

# Package: saasCNV (via r-universe)

September 17, 2024

**Version** 0.3.4

**Date** 2016-05-10

**Title** Somatic Copy Number Alteration Analysis Using Sequencing and SNP Array Data

**Author** Zhongyang Zhang [aut, cre], Ke Hao [aut], Nancy R. Zhang [ctb]

**Maintainer** Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**Depends** R (>= 2.10), RANN, DNACopy

**Description** Perform joint segmentation on two signal dimensions derived from total read depth (intensity) and allele specific read depth (intensity) for whole genome sequencing (WGS), whole exome sequencing (WES) and SNP array data.

**License** GPL (>= 2)

**URL** <https://zhangz05.u.hpc.mssm.edu/saasCNV/>

**NeedsCompilation** no

**Date/Publication** 2016-05-18 02:04:56

**Repository** <https://zhangzhongyang16.r-universe.dev>

**RemoteUrl** <https://github.com/cran/saasCNV>

**RemoteRef** HEAD

**RemoteSha** 2b74d199913425db6929decc04118ebbd55b1803

## Contents

saasCNV-package . . . . .	2
cnv.call . . . . .	3
cnv.data . . . . .	4
diagnosis.cluster.plot . . . . .	5
diagnosis.seg.plot.chr . . . . .	7
GC.adjust . . . . .	8
genome.wide.plot . . . . .	9
internals . . . . .	10
joint.segmentation . . . . .	10

merging.segments . . . . .	12
NGS.CNV . . . . .	13
reannotate.CNV.res . . . . .	16
SNP.CNV . . . . .	17
snp.cnv.data . . . . .	20
snp.refine.boundary . . . . .	21
vcf2txt . . . . .	23

<b>Index</b>	<b>25</b>
--------------	-----------

---

saasCNV-package	<i>Somatic Copy Number Alteration Analysis Using Sequencing and SNP Array Data</i>
-----------------	--

---

## Description

Perform joint segmentation on two signal dimensions derived from total read depth (intensity) and allele specific read depth (intensity) for whole genome sequencing (WGS), whole exome sequencing (WES) and SNP array data.

## Details

Package: saasCNV  
 Type: Package  
 Version: 0.3.4  
 Date: 2016-05-10  
 License: GPL (>= 2)

See the vignettes of the package for more details.

## Author(s)

Zhongyang Zhang [aut, cre], Ke Hao [aut], Nancy R. Zhang [ctb]  
 Maintainer: Zhongyang Zhang <zhongyang.zhang@mssm.edu>

## References

Zhang, Z. and Hao, K. (2015) SAAS-CNV: A joint segmentation approach on aggregated and allele specific signals for the identification of somatic copy number alterations with next-generation sequencing Data. *PLoS Computational Biology*, **11(11)**:e1004618.

Zhang, N. R., Siegmund, D. O., Ji, H., Li, J. Z. (2010) Detecting simultaneous changepoints in multiple sequences. *Biometrika*, **97(3)**:631–645.

## See Also

DNACopy

## Examples

```
## See the vignettes of the package for examples.
```

---

cnv.call	<i>CNV Calling from Sequencing Data</i>
----------	---

---

## Description

Assign SCNA state to each segment directly from joint segmentation or from the results after segments merging step.

## Usage

```
cnv.call(data, sample.id, segs.stat, maxL = NULL, N = 1000,
         pvalue.cutoff = 0.05, seed = NULL,
         do.manual.baseline=FALSE,
         log2mBAF.left=NULL, log2mBAF.right=NULL,
         log2ratio.bottom=NULL, log2ratio.up=NULL)
```

## Arguments

<code>data</code>	a data frame containing log2ratio and log2mBAF data generated by <a href="#">cnv.data</a> .
<code>sample.id</code>	sample ID to be displayed.
<code>segs.stat</code>	a data frame containing segment locations and summary statistics resulting from <a href="#">joint.segmentation</a> or <a href="#">merging.segments</a> .
<code>maxL</code>	integer. The maximum length in terms of number of probes a bootstrapped segment may span. Default is NULL. If NULL, It will be automatically specified as 1/100 of the number of data points.
<code>N</code>	the number of replicates drawn by bootstrap.
<code>pvalue.cutoff</code>	a p-value cut-off for CNV calling.
<code>seed</code>	integer. Random seed can be set for reproducibility of results.
<code>do.manual.baseline</code>	logical. If baseline adjustment to be done manually. Default is FALSE.
<code>log2mBAF.left</code> , <code>log2mBAF.right</code> , <code>log2ratio.bottom</code> , <code>log2ratio.up</code>	left, right, bottom and up boundaries to be specified manually by a visual inspectio of 2-D diagnosis plot generated by <a href="#">diagnosis.cluster.plot</a> . These parameters are active when <code>do.manual.baseline=TRUE</code> .

## Details

The baseline adjustment step is incorporated implicitly in the function.

**Value**

A few more columns have been added to the data frame resulting from [joint.segmentation](#) or [merging.segments](#), which summarize the baseline adjusted median log2ratio, log2mBAF, p-values and CNV state for each segment.

