Identifying Copy Number Polymorphisms

Jacob Carey, Steven Cristiano, and Robert Scharpf

October 13, 2015

Contents

1 Introduction

Identify consensus start and stop coordinates of a copy number polymorphism

The collection of copy number variants (CNVs) identified in a study can be encapulated in a GRangesList, where each element is a GRanges of the CNVs identified for an individual. (For a study with 1000 subjects, the GRangesList object would have length 1000 if each individual had 1 or more CNVs.) For regions in which CNVs occur in more than 2 percent of study participants, the start and end boundaries of the CNVs may differ because of biological differences in the CNV size as well as due to technical noise of the assay and the uncertainty of the breakpoints identified by a segmentation of the genomic data. Among subjects with a CNV called at a given locus, the `consensusCNP` function identifies the largest region that is copy number variant in half of these subjects.

2 Simulating CNP data

Included in the CNPBayes package are objects of class `SnpArrayExperiment` and `GRangesList`. We begin by loading the necessary libraries and data.

```
se <- readRDS(system.file("extdata", "simulated_se.rds", package="CNPBayes"))
grl <- readRDS(system.file("extdata", "grl_deletions.rds", package="CNPBayes"))
suppressMessages(library(CNPBayes))
suppressMessages(library(VanillaICE))
```

The object `se` contains log R ratios and B Allele Frequencies, and the object `grl` is a `GRangesList` of simulated deletions.

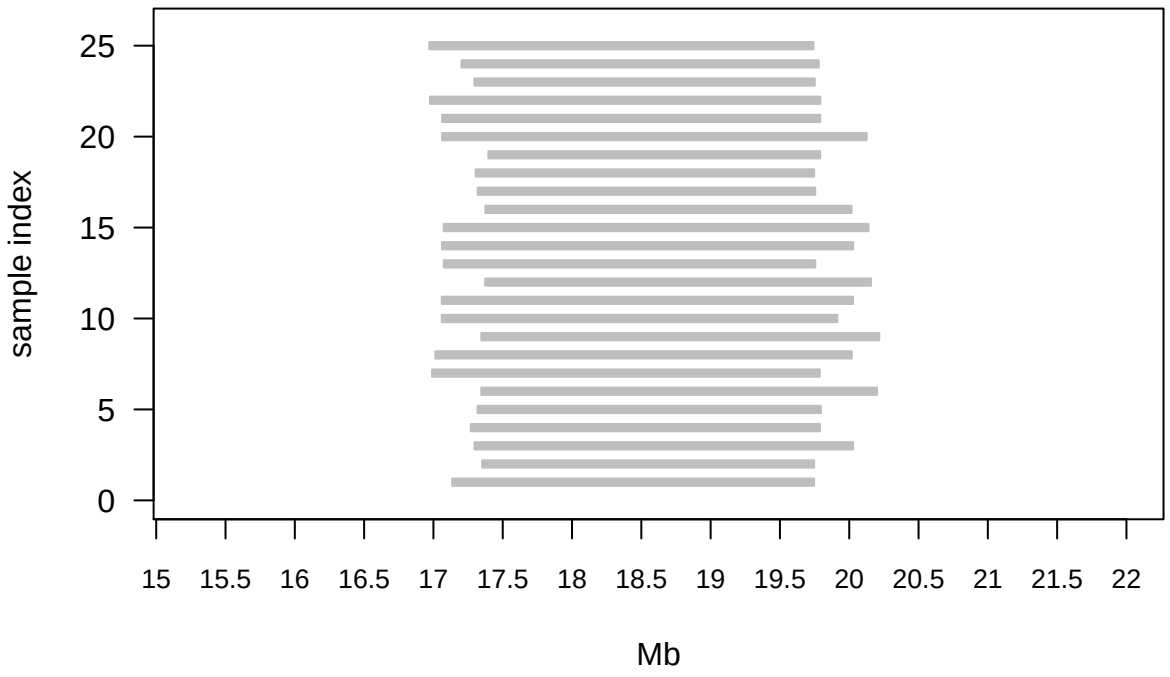
3 Finding CNPs

After reading this saved data, we visualize the CNVs.

```
cnv.region <- consensusCNP(grl[1], max.width=5e6)
## unlist GRangesList...
## find consensus regions...
## .
## Dropping CNV regions failing min.width and max.width criteria. See ?consensusCNP to relax these setting
```

```
i <- subjectHits(findOverlaps(cnv.region, rowRanges(se)))

xlim <- c(min(start(se)), max(end(se)))
par(las=1)
plot(0, xlim=xlim, ylim=c(0, 26), xlab="Mb", ylab="sample index", type="n",
     xaxt="n")
at <- pretty(xlim, n=10)
axis(1, at=at, labels=round(at/1e6, 1), cex.axis=0.8)
rect(start(grl), seq_along(grl)-0.2, end(grl), seq_along(grl)+0.2,
     col="gray", border="gray")
```



Before further analysis can be performed, the log R ratios in a CNV region must be summarized to a one dimensional object (Cardin et al. 2011). Within each CNP locus, log R ratios at each SNP are summarized by sample. Below are examples of using the median and the first principal component to create a one dimensional summary.

