Package ‘cghseg’

January 3, 2016

Type Package
Title Segmentation Methods for Array CGH Analysis
Version 1.0.2-1
Depends methods, parallel
Date 2012-12-06
Author Franck Picard, Mark Hoebeke, Emilie Lebarbier, Vincent Miele,
      Guillem Rigaill, Stephane Robin
Maintainer ORPHANED
Description cghseg is an R package dedicated to the analysis of CGH
      profiles using segmentation models.
License GPL (>= 2)
NeedsCompilation yes
X-CRAN-Original-Maintainer Franck Picard <franck.picard@univ-lyon1.fr>
X-CRAN-Comment Orphaned on 2013-11-12 as the maintainer repeatedly
      failed to respond to requests for valid portable C++ code.
Repository CRAN
Date/Publication 2016-01-03 10:39:51

R topics documented:

alpha<- ................................................................. 2
beta<- ............................................................... 3
calling<- ............................................................. 4
CGHdata-class ......................................................... 5
CGHoptions-class ...................................................... 6
CGHresults-class ...................................................... 7
GCnorm<- ............................................................. 9
getbqoutliers ......................................................... 9
getbp ................................................................. 10
getlevels ............................................................. 11
getmultiKmax ......................................................... 12
Description
changes slot alpha of a CGHoptions object

Arguments

.Object        an object of the CGHoptions class

Details

CGHo["alpha"] sets the proportion of length of each profile that will be used for . If \( n_i \) is the length of profile \( i \) then

\[
\text{unikmax}[i] = \alpha n_i
\]

The speed of execution of the method decreases with \( K_{\text{max}} \). Each \( K_{\text{max}} \) must fulfill some conditions wrt \( \text{CGH}_o \):

\[
\text{unikmax}[i] > \text{nlevels} \\
\text{unikmax}[i] < n_i
\]

Author(s)
F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References
Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

See Also
beta, getuniKmax
Examples

```r
beta <-
set.seed(1)
simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
CGHd = new("CGHdata",Y=simul$Y)
CGHo = new("CGHoptions")
alpha(CGHo) = 0.5
```

---

Description

```
beta<-
Replace slot beta
```

changes slot beta of a CGHoptions object

Arguments

```
. Object an object of the CGHoptions class
```

Details

```
CGHo["beta"] sets the proportion of each Kmax in multiKmax. If uniKmax[[i]] is the Kmax of profile i then

multiKmax = \beta \times \sum_i uniKmax[[i]]

MultiKmax must be greater than the number of profiles, but should not be too small since it should be greater than the selected number of segments (unknown).
```

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

See Also

alpha, getmultiKmax

Examples

```r
#set.seed(1)
#simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
#CGHd = new("CGHdata",Y=simul$Y)
#CGHo = new("CGHoptions")
#beta(CGHo) = 0.5
```
calling<-  

Replace slot calling

Description

changes slot calling of a CGHoptions object

Arguments

.Object an object of the CGHoptions class

Details

CGHo["calling"] determines if calling should be done during segmentation (logical). The number of levels is fixed and set in CGHo["nlevels"]

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

See Also

nlevels

Examples

#set.seed(1)
#simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
#CGHd = new("CGHdata",Y=simul$Y)
#CGHo = new("CGHoptions")
#calling(CGHo) = TRUE
#CGHr = multiseg(CGHd,CGHo)
The CGHdata class is built to store the dataset.

Objects from the Class

Objects can be created by calls of the form `new("CGHdata", Y)`.

- **Y**: data frame containing recorded values. One column per profile (of the same length), with patient IDs as column names. If the data contains probeID, then one column of Y must be named `probeID`. If the data contains the genomic position, then one column of Y must be named `genomic.position`. If the data contains GC content of the probe then one column of Y must be named `gccontent`.

The dataset is checked while constructing the class. If colnames are not `gccontent`, `genomic.position`, `probeID`, all columns will be considered as microarray signals.

Slots

- **Y**: Contains the data one field per patient (list)
- **probeID**: Contains the ID of each probe (factor)
- **genomic.position**: Contains the genomic position of each probe (numeric)
- **gccontent**: Contains the GC content of each probe (numeric)

Methods

- **[** Accessor to slots of the class
- **initialize** to construct the class
- **getuniKmax** Assesses the maximum number of segments per profile
- **getmultiKmax** Assesses the maximum total number of segments
- **multiseg** Joint segmentation of multiple CGH profiles
- **uniseg** Separate segmentation single CGH profiles
- **print** print class object
- **show** show class object
- **summary** summary of class object

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin
CGHoptions-class

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

Examples

```r
#set.seed(1)
simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
CGHd = new("CGHdata",Y=simul$Y)
show(CGHd)
summary(CGHd)
```

---

CGHoptions-class  
Class "CGHoptions"

Description

The CGHoptions class is built to store informations regarding the methods used in the cghseg package.

Objects from the Class

Objects can be created by calls of the form `new("CGHoptions")`.