**Author(s)**

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**See Also**

[joint.segmentation](#), [merging.segments](#), [cnv.data](#)

**Examples**

```
data(seq.data)
data(seq.segs.merge)

## Not run:
seq.cnv <- cnv.call(data=seq.data, sample.id="PT116",
                   segs.stat=seq.segs.merge, maxL=2000, N=1000,
                   pvalue.cutoff=0.05)

## End(Not run)

## how the results look like
data(seq.cnv)
head(seq.cnv)
```

---

cnv.data

*Construct Data Frame for CNV Inference with NGS Data*

---

**Description**

Transform read depth information into log2ratio and log2mBAF that we use for joint segmentation and CNV calling.

**Usage**

```
cnv.data(vcf, min.chr.probe = 100, verbose = FALSE)
```

**Arguments**

vcf	a data frame constructed from a vcf file. See <a href="#">vcf2txt</a> .
min.chr.probe	the minimum number of probes tagging a chromosome for it to be passed to the subsequent analysis.
verbose	logical. If more details to be output. Default is FALSE.

**Value**

A data frame containing the log2raio and log2mBAF values for each probe site.

**Author(s)**

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**References**

Staaf, J., Vallon-Christersson, J., Lindgren, D., Juliusson, G., Rosenquist, R., Hoglund, M., Borg, A., Ringner, M. (2008) Normalization of Illumina Infinium whole-genome SNP data improves copy number estimates and allelic intensity ratios. *BMC bioinformatics*, **9**:409.

**See Also**

[vcf2txt](#)

**Examples**

```
## load a data frame constructed from a vcf file with vcf2txt

## Not run:
## download vcf_table.txt.gz
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/vcf_table.txt.gz"
tryCatch({download.file(url=url, destfile="vcf_table.txt.gz")
          }, error = function(e) {
          download.file(url=url, destfile="vcf_table.txt.gz", method="curl")
          })
## If download.file fails to download the data, please manually download it from the url.

vcf_table <- read.delim(file="vcf_table.txt.gz", as.is=TRUE)
seq.data <- cnv.data(vcf=vcf_table, min.chr.probe=100, verbose=TRUE)

## End(Not run)

## see how seq.data looks like
data(seq.data)
head(seq.data)
```

---

diagnosis.cluster.plot

*Visualize Genome-Wide SCNA Profile in 2D Cluster Plot*

---

**Description**

An optional function to visualize genome-wide SCNA Profile in log2mBAF-log2ratio 2D cluster plot.

**Usage**

```
diagnosis.cluster.plot(segs, chrs, min.snps, max.cex = 3, ref.num.probe = NULL)
```

**Arguments**

segs	a data frame containing segment location, summary statistics and SCNA status resulting from <a href="#">cnv.call</a> .
chrs	the chromosomes to be visualized. For example, 1:22.
min.snps	the minimum number of probes a segment span.
max.cex	the maximum of cex a circle is associated with. See details.
ref.num.probe	integer. The reference number of probes against which a segment is compared in order to determine the cex of the segment to be displayed. Default is NULL. If NULL, It will be automatically specified as 1/100 of the number of data points.

**Details**

on the main log2mBAF-log2ratio panel, each circle corresponds to a segment, with the size reflecting the length of the segment; the color code is specified in legend; the dashed gray lines indicate the adjusted baselines. The side panels, corresponding to log2ratio and log2mBAF dimension respectively, show the distribution of median values of each segment weighted by its length.

**Value**

An R plot will be generated.

**Author(s)**

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**See Also**

[joint.segmentation](#), [cnv.call](#), [diagnosis.seg.plot.chr](#), [genome.wide.plot](#)

**Examples**

```
data(seq.data)
data(seq.cnv)

diagnosis.cluster.plot(segs=seq.cnv,
                      chrs=sub("^chr", "", unique(seq.cnv$chr)),
                      min.snps=10, max.cex=3, ref.num.probe=1000)
```

---

`diagnosis.seg.plot.chr`*Visualize Segmentation Results for Diagnosis*

---

## Description

The results from joint segmentation and segments merging are visualized for the specified chromosome.

## Usage

```
diagnosis.seg.plot.chr(data, segs, sample.id = "Sample", chr = 1, cex = 0.3)
```

## Arguments

<code>data</code>	a data frame containing log2ratio and log2mBAF data generated by <a href="#">cnv.data</a> .
<code>segs</code>	a data frame containing segment locations and summary statistics resulting from <a href="#">joint.segmentation</a> or <a href="#">merging.segments</a> .
<code>sample.id</code>	sample ID to be displayed in the title of the plot.
<code>chr</code>	the chromosome number (e.g. 1) to be visualized.
<code>cex</code>	a numerical value giving the amount by which plotting text and symbols should be magnified relative to the default. It can be adjusted in order to make the plot legible.

## Value

An R plot will be generated.

## Author(s)

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

## See Also

[joint.segmentation](#), [merging.segments](#), [cnv.data](#)

## Examples

```
## visual diagnosis of joint segmentation results
data(seq.data)
data(seq.segs)
diagnosis.seg.plot.chr(data=seq.data, segs=seq.segs,
                      sample.id="Joint Segmentation",
                      chr=1, cex=0.3)

## visual diagnosis of results from merging step
data(seq.segs.merge)
```

```
diagnosis.seg.plot.chr(data=seq.data, segs=seq.segs.merge,
                      sample.id="After Segments Merging Step",
                      chr=1, cex=0.3)
```

---

GC.adjust

*GC Content Adjustment*


---

### Description

This function adjusts log2ratio by GC content using LOESS.

