Package ‘CGHbase’
January 30, 2017

Type Package
Title CGHbase: Base functions and classes for arrayCGH data analysis.
Version 1.34.0
Date 2015-12-01
Author Sjoerd Vosse, Mark van de Wiel
Maintainer Mark van de Wiel <mark.vdwiel@vumc.nl>
Depends R (>= 2.10), methods, Biobase (>= 2.5.5), marray
Description Contains functions and classes that are needed by arrayCGH packages.
License GPL
   methods-cghSeg.R methods-cghCall.R methods-cghRegions.R
biocViews Infrastructure, Microarray, CopyNumberVariation
URL https://github.com/tgac-vumc/CGHbase
BugReports https://github.com/tgac-vumc/CGHbase/issues
NeedsCompilation no

R topics documented:

CGHbase-package ................................................. 2
avedist ....................................................... 2
cghCall ....................................................... 3
cghRaw ......................................................... 5
cghRegions .................................................... 7
cghSeg ......................................................... 8
chromosomes ................................................... 10
copynumber ................................................... 11
frequencyPlot .................................................. 12
frequencyPlotCalls ............................................. 13
make_cghRaw .................................................. 14
plot.cghRaw ................................................... 14
probloss ....................................................... 15
regions ......................................................... 16
summaryPlot .................................................... 17
Wilting ......................................................... 18
CGHbase-package

CGHbase: Base functions and classes for arrayCGH data analysis.

Description

CGHbase: Base functions and classes for arrayCGH data analysis.

Details

Main infrastructural classes: cghRaw, cghSeg, cghCall. Full help on methods and associated functions is available from within class help pages.


Author(s)

Sjoerd Vosse <sjoerdvos@yahoo.com>

avedist

Retrieve regions information from cghRegions object.

Description

This function accesses the regions information stored in the featureData of an object derived from the cghRegions-class.

Usage

avedist(object)
nclone(object)

Arguments

object Object derived from class cghRegions

Value

avedist returns a vector containing the Average L1-distance of clone signatures to the medoid signature; nclone returns a vector containing the number of clones that is included in each region;

Author(s)

Sjoerd Vosse
See Also
cghRegions-class

cghCall

Class to contain and describe called array comparative genomic hybridization data.

Description

Container for aCGH data and experimental metadata. cghCall class is derived from eSet, and requires the following matrices of equal dimension as assayData members:

- copynumber
- segmented
- calls
- probloss
- probnorm
- probgain

Furthermore, columns named Chromosome, Start, and End are required as featureData members, containing feature position information.

Extends

Directly extends class eSet.

Creating Objects

ew('cghCall', phenoData = [AnnotatedDataFrame], experimentData = [MIAME], annotation = [character], copynumber = [matrix], probloss = [matrix], probnorm = [matrix], probgain = [matrix], featureData = [AnnotatedDataFrame], ...)

An object of class cghCall is generally obtained as output from CGHcall.

Slots

Inherited from eSet:

assayData: Contains matrices with equal dimensions, and with column number equal to nrow(phenoData).

assayData must contain the following matrices

- copynumber
- segmented
- calls
- probloss
- probnorm
- probgain

with rows representing array probes and columns representing samples. Additional matrices of identical size (e.g., representing measurement errors) may also be included in assayData.

Class: AssayData-class

phenoData: See eSet
featureData: An AnnotatedDataFrame with columns Chromosome, Start, and End containing array element position data.

experimentData: See eSet

annotation: See eSet

Methods

Class-specific methods.

`copynumber(cghCall), copynumber(cghCall,matrix)`<- Access and set elements named `copynumber` in the AssayData-class slot.

`segmented(cghCall), segmented(cghCall,matrix)`<- Access and set elements named `segmented` in the AssayData-class slot.

`calls(cghCall), calls(cghCall,matrix)`<- Access and set elements named `calls` in the AssayData-class slot.

`probloss(cghCall), probloss(cghCall,matrix)`<- Access and set elements named `probloss` in the AssayData-class slot.

`probnorm(cghCall), probnorm(cghCall,matrix)`<- Access and set elements named `probnorm` in the AssayData-class slot.

