CONY manual

(R program)

R topics documented:

CONY.R 1

AdjRD 4

CalRD 5

ComResult 7

EstPar 9

RunCONY 10

UsedRD 13

WindowInfo 15

CONY.R *Copy number variation detection tool via a Bayesian procedure*

Description

CONY, a copy number variation (CNV) detection tool via a Bayesian procedure, adopts a hierarchical model and an efficient reversible jump Markov chain Monte Carlo (RJMCMC) inference algorithm for whole genome sequencing read depths data. CONY can be applied not only to an individual for estimating the absolute number of copies but also to case-control pairs for detecting patient specific relative variations.

Details

CONY is used to identify CNVs from sequencing through the several steps, including windows definition and information summary (`WindowInfo` function), read depth calculation, adjustment, and transformation (`CalRD`, `AdjRD`, `UsedRD` functions), parameter settings for Bayesian hierarchical model and RJMCMC (`EstPar` function), RJMCMC simulation (`RunCONY` function), and CNV regions identification (`ComResult` function).

Chromosome 20 from two samples (NA12156 and NA12878) provided by 1000 Genomes project are taken for examples. For the single sample analysis, NA12878 is used. For the paired samples analysis, NA12878 is described as case and NA12156 as control. Samples' reads that have been mapped to hg19 reference genome with default adjustments are downloaded from 1000 Genomes project ftp (http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/phase3/data). The corresponding data are also provided in https://github.com/weiyuchung/CONY.

References

Consortium, G. P., 2010 A map of human genome variation from population-scale sequencing. Nature 467**:** 1061-1073.

Wei, Y-C and Huang G-H. CONY: A Comprehensive Bayesian Procedure for Detecting Copy Number Variations from Sequencing Read Depths.

Examples

```
## Detect absolute copy number for single sample analysis
CONY.TempRegion=WindowInfo (target.df=
    as.data.frame(matrix(c(20,1,63025520),1,3,dimnames=list(c(
    "1"),c("seqname","start","end")))),RefFaFileName="chr20.fa
    ",WindowSize=100)

# The file (NA12878.chrom20.ILLUMINA.bwa.CEU.
    low_coverage.20121211.chr20.sorted.rmdup.bam) provides
    the sorted reads after removing PCR duplication via
    SAMTools sort and rmdup functions.
# The file (NA12878.low.chr20.Used.txt) provides base read
    depths, in which nucleotide with base-calling quality
    score less than 30 and reads with mapping quality scores
    less than 30 are filter out via SAMTools mpileup function.
CalRD(TempRegion=CONY.TempRegion, CRDMethod="SumUp",
    SampleBamFileName="NA12878.chrom20.ILLUMINA.bwa.CEU.low_co
    verage.20121211.chr20.sorted.rmdup.bam",
    MPileCountFileName=
    "NA12878.low.chr20.Used.txt",SampleName="NA12878",TargetCh
    r="chr20", WindowSize=100)

AdjRD (CRDMethod= "SumUp", TargetChr="chr20",
    SampleName="NA12878")

UsedRD(CRDMethod="SumUp", AnaMethod= "Single",TargetChr="chr20",
    SampleName= "NA12878")

EstPar(CRDMethod="SumUp", AnaMethod="Single", TargetChr="chr20",
    SampleName="NA12878",NCN=5)

RunCONY(CRDMethod="SumUp", AnaMethod="Single",
    TargetChr="chr20", SampleName="NA12878", RunTime =
    300000,BurnN = 5000,RTN = 1000,BCPoint = 20,
    FragLength=500000)

ComResult (CRDMethod="SumUp", AnaMethod="Single",
    TargetChr="chr20", SampleName="NA12878")
```

Detect relative copy number for paired samples analysis

```
CONY.TempRegion=WindowInfo (target.df=
    as.data.frame(matrix(c(20,1,63025520),1,3,dimnames=list(c("
    1"),c("seqname","start","end")))),RefFaFileName="chr20.fa",
    WindowSize=100)
```