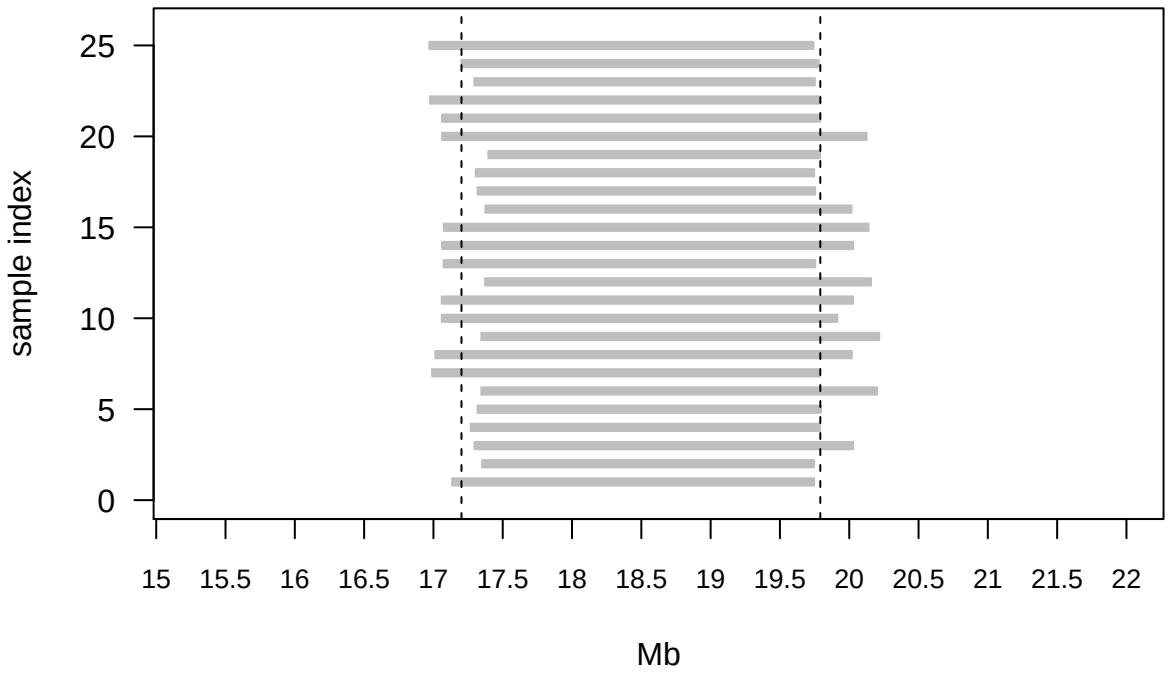
3.1 CNP Boundaries with Median Summary

To summarize samples within a CNV locus by the median log R ratios, we define the largest region that spans 50 percent of the samples using the function `consensusCNP`. Using log R ratios of SNPs contained in this region, the median is

taken across samples. Because the deletions in this example are large (> 2Mb), we specify a large value for `max.width` to avoid filtering these CNVs.

```
cnv.region <- consensusCNP(grl, max.width=5e6)
## unlist GRangesList...
## find consensus regions...
## .
## Dropping CNV regions failing min.width and max.width criteria. See ?consensusCNP to relax these setting
```

```
i <- subjectHits(findOverlaps(cnv.region, rowRanges(se)))
med.summary <- matrixStats::colMedians(lrr(se)[i, ], na.rm=TRUE)
par(las=1)
plot(0, xlim=xlim, ylim=c(0, 26), xlab="Mb", ylab="sample index", type="n",
     xaxt="n")
at <- pretty(xlim, n=10)
axis(1, at=at, labels=round(at/1e6, 1), cex.axis=0.8)
rect(start(grl), seq_along(grl)-0.2, end(grl), seq_along(grl)+0.2,
     col="gray", border="gray")
abline(v=c(start(cnv.region), end(cnv.region)), lty=2)
```