`probgain(cghCall), probgain(cghCall,matrix)`<- Access and set elements named `probgain` in the AssayData-class slot.

`chromosomes, bpstart, bpend` Access the chromosomal positions stored in `featureData`

`plot` Create a plot containing log2ratios, segments and call probabilities ordered by chromosomal position. EXTRA OPTIONS PLUS DEFAULTS: `dotres=10`. Every `dotres`-th log2-ratio is plotted. `dotres=1` plots all data. However, higher values save a lot of space and allow quicker browsing of the plots. `ylim=c(-5,5)`: limits of the y-axis. `gaincol='green'; losscol='red'; ampcol='darkgreen'; dlcol='darkred'`: Colors used for gain, loss (bars) and amplifications, double loss (tick marks). `build='GRCh37'`: Build of human genome used for determining positions of centromeres

`plot.summary` Create a plot summarizing the call probabilities of all samples

`frequencyPlotCalls` Create a frequency plot summarizing the calls of all samples

See `eSet` for derived methods.

Author(s)

Sjoerd Vosse

See Also

eSet-class, cghRaw-class, cghSeg-class

Examples

```r
# create an instance of cghCall
new("cghCall")

# create an instance of cghCall through \code{\link(ExpandCGHcall)}
## Not run:
data(Wilting)
rawcgh <- make_cghSeg(Wilting)
```
normalized <- normalize(rawcgh)
segmented <- segmentData(normalized)
perc.tumor <- rep(0.75, 3)
listcalled <- CGHcall(segmented,cellularity=perc.tumor)
called <- ExpandCGHcall(listcalled,segmented)

# plot the first sample. Default only every 10th log2-ratio is plotted (dotres=10). Adjust using dotres= option
plot(called[,1])
# plot the first chromosome of the first sample
plot(called[chromosomes(called)==1,1])

# get the copynumber values of the third and fourth sample
log2ratios <- copynumber(called[,3:4])

# get the names of the samples
sampleNames(called)

# get the names of the array elements
featureNames(called)

## End(Not run)

cghRaw

Class to contain and describe raw or normalized array comparative genomic hybridization data.

Description

Container for aCGH data and experimental metadata. cghRaw class is derived from eSet, and requires a matrix named copynumber as assayData member. Furthermore, columns named Chromosome, Start, and End are required as featureData members, containing feature position information.

Extends

Directly extends class eSet.

Creating Objects

new('cghRaw', phenoData = [AnnotatedDataFrame], experimentData = [MIAME], annotation = [character], copynumber = [matrix], featureData = [AnnotatedDataFrame], ...

make_cghRaw is a function to convert a dataframe or textfile to an object of class cghRaw. The input should be either a dataframe or a tabseparated textfile (textfiles must contain a header). The first three columns should contain the name, chromosome and position in bp for each array target respectively. The chromosome and position column must contain numbers only. Following these is a column with log2 ratios for each of your samples. If the input type is a textfile, missing values should be represented as 'NA' or an empty field.

Slots

Inherited from eSet:
assayData: Contains matrices with equal dimensions, and with column number equal to nrow(phenoData).
assayData must contain a matrix copynumber with rows representing array probes and columns representing samples. Additional matrices of identical size (e.g., representing measurement errors) may also be included in assayData. Class: AssayData-class
phenoData: See eSet
featureData: An AnnotatedDataFrame with columns Chromosome, Start, and End containing array element position data.
experimentData: See eSet
annotation: See eSet

Methods
Class-specific methods.
copynumber(cghRaw), copynumber(cghRaw,matrix) <- Access and set elements named copynumber in the AssayData-class slot.
chromosomes, bpstart, bpend Access the chromosomal positions stored in featureData
plot Create a plot containing log2ratios ordered by chromosomal position
See eSet for derived methods. Annotation functionality is not yet supported.

