Package ‘CNAnorm’

March 28, 2017

Version 1.20.0
Date 2012-08-06
Title A normalization method for Copy Number Aberration in cancer samples
Author Stefano Berri <sberri@illumina.com>, Henry M. Wood <H.M.Wood@leeds.ac.uk>, Arief Gusnanto <a.gusnanto@leeds.ac.uk>
Maintainer Stefano Berri <sberri@illumina.com>
Depends R (>= 2.10.1), methods
Description Performs ratio, GC content correction and normalization of data obtained using low coverage (one read every 100-10,000 bp) high throughput sequencing. It performs a “discrete” normalization looking for the ploidy of the genome. It will also provide tumour content if at least two ploidy states can be found.
License GPL-2
Imports DNAcopy
URL http://www.r-project.org.
LazyLoad yes
biocViews CopyNumberVariation, Sequencing, Coverage, Normalization, WholeGenome, DNASEq, GenomicVariation
NeedsCompilation yes

R topics documented:

  addDNACopy ................................................................. 2
  addSmooth ................................................................. 3
  chromosomesPosition ..................................................... 4
  CN ......................................................................... 5
  CNAnorm-class ............................................................ 6
  CNAnormWorkflow ........................................................ 7
addDNACopy

Methods for Function addDNACopy in Package ‘CNAnorm’

Description

addSmooth segment ratio values in Package 'CNAnorm' using DNACopy

Usage

## S4 method for signature 'CNAnorm'
addDNACopy(object, independent.arms = FALSE, ideograms = NULL,
DNAcopy.smooth = list(), DNAcopy.weight = character(), DNAcopy.segment = list())

Arguments

object
An object of Class "CNAnorm"

independent.arms
Boolean. If TRUE chromosomes arms will be treated as independent, ideograms
must be provided

ideograms
A data frame containing information about banding. See ?hg19_hs_ideogr for
more information

DNAcopy.smooth
A list of parameters to be passed to function 'smooth.CNA' in package DNA-
copy

DNAcopy.weight
A character value of one of these values. ‘poisson’, ‘gaussian’, ‘twoTailQuan-
tile’, ‘oneTailQuintile’. It specifies a way to give weight to each window de-
pending on how much coverage in the normal deviates from the median for that
chromosome. Options are listed in decreasing order of stringency. See Details

DNAcopy.segment
A list of parameters to be passed to function 'segment' in package DNAcopy.
Parameters ‘weights’ and ‘verbose’ are not accepted
addSmooth

Details

'poisson': windows with coverage more or less than 2*sqrt(mean) from the mean are weighted
down. The most stringent. Recommended for unbiased genome wide sequencing.
'gaussian': windows with coverage more or less than 2*sd from the median are weighted down.
Recommended for genome wide sequencing where coverage in the normal is far from poisson
distribution.
'twoTailQuantile': windows with coverage outside 5-95th quantile are weighted down. Recom-
mended when coverage is far from a normal distribution - such as capture experiments -
'oneTailQuantile': windows with coverage lower than 5th quantile are weighted down. Recom-
mended when coverage is far from a normal distribution - such as capture experiments. Does not
weight down windows with very high coverage.

Value

An object of class "CNAnorm"

Methods

signature(object = "CNAnorm") Segment ratio values on an object of class "CNAnorm". Re-
turns the same object with extra slots (segMean, segID)

Author(s)

Stefano Berri <s.berri@leeds.ac.uk> and Arief Gusnanto <a.gusnanto@leeds.ac.uk>

References

the analysis of array CGH data. Bioinformatics

See Also

segMean, CNAnorm-class, DNAcopy, hg19_hs_ideogr

Examples

data(LS041)
CN <- dataFrame2object(LS041)
CN <- addDNAcopy(CN)

Methods for Function addSmooth in Package ‘CNAnorm’

Description

addSmooth segment and smooth perform ratio values in Package ‘CNAnorm’

Usage

## S4 method for signature 'CNAnorm'
addSmooth(object, lambda = 7, ...)
Arguments

object  An object of Class "CNAnorm"
lambda  Degree of smoothness. See reference for more details
...  Further arguments to pass to the function smoothseg

Value

An object of class "CNAnorm"

Methods

signature(object = "CNAnorm") Segment and smooth perform ratio values on an object of
class "CNAnorm". Returns the same object with extra slot (ratio.s)

Author(s)

Stefano Berri <s.berri@leeds.ac.uk> and Arief Gusnanto <a.gusnanto@leeds.ac.uk>

References

segmentation approach for array CGH data analysis. Bioinformatics

See Also

ratio.s, CNAnorm-class

Examples

data(LS041)
CN <- dataFrame2object(LS041)
CN.gcNorm <- gcNorm(CN, exclude = c("chrX", "chrY", "chrM"))
CN.smooth <- addSmooth(CN)

Description

chrs returns/set the name of chromosomes/contigs
arms retruns the name of the chromosome and arm. A data frame containing ideogram information
has to be provided. See ?hg19_hs_ideogr for an example
pos returns/set the position of starting window. Be careful! If you need to change data, it is better
to change the input data and start over.

