiles), 13, 13)

```

## 2 Preprocessing and genotyping.

The preprocessing steps for copy number estimation includes quantile normalization of the raw intensities for each probe and a step that summarizes the intensities of multiple probes at a single locus. For example, the Affymetrix 6.0 platform has 3 or 4 identical probes at each polymorphic locus and the normalized intensities are summarized by a median. For the nonpolymorphic markers on Affymetrix 6.0, only one probe per locus is available and the summarization step is not needed. After preprocessing the arrays, the `crmm` package estimates the genotype using the CRLMM algorithm and provides a confidence score for the genotype calls. The function `genotype` performs both the preprocessing and genotyping.

```

> cnSet <- genotype(celFiles, batch = plates, cdfName = cdfName)

```

The value returned by `genotype` is an instance of the class `CNSet`. The normalized intensities, genotype calls, and confidence scores are stored as `ff` objects in the `assayData` slot. A concise summary of this object can be obtained through the `print` or `show` methods.

```

> print(cnSet)

```

```

CNSet (assayData/batchStatistics elements: ff_matrix)
CNSet (storageMode: lockedEnvironment)
assayData: 1852215 features, 180 samples
  element names: alleleA, alleleB, call, callProbability
protocolData
  rowNames: NA06985_GW6_C.CEL NA06991_GW6_C.CEL ...
            NA19240_GW6_Y.CEL (180 total)
  varLabels: ScanDate
  varMetadata: labelDescription
phenoData
  sampleNames: NA06985_GW6_C.CEL NA06991_GW6_C.CEL ...
              NA19240_GW6_Y.CEL (180 total)
  varLabels: SKW SNR gender
  varMetadata: labelDescription
featureData
  featureNames: SNP_A-2131660 SNP_A-1967418 ... CN_954736
              (1852215 total)
  fvarLabels: chromosome position isSnp
  fvarMetadata: labelDescription
experimentData: use 'experimentData(object)'
Annotation: genomewidesnp6
batch:      C 90, Y 90
batchStatistics: 29 elements, 1852215 features, 2 batches

```

Note that the object is relatively small as the intensities and genotype calls are stored on disk rather than in active memory.

```
> print(object.size(cnSet), units = "Mb")
```

```
134.3 Mb
```

We save the `cnSet` object in a local directory for subsequent copy number analysis in the `copynumber` vignette.

```
> saveObject <- function(outdir, cnSet) {  
  save(cnSet, file = file.path(outdir, "cnSet.rda"))  
  TRUE  
}  
> (cnset.saved <- saveObject(outdir, cnSet))
```

Users can proceed to the `copynumber` vignette for copy number analyses. See the `Infrastructure` vignette for additional details on the `CNSet` class, including an overview of the available accessors.

### 3 Session information

```
> toLatex(sessionInfo())
```

- R version 2.14.0 Under development (unstable) (2011-03-31 r55220), x86\_64-unknown-linux-gnu
- Locale: LC\_CTYPE=en\_US.iso885915, LC\_NUMERIC=C, LC\_TIME=en\_US.iso885915, LC\_COLLATE=en\_US.iso885915, LC\_MONETARY=C, LC\_MESSAGES=en\_US.iso885915, LC\_PAPER=en\_US.iso885915, LC\_NAME=C, LC\_ADDRESS=C, LC\_TELEPHONE=C, LC\_MEASUREMENT=en\_US.iso885915, LC\_IDENTIFICATION=C
- Base packages: base, datasets, graphics, grDevices, methods, stats, tools, utils
- Other packages: Biobase 2.11.9, bit 1.1-6, cacheSweave 0.4-5, cmlmm 1.9.23, ff 2.2-1, filehash 2.1-1, oligoClasses 1.13.22, stashR 0.3-3
- Loaded via a namespace (and not attached): affyio 1.19.2, annotate 1.29.3, AnnotationDbi 1.13.18, Biostrings 2.19.12, DBI 0.2-5, digest 0.4.2, ellipse 0.3-5, genefilter 1.33.1, IRanges 1.9.27, mvtnorm 0.9-96, preprocessCore 1.13.6, RSQLite 0.9-4, splines 2.14.0, survival 2.36-5, xtable 1.5-6