Slots

- **select**: Model selection to select the number of segments; "mBIC","none" (character)
- **calling**: calling of segments (logical)
- **wavenorm**: Wave normalization; "none","position","spline" (character)
- **Gcnorm**: GCcontent normalization; "none","linear" (character)
- **nblevels**: Number of levels when calling (numeric)
- **alpha**: Proportion of the individual signal length that gives uniKmax.
- **beta**: Proportion of the sum of uniKmax that gives multiKmax.
- **itermax**: Maximal number of iterations
- **nbprocs**: Number of available computer processes

Prototype

```r
CGHo = new('CGHoptions')
CGHo@select = "mBIC"
CGHo@calling = FALSE
CGHo@wavenorm = "position"
CGHo@Gcnorm = "none"
CGHo@nblevels = 3
CGHo@alpha = 0.2
CGHo@beta = 0.5
CGHo@itermax = Inf
CGHo@nbprocs = 1
```
Methods

[ Accessor to slots of the class
initialize to construct the class
print Print CGHo
show Show CGHo
select<- Modification of slot select; select(CGHo)<-
calling<- Modification of slot calling; calling(CGHo)<-
wavenorm<- Modification of slot wavenorm; wavenorm(CGHo)<-
GCnorm<- Modification of slot GCnorm; GCnorm(CGHo)<-
nblevels<- Modification of slot nblevels; nblevels(CGHo)<-
alpha<- Modification of slot alpha; alpha(CGHo)<-
beta<- Modification of slot beta; beta(CGHo)<-
itermax<- Modification of slot itermax; itermax(CGHo)<-
nbprocs<- Modification of slot nbproc; nbproc(CGHo)<-

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

Examples

#CGHo = new("CGHoptions")
#showClass("CGHoptions")
Slots

- **mu**: list of dataframes containing the position of breakpoints and estimated means (list)
- **theta**: List of effects used for normalization: waveffect, GCeffect (listOrNULL)
- **loglik**: Log-likelihood of the selected model (numeric)
- **nbiter**: number of iterations (numeric)
- **from**: Name of the function that produced the results (character)
- **probeID**: Contains the ID of each probe (factor)
- **genomic.position**: Contains the genomic position of each probe (numeric)
- **options**: list of options (of class CGHoptions)

Methods

- **initialize** to construct the class
- **summary** summary of CGHr
- **getbp** Extract the frequency of breakpoints
- **getsegprofiles** Extract model predictions
- **getlevels** Extract levels of segments
- **getbgoutliers** Identification of outliers in the background intensity

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

Examples

```r
#set.seed(1)
#simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
#CGHd = new("CGHdata",Y=simul$Y)
#CGHo = new("CGHoptions")
#CGHr = multiseg(CGHd,CGHo)
#bplist = getbp(CGHr)
#segprofiles = getsegprofiles(CGHr)

#calling(CGHo) = TRUE
#CGHr = multiseg(CGHd,CGHo)
#bplist = getbp(CGHr)
#levelslist = getlevels(CGHr)
#segprofiles = getsegprofiles(CGHr)
```
Description

changes slot GCnorm of a CGHoptions object

Arguments

Object an object of the CGHoptions class

Details

cgho["GCnorm"] determines if the bias due to GC content should be removed from the data:
"linear" for linear regression (degree 2) "none" for no correction

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

getbgoutliers

Description

Identification of outliers in the background intensity

Usage

getbgoutliers(.Object, CGH.smooth, fdr)

Arguments

.Object an object of the CGHresults class for which CGH["wavenorm"]="position"

CGH.smooth an object of the CGHresults class for which CGH["wavenorm"] is "spline"

fdr fdr level to control for multiple testing

Value

bgoutliers dataframe containing background outliers
getbp

Breakpoints frequencies

Description
Gives the frequency of breakpoints

Usage
getbp(.Object)

Arguments

.Object object of class CGHresults

Value

bplist list of dataframes containing 0/1 for the position of the breaks

Author(s)
F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References
Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)
getlevels

Examples

getlevels

#set.seed(1)
#simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
#CGHd = new("CGHdata",Y=simul$Y)
#CGHo = new("CGHoptions")
#CGHr = multiseg(CGHd,CGHo)
#bplist = getbp(CGHr)

getlevels

<table>
<thead>
<tr>
<th>Levels frequency</th>
</tr>
</thead>
</table>

Description

Gives the calls status at every position

Usage

getlevels(.Object)

Arguments

.Object an object of the CGHresults class

Value

levelslist list of dataframes containing calls for each position

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

Examples

#set.seed(1)
#simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
#CGHd = new("CGHdata",Y=simul$Y)
#CGHo = new("CGHoptions")
#calling(CGHo) = TRUE
#CGHr = multiseg(CGHd,CGHo)
#levelslist = getlevels(CGHr)
Determining the maximum number of segments for joint segmentation

Description

Determining the total maximum number of segments according to the maximum number of segments for each profile

Usage

getmultiKmax(.Object, CGHo, uniKmax=NULL, multiKmax=NULL)

Arguments

/Object/ an object of the CGHdata class  
CGHo an object of the CGHoptions class  
uniKmax NULL if no value is proposed, list of Kmax for each profile  
multiKmax NULL or a proposed value for multiKmax, to check validity

Details

CGHo[“beta”] sets the proportion of each Kmax in multiKmax. If uniKmax[i] is the Kmax of profile i then

multiKmax = \beta \times \sum_{i} uniKmax[i]

MultiKmax must be greater than the number of profiles, but should not be too small since it should be greater than the selected number of segments (unknown).

Value

multiKmax total number of segments for joint segmentation

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)
getsegprofiles

Examples

# set.seed(1)
# simul = simulprofiles(M=5, n=100, k.mean=2, SNR=5, lambda=1)
# CGHd = new("CGHdata", Y=simul$Y)
# CGHo = new("CGHoptions")

## Default initialization

# multikmax = getmultikmax(CGHd, CGHo)

## Modification of all fields:
## to set multikmax to 50% of the sum of Kmax of each profile