Usage

chrs(object)
## S4 method for signature 'CNAnorm'
pos(object, show = "start")
## S4 method for signature 'CNAnorm'
arms(object, banding_df)
Arguments

object An object of Class \"CNAnorm\"
show The position to show: \textquotesingle start\textquotesingle , \textquotesingle end\textquotesingle 
banding_df A data frame with information about ideogram

Value

chrs and arms return a character vector, pos returns a numeric vector

Author(s)

Stefano Berri <s.berri@leeds.ac.uk>

See Also

gcNorm, CNAnorm-class, hg19_hs_ideogr

Examples

data(LS041)
data(hg19_hs_ideogr)
CN <- DataFrame2object(LS041)
dataFrameNames <- as.character(LS041$Chr)
objectNames <- chrs(CN)
armNames <- arms(CN, hg19_hs_ideogr)
# check the names are, indeed, the same
all(dataFrameNames == objectNames)
# make shorter names, drop the first three letters (\textquotesingle chr\textquotesingle )
shortNames <- substr(chrs(CN), 4, nchar(chrs(CN)))
chrs(CN) <- shortNames

# retrieve all new names
unique(chrs(CN))
unique(armNames)

---

A \textit{CNAnorm} object with information about most abundant ploidy states, obtained from data LS041.

Description

This data is to provide an object to use in several examples without having to wait for computing it. To see how it was generated, see documentation of function peakPloidy.

Usage

data(CN)

Format

A \textit{CNAnorm} object
Class "CNAnorm"

Description
Class to Contain and Describe copy number aberration (CNA) data from low coverage (approx 0.01 - 0.5X) Next Generation Sequencing

Objects from the Class
Objects can be created by calls of the form `new("CNAnorm", InData)`.

Slots
- **InData**: Object of class "InData". Contains input data provided by the user. All slots have same length. Each element describe one window. See Class "InData"
- **DerivData**: Object of class "DerivData". Contains data derived from "InData". It can be Retrieved by the user, but methods should be used to populate "DerivData". All slots have same length as input data. See Class DerivData
- **Res**: Object of class "Res". Contains slots with obtained results. See Class "Res"
- **Params**: Object of class "Params". Contains crucial parameters passed to some of the methods for reusing in later steps or for documentation.

Methods
Summary of methods for class "CNAnorm". Type "methods ? methodName" for more details about methodName.

- **chrs** signature(object = "CNAnorm"): Retrieve Chromosomes/contig name
- **chrs<-** signature(object = "CNAnorm"): Set Chromosomes/contig name
- **guessPeaksAndPloidy** signature(object = "CNAnorm"): Estimate ploidy of the sample, tumor content and other results saved in Slot "Res"
- **length** signature(x = "CNAnorm"): Returns number of element/windows
- **[** signature(x = "CNAnorm"): Produce on object of class "CNAnorm" with a subser of windows
- **plotGenome** signature(object = "CNAnorm"): Plot annotated normalized genome copy number
- **plotPeaks** signature(object = "CNAnorm"): Plot peaks and estimated/validated ploidy
- **pos** signature(object = "CNAnorm"): Retrieve Chromosomes/contig position
- **pos<-** signature(object = "CNAnorm"): Set Chromosomes/contig position
- **ratio** signature(object = "CNAnorm"): Retrieve ratio (Test/Control). If gcNorm was called, ratio is GC normalized
- **ratio.n** signature(object = "CNAnorm"): Retrieve normalized ratio (not smoothed)
- **ratio.s** signature(object = "CNAnorm"): Retrieve smoothed ratio
- **ratio.n.s** signature(object = "CNAnorm"): Retrieve normalized smoothed ratio
- **segMean** signature(object = "CNAnorm"): Retrieve segmented ratio (as provided by DNA-copy)
- **segMean.n** signature(object = "CNAnorm"): Retrieve normalized segmented ratio
**CNA.normWorkflow**

**Author(s)**
Stefano Berri <s.berri@leeds.ac.uk> and Arief Gusnanto <a.gusnanto@leeds.ac.uk>

**References**
CNA-norm: Discrete Normalization of Copy Number Alteration data from clinical samples (in preparation)

**See Also**
*InData, DerivData* for documentation on the slots.

**Examples**
```r
data(LS041)
CNA <- new("CNAnorm", InData = new("InData", Chr = as.character(LS041$Chr), Pos = LS041$Pos, Test = LS041$Test, Norm = LS041$Norm, GC = LS041$GC))
```

---

**Description**
This function is a wrapper to use for a fully automated CNA.norm workflow where interactivity is not required. It contains MOST possible parameters. Defaults are set to run a standard and conservative workflow.