Author(s)
Sjoerd Vosse

See Also
eSet-class, cghSeg-class, cghCall-class

Examples

# create an instance of cghRaw
new("cghRaw")

# create an instance of cghRaw from a dataframe
data(Wilting)
rawcgh <- make_cghRaw(Wilting)

# plot the first sample
plot(rawcgh[,1])
# first three chromosomes
plot(rawcgh[chromosomes(rawcgh)==1,1])

# get the copynumber values of the third and fourth sample
log2ratios <- copynumber(rawcgh[,3:4])

# get the names of the samples
sampleNames(rawcgh)

# get the names of the array elements
featureNames(rawcgh)
**cghRegions**

Class to contain and describe array comparative genomic hybridization regions data.

**Description**

Container for aCGH regions data and experimental metadata. `cghRegions` class is derived from `eSet`, and requires a matrix named `regions` as assayData member. Furthermore, columns named `Chromosome`, `Start`, `End`, `Nclone`, and `Avedist` are required as featureData members, containing region and position information.

**Extends**

Directly extends class `eSet`.

**Creating Objects**

```r
new('cghRegions', phenoData = [AnnotatedDataFrame], experimentData = [MIAME], annotation = [character], regions = [matrix], featureData = [AnnotatedDataFrame], ...)
```

An object of this class is generally obtained by running the function `CGHregions`.

**Slots**

Inherited from `eSet`:

- **assayData**: Contains matrices with equal dimensions, and with column number equal to `nrow(phenoData)`. `assayData` must contain a matrix `regions` with rows representing regions and columns representing samples. Additional matrices of identical size (e.g., representing measurement errors) may also be included in `assayData`. Class: `AssayData`

- **phenoData**: See `eSet`

- **featureData**: An `AnnotatedDataFrame` with columns `Chromosome`, `Start`, `End`, `Nclone`, and `Avedist` containing region and position information.

- **experimentData**: See `eSet`

- **annotation**: See `eSet`

**Methods**

Class-specific methods.

- `regions(cghRegions), regions(cghRegions,matrix)<-` Access and set elements named `regions` in the `AssayData-class` slot.

- `chromosomes`, `bpstart`, `bpend`, `nclone`, `avedist` Access the region and position information stored in `featureData`

- **plot.cghRegions** Create a plot displaying chromosomes on the Y-axis and base pair position on the X-axis. A new region is displayed by a slight jump with respect to the previous region. Each region is displayed as a bi-colored segment, the lower and upper part of which correspond to the proportions `pl` and `pg` of samples with a loss (red) or gain (green), respectively. The color coding is displayed as well: 1: `pl` (pg) < 10%; 2: 10% = `pl` (pg) < 30%; 3: 30% = `pl` (pg) < 50%; 4: `pl` (pg) = 50%.

- **frequencyPlot** Create a frequency plot

See `eSet` for derived methods. Annotation functionality is not yet supported.
cghSeg

Author(s)

Sjoerd Vosse

See Also

eSet, cghRaw-class, cghSeg-class, cghCall-class

Examples

```r
# create an instance of cghRegions
new("cghRegions")

# load an instance of cghRegions
data(WiltingRegions)

# plot all region data
plot(WiltingRegions)
# make a frequency plot
frequencyPlot(WiltingRegions)

# extract the region values
values <- regions(WiltingRegions)

# get the names of the samples
sampleNames(WiltingRegions)
```

cghSeg

Class to contain and describe segmented array comparative genomic hybridization data.

Description

Container for aCGH data and experimental metadata. cghSeg class is derived from eSet, and requires a matrix named copynumber as well as a matrix named segmented as assayData members of equal dimensions. Furthermore, columns named Chromosome, Start, and End are required as featureData members, containing feature position information.

Extends

Directly extends class eSet.

Creating Objects

```r
new('cghSeg', phenoData = [AnnotatedDataFrame], experimentData = [MIAME], annotation = [char]
```

An object of class cghSeg is generally obtained as output from segmentData.
**Slots**

Inherited from eSet:

- **assayData**: Contains matrices with equal dimensions, and with column number equal to \texttt{nrow(phenoData)}. assayData must contain matrices copynumber and segmented with rows representing array probes and columns representing samples. Additional matrices of identical size (e.g., representing measurement errors) may also be included in assayData. Class: \texttt{AssayData-class}

- **phenoData**: See \texttt{eSet}

- **featureData**: An \texttt{AnnotatedDataFrame} with columns Chromosome, Start, and End containing array element position data.