 # The file descriptions for NA12156 are the same as NA12878

```
CalRD(TempRegion=CONY.TempRegion, CRDMethod="SumUp",
  SampleBamFileName="NA12878.chrom20.ILLUMINA.bwa.CEU.low_cover
  age.20121211.chr20.sorted.rmdup.bam", MPileCountFileName=
  "NA12878.low.chr20.Used.txt",SampleName="NA12878",TargetChr="
  chr20", WindowSize=100)
AdjRD (CRDMethod= "SumUp", TargetChr="chr20",
  SampleName="NA12878")

CalRD(TempRegion=CONY.TempRegion, CRDMethod="SumUp",
  SampleBamFileName="NA12156.chrom20.ILLUMINA.bwa.CEU.low_cover
  age.20120522.chr20.sorted.rmdup.bam", MPileCountFileName=
  "NA12156.low.chr20.Used.txt",SampleName="NA12156",TargetChr="
  chr20", WindowSize=100)
AdjRD (CRDMethod= "SumUp", TargetChr="chr20",
  SampleName="NA12156")

UsedRD(CRDMethod="SumUp", AnaMethod= "Paired",TargetChr="chr20",
  SampleName="NA12878",ControlName="NA12156")

EstPar(CRDMethod="PointR", AnaMethod="Paired", TargetChr="chr20",
  SampleName="NA12878",NCN=3)

RunCONY(CRDMethod="SumUp", AnaMethod="Paired", TargetChr="chr20",
  SampleName="NA12878", RunTime = 300000,BurnN = 5000,RTN =
  1000,BCPoint = 20, FragLength=500000)

ComResult (CRDMethod="SumUp", AnaMethod="Paired",
  TargetChr="chr20", SampleName="NA12878")
```

AdjRD *Window read depth adjustment*

Description

Raw window read depths are adjusted for the percentages of indefinable bases and G- and C-contents.

Details

Two major biases (percentage of indefinable bases and GC-contents) should be adjusted for raw window read depths to purify the evidence of CNVs. The file with window information and raw read depths (generated from `CalRD` function) is used. A simple adjustment is used for the indefinable-bases issue and local regression model is adopted for the GC-content.

Usage

```
AdjRD(CRDMethod=c("SumUp","PointR"),
 TargetChr,
 SampleName)
```

Argument

`CRDMethod`	`character`, the method for window read depth calculation. `"SumUp"` for sum up the base read depth and `"PointR"` for count the number of reads which middle or start position located in the specific window
`TargetChr`	`character`, the target chromosome.
`SampleName`	`character`, the name of the case sample

Value

Adjusted window read depths are added to the window information file (generated from `CalRD` function). The file with 8 columns includes the chromosome name (`seqname`), start position (`start`), end position (`end`), width length (`width`), percentage of indefinable base (`nonAmb`), GC

percentage (GC), raw window read depth (ARD), and adjusted window read depth (AdjRD) for each window. The output is saved as a text file (CONY.2-TempRegion.*.AdjRD.txt) and it would be used for the downstream steps.

Example

```
AdjRD(CRDMethod= "SumUp", TargetChr="chr20",
    SampleName="NA12878")
```

CalRD *Window read depth calculation for each sample*

Description

Two options are provided for calculating window read depths. In CONY approach, we suggest that window read depths are summed up from base depths. The traditional method, window read depths are counted from the number of reads which middle or start position located in the specific window, is also available.

Usage

```
CalRD(TempRegion=CONY.TempRegion,
 CRDMethod=c("SumUp","PointR"),
 SampleBamFileName,
 MPileCountFileName,
 SampleName,
 TargetChr,
 WindowSize=100)
```

Argument

TempRegion	data.frame, window information file that derived from WindowInfo function
CRDMethod	character, the method for window read depth calculation. "SumUp" for sum up the

	base read depth and "PointR" for count the number of reads which middle or start position located in the specific window
SampleBamFileName	character, the name of the alignment read file (.bam). The command is essential for CRDMethod="PointR" only
MPileCountFileName	character, the name of base read depth file (.txt). The command is essential for CRDMethod="SumUp" only
SampleName	character, the name of the case sample
TargetChr	character, the target chromosome
WindowSize	numeric, window size with default 100 (bp)

Value

Window read depths are added to the window information file (generated from WindowInfo function). The file with 7 columns includes the name of chromosome (seqname), start position (start), end position (end), width length (width), percentage of indefinable base (nonAmb), GC percentage (GC), and raw window read depth (ARD) for each window. The output is saved as a text file (CONY.1-TempRegion.*.RD.txt) and it would be used for the downstream steps.