**Usage**
```r
CNA.normWorkflow(dataFrame, gc.do=FALSE, gc.exclude=character(0), gc.maxNumPoints=10000, smooth.do=TRUE, smooth.lambda=7, smooth.other=list(), peak.method="closest", peak.exclude=character(0), peak.ploidyToTest=12, peak.sd=5, peak.dThresh=0.01, peak.n=2048, peak.adjust=.9, peak.force.smooth=TRUE, peak.reg=FALSE, peak.ds=1.5, peak.zero.count=FALSE, peak.other=list(), DNAcopy.do=TRUE, DNAcopy.independent.arms=FALSE, DNAcopy.ideograms=NULL, DNAcopy.smooth=list(), DNAcopy.segment=list(), DNAcopy.weight=character(), dNorm.normBy=NULL)
```

**Arguments**
- All arguments are explained in the relative functions
- A data frame with columns Chr, Pos, Test, Norm and optional GC. See `dataFrame2object`
- `dataFrame`
- `gc.do` Specify if GC correction need to be done. See `gcNorm`
- `gc.exclude`
- `gc.maxNumPoints`
- `smooth.do` Specify if smoothing need to be done. See `addSmooth`
- `smooth.lambda`
- `peak.method`  
- `peak.exclude`
- `peak.ploidyToTest`
- `peak.sd`
- `peak.dThresh`
- `peak.n`
- `peak.adjust`
- `peak.force.smooth`
- `peak.reg`
- `peak.ds`
- `peak.zero.count`
- `peak.other`
- `DNAcopy.do`  
- `DNAcopy.independent.arms`
- `DNAcopy.ideograms`
- `DNAcopy.smooth`
- `DNAcopy.segment`
- `DNAcopy.weight`
- `dNorm.normBy`
smooth.other  A list of other parameters to pass to the smoothing function. See addSmooth
peak.method  See peakPloidy
peak.exclude  See peakPloidy
peak.ploidyToTest  
  See peakPloidy
peak.sd  See peakPloidy
peak.dThresh  See peakPloidy
peak.n  See peakPloidy
peak.adjust  See peakPloidy
peak.force.smooth  
  See peakPloidy
peak.reg  See peakPloidy
peak.ds  See peakPloidy
peak.zero.count  
  See peakPloidy
peak.other  A list of other parameters to be passed to functions for peak detection. See peakPloidy
DNAcopy.do  Specify if segmentation with DNAcopy need to be done. See addDNACopy
DNAcopy.independent.arms  
  See addDNACopy
DNAcopy.ideograms  
  See addDNACopy
DNAcopy.smooth  See addDNACopy
DNAcopy.segment  
  See addDNACopy
DNAcopy.weight  See addDNACopy
dNorm.normBy  See discreteNorm

Value
An object of Class "CNAnorm"

Author(s)
Stefano Berri <s.berri@leeds.ac.uk>

See Also
dataFrame2object, gcNorm, addSmooth, peakPloidy, addDNACopy, discreteNorm

Examples
data(LS041)
CN <- CNAnormWorkflow(LS041)
**dataFrame2object**

*Convert a data frame into an object of Class "CNAnorm"*

**Description**

Convert a data frame with column: Chr, Pos, Test, Norm and optional GC into object of class "CNAnorm"

**Usage**

`dataFrame2object(dataFrame)`

**Arguments**

- `dataFrame` A data frame with columns Chr, Pos, Test, Norm and optional GC

**Value**

An object of Class "CNAnorm"

**Author(s)**

Stefano Berri <s.berri@leeds.ac.uk>

**See Also**

`CNAnorm-class, InData-class, data.frame`

**Examples**

```r
data(LS041)
CN <- dataFrame2object(LS041)
```

---

**DerivData-class**

*Class "DerivData"*

**Description**

A Class containing data derived from InData used for further computation and plotting.