### Usage

```
GC.adjust(data, gc, maxNumDataPoints = 10000)
```

### Arguments

data	A data frame generated by <a href="#">cnv.data</a> or <a href="#">snp.cnv.data</a> .
gc	A data frame containing three columns: chr, position and GC. See the example data below for details.
maxNumDataPoints	The maximum number of data points used for loess fit. Default is 10000.

### Details

The method for GC content adjustment was adopted from CNAnorm (Gusnato et al. 2012).

### Value

A data frame containing the log2ratio (GC adjusted) and log2mBAF values for each probe site in the same format as generated by [cnv.data](#) or [snp.cnv.data](#). The original log2ratio is renamed as log2ratio.woGCAdj. The GC-adjusted log2ratio is named as log2ratio.

### Note

This function is optional in the analysis pipeline and is now in beta version.

### Author(s)

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

### References

Gusnanto, A, Wood HM, Pawitan Y, Rabbitts P, Berri S (2012) Correcting for cancer genome size and tumour cell content enables better estimation of copy number alterations from next-generation sequence data. *Bioinformatics*, **28**:40-47.



**See Also**

[cnv.data](#), [snp.cnv.data](#)

**Examples**

```
## CNV data generated by cnv.data
data(seq.data)
head(seq.data)

## Not run:
## an example GC content file
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/GC_1kb_hg19.txt.gz"
tryCatch({download.file(url=url, destfile="GC_1kb_hg19.txt.gz")
}, error = function(e) {
  download.file(url=url, destfile="GC_1kb_hg19.txt.gz", method="curl")
})
## If download.file fails to download the data, please manually download it from the url.

gc <- read.delim(file = "GC_1kb_hg19.txt.gz", as.is=TRUE)
head(gc)

## GC content adjustment
seq.data <- GC.adjust(data = seq.data, gc = gc, maxNumDataPoints = 10000)
head(seq.data)

## End(Not run)
```

---

genome.wide.plot

*Visualize Genome-Wide SCNA Profile*

---

**Description**

An optional function to visualize genome-wide SCNA Profile.

**Usage**

```
genome.wide.plot(data, segs, sample.id, chrs, cex = 0.3)
```

**Arguments**

data	a data frame containing log2ratio and log2mBAF data generated by <a href="#">cnv.data</a> .
segs	a data frame containing segment location, summary statistics and SCNA status resulting from <a href="#">cnv.call</a> .
sample.id	sample ID to be displayed in the title of the plot.
chrs	the chromosomes to be visualized. For example, 1:22.
cex	a numerical value giving the amount by which plotting text and symbols should be magnified relative to the default. It can be adjusted in order to make the plot legible.

**Details**

On the top panel, the log2ratio signal is plotted against chromosomal position and on the panels below, the log2mBAF, tumor mBAF, and normal mBAF signals. The dots, each representing a probe data point, are colored alternately to distinguish chromosomes. The segments, each representing a DNA segment resulting from the joint segmentation, are colored based on inferred copy number status.

**Value**

An R plot will be generated.

**Author(s)**

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**See Also**

[joint.segmentation](#), [cnv.call](#), [diagnosis.seg.plot.chr](#), [diagnosis.cluster.plot](#)

**Examples**

```
data(seq.data)
data(seq.cnv)

genome.wide.plot(data=seq.data, segs=seq.cnv,
                 sample.id="PT116",
                 chrs=sub("^chr", "", unique(seq.cnv$chr)),
                 cex=0.3)
```

---

internals

*Internal Functions and Data*

---

**Description**

These are the functions and data to which the users do not need to directly get access.

---

joint.segmentation

*Joint Segmentation on log2ratio and log2mBAF Dimensions*

---

**Description**

We employ the algorithm developed by (Zhang et al., 2010) to perform joint segmentation on log2ratio and log2mBAF dimensions. The function outputs the starting and ending points of each CNV segment as well as some summary statistics.

**Usage**

```
joint.segmentation(data, min.snps = 10, global.pval.cutoff = 1e-04,  
  max.chpts = 30, verbose = TRUE)
```

**Arguments**

data	a data frame containing log2ratio and log2mBAF data generated by <a href="#">cnv.data</a> .
min.snps	the minimum number of probes a segment needs to span.
global.pval.cutoff	the p-value cut-off a (or a pair) of change points to be determined as significant in each cycle of joint segmentation.
max.chpts	the maximum number of change points to be detected for each chromosome.
verbose	logical. If more details to be output. Default is TRUE.

**Value**

A data frame containing the starting and ending points of each CNV segment as well as some summary statistics.

**Author(s)**

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**References**

Zhang, N. R., Siegmund, D. O., Ji, H., Li, J. Z. (2010) Detecting simultaneous changepoints in multiple sequences. *Biometrika*, **97**:631–645.

**See Also**

[cnv.data](#)

**Examples**

```
data(seq.data)  
  
## Not run:  
seq.segs <- joint.segmentation(data=seq.data, min.snps=10,  
  global.pval.cutoff=1e-4, max.chpts=30,  
  verbose=TRUE)  
  
## End(Not run)  
  
## how the joint segmentation results look like  
data(seq.segs)  
head(seq.segs)
```

---

merging.segments	<i>Merge Adjacent Segments</i>
------------------	--------------------------------

---

### Description

It is an option to merge adjacent segments, for which the median values in either or both log2ratio and log2mBAF dimensions are not substantially different. For WGS and SNP array, it is recommended to do so.