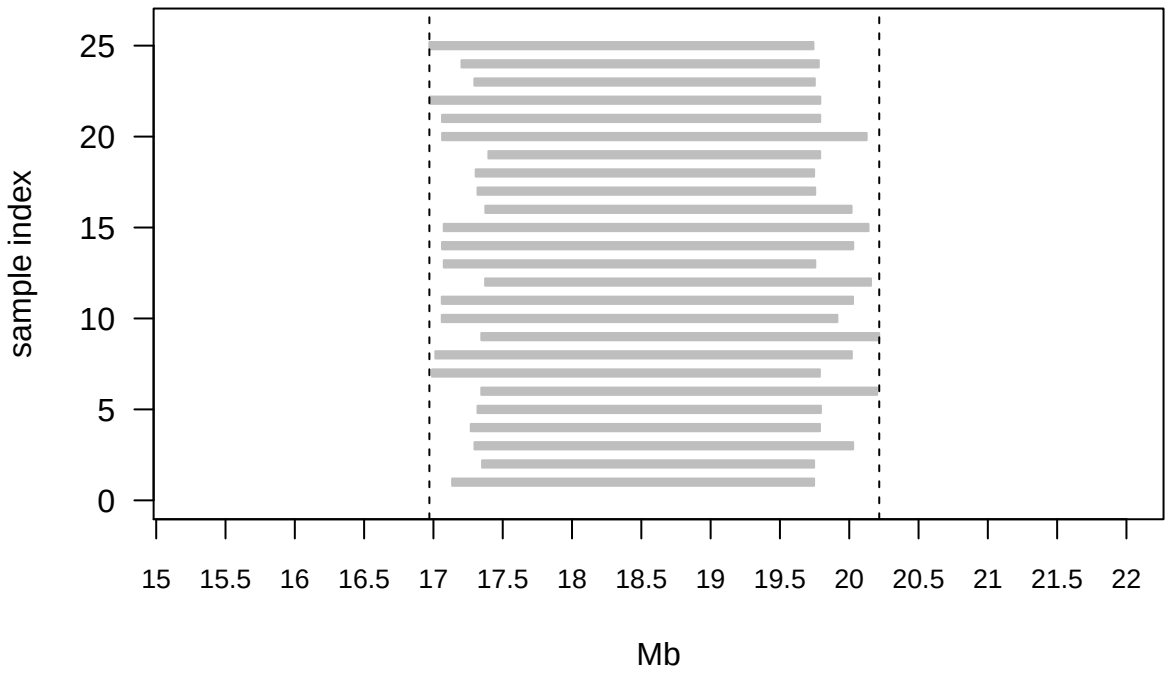
3.2 CNP Boundaries with PCA Summary

Another method for summarizing the log R ratops is by the first principal component on the markers for the entire region (Cardin et al. 2011). A possible disadvantage of this approach is that the scale of the loadings makes it more difficult to interpret the copy number of the mixture components. Instead, the median log R ratio is adequate and retains the original scale. An advantage of the principal component method is that the minimum `start` and maximum `end` can be

used to define the CNV region. The principal component method should downweight markers that are not consistent with the Copy Number Variation.

```
cnv.region2 <- reduce(unlist(grl))
i.pc <- subjectHits(findOverlaps(cnv.region2, rowRanges(se)))
x <- lrr(se)[i.pc, ]
nas <- rowSums(is.na(x))
na.index <- which(nas > 0)
x <- x[-na.index, , drop=FALSE]
pc.summary <- prcomp(t(x))$x[, 1]
meds.for.pc <- matrixStats::colMedians(x, na.rm=TRUE)
if(cor(pc.summary, meds.for.pc) < 0) pc.summary <- -1*pc.summary

par(las=1)
plot(0, xlim=xlim, ylim=c(0, 26), xlab="Mb", ylab="sample index", type="n",
     xaxt="n")
at <- pretty(xlim, n=10)
axis(1, at=at, labels=round(at/1e6, 1), cex.axis=0.8)
rect(start(grl), seq_along(grl)-0.2, end(grl), seq_along(grl)+0.2,
     col="gray", border="gray")
abline(v=c(start(cnv.region2),
           end(cnv.region2)), lty=2)
```