**Objects from the Class**

Objects can be created by calls of the form `new("DerivData")`. However DerivData is typically populated using methods. If a slot has not been populated yet, it has zero length, otherwise slots have the same length as InData.
discreteNorm

Slots

ratio: Numeric vector with ratio Test/Normal. Optionally GC corrected.
ratios: Numeric vector with smoothed ratio.
ratio.n: Numeric vector with normalized ratio.
ratio.s.n: Numeric vector with normalized and smoothed ratio.
segID: Numeric vector with ID of segmented data (as provided by DNACopy). Each segment has a different ID.
segMean: Numeric vector with mean value of the segment (as provided by DNACopy.)
segMean.n: Numeric vector with normalized segMean.
ok4density: Logical vector. Specify which values have been used to calculate density.

Methods

length signature(x = "DerivData"): Returns number of windows.

Author(s)
Stefano Berri and Arief Gusnanto

References


See Also

CNAnorm, InData-class

Examples

data(LS041)
inObject <- new("InData", Chr = as.character(LS041$Chr),
    Pos = LS041$Pos, Test = LS041$Test, Norm = LS041$Norm,
    GC = LS041$GC)
CNA <- new("CNAnorm", InData = inObject)

discreteNorm Methods for Function addSmooth in Package 'CNAnorm'

discreteNorm performs normalization of data using information on ploidy. Implicitly calls "validation" if no validation was performed

Usage

## S4 method for signature 'CNAnorm'
discreteNorm(object, normBy = object)
exportTable

Arguments

object An object of Class "CNAnorm" to normalize

normBy An object of Class "CNAnorm" used to set normalization. It is possible, for instance, to normalize a sample at high resolution, using information obtained from the same sample at low resolution

Value

An object of class "CNAnorm"

Author(s)

Stefano Berri <s.berri@leeds.ac.uk> and Arief Gusnanto <a.gusnanto@leeds.ac.uk>

References


See Also

validation, peakPloidy

Examples

data(CN)
# see peakPloidy documentation to know how object CN is created
CN <- discreteNorm(CN)

exportTable

Methods for Function exportTable in Package 'CNAnorm'

Description

exportTable write a table with normalised values of each window. A wrapper to "write.table"

Usage

## S4 method for signature 'CNAnorm'
exportTable(object, file = "CNAnorm_table.tab", show = 'ratio',
sep = "\t", row.names = FALSE, ...)

Arguments

object an object of Class "CNAnorm"

file name of the file to save to

show what should be reported in the table: "ratio": the normalized ratio (a value of 1 means diploid). "ploidy": the same as ratio * 2. "center": report ratio centered on the most abundant copy. Ratio of 1 means that the most abundant "state" is centered to 1
The field separator string.

either a logical value indicating whether the row number should be written or a character vector of row names to be written.

Extra arguments to be passed to "write.table"

It produces a tab delimited text file with the following columns:

- **Chr**: Chromosome/contig name.
- **Pos**: Starting position of the window.
- **Ratio**: Ratio Test/Normal for each window after GC correction.
- **Ratio.n**: Ratio Test/Normal or ploidy for each window after normalisation.
- **Ratio.s.n**: Smoothed and normalised ratio Test/Normal or ploidy for each window.
- **SegMean**: Mean of the segment this window belongs to.
- **SegMean.n**: Normalised mean ratio Test/Normal or ploidy of the segment this window belongs to.

An object of class "CNAnorm"

Stefano Berri <s.berri@leeds.ac.uk>

**See Also**

write.table

**Examples**

data(CN)
CN <- validation(CN)
CN <- discreteNorm(CN)
exportTable(CN, file = "CNAnorm_table.tab", show = 'ploidy')

---

**Description**

gcNorm perform GC content normalization on ratio Test/Normal in Package ‘CNAnorm’

**Usage**

```r
## S4 method for signature 'CNAnorm'
gcNorm(object, exclude = character(0), maxNumPoints = 10000)
```
Arguments

object: An object of Class "CNAnorm"
exclude: A character vector with name of chromosomes/contigues not to use to calculate GC content correction. All genome, however, will be corrected
maxNumPoints: Maximum number of data points to fit the loess correction. For computational purposes, if the number of points in ratio(object) is greater than maxNumPoints, only maxNumPoints randomly selected will be used

Value

An object of class "CNAnorm"

Methods

signature(object = "CNAnorm") Perform GC content correction on an object of class "CNAnorm". Returns the same object with corrected ratio

Author(s)

Stefano Berri <s.berri@leeds.ac.uk>

See Also

loess, CNAnorm-class, ratio

Examples

data(LS041)
CN <- DataFrame2object(LS041)
# correct for GC content, but ignoring data from sex chromosomes and # mitochondria
CN.gcNorm <- gcNorm(CN, exclude = c("chrX", "chrY", "chrM"))

---

gPar

An object with the default graphical parameters

Description

This data object is used by some plotting methods and contains the default values. User can change graphical parameters by changing this object