- **experimentData**: See \texttt{eSet}

- **annotation**: See \texttt{eSet}

**Methods**

Class-specific methods.

copynumber(cghSeg), copynumber(cghSeg,matrix)<- Access and set elements named copynumber in the AssayData-class slot.

segmented(cghSeg), segmented(cghSeg,matrix)<- Access and set elements named segmented in the AssayData-class slot.

- **chromosomes**, **bpstart**, **bpend** Access the chromosomal positions stored in featureData

- **plot**: Create a plot containing log2ratios and segments ordered by chromosomal position. TWO EXTRA OPTIONS PLUS DEFAULTS: dotres=10. Every dotres-th log2-ratio is plotted. dotres=1 plots all data. However, higher values save a lot of space and allow quicker browsing of the plots. ylimit=c(-2,5): limits of the y-axis

See \texttt{eSet} for derived methods.

**Author(s)**

Sjoerd Vosse

**See Also**

\texttt{eSet-class}, \texttt{cghRaw-class}, \texttt{cghCall-class}

**Examples**

```r
# create an instance of cghSeg
new("cghSeg")

# create an instance of cghSeg through \code{segmentData}
## Not run:
data(Wilting)
rawcgh <- make_cghSeg(Wilting)
normalized <- normalize(rawcgh)
segmented <- segmentData(normalized)

# plot the first sample. Default only every 10th log2-ratio is plotted (dotres=10). Adjust using dotres= option
plot(segmented[,1])
# first three chromosomes
plot(segmented[chromosomes(segmented)<=3,1])
```
chromosomes

Retrieve feature position data from cgh objects.

Description
These generic functions access the position data stored in the featureData of an object derived from the cghRaw-class, cghSeg-class or cghCall-class.

Usage
chromosomes(object)
bpstart(object)
bpend(object)

Arguments
object Object derived from class cghRaw, cghSeg, or cghCall

Value
chromosomes returns a vector of chromosome numbers; bpstart returns a vector of basepair start positions; bpend returns a vector of basepair end positions;

Author(s)
Sjoerd Vosse

See Also
cghRaw-class, cghSeg-class, cghCall-class
copynumber

Retrieve copynumber data from cgh objects.

Description

These generic functions access the copynumber values of assay data stored in an object derived from the cghRaw-class, cghSeg-class or cghCall-class.

Usage

```r
copynumber(object)
copynumber(object) <- value
segmented(object)
segmented(object) <- value
calls(object)
calls(object) <- value
```

Arguments

- `object` Object derived from class cghRaw, cghSeg, or cghCall
- `value` Matrix with rows representing features and columns samples.

Value

`copynumber` returns a matrix of copynumber values;

Author(s)

Sjoerd Vosse

See Also

cghRaw-class, cghSeg-class, cghCall-class

Examples

```r
data(WiltingCalled)
log2ratios <- copynumber(WiltingCalled)
segments <- segmented(WiltingCalled)
calls <- calls(WiltingCalled)
```
Description
This function creates a frequency plot for aCGH regions.

Usage
frequencyPlot(x, y, ...)

Arguments
x An object of class cghRegions.
y This argument is not used and should be missing.
... Arguments plot.

Details
We find plotted on the x-axis the array probes sorted by chromosomal position. The vertical bars represent the frequency of gains and losses across your samples. The black bars represent gains, the gray bars represent losses.

Value
This function creates a plot.

Author(s)
Mark van de Wiel and Sjoerd Vosse

References

Examples
## Not run:
data(WiltingRegions)
frequencyPlot(WiltingRegions)
## End(Not run)
**frequencyPlotCalls**  

**Visualization of aCGH profiles.**

**Description**  
This function creates a frequency plot for aCGH profiles.

**Usage**  

```r
frequencyPlotCalls(x, main='Frequency Plot', gaincol='blue', losscol='red', misscol=NA, build='GRCh37', ...)  
```

**Arguments**  

- `x` An object of class `cghCall`.
- `main` Title of plot.
- `gaincol` Color to use for gains.
- `losscol` Color to use for losses.
- `misscol` Missings.
- `build` Build of Humane Genome. Either `GRCh37`, `GRCh36`, `GRCh35` or `GRCh34`.
- `...` Arguments to `plot`.

