Smoothing raw copy number estimates

Rob Scharpf

March 29, 2011

Abstract

This vignette describes how segmentation algorithms and hidden Markov models implemented in the R packages DNAcopy and VanillaICE packages, respectively, can be interfaced with the crlmm raw copy number estimates. This vignette assumes successfull completion of the copynumber vignette.

1 Set up

As in the previous vignettes, we load the required libraries and specify a path for storing output files.

```
R> library(ff)
R> library(crlmm)
R> library(cacheSweave)
R> require(DNAcopy)
R> require(VanillaICE)
R> if (getRversion() < "2.13.0") {
     rpath <- getRversion()
 } else rpath <- "trunk"
R> outdir <- paste("/thumper/ctsa/snpmicroarray/rs/ProcessedData/crlmm/",
     rpath, "/copynumber_vignette", sep = "")
```

```
R> ldPath(outdir)
R> setCacheDir(outdir)
R> ocProbesets(50000)
R> ocSamples(200)
```

We begin by loading the `cnSet` object created by the `AffymetrixPreprocessCN` vignette.

```
R> if (!exists("cnSet")) load(file.path(outdir, "cnSet.rda"))
```

2 Interfacing with the DNAcopy and VanillaICE packages

This section is incomplete.

As discussed in the `copynumber` vignette, we create an instance of *oligoSnpSet* class by using the method `as` for subsets of the markers to keep the RAM at manageable levels (one can specify smaller values of ocProbesets() to further reduce the RAM).

```
R> marker.index <- which(!is.na(chromosome(cnSet)))
R> marker.index <- marker.index[order(chromosome(cnSet)[marker.index],
     position(cnSet)[marker.index])]
R> marker.indices <- splitIndicesByLength(marker.index,
     ocProbesets())
R> cbs.results <- hmm.results <- vector("list", length(marker.indices))
R> open(cnSet)
```

```
R> for (i in 1:5) {
     cnset.subset <- cnSet[marker.indices[[i]], seq(length = ncol(cnSet))]
     system.time(oligoset <- as(cnset.subset, "oligoSnpSet"))
     rm(cnset.subset)
     gc()
     stopifnot(class(copyNumber(oligoset)) == "matrix")
     CNA.object <- CNA(genomdat = copyNumber(oligoset),
         chrom = chromosome(oligoset), maploc = position(oligoset),
         data.type = "logratio", sampleid = sampleNames(oligoset))
     smu.object <- smooth.CNA(CNA.object)
     rm(CNA.object)
     gc()
     cbs.results[[i]] <- segment(smu.object)
     sample.index <- (1:(ncol(smu.object) - 2)) + 2
     copyNumber(oligoset) <- as.matrix(smu.object[, sample.index])
     sds <- robustSds(copyNumber(oligoset))
     cnConfidence(oligoset) <- 1/sds
     hmmOpts <- hmm.setup(oligoset, c("hom-del", "hem-del",
         "normal", "amp1copy", "amp2copy"), copynumberStates = c(0:4),
         normalIndex = 3, log.initialP = rep(log(1/5),
             5), prGenotypeHomozygous = c(0.8, 0.99, 0.7,
             0.75, 0.75))
     hmm.results[[i]] <- hmm(oligoset, hmmOpts, verbose = FALSE,
         TAUP = 1e+10)
     rm(oligoset, sds, smu.object, CNA.object, hmmOpts)
     gc()
 }
R> close(cnSet)
R> save(hmm.results, file = file.path(outdir, "hmm.results.rda"))
R> save(cbs.results, file = file.path(outdir, "cbs.results.rda"))
```