The object consists of several layers refering to different plots and different properties. Here an indicative description

gPar$genome: parameters here refer to the plot produced by plotGenome
graphical parameters: see ?par
$colors: specify colors $cex: specify relative size - points, text... $lwd: specify line width $lty: specify line type - solid, dashed $mar: specify margins

Usage

data(gPar)
InData-class

Format
A S3 object

hg19_hs_ideogr  
An object with the ideogram information for homo sapiens - hg19

Description
This is bundles data that can be provided to plotGenome in order to plot location of the centromere. In future release it might be used to produce an ideogram too

Usage
data(hg19_hs_ideogr)

Format
A data.frame

InData-class  
Class "InData"~~~

Description
A Class containing input data for CNA

Objects from the Class
Objects can be created by calls of the form new("InData", Chr, Pos, Test, Norm, ratio, GC).

Slots
Chr: Object of class "character". Name of the Chromosomes/Contigs of each window.
Pos: Object of class "numeric". Starting position of the each window.
Test: Object of class "numeric". Number of reads from Test in each window.
Norm: Object of class "numeric". Number of reads from Normal (Control) in each window.
ratio: Object of class "numeric". Ratio Test/Control in each window. Automatically computed if Test and Norm are provided, or user generated if Test and Norm are not know.
GC: Object of class "numeric". GC content of each window.

Methods
length signature(x = "InData"): Returns number of windows from input data.

Author(s)
Stefano Berri
References


See Also

CNAnorm

Examples

data(LS041)
inObject <- new("InData", Chr = as.character(LS041$Chr), Pos = LS041$Pos, Test = LS041$Test, Norm = LS041$Norm, GC = LS041$GC)
CNA <- new("CNAnorm", InData = inObject)

LS041

Mapped reads in tumor and matched blood for patient LS041

Description

This data set provide reads in tumor and matched blood for patient LS041. Each row has information about non-overlapping window across the genome. In particular it reports: chromosome name, starting position of the window (1 based), number of mapped reads in the test (lung tumor), number of reads in the control (matched blood) and GC content of the window.

Usage

data(LS041)

Format

A dataframe

References

Description

A Class containing some Parameters used in the analysis. Not heavily used at the moment.

Objects from the Class

Objects can be created by calls of the form new("Params"), it is usually iniiziated and populated with methods and functions of class CNAnorm.

Slots

- method: variable of class "character". Record if the peakPloidy function was called using density or mixture.
- density.n: The variable "n" used when calling peakPloidy. This variable is saved so that can be used later for drawing plots.
- density.adjust: The variable "adjust" used when calling peakPloidy. This variable is saved so that can be used later for drawing plots.
- gc.excludeFromGCNorm: Vector of class "character". Name of the Chromosomes/Contigs not used for GC content correction.
- gc.maxNumPoints: One element vector of class "numeric". Specify how many points to use for GC correction.
- gp.excludeFromDensity: Vector of class "character". Name of the Chromosomes/Contigs not used for peak guessing.

Methods

- length signature(x = "Params")

Author(s)

Stefano Berri

References


See Also

CNAnorm

Examples

data(LS041)
inObject <- new("InData", Chr = as.character(LS041$Chr), Pos = LS041$Pos,
Test = LS041$Test, Norm = LS041$Norm, GC = LS041$GC)
CNA <- new("CNAnorm", InData = inObject)
Methods for Function `peakPloidy` in Package ‘CNAnorm’

Description

`peakPloidy` Estimate most likely ploidy of genome looking at distribution of smoothed ratio.

Usage

```r
## S4 method for signature 'CNAnorm'
peakPloidy(object, method = 'mixture', exclude = character(0),
            ploidyToTest = 12, sd = 5, dThresh = 0.01, n = 2048, adjust = .9, force.smooth = TRUE,
            reg = FALSE, ds = 1.5, zero.cont = FALSE, ...)
```