**Details**  
We find plotted on the x-axis the array probes sorted by chromosomal position. The vertical bars represent the frequency of gains or losses.

**Value**  
This function creates a plot.

**Author(s)**  
Sjoerd Vosse & Mark van de Wiel

**References**  

**Examples**  
```r
## Not run:
data(Wilting)
rawcgh <- make_cghSeg(Wilting)
normalized <- normalize(rawcgh)
segmented <- segmentData(normalized)
called <- CGHcall(segmented, cellularity = rep(0.75, 3))
frequencyPlotCalls(called)
## End(Not run)
```
**make_cghRaw**

Convert a dataframe or textfile to an object of class cghRaw.

**Description**

This function converts a dataframe of appropriate format to an object of class cghRaw.

**Usage**

```r
make_cghRaw(input)
```

**Arguments**

- `input` Either a dataframe or character string containing a filename. See details for the format.

**Details**

The input should be either a dataframe or a tabseparated textfile (textfiles must contain a header). The first four columns should contain the name, chromosome and the start and end position in bp for each array target respectively. The chromosome and position column must contain numbers only. Following these is a column with log2 ratios for each of your samples. If the input type is a textfile, missing values should be represented as 'NA' or an empty field.

**Value**

This function returns an object of class `cghRaw-class`.

**Author(s)**

Sjoerd Vosse & Mark van de Wiel

**Examples**

```r
data(Wilting)
## Convert to \code{\link{cghRaw}} object
cgh <- make_cghRaw(Wilting)
```

**plot.cghRaw**

Plot aCGH data.

**Description**

Please see the class descriptions for more details on the plot functions.
probloss

Usage

## S3 method for class 'cghRaw'
plot(x, y, ...)

## S3 method for class 'cghSeg'
plot(x, y, ...)

## S3 method for class 'cghCall'
plot(x, y, ...)

## S3 method for class 'cghRegions'
plot(x, y, ...)

Arguments

x An object of class cghRaw, cghSeg, cghCall, or cghSeg.
y This argument is not used and should be missing.
... Arguments plot.

Author(s)

Sjoerd Vosse

See Also

cghRaw-class, cghSeg-class, cghCall-class, cghRegions-class

probloss Retrieve call probabilities from a cghCall object.

Description

These generic functions access the call probabilities from assay data stored in an object derived from the cghCall-class.

Usage

probloss(object)
probloss(object) <- value
probloss(object)
probloss(object) <- value
probnorm(object)
probnorm(object) <- value
probgain(object)
probgain(object) <- value
probamp(object)
probamp(object) <- value

Arguments

object Object derived from class cghCall
value Matrix with rows representing features and columns samples.
regions

Value

`probloss` returns matrix of call probabilities;

Author(s)

Sjoerd Vosse

See Also

`cghCall-class`

---

**Description**

This function accesses the regions values of assay data stored in an object derived from the `cghRegions-class`.

**Usage**

```r
regions(object)
regions(object) <- value
```

**Arguments**

- `object` Object derived from class `cghRegions`
- `value` Matrix with rows representing features and columns samples.

**Value**

`regions` returns a matrix of regions values;

**Author(s)**

Sjoerd Vosse

**See Also**

`cghRegions-class`
summaryPlot

Visualization of aCGH profiles.

Description
This function creates a summary plot for aCGH profiles.

Usage
summaryPlot(x, main, gaincol, losscol, misscol, build, ...)

Arguments

- **x**: An object of class `cghCall`.
- **main**: Title of plot.
- **gaincol**: Color to use for gains.
- **losscol**: Color to use for losses.
- **misscol**: Missings.
- **build**: Build of Humane Genome. Either GRCh37, GRCh36, GRCh35 or GRCh34.
- **...**: Arguments `plot`.