Imports

ExomeCopy and IRanges R package

References

Lawrence, M., W. Huber, H. Pages, P. Aboyoun, M. Carlson *et al.*, 2013 Software for computing and annotating genomic ranges. PLoS Comput Biol 9**:** e1003118.

Love, M., M. M. Love, D. IRanges, R. GenomicRanges, S. Biostrings *et al.*, 2013 Package 'exomeCopy'.

Example

```
## For SumUp method
# The file (NA12878.low.chr20.Used.txt) provides base read
   depths, in which nucleotide with base-calling quality
   score less than 30 and reads with mapping quality scores
   less than 30 are filter out via SAMTools mpileup function.
 CalRD(TempRegion=CONY.TempRegion, CRDMethod="SumUp",
    SampleBamFileName=NA, MPileCountFileName=
    "NA12878.low.chr20.Used.txt",SampleName="NA12878",TargetCh
    r="chr20", WindowSize=100)

## For PointR method
# The file (NA12878.chrom20.ILLUMINA.bwa.CEU.
   low_coverage.20121211.chr20.sorted.rmdup.bam) provides
   the sorted reads after removing PCR duplication via
   SAMTools sort and rmdup functions.
CalRD(TempRegion=CONY.TempRegion, CRDMethod="PointR",
  SampleBamFileName="NA12878.chrom20.ILLUMINA.bwa.CEU.low_cover
  age.20121211.chr20.sorted.rmdup.bam", MPileCountFileName=
  NA,SampleName="NA12878",TargetChr="chr20", WindowSize=100)
```

ComResult *Copy number variation regions identification*

Description

Estimated copy number (CN) status via RJMCMC simulation and zero status from preprocessing step are combined to identify copy number variation regions.

Usage

```
ComResult(CRDMethod=c("SumUp","PointR"),
 AnaMethod=c("Single","Paired"),
 TargetChr,
 SampleName)
```

Value

The estimated CN state of each window for all lane are combined as one file (`CONY.Result.*.Window.txt`). Finally, the identified copy number regions are provided (`CONY.Result.*.CNRegionAll.txt`) with 3 columns, including start position (`start`), end position (`end`), and CN status (`CN`) of each copy number regions.

Argument

`CRDMethod`	`character`, the method for window read depth calculation. `"SumUp"` for sum up the base read depth and `"PointR"` for count the number of reads which middle or start position located in the specific window
`AnaMethod`	`character`, the sample designs. `"Single"` for detecting absolute number of copies (single sample analysis) and `"Paired"` for detecting patient specific relative CNVs (paired samples analysis)
`TargetChr`	`character`, the target chromosome
`SampleName`	`character`, the name of case sample

Examples

```
## For single sample analysis
 ComResult (CRDMethod="SumUp", AnaMethod="Single",
    TargetChr="chr20", SampleName="NA12878")
```

```
## For paired samples analysis (NA12878 as case and NA12156
    as control)
ComResult (CRDMethod="SumUp", AnaMethod="Paired",
  TargetChr="chr20", SampleName="NA12878")
```

EstPar *Parameter settings for Bayesian hierarchical model and RJMCMC*

Description

Parameters (mean, variance, and proportion of each CN status) for RJMCMC simulation are estimated. Please see the main manuscript and supplementary of CONY for the details of model and hyper-parameters settings.

Usage

```
EstPar(CRDMethod=c("SumUp","PointR"),
 AnaMethod=c("Single","Paired"),
 TargetChr,
 SampleName,
 NCN)
```

Argument

CRDMethod	character, the method for window read depth calculation. "SumUp" for sum up the base read depth and "PointR" for count the number of reads which middle or start position located in the specific window
AnaMethod	character, the sample designs. "Single" for detecting absolute number of copies (single sample analysis) and "Paired" for detecting patient specific relative CNVs (paired samples analysis)

TargetChr	character, the target chromosome
SampleName	character, the name of case sample
NCN	numeric, the number of copy number state category. For single sample analysis, the suggested NCN is 5 for absolute number of copy 1 to 5. For paired sample analysis, the suggested NCN is 3 for deletion, normal, and duplication states.