Arguments

- `object` An object of Class "CNAnorm"
- `exclude` A character vector with names of Chromosomes/Contigs not to use to estimate ploidy.
- `method` A character element matching either "mixture", "density", "median", "mode" or "closest". "mixture" will fit a mixture model to find peaks; "density" will use the density function to find peaks; "median" "mode" and "closest" will only find one peak at the median, mode or peak closest to the median respectively. No tumour content or reliable estimated ploidy will be provided. These methods are meant to perform a more "standard" normalisation, without stretching the data. Suggested for germline CNV or a fully automated process that does not require a normalisation on integer copy number or for highly heterogeneous sample where such normalisation would not be possible. Non ambiguous partial matches can be used.
- `ploidyToTest` Maximum ploidy allowed. **Warnings!** Computation time increases exponentially with this parameter if using "density".
- `adjust` The "adjust" parameter passed to the `density` function.
- `n` The "n" parameter passed to the `density` function.
- `force.smooth` If the input object does not have smoothed ratio, it will smooth using "addSmooth". It is highly recommended to use "force.smooth = TRUE"
- `sd` Parameter to filter outliers. Values greater than i median + sd * standard deviations will be ignored while detecting peaks and ploidy.
- `dThresh` Parameter to filter outliers. Values with a density lower than max(density)*dThresh will be ignored while detecting peaks and ploidy.
- `reg` Parameter for mixture model: If set TRUE, the starting point for EM algorithm will be optimized through several methods including regular grid on the ratio distribution. The default is FALSE, by which the starting values are taken from the quantiles of the distribution.
- `ds` Parameter for mixture model: A constraint in EM algorithm of minimum distance between mean estimates, in terms of median standard deviation of the mixture components.
- `zero.cont` Parameter for mixture model: An argument for the mixture model. If set TRUE, the EM algorithm considers exactly-zero ratios as a mixture component.
- `...` Extra parameters to be passed to functions for peak detection, specific to each of the methods (density or mixture), see below for details.
An object of class "CNAnorm"

Other optional parameters to be passed (…)

**mixture method**

**density method**

**peakRatio** Threshold used to call a peak. Peaks smaller than maxPeakHeight/peakRatio are not considered peaks.

**spacingTolerance** A parameter smaller than 1 which describes how strict the program should be on alternative solutions. Only solution with the best $R^2 \geq max(R^2) \times spacingTolerance$ will be considered as valid.

**interceptRatio** Minimum value for the intercept of the linear model. Ideally, should be zero, but the default allows a little flexibility.

Stefano Berri <s.berri@leeds.ac.uk> and Arief Gusnanto <a.gusnanto@leeds.ac.uk>


**See Also**

CNAnorm-class, density

**Examples**

data(LS041)
CN <- dataFrame2object(LS041)
chr2skip <- c("chrY", "chrM")
CN <- gcNorm(CN, exclude = chr2skip)
CN <- addSmooth(CN, lambda = 3)
CN <- peakPloidy(CN, exclude = chr2skip)
# this object CN is what you obtain when you load
# data(CN)

plotGenome plot normalized ratio and optionally segmented and/or smoothed normalized ratio values in Package ‘CNAnorm’. It also shows annotation.
## Usage

```r
## S4 method for signature 'CNAnorm'
plotGenome(object, maxRatio = 8, minRatio = -1,
    superimpose = character(0), gPar = NULL, numHorLables = 10,
    colorful = FALSE, show.centromeres = TRUE, idiogram = NULL, fixVAxes = FALSE,
    supLineColor = character(0), supLineCex = character(0), dot.cex = .2, ...)
```

### Arguments

- **object**: An object of Class "CNAnorm"
- **maxRatio**: The maximum ratio to be shown on the plot. Values or ratio greater than maxRatio will be displayed as green triangulars
- **minRatio**: The minimum ratio to be shown on the plot. Values or ratio smaller than minRatio will be displayed as green triangulars
- **superimpose**: A character vector with one or both of the following: "smooth", "DNACopy"
- **numHorLables**: Number of maximum horizontal labels. The function will try to annotate numHorLables so that they are approximately equally spaced.
- **colorful**: A switch to decide if the background dots representing the ratio of each window should be gray or colored according their value in relation to the peak closest to the median
- **show.centromeres**: A switch to decide if location of centromere are displayed on the graph. The location of the centromere is stored in idiogram
- **idiogram**: A data frame containing banding information. if NULL -default- human information will be loaded by data(hg19_hs_ideogr)
- **fixVAxes**: A switch to decide if the vertical axes should be fixed to minRatio and maxRatio or fit the data within minRatio and maxRatio.
- **gPar**: a S3 object with all graphical parameters. If NULL (default) data(gPar) is called
- **supLineColor**: A three element character vector with colors to be used for first superimposed line, second superimposed line, normalized ratio dots. If none is provided, defaults are: c("black", "cyan", "grey60")
- **supLineCex**: A two element vector with cex value to be used for width of first superimposed line and second superimposed line. If none is provided, defaults are: c(0.5, 0.5)
- **dot.cex**: size of the dots in the plot
- **...**: Further arguments to pass to the function plot

### Value

An object of class "CNAnorm"