Details
We find plotted on the x-axis the array probes sorted by chromosomal position. The vertical bars represent the average probability that the positions they cover are gained (green bars) or lost (red bars). The green bars represent gains, the red bars represent losses.

Value
This function creates a plot.

Author(s)
Sjoerd Vosse & Mark van de Wiel

References

Examples
```r
## Not run:
data(Wilting)
rawcgh <- make_cghSeg(Wilting)
normalized <- normalize(rawcgh)
segmented <- segmentData(normalized)
called <- CGHcall(segmented, cellularity = rep(0.75, 3))
summaryPlot(called)

## End(Not run)
```
**Wilting**

*Cervical cancer arrayCGH data*

**Description**

A dataframe containing 4709 rows and 8 columns with arrayCGH data.

**Usage**

Wilting

**Format**

A dataframe containing the following 8 columns:

- **Name** The unique identifiers of array elements.
- **Chromosome** Chromosome number of each array element.
- **Position** Chromosomal position in bp of each array element.
- **AdCA10** Raw log2 ratios for cervical cancer sample AdCA10.
- **SCC27** Raw log2 ratios for cervical cancer sample SCC27.
- **SCC32** Raw log2 ratios for cervical cancer sample SCC32.
- **SCC36** Raw log2 ratios for cervical cancer sample SCC36.

**Source**


**WiltingCalled**

*Cervical cancer arrayCGH data called with CGHcall*

**Description**

Cervical cancer arrayCGH data called with CGHcall with default settings, containing 3552 features for 5 samples.

**Usage**

WiltingCalled

**Format**

An object of class cghCall
WiltingNorm

Source


WiltingNorm

Normalized log2 ratios from cervical cancer arrayCGH data.

Description
Normalized log2 ratios from cervical cancer arrayCGH data, containing 3552 features for 5 samples. These data have been normalized using the normalize function with default settings.

Usage
WiltingCalled

Format
An object of class cghRaw.

Source

WiltingRaw

Raw log2 ratios from cervical cancer arrayCGH data.

Description
Raw log2 ratios from cervical cancer arrayCGH data, containing 3552 features for 5 samples. These data have been preprocessed using preprocess.

Usage
WiltingCalled

Format
An object of class cghRaw.
WiltingSeg

Source


WiltingRegions

Regions of cervical cancer arrayCGH data as defined by CGHregions

Description

Regions of cervical cancer arrayCGH data as defined by CGHregions with default settings, containing 90 regions over 5 samples.

Usage

WiltingRegions

Format

An object of class cghRegions

Source


WiltingSeg

Segmented log2 ratios from cervical cancer arrayCGH data.

Description

Segmented log2 ratios from cervical cancer arrayCGH data, containing 3552 features for 5 samples. These data have been segmented using segmentData with default settings.

Usage

WiltingCalled

Format

An object of class cghSeg.
Increased gene copy numbers at chromosome 20q are frequent in both squamous cell carcinomas and adenocarcinomas of the cervix. *Journal of Pathology, 210*, 258-259.
Index

*Topic** classes
  cghCall, 3
cghRaw, 5
cghRegions, 7
cghSeg, 8

*Topic** datasets
  Wilting, 18
  WiltingCalled, 18
  WiltingNorm, 19
  WiltingRaw, 19
  WiltingRegions, 20
  WiltingSeg, 20

*Topic** manip
  avedist, 2
  chromosomes, 10
  copynumber, 11
  probloss, 15
  regions, 16

*Topic** misc
  frequencyPlot, 12
  frequencyPlotCalls, 13
  make_cghRaw, 14
  summaryPlot, 17