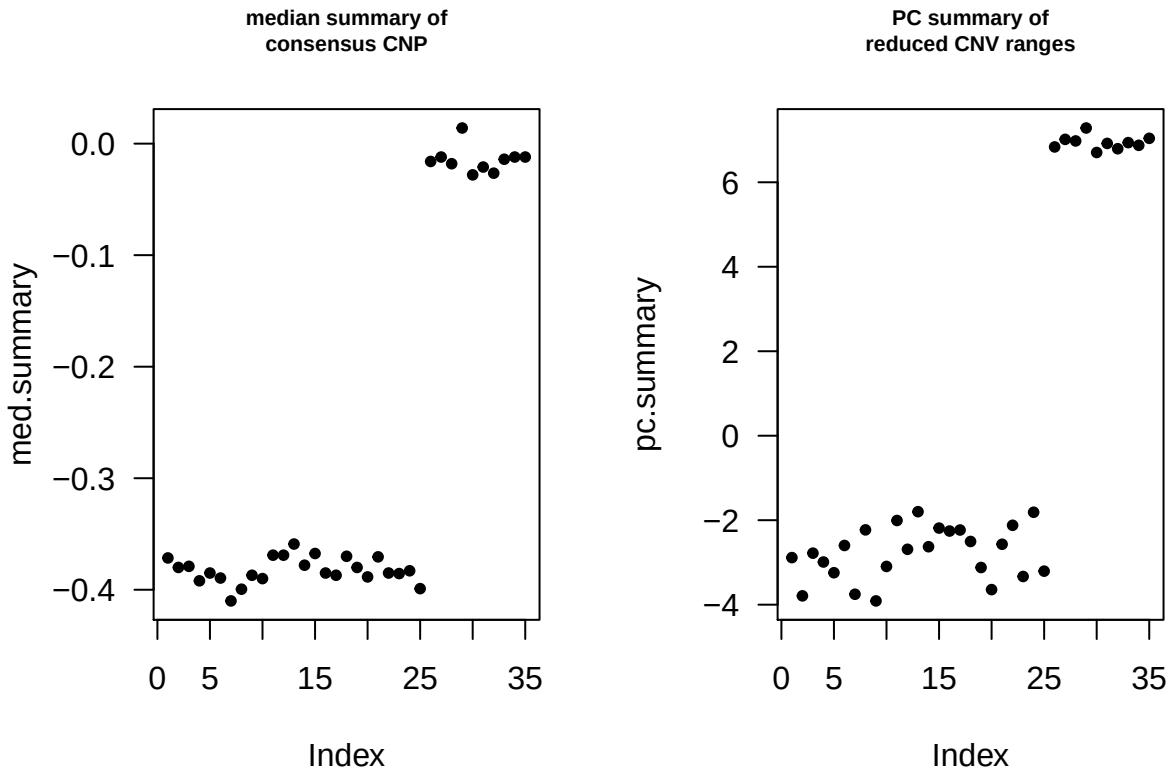


Note that boundaries of created by principal component summary method are wider than those in the above median summary method. In this example, the boundaries are not significantly different, but in samples with ragged starts and ends, the PCA method will provide greater coverage.

3.3 Summary Plots

Finally, we plot the one dimensional summaries.

```
par(mfrow=c(1,2), las=1)
plot(med.summary, main="median summary of\nconsensus CNP", cex.main=0.7, pch=20)
plot(pc.summary, main="PC summary of\nreduced CNV ranges", cex.main=0.7, pch=20)
```



3.4 Filtering

Loci with no additions or deletions can be identified as normal distribution. If a loci has log R ratios drawn from a normal distribution, then the loci should not be included in the model fitting process. Removing these unneeded loci can reduce the computational burden. An easy way to identify such loci is by using the Shapiro-Wilk test as a test of normality (Shapiro and Wilk 1965). To be conservative, we suggest retaining only those loci which have a p value < 0.1.

```
shapiro.test(med.summary)
##
##  Shapiro-Wilk normality test
##
## data:  med.summary
## W = 0.63062, p-value = 3.652e-08
```

3.5 Mixture Model

After deciding which one dimensional summary to use and summary of the data is complete, one can proceed to fitting a `MixtureModel` to the data. Models can be hierarchical over the batches, or marginal over the batches. Please refer to the CNPBayes vignette for fitting a `MixtureModel`.

References

Cardin, Niall, Chris Holmes, Peter Donnelly, and Jonathan Marchini. 2011. "Bayesian Hierarchical Mixture Modeling to Assign Copy Number from a Targeted CNV Array." *Genet. Epidemiol.* doi:10.1002/gepi.20604.

Shapiro, S. S., and M. B. Wilk. 1965. "An Analysis of Variance Test for Normality (Complete Samples)." *Biometrika* 52 (3-4). Oxford University Press (OUP): 591–611. doi:10.1093/biomet/52.3-4.591.