### Author(s)

Stefano Berri <s.berri@leeds.ac.uk> and Arief Gusnanto <a.gusnanto@leeds.ac.uk>

### See Also

plot, par, peakPloidy, gPar, hg19_hs_ideogr
Examples

```r
data(CN)
# see peakPloidy documentation to know how object CN is created
CN <- addDNACopy(CN)
CN <- validation(CN)
CN <- discreteNorm(CN)
plotGenome(CN, superimpose = 'DNACopy')
```

Description

`plotPeaks` plot annotated distribution of ratio Test/Normal

Usage

```r
## S4 method for signature 'CNAnorm'
plotPeaks(object, special1 = character(0), special2 = character(0),
          show = c('suggested', 'validated'), ...)
```

Arguments

- **object**: An object of Class "CNAnorm"
- **special1**: The chromosome/contig whose distribution will be shown with a different color
- **special2**: The chromosome/contig whose distribution will be shown with a different color
- **show**: A character vector with one or both of the following: "suggested", "validated". Specify what need to be plotted
- **...**: Further arguments to pass to the function `plot`

Author(s)

Stefano Berri <s.berri@leeds.ac.uk>

See Also

`plot`, `validation`, `peakPloidy`

Examples

```r
data(CN)
# see peakPloidy documentation to know how object CN is created
plotPeaks(CN, special1 = 'chrX', special2 = 'chrY')
```
Methods for Function ratio in Package ‘CNAnorm’

Description

ratio returns the Test/Normal ratio from an object of class CNAnorm. ratio is corrected for GC content if gcNorm was called.
	ratio.n returns the Test/Normal normalized ratio from an object of class CNAnorm after normalization. Its input is ratio(object)
	ratio.s returns the Test/Normal smoothed ratio from an object of class CNAnorm. Its input is ratio(object)
	ratio.s.n returns the Test/Normal smoothed and normalized ratio from an object of class CNAnorm. Its input is ratio.s(object)

segMean returns the mean of the segments as produced by DNACopy

segMean.n returns the normalized mean of the segments

Usage

ratio(object)
ratio.n(object)
ratio.s(object)
ratio.s.n(object)
segMean(object)
segMean.n(object)

Arguments

object An object of Class "CNAnorm"

Value

A numeric vector

Author(s)

Stefano Berri <s.berri@leeds.ac.uk>

See Also

gcNorm, CNAnorm-class, DNACopy

Examples

data(LS041)
CN <- dataFrame2object(LS041)
ratio.original <- ratio(CN)
CN.gcNorm <- gcNorm(CN, exclude = c("chrX", "chrY", "chrM"))
ratio.gc.corrected <- ratio(CN.gcNorm)
retrieve peaks and ploidy

Methods for Function to retrieve suggested/validated ploidy and peaks in Package ‘CNAnorm’

Description

sugg.peaks returns the location of peaks before normalization as suggested by peakPloidy.
sugg.ploidy returns the ploidy of the peaks as suggested by peakPloidy.
valid.peaks returns the location of peaks before normalization as validated after calling method "validation"
valid.ploidy returns the validated ploidy of the peaks as validated after calling method "validation"

Usage

sugg.peaks(object)
sugg.ploidy(object)
valid.peaks(object)
valid.ploidy(object)

Arguments

object An object of Class "CNAnorm" after method "peakPloidy" was called

Value

A numeric vector

Author(s)

Stefano Berri <s.berri@leeds.ac.uk>

See Also

gcNorm, CNAnorm-class, DNAcopy

Examples

data(CN)
# see peakPloidy documentation to know how object CN is created
plot(sugg.ploidy(CN), sugg.peaks(CN))
Methods for Function `addSmooth` in Package ‘CNAnorm’

Description
validation segment and smooth perform ratio values in Package ‘CNAnorm’

Usage
```
## S4 method for signature 'CNAnorm'
validation(object, peaks = sugg.peaks(object),
          ploidy = sugg.ploidy(object))
```

Arguments
- `object`: An object of Class “CNAnorm”
- `peaks`: The user validated location (ratio Test/Normal) of the peaks before normalization.
- `ploidy`: The user validated ploidy of the peaks before normalization.

