hsegHMM Package

August 24, 2018

Contents

1 Introduction

Characterizing somatic copy number alterations (SCNAs) is important for understanding tumorigenesis, cancer etiology and prognosis. In normal cells, two copies of chromosome are inherited from both parents. In contrast, tumor cells frequently contain alterations in copy numbers across the chromosomes, such as deletions, insertions, or amplifications among others. In addition, tumor tissues always contain normal cells (reduced tumor purity) and frequently show an abnormal number of chromosomes (aneuploidy). These characteristics of the cancer genome and tissue heterogeneity complicate the estimation of SCNAs.

This package considers next-generation sequencing (NGS) platform-based data for studying SCNAs. The NGS technology provides high resolution at the single basepair, which comes with mapping bias and the tendency for hypersegmentation. Mapping bias occurs from higher mapping rates for the reference allele than those for the variant allele at heterozygous loci. This bias leads to incorrect interpretations of allele-specific SCNAs. Hypersegmentation is also a major challenge in NGS-based allele-specific SCNA.

This package utilizes a novel hidden Markov modeling approach (hsegHMM) for allele-specific SCNA analysis accounting for the hypersegmentation and simultaneously conducts the segmentation and genotype mixture modeling required to identify SCNAs across chromosomes.

2 Example using simulated TCGA data and the facets package

Load the facets and hsegHMM packages

```
> library(facets)
> library(hsegHMM)
```

Get the path to the data.

```
> datafile <- system.file("sampleData", "facets_data.csv.gz", package="hsegHMM")
```

Read in the data

```
> tcga <- readSnpMatrix(datafile)
```

Set a seed and pre-process the data

```
> set.seed(2017)
> xx <- preProcSample(tcga,ndepth=5)
```

Process the data to get log(ratio) and log(OR) values

```
> oo <- procSample(xx,cval=150)
```

Pull out log(ratio) and log(OR) values

```
> inputs <- oo$jointseg[,11:12]
> lr <- inputs[,1]
> logor <- inputs[,2]
```

For faster convergence, take a subset.

```
> idx_thin <- seq(1, length(lr), 30)
> lr       <- lr[idx_thin]
> logor    <- logor[idx_thin]
```

Call the hsegHMM main function. Note that stopTol is set to 1 for faster convergence.

```
> ret <- hsegHMM_T(lr, logor, purity=0.8, ploidy=1.5, logR.var=0.5,
+                      logOR.var=0.5, df=3, stopTol=1)
```

```
Iteration 0: loglike = -4286.56721665417
Iteration 1: loglike = -1171.19237173397
Iteration 2: loglike = -791.419854848803
Iteration 3: loglike = -789.748868699618
Iteration 4: loglike = -789.205805820543
```

Get the genotype states and copy number

```
> gtype <- ret$genoStates
> ctz0  <- ret$copyNumber
```

Get the genotype status which gives the maximum posterior probability at each location

```
> idx_hgtype <- ret$which.max.post.prob
```

Get the copy number at the maximum posterior probability

```
> hat_ctz    <- ctz0[idx_hgtype]
```

Get the expectation of logR and logOR based on estimates from hsegHMM

```
> hat_logr  <- ret$logR_hat
> hat_logor <- ret$logOR_hat
```

Create a plot for the tumor copy number profile across chromosomes. The blue dots are observed values and red bars are estimates. The first two panels show the profiles of logR and logOR over the entire chromosomes. The last two panels indicate estimated copy numbers and genotype for each sequence over the entire chromosomes.

```
> par(mfrow=c(4,1))
> plot(1:length(lr), lr, pch=20, col="blue", cex=0.5,
+   xlab="genetic location", cex.lab=1.5, ylab="logR")
> points(1:length(hat_logr),hat_logr, pch=20, col="red", cex=0.5)
> plot(1:length(logor), logor, pch=20, col="blue", ylab="logOR",
+    cex.lab=1.5,xlab="genetic location",cex=0.5,
+    ylim=c(min(na.omit(logor)),max(na.omit(logor))))
> points(1:length(hat_logor), hat_logor, pch=20,col="red", cex=0.5)
> points(1:length(hat_logor), -hat_logor, pch=20,col="red", cex=0.5)
> plot(1:length(hat_ctz), hat_ctz, pch=20, ylab="copy number",cex.lab=1.5,
+   xlab="genetic location", ylim=c(0,max(hat_ctz)),col="red", cex=0.5)
> plot(1:length(idx_hgtype), idx_hgtype-1, pch=20, ylab="",cex.lab=1.5,
+  xlab="genetic location",ylim=c(0,max(idx_hgtype-1)), yaxt="n",
+  col="red", cex=0.5)
> axis(2, at=c(0:max(idx_hgtype-1)), labels=gtype[1:max(idx_hgtype)], las=2)
```

3 Session Information

```
> sessionInfo()

R version 3.5.0 (2018-04-23)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: CentOS Linux 7 (Core)

Matrix products: default
BLAS/LAPACK: /usr/local/intel/compilers_and_libraries_2018.1.163/linux/mkl/lib/intel64_lin

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=en_US.UTF-8        LC_COLLATE=C
 [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] hsegHMM_0.1.0   facets_0.5.14   pctGCdata_0.2.0

loaded via a namespace (and not attached):
[1] compiler_3.5.0 tools_3.5.0
```