Value

Parameters information for each copy number state are provided as a text file (CONY.3-GroupSumm.*.txt). Four columns represent as the copy number state (GroupCNV), mean RDS (GroupMean), variance RDS (GroupVar), and proportion (GroupPro) of each state.

Examples

```
## For single sample analysis
EstPar(CRDMethod="SumUp", AnaMethod="Single", TargetChr="chr20",
    SampleName="NA12878",NCN=5)

## For paired samples analysis
EstPar(CRDMethod="SumUp", AnaMethod="Paired", TargetChr="chr20",
    SampleName="NA12878",NCN=3)
```

RunCONY *Running the RJMCMC algorithm*

Description

RJMCMC procedure for estimating copy number status of each window is worked with parallel operation

Details

Based on the novel moving procedure of RJMCMC in CONY, copy number status for each window is estimated. We recommend that the whole genome sequencing should separate the genome into several fragments for analysis one at a time. The option fragment length is 500,000 bases.

Usage

```
RunCONY(CRDMethod=c("SumUp","PointR"),
 AnaMethod=c("Single","Paired")
 TargetChr,
 SampleName,
 RunTime = 300000,
 BurnN = 5000,
 RTN = 1000,
 BCPoint = 20,
 FragLength=500000)
```

Argument

CRDMethod	character, the method for window read depth calculation. "SumUp" for sum up the base read depth and "PointR" for count the number of reads which middle or start position located in the specific window
AnaMethod	character, the sample designs. "Single" for detecting absolute number of copies (single sample analysis) and "Paired" for detecting patient specific relative CNVs (paired samples analysis)
TargetChr	character, the target chromosome
SampleName	character, the name of case sample

`RunTime`	`numeric`, maximum number of iterations. The default is 300,000
`BurnN`	`numeric`, the number of burn out iteration in RJMCMC. The default is 5,000
`RTN`	`numeric`, the number of iterations for evaluating the status stable in RJMCMC. The default is 1,000
`BCPoint`	`numeric`, Bayes factor threshold with default 20
`FragLength`	`numeric`, the length of analytic fragments for each lane. The suggested default is 500,000 (bps) at a time. RJMCMC would be run with several lanes simultaneously via snow package. The number of lanes is total number of analytic windows/ (fragment length/ window size).

Value

The estimated copy number statuses and corresponding window information of each lane are generated. The file with 11 columns includes the chromosome name (`seqname`), start position (`start`), end position (`end`), width length (`width`), percentage of indefinable base (`nonAmb`), GC percentage (`GC`), raw window read depth (`ARD`), adjusted window read depth (`AdjRD`), RDSs (`RD`), target information (`target`), and estimated copy number status (`CN`) for each window. The output is saved as a text file (`CONY.4-Result.*.txt`).

Imports

snow package

Reference

Tierney, L., A. J. Rossini and N. Li, 2009 Snow: A parallel computing framework for the R system. International Journal of Parallel Programming 37**:** 78-90.

Examples

```
## For single sample analysis
 RunCONY(CRDMethod="SumUp", AnaMethod="Single",
     TargetChr="chr20", SampleName="NA12878", RunTime =
     300000,BurnN = 5000,RTN = 1000,BCPoint = 20,
     FragLength=500000)


## For paired samples analysis
RunCONY(CRDMethod="SumUp", AnaMethod="Paired", TargetChr="chr20",
  SampleName="NA12878", RunTime = 300000,BurnN = 5000,RTN =
  1000,BCPoint = 20, FragLength=500000)
```

`UsedRD` *Window read depth transformation to read depth signal (RDS)*

Description

Transformed window read depths signals (RDSs) for the downstream RJMCMC simulation are calculated through this function. The potential CNVs with zero copy and the list of windows with too many indefinable bases are also provided.

Details

Two sample designs are available for CONY, including single sample analysis for detecting absolute copy numbers and paired samples analysis for detecting patient specific relative CNVs. Adjusted window read depths (generated from the `AdjRD` function) are transformed to RDSs by logarithm (single sample analysis) or log-ratio (paired samples analysis) equations for the downstream RJMCMC simulation.

Non-informative windows would not be concluded in the following RJMCMC simulation, including windows with zero read depths and more than half of

indefinable bases. Windows with zero adjusted window read depths are set as the potential CNVs with state 0, and with more than half of indefinable bases are excluded because of insufficient information. The lists of non-informative windows are provided.

Usage

```
UsedRD(CRDMethod=c("SumUp","PointR"),
 AnaMethod= c("Single","Paired"),
 TargetChr,
 SampleName,
 ControlName)
```

Argument

`CRDMethod`	`character`, the method for window read depth calculation. `"SumUp"` for sum up the base read depth and `"PointR"` for count the number of reads which middle or start position located in the specific window
`AnaMethod`	`character`, the sample designs. `"Single"` for detecting absolute number of copies (single sample analysis) and `"Paired"` for detecting patient specific relative CNVs (paired samples analysis)
`TargetChr`	`character`, the target chromosome
`SampleName`	`character`, the name of case sample
`ControlName`	`character`, the name of control sample. The command is essential for `AnaMethod ="Paired"` only.

Value

The RDSs and corresponding window information would be generated through the `UsedRD` function. The file with 10 columns includes the chromosome name (`seqname`), start position (`start`), end position (`end`), width length (`width`), percentage of indefinable base (`nonAmb`), GC content (`GC`), raw window read depth (`ARD`), adjusted window read depth (`AdjRD`), RDSs (`RD`), and target information (`target`) for each window. The output is saved as a text file (`CONY.3-TempRegion.*.UsedRD.txt`).

Lists of non-informative windows are saved as text files, including windows with more than half of indefinable bases (`CONY.3-NonInfRegion.*.txt`), zero read depths for case sample (`CONY.3-CN0Region.*.txt`) and for control sample (`CONY.3-CNGRegion.*.txt`)

Examples

```
## For single sample analysis
 UsedRD(CRDMethod="SumUp", AnaMethod= "Single",TargetChr="chr20",
     SampleName= "NA12878")

## For paired samples analysis (NA12878 as case and NA12156
    as control)
UsedRD(CRDMethod="SumUp", AnaMethod= "Paired",TargetChr="chr20",
  SampleName="NA12878",ControlName="NA12156")
```

`WindowInfo` *Windows definition and information summary*

Description

Based on the reference genome, the sliding non-overlap windows are defined and the percentage of G, C and indefinable code are calculated.

Usage

```
WindowInfo(target.df, RefFaFileName ,WindowSize=100)
```

Argument

`target.df`	`data.frame`, the analytic regions information. Three columns are seqname of chromosome, start and end position. `.bed` file is available
`RefFaFileName`	`character`, reference genome as FASTA format (.fa) with one chromosome at a time
`WindowSize`	`numeric`, window size with default 100 (bp)

Value

The function outputs a window information file with 6 columns. It includes the name of chromosome (`seqname`), start position (`start`), end position (`end`), width length (`width`), percentage of indefinable base (`nonAmb`), and GC percentage (`GC`) for each window. The output is saved as a text file (`CONY.TempRegion.txt`) and it would be used for the following steps.

Imports

ExomeCopy and IRanges

References

Lawrence, M., W. Huber, H. Pages, P. Aboyoun, M. Carlson *et al.*, 2013 Software for computing and annotating genomic ranges. PLoS Comput Biol 9**:** e1003118.

Love, M., M. M. Love, D. IRanges, R. GenomicRanges, S. Biostrings *et al.*, 2013 Package 'exomeCopy'.

Example

```
# Chromosome 20 from human genome 19 (hg19) is used to the
  reference genome
 CONY.TempRegion=WindowInfo(target.df=
    as.data.frame(matrix(c(20,1,63025520),1,3,dimnames=list(c(
    "1"),c("seqname","start","end")))),RefFaFileName="chr20.fa
    ",WindowSize=100)
```