*Topic** package
  CGHbase-package, 2

AnnotatedDataFrame, 4, 6, 7, 9
AssayData, 7
avedist, 2
avedist, cghRegions-method (cghRegions), 7

bpend (chromosomes), 10
bpend, cghCall-method (cghCall), 3
bpend, cghRaw-method (cghRaw), 5
bpend, cghRegions-method (cghRegions), 7
bpend, cghSeg-method (cghSeg), 8
bpstart (chromosomes), 10
bpstart, cghCall-method (cghCall), 3
bpstart, cghRaw-method (cghRaw), 5
bpstart, cghRegions-method (cghRegions), 7
bpstart, cghSeg-method (cghSeg), 8
calls (copynumber), 11
calls, cghCall-method (cghCall), 3
calls<- (copynumber), 11
calls<-, cghCall, matrix-method (cghCall), 3
CGHbase (CGHbase-package), 2
CGHbase-package, 2
CGHcall, 3, 18
cghCall, 2, 3, 13, 15, 17, 18
cghCall-class (cghCall), 3
cghRaw, 2, 5, 15, 19
cghRaw-class (cghRaw), 5
CGHregions, 7, 20
cghRegions, 7, 12, 20
cghRegions-class (cghRegions), 7
cghSeg, 2, 8, 15, 20
cghSeg-class (cghSeg), 8
chromosomes, 10
chromosomes, cghCall-method (cghCall), 3
chromosomes, cghRaw-method (cghRaw), 5
chromosomes, cghRegions-method (cghRegions), 7
chromosomes, cghSeg-method (cghSeg), 8
copynumber, 11
copynumber, cghCall-method (cghCall), 3
copynumber, cghRaw-method (cghRaw), 5
copynumber, cghRegions-method (cghRegions), 7
copynumber<-, cghCall, matrix-method (cghCall), 3
copynumber<-, cghRaw, matrix-method (cghRaw), 5
copynumber<-, cghSeg, matrix-method (cghSeg), 8
eSet, 3–9
frequencyPlot, 12
frequencyPlot, cghRegions, missing-method (cghRegions), 7
frequencyPlotCalls, 13
frequencyPlotCalls, cghCall, missing-method (cghCall), 3
initialize, cghCall-method (cghCall), 3
initialize, cghRaw-method (cghRaw), 5
initialize, cghRegions-method (cghRegions), 7
initialize, cghSeg-method (cghSeg), 8
make_cghRaw, 14
nclone (avedist), 2
nclone, cghRegions-method (cghRegions), 7
normalize, 19
plot, cghCall, missing-method (cghCall), 3
plot, cghRaw, missing-method (cghRaw), 5
plot, cghSeg, missing-method (cghSeg), 8
plot, cghCall (plot, cghRaw), 14
plot, cghRaw, 14
plot, cghRegions (plot, cghRaw), 14
plot, cghRegions, cghRegions, missing-method (cghRegions), 7
plot, cghSeg (plot, cghRaw), 14
plot.summary, cghCall, missing-method (cghCall), 3
preprocess, 19
probamp (probloss), 15
probamp, cghCall-method (cghCall), 3
probamp<-, (probloss), 15
probamp<-, cghCall, matrix-method (cghCall), 3
probloss (probloss), 15
probloss, cghCall-method (cghCall), 3
probloss<-, (probloss), 15
probloss<-, cghCall, matrix-method (cghCall), 3
probgain (probloss), 15
probgain, cghCall-method (cghCall), 3
probgain<-, (probloss), 15
probgain<-, cghCall, matrix-method (cghCall), 3
probloss, 15
probloss, cghCall-method (cghCall), 3
probloss<-, (probloss), 15
probloss<-, cghCall, matrix-method (cghCall), 3
probnorm (probloss), 15
probnorm, cghCall-method (cghCall), 3
probnorm<-, (probloss), 15
probnorm<-, cghCall, matrix-method (cghCall), 3
regions, 16
regions, cghRegions-method (cghRegions), 7
regions<-, (regions), 16
regions<-, cghRegions, matrix-method (cghRegions), 7
segmentData, 8, 20
segmented (copynumber), 11
segmented, cghCall-method (cghCall), 3
segmented, cghSeg-method (cghSeg), 8
segmented<-, (copynumber), 11
segmented<-, cghCall, matrix-method (cghCall), 3
segmented<-, cghSeg, matrix-method (cghSeg), 8
summaryPlot, 17
Wilting, 2, 18
WiltingCalled, 2, 18
WiltingNorm, 2, 19
WiltingRaw, 2, 19
WiltingRegions, 20
WiltingSeg, 2, 20