Value
An object of class "CNAnorm"

Note
It is implicitly called by `discreteNorm` if no validation was manually performed

Author(s)
Stefano Berri <s.berri@leeds.ac.uk>

See Also
`ratio.s`, `CNAnorm-class`

Examples
```
data(CN)
# see peakPloidy documentation to know how object CN is created
CN <- validation(CN)
```
Index

*Topic classes
  CNAnorm-class, 6
  DerivData-class, 9
  InData-class, 14
  Params-class, 16

*Topic convert, data.frame
dataFrame2object, 9

*Topic datasets, dataframe
  LS041, 15

*Topic datasets, object
  CN, 5
  gPar, 13
  hg19_hs_ideogr, 14

*Topic methods, accessor, get, set
  ratio, 21
  retrieve peaks and ploidy, 22

*Topic methods, accessor, get
  chromosomesPosition, 4

*Topic methods, export, dataframe, table
  exportTable, 11

*Topic methods, normalization
  gcNorm, 12
  peakPloidy, 17

*Topic methods, segmentation, smoothing
  addDNACopy, 2
  addSmooth, 3

*Topic workflow
  CNAnormWorkflow, 7
  [,CNAnorm-method (CNAnorm-class), 6
  chromosomesPosition, 4
  chrs (chromosomesPosition), 4
  chrs, CNAnorm-method (CNAnorm-class), 6
  chrs-methods (chromosomesPosition), 4
  chrs< (chromosomesPosition), 4
  chrs<-, CNAnorm-method (CNAnorm-class), 6
  chrs<--methods (chromosomesPosition), 4
  CN, 5
  CNAnorm, 10, 15, 16
  CNAnorm (CNAnorm-class), 6
  CNAnorm-class, 6
  CNAnormWorkflow, 7

  data.frame, 9
  dataFrame2object, 8, 9
  density, 18
  DerivData, 7
  DerivData-class, 9
  discreteNorm, 8, 10
  discreteNorm, CNAnorm-method (discreteNorm), 10
  discreteNorm-methods (discreteNorm), 10
  exportTable, 11
  exportTable, CNAnorm-method (exportTable), 11
  exportTable-methods (exportTable), 11
  gcNorm, 5, 8, 12, 21, 22
  gcNorm, CNAnorm-method (gcNorm), 12
  gcNorm-methods (gcNorm), 12
  gPar, 13, 19
  hg19_hs_ideogr, 3, 5, 14, 19

  InData, 7
  InData-class, 14

  length, CNAnorm-method (CNAnorm-class), 6
  length, DerivData-method (DerivData-class), 9
  length, InData-method (InData-class), 14
  loess, 13
  LS041, 15
INDEX

par, 19
Params-class, 16
peakPloidy, 8, 11, 17, 19, 20
peakPloidy,CNAnorm-method (peakPloidy), 17
peakPloidy-methods (peakPloidy), 17
plot, 19, 20
plotGenome, 18
plotGenome,CNAnorm-method (plotGenome), 18
plotGenome-methods (plotGenome), 18
plotPeaks, 20
plotPeaks,CNAnorm-method (plotPeaks), 20
plotPeaks-methods (plotPeaks), 20
pos (chromosomesPosition), 4
pos,CNAnorm-method (chromosomesPosition), 4
pos-methods (chromosomesPosition), 4
pos<-(chromosomesPosition), 4
pos<-,CNAnorm-method (CNAnorm-class), 6
pos<--methods (chromosomesPosition), 4
ratio, 13, 21
ratio,CNAnorm-method (CNAnorm-class), 6
ratio-methods (ratio), 21
ratio.n (ratio), 21
ratio.n,CNAnorm-method (CNAnorm-class), 6
ratio.n-methods (ratio), 21
ratio.s, 4, 23
ratio.s (ratio), 21
ratio.s,CNAnorm-method (CNAnorm-class), 6
ratio.s-methods (ratio), 21
ratio.s.n (ratio), 21
ratio.s.n,CNAnorm-method (CNAnorm-class), 6
ratio.s.n-methods (ratio), 21
retrieve peaks and ploidy, 22
segMean, 3
segMean (ratio), 21
segMean,CNAnorm-method (CNAnorm-class), 6
segMean-methods (ratio), 21
segMean.n (ratio), 21
segMean.n,CNAnorm-method (CNAnorm-class), 6
segMean.n-methods (ratio), 21
sugg.peaks (retrieve peaks and ploidy), 22
sugg.peaks,CNAnorm-method (CNAnorm-class), 6
sugg.peaks-methods (retrieve peaks and ploidy), 22
sugg.ploidy (retrieve peaks and ploidy), 22
sugg.ploidy,CNAnorm-method (CNAnorm-class), 6
sugg.ploidy-methods (retrieve peaks and ploidy), 22
valid.peaks (retrieve peaks and ploidy), 22
valid.peaks,CNAnorm-method (CNAnorm-class), 6
valid.peaks-methods (retrieve peaks and ploidy), 22
valid.ploidy (retrieve peaks and ploidy), 22
valid.ploidy,CNAnorm-method (CNAnorm-class), 6
valid.ploidy-methods (retrieve peaks and ploidy), 22
validation, 11, 20, 23
validation,CNAnorm-method (validation), 23
validation-methods (validation), 23
write.table, 12