### Usage

```
merging.segments(data, segs.stat, use.null.data = TRUE,
                 N = 1000, maxL = NULL, merge.pvalue.cutoff = 0.05,
                 do.manual.baseline=FALSE,
                 log2mBAF.left=NULL, log2mBAF.right=NULL,
                 log2ratio.bottom=NULL, log2ratio.up=NULL,
                 seed = NULL,
                 verbose = TRUE)
```

### Arguments

data	a data frame containing log2ratio and log2mBAF data generated by <a href="#">cnv.data</a> .
segs.stat	a data frame containing segment locations and summary statistics resulting from <a href="#">joint.segmentation</a> .
use.null.data	logical. If only data for probes located in normal copy segments to be used for bootstrapping. Default is TRUE. If a more aggressive merging is needed, it can be switched to FALSE.
N	the number of replicates drawn by bootstrap.
maxL	integer. The maximum length in terms of number of probes a bootstrapped segment may span. Default is NULL. If NULL, It will be automatically specified as 1/100 of the number of data points.
merge.pvalue.cutoff	a p-value cut-off for merging. If the empirical p-value is greater than the cut-off value, the two adjacent segments under consideration will be merged.
do.manual.baseline	logical. If baseline adjustment to be done manually. Default is FALSE.
log2mBAF.left, log2mBAF.right, log2ratio.bottom, log2ratio.up	left, right, bottom and up boundaries to be specified manually by a visual inspection of 2-D diagnosis plot generated by <a href="#">diagnosis.cluster.plot</a> . These parameters are active when do.manual.baseline=TRUE.
seed	integer. Random seed can be set for reproducibility of results.
verbose	logical. If more details to be output. Default is TRUE.

### Value

A data frame with the same columns as the one generated by [joint.segmentation](#).

**Author(s)**

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**See Also**

[cnv.data](#), [joint.segmentation](#)

**Examples**

```
data(seq.data)
data(seq.segs)

## Not run:
seq.segs.merge <- merging.segments(data=seq.data, segs.stat=seq.segs,
                                   use.null.data=TRUE,
                                   N=1000, maxL=2000,
                                   merge.pvalue.cutoff=0.05, verbose=TRUE)

## End(Not run)

## how the results look like
data(seq.segs.merge)
head(seq.segs.merge)
```

---

NGS.CNV

*CNV Analysis Pipeline for WGS and WES Data*

---

**Description**

All analysis steps are integrate into a pipeline. The results, including visualization plots are placed in a directory as specified by user.

**Usage**

```
NGS.CNV(vcf, output.dir, sample.id,
        do.GC.adjust = FALSE,
        gc.file = system.file("extdata", "GC_1kb_hg19.txt.gz", package="saasCNV"),
        min.chr.probe = 100, min.snps = 10,
        joint.segmentation.pvalue.cutoff = 1e-04, max.chpts = 30,
        do.merge = TRUE, use.null.data = TRUE,
        num.perm = 1000, maxL = NULL,
        merge.pvalue.cutoff = 0.05,
        do.cnvcall.on.merge = TRUE,
        cnvcall.pvalue.cutoff = 0.05,
        do.plot = TRUE, cex = 0.3, ref.num.probe = NULL,
        do.gene.anno = FALSE,
        gene.anno.file = NULL,
        seed = NULL,
        verbose = TRUE)
```

**Arguments**

<code>vcf</code>	a data frame constructed from a vcf file. See <a href="#">vcf2txt</a> .
<code>output.dir</code>	the directory to which all the results will be located.
<code>sample.id</code>	sample ID to be displayed in the data frame of the results and the title of some diagnosis plots.
<code>do.GC.adjust</code>	logical. If GC content adjustment on <code>log2ratio</code> to be carried out. Default is FALSE. See <a href="#">GC.adjust</a> for details.
<code>gc.file</code>	the location of tab-delimit file with GC content (averaged per 1kb window) information. See <a href="#">GC.adjust</a> for details.
<code>min.chr.probe</code>	the minimum number of probes tagging a chromosome for it to be passed to the subsequent analysis.
<code>min.snps</code>	the minimum number of probes a segment needs to span.
<code>joint.segmentation.pvalue.cutoff</code>	the p-value cut-off one (or a pair) of change points to be determined as significant in each cycle of joint segmentation.
<code>max.chpts</code>	the maximum number of change points to be detected for each chromosome.
<code>do.merge</code>	logical. If segments merging step to be carried out. Default is TRUE.
<code>use.null.data</code>	logical. If only data for probes located in normal copy segments to be used for bootstrapping. Default is TRUE. If a more aggressive merging is needed, it can be switched to FALSE.
<code>num.perm</code>	the number of replicates drawn by bootstrap.
<code>maxL</code>	integer. The maximum length in terms of number of probes a bootstrapped segment may span. Default is NULL. If NULL, It will be automatically specified as 1/100 of the number of data points.
<code>merge.pvalue.cutoff</code>	a p-value cut-off for merging. If the empirical p-value is greater than the cut-off value, the two adjacent segments under consideration will be merged.
<code>do.cnvcall.on.merge</code>	logical. If CNV call to be done for the segments after merging step. Default is TRUE. If TRUE, CNV call will be done on the segments resulting directly from joint segmentation without merging step.
<code>cnvcall.pvalue.cutoff</code>	a p-value cut-off for CNV calling.
<code>do.plot</code>	logical. If diagnosis plots to be output. Default is TRUE.
<code>cex</code>	a numerical value giving the amount by which plotting text and symbols should be magnified relative to the default. It can be adjusted in order to make the plot legible.
<code>ref.num.probe</code>	integer. The reference number of probes against which a segment is compared in order to determine the <code>cex</code> of the segment to be displayed. Default is NULL. If NULL, It will be automatically specified as 1/100 of the number of data points.
<code>do.gene.anno</code>	logical. If gene annotation step to be performed. Default is FALSE.
<code>gene.anno.file</code>	a tab-delimited file containing gene annotation information. For example, Ref-Seq annotation file which can be found at UCSC genome browser.

seed integer. Random seed can be set for reproducibility of results.  
 verbose logical. If more details to be output. Default is TRUE.

### Details

See the vignettes of the package for more details.

### Value

The results, including visualization plots are placed in subdirectories of the output directory `output.dir` as specified by user.

### Author(s)

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

### References

Zhongyang Zhang and Ke Hao. (2015) SAAS-CNV: A Joint Segmentation Approach on Aggregated and Allele Specific Signals for the Identification of Somatic Copy Number Alterations with Next-Generation Sequencing Data. *PLoS Computational Biology*, 11(11):e1004618.

### See Also

[vcf2txt](#), [cnv.data](#), [joint.segmentation](#), [merging.segments](#), [cnv.call](#), [diagnosis.seg.plot.chr](#), [genome.wide.plot](#), [diagnosis.cluster.plot](#)

### Examples

```
## Not run:
## NGS pipeline analysis
## download vcf_table.txt.gz
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/vcf_table.txt.gz"
tryCatch({download.file(url=url, destfile="vcf_table.txt.gz")
  }, error = function(e) {
  download.file(url=url, destfile="vcf_table.txt.gz", method="curl")
})
## If download.file fails to download the data, please manually download it from the url.

vcf_table <- read.delim(file="vcf_table.txt.gz", as.is=TRUE)

## download refGene_hg19.txt.gz
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/refGene_hg19.txt.gz"
tryCatch({download.file(url=url, destfile="refGene_hg19.txt.gz")
  }, error = function(e) {
  download.file(url=url, destfile="refGene_hg19.txt.gz", method="curl")
})
## If download.file fails to download the data, please manually download it from the url.

sample.id <- "WES_0116"
output.dir <- file.path(getwd(), "test_saasCNV")
```

```

NGS.CNV(vcf=vcf_table, output.dir=output.dir, sample.id=sample.id,
        min.chr.probe=100,
        min.snps=10,
        joint.segmentation.pvalue.cutoff=1e-4,
        max.chpts=30,
        do.merge=TRUE, use.null.data=TRUE, num.perm=1000, maxL=2000,
        merge.pvalue.cutoff=0.05,
        do.cnvcall.on.merge=TRUE,
        cnvcall.pvalue.cutoff=0.05,
        do.plot=TRUE, cex=0.3, ref.num.probe=1000,
        do.gene.anno=TRUE,
        gene.anno.file="refGene_hg19.txt.gz",
        seed=123456789,
        verbose=TRUE)

## End(Not run)

```

---

reannotate.CNV.res      *Gene Annotation*

---

## Description

An optional function to add gene annotation to each CNV segment.

## Usage

```
reannotate.CNV.res(res, gene, only.CNV = FALSE)
```

## Arguments

res	a data frame resulting from <a href="#">cnv.call</a> .
gene	a data frame containing gene annotation information.
only.CNV	logical. If only segment assigned to gain/loss/LOH to be annotated and output. Default is FALSE.

## Details

The RefSeq gene annotation file can be downloaded from UCSC Genome Browser.

## Value

A gene annotation column have been add to the data frame resulting from [cnv.call](#).

## Author(s)

Zhongyang Zhang <zhongyang.zhang@mssm.edu>



**See Also**

[joint.segmentation](#), [cnv.call](#)

**Examples**

```
## Not run:
## An example of RefSeq gene annotation file,
## the original version of which can be downloaded from UCSC Genome Browser
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/refGene_hg19.txt.gz"
tryCatch({download.file(url=url, destfile="refGene_hg19.txt.gz")
          }, error = function(e) {
          download.file(url=url, destfile="refGene_hg19.txt.gz", method="curl")
          })
## If download.file fails to download the data, please manually download it from the url.

gene.anno <- read.delim(file="refGene_hg19.txt.gz", as.is=TRUE, comment.char="")
data(seq.cnv)
seq.cnv.anno <- reannotate.CNV.res(res=seq.cnv, gene=gene.anno, only.CNV=TRUE)

## End(Not run)
```

---

 SNP.CNV

*CNV Analysis Pipeline for SNP array Data*


---

**Description**

All analysis steps are integrate into a pipeline. The results, including visualization plots are placed in a directory as specified by user.

**Usage**

```
SNP.CNV(snp, output.dir, sample.id,
        do.GC.adjust = FALSE,
        gc.file = system.file("extdata", "GC_1kb_hg19.txt.gz", package="saasCNV"),
        min.chr.probe = 100, min.snps = 10,
        joint.segmentation.pvalue.cutoff = 1e-04, max.chpts = 30,
        do.merge = TRUE, use.null.data = TRUE,
        num.perm = 1000, maxL = NULL,
        merge.pvalue.cutoff = 0.05,
        do.cnvcall.on.merge = TRUE,
        cnvcall.pvalue.cutoff = 0.05,
        do.boundary.refine = FALSE,
        do.plot = TRUE, cex = 0.3,
        ref.num.probe = NULL,
        do.gene.anno = FALSE,
        gene.anno.file = NULL,
        seed = NULL, verbose = TRUE)
```

**Arguments**

<code>snp</code>	a data frame constructed from a text file with LRR and BAF information.
<code>output.dir</code>	the directory to which all the results will be located.
<code>sample.id</code>	sample ID to be displayed in the data frame of the results and the title of some diagnosis plots.
<code>do.GC.adjust</code>	logical. If GC content adjustment on <code>log2ratio</code> to be carried out. Default is FALSE. See <a href="#">GC.adjust</a> for details.
<code>gc.file</code>	the location of tab-delimit file with GC content (averaged per 1kb window) information. See <a href="#">GC.adjust</a> for details.
<code>min.chr.probe</code>	the minimum number of probes tagging a chromosome for it to be passed to the subsequent analysis.
<code>min.snps</code>	the minimum number of probes a segment needs to span.
<code>joint.segmentation.pvalue.cutoff</code>	the p-value cut-off one (or a pair) of change points to be determined as significant in each cycle of joint segmentation.
<code>max.chpts</code>	the maximum number of change points to be detected for each chromosome.
<code>do.merge</code>	logical. If segments merging step to be carried out. Default is TRUE.
<code>use.null.data</code>	logical. If only data for probes located in normal copy segments to be used for bootstrapping. Default is TRUE. If a more aggressive merging is needed, it can be switched to FALSE.
<code>num.perm</code>	the number of replicates drawn by bootstrap.
<code>maxL</code>	integer. The maximum length in terms of number of probes a bootstrapped segment may span. Default is NULL. If NULL, It will be automatically specified as 1/100 of the number of data points.
<code>merge.pvalue.cutoff</code>	a p-value cut-off for merging. If the empirical p-value is greater than the cut-off value, the two adjacent segments under consideration will be merged.
<code>do.cnvcall.on.merge</code>	logical. If CNV call to be done for the segments after merging step. Default is TRUE. If TRUE, CNV call will be done on the segments resulting directly from joint segmentation without merging step.
<code>cnvcall.pvalue.cutoff</code>	a p-value cut-off for CNV calling.
<code>do.boundary.refine</code>	logical. If the segment boundaries based on the grid of heterozygous probes to be refined by all probes with LRR data. Default is FALSE. We do not recommend to perform this step except in the case that the segment boundaries need to be aligned well on the same grid of probes for downstream analysis.
<code>do.plot</code>	logical. If diagnosis plots to be output. Default is TRUE.
<code>cex</code>	a numerical value giving the amount by which plotting text and symbols should be magnified relative to the default. It can be adjusted in order to make the plot legible.

<code>ref.num.probe</code>	integer. The reference number of probes against which a segment is compared in order to determine the cex of the segment to be displayed. Default is NULL. If NULL, It will be automatically specified as 1/100 of the number of data points.
<code>do.gene.anno</code>	logical. If gene annotation step to be performed. Default is FALSE.
<code>gene.anno.file</code>	a tab-delimited file containing gene annotation information. For example, Ref-Seq annotation file which can be found at UCSC genome browser.
<code>seed</code>	integer. Random seed can be set for reproducibility of results.
<code>verbose</code>	logical. If more details to be output. Default is TRUE.

### Details

See the vignettes of the package for more details.

### Value

The results, including visualization plots are placed in subdirectories of the output directory `output.dir` as specified by user.

### Author(s)

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

### References

Zhongyang Zhang and Ke Hao. (2015) SAAS-CNV: A Joint Segmentation Approach on Aggregated and Allele Specific Signals for the Identification of Somatic Copy Number Alterations with Next-Generation Sequencing Data. *PLoS Computational Biology*, 11(11):e1004618.

### See Also

[NGS.CNV](#), [snp.cnv.data](#), [joint.segmentation](#), [merging.segments.cnv.call](#), [diagnosis.seg.plot.chr](#), [genome.wide.plot](#), [diagnosis.cluster.plot](#), [snp.refine.boundary](#)

### Examples

```
## Not run:
## the pipeline for SNP array analysis
## download snp_table.txt.gz
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/snp_table.txt.gz"
tryCatch({download.file(url=url, destfile="snp_table.txt.gz")
}, error = function(e) {
  download.file(url=url, destfile="snp_table.txt.gz", method="curl")
})
## If download.file fails to download the data, please manually download it from the url.

snp_table <- read.delim(file="snp_table.txt.gz", as.is=TRUE)

## download refGene_hg19.txt.gz
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/refGene_hg19.txt.gz"
tryCatch({download.file(url=url, destfile="refGene_hg19.txt.gz")
```

```

    }, error = function(e) {
      download.file(url=url, destfile="refGene_hg19.txt.gz", method="curl")
    })
## If download.file fails to download the data, please manually download it from the url.

sample.id <- "SNP_0116"
output.dir <- file.path(getwd(), "test_saasCNV")

SNP.CNV(snp=snp_table, output.dir=output.dir, sample.id=sample.id,
        min.chr.probe=100,
        min.snps=10,
        joint.segmentation.pvalue.cutoff=1e-4,
        max.chpts=30,
        do.merge=TRUE, use.null.data=TRUE, num.perm=1000, maxL=5000,
        merge.pvalue.cutoff=0.05,
        do.cnvcall.on.merge=TRUE,
        cnvcall.pvalue.cutoff=0.05,
        do.boundary.refine=TRUE,
        do.plot=TRUE, cex=0.3, ref.num.probe=5000,
        do.gene.anno=TRUE,
        gene.anno.file="refGene_hg19.txt.gz",
        seed=123456789,
        verbose=TRUE)

## End(Not run)

```

---

snp.cnv.data

---

*Construct Data Frame for CNV Inference with SNP Array Data*


---

### Description

Transform LRR and BAF information into log2ratio and log2mBAF that we use for joint segmentation and CNV calling.

### Usage

```
snp.cnv.data(snp, min.chr.probe = 100, verbose = FALSE)
```

### Arguments

snp	a data frame with LRR and BAF information from SNP array. See the example below for details.
min.chr.probe	the minimum number of probes tagging a chromosome for it to be passed to the subsequent analysis.
verbose	logical. If more details to be output. Default is FALSE.

### Value

A data frame containing the log2raio and log2mBAF values for each probe site.

**Author(s)**

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**References**

Staaf, J., Vallon-Christersson, J., Lindgren, D., Juliusson, G., Rosenquist, R., Hoglund, M., Borg, A., Ringner, M. (2008) Normalization of Illumina Infinium whole-genome SNP data improves copy number estimates and allelic intensity ratios. *BMC bioinformatics*, **9**:409.

**See Also**

[cnv.data](#)

**Examples**

```
## Not run:
## an example data with LRR and BAF information
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/snp_table.txt.gz"
tryCatch({download.file(url=url, destfile="snp_table.txt.gz")
          }, error = function(e) {
          download.file(url=url, destfile="snp_table.txt.gz", method="curl")
          })
## If download.file fails to download the data, please manually download it from the url.

snp_table <- read.delim(file="snp_table.txt.gz", as.is=TRUE)
snp.data <- snp.cnv.data(snp=snp_table, min.chr.probe=100, verbose=TRUE)

## see how seq.data looks like
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/snp.data.RData"
tryCatch({download.file(url=url, destfile="snp.data.RData")
          }, error = function(e) {
          download.file(url=url, destfile="snp.data.RData", method="curl")
          })
## If download.file fails to download the data, please manually download it from the url.

load("snp.data.RData")
head(snp.data)

## End(Not run)
```

---

snp.refine.boundary     *Refine Segment Boundaries*

---

**Description**

Refine the segment boundaries based on the grid of heterozygous probes by all probes with LRR data. We do not recommend to perform this step except in the case that the segment boundaries need to be aligned well on the same grid of probes for downstream analysis.

**Usage**

```
snp.refine.boundary(data, segs.stat)
```

**Arguments**

`data` a data frame containing log2ratio and log2mBAF data generated by [snp.cnv.data](#).  
`segs.stat` a data frame containing segment locations and summary statistics resulting from [cnv.call](#).

**Value**

A data frame with the same columns as the one generated by [cnv.call](#) with the columns `posStart`, `posEnd`, `length`, `chrIdxStart`, `chrIdxEnd` and `numProbe` updated accordingly.

**Author(s)**

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**See Also**

[snp.cnv.data](#), [cnv.call](#)

**Examples**

```
## Not run:
## download snp.data.RData
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/snp.data.RData"
tryCatch({download.file(url=url, destfile="snp.data.RData")
}, error = function(e) {
  download.file(url=url, destfile="snp.data.RData", method="curl")
})
## If download.file fails to download the data, please manually download it from the url.

load("snp.data.RData")
data(snp.cnv)
snp.cnv.refine <- snp.refine.boundary(data=snp.data, segs.stat=snp.cnv)

## End(Not run)

## how the results look like
data(snp.cnv.refine)
head(snp.cnv.refine)
```

---

`vcf2txt`*Covert VCF File to A Data Frame*

---

**Description**

It parses a VCF file and extract necessary information for CNV analysis.

**Usage**

```
vcf2txt(vcf.file, normal.col = 10, tumor.col = 11, MQ.cutoff = 30)
```

**Arguments**

<code>vcf.file</code>	vcf file name.
<code>normal.col</code>	the number of the column in which the genotype and read depth information of normal tissue are located in the vcf file.
<code>tumor.col</code>	the number of the column in which the genotype and read depth information of tumor tissue are located in the vcf file.
<code>MQ.cutoff</code>	the minimum criterion of mapping quality.

**Details**

Note that the first 9 columns in vcf file are mandatory, followed by the information for called variant starting from the 10th column.

**Value**

A data frame of detailed information about each variant, including chrosome position, reference and alternative alleles, genotype and read depth carrying reference and alternative alleles for normal and tumor respectively.

**Author(s)**

Zhongyang Zhang <zhongyang.zhang@mssm.edu>

**References**

Danecek, P., Auton, A., Abecasis, G., Albers, C. A., Banks, E., DePristo, M. A., Handsaker, R. E., Lunter, G., Marth, G. T., Sherry, S. T., et al. (2011) The variant call format and VCFtools. *Bioinformatics*, **27**:2156–2158.

<http://www.1000genomes.org/node/101>

## Examples

```
## Not run:
## an example VCF file from WES
## download WES_example.vcf.gz
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/WES_example.vcf.gz"
tryCatch({download.file(url=url, destfile="WES_example.vcf.gz")
}, error = function(e) {
  download.file(url=url, destfile="WES_example.vcf.gz", method="curl")
})
## If download.file fails to download the data, please manually download it from the url.

## convert Vcf file to a data frame
vcf_table <- vcf2txt(vcf.file="WES_example.vcf.gz", normal.col=9+1, tumor.col=9+2)

## see how vcf_table looks like
## download vcf_table.txt.gz
url <- "https://zhangz05.u.hpc.mssm.edu/saasCNV/data/vcf_table.txt.gz"
tryCatch({download.file(url=url, destfile="vcf_table.txt.gz")
}, error = function(e) {
  download.file(url=url, destfile="vcf_table.txt.gz", method="curl")
})
## If download.file fails to download the data, please manually download it from the url.

vcf_table <- read.delim(file="vcf_table.txt.gz", as.is=TRUE)
head(vcf_table)

## End(Not run)
```



# Index

- \* **CBS**
    - joint.segmentation, 10
  - \* **CNV call**
    - cnv.call, 3
  - \* **CNV**
    - cnv.call, 3
    - cnv.data, 4
    - GC.adjust, 8
    - NGS.CNV, 13
    - reannotate.CNV.res, 16
    - SNP.CNV, 17
    - snp.cnv.data, 20
    - snp.refine.boundary, 21
  - \* **GC content**
    - GC.adjust, 8
  - \* **NGS**
    - NGS.CNV, 13
  - \* **SCNA**
    - diagnosis.cluster.plot, 5
    - genome.wide.plot, 9
  - \* **SNP array**
    - SNP.CNV, 17
    - snp.refine.boundary, 21
  - \* **VCF**
    - vcf2txt, 23
  - \* **annotation**
    - reannotate.CNV.res, 16
  - \* **cluster**
    - diagnosis.cluster.plot, 5
  - \* **diagnosis**
    - diagnosis.cluster.plot, 5
    - diagnosis.seg.plot.chr, 7
    - genome.wide.plot, 9
  - \* **gene**
    - reannotate.CNV.res, 16
  - \* **joint segmentation**
    - joint.segmentation, 10
  - \* **merge**
    - merging.segments, 12
  - \* **pipeline**
    - NGS.CNV, 13
    - SNP.CNV, 17
  - \* **segmentation**
    - diagnosis.seg.plot.chr, 7
    - joint.segmentation, 10
    - merging.segments, 12
  - \* **vcf**
    - vcf2txt, 23
  - \* **visualization**
    - diagnosis.cluster.plot, 5
    - genome.wide.plot, 9
- BAF2mBAF (internals), 10
- check.overlap (internals), 10
- cnv.call, 3, 6, 9, 10, 15–17, 19, 22
- cnv.data, 3, 4, 4, 7–9, 11–13, 15, 21
- cnv.data.chr (internals), 10
- compute.baseline (internals), 10
- compute.var (internals), 10
- computeBeta (internals), 10
- computeMoments (internals), 10
- computeTiltDirect (internals), 10
- ComputeZ.fromS.R (internals), 10
- computeZ.onechange (internals), 10
- computeZ.squarewave.sample (internals), 10
- dchi (internals), 10
- delta.sd (internals), 10
- diagnosis.cluster.plot, 3, 5, 10, 12, 15, 19
- diagnosis.QQ.plot (internals), 10
- diagnosis.seg.plot.chr, 6, 7, 10, 15, 19
- fcompute.max.Z (internals), 10
- fmscbs (internals), 10
- fscan.max (internals), 10
- GC.adjust, 8, 14, 18
- genome.wide.plot, 6, 9, 15, 19

getCutoffMultisampleWeightedChisq  
    (internals), 10

impute.missing.data (internals), 10  
internals, 10

joint.segmentation, 3, 4, 6, 7, 10, 10, 12,  
    13, 15, 17, 19

matrix.max (internals), 10  
merging.segments, 3, 4, 7, 12, 15, 19  
merging.segments.chr (internals), 10  
Mode (internals), 10  
mscbs.classify (internals), 10

NGS.CNV, 13, 19

pmarg.sumweightedchisq (internals), 10  
pvalueMultisampleWeightedChisq  
    (internals), 10

reannotate.CNV.res, 16

saasCNV (saasCNV-package), 2  
saasCNV-package, 2  
seg.summary (internals), 10  
seq.cnv (internals), 10  
seq.data (internals), 10  
seq.segs (internals), 10  
SNP.CNV, 17  
snp.cnv (internals), 10  
snp.cnv.data, 8, 9, 19, 20, 22  
snp.data (internals), 10  
snp.refine.boundary, 19, 21  
snp.segs (internals), 10  
snp\_table (internals), 10

vcf2txt, 4, 5, 14, 15, 23  
vcf\_table (internals), 10  
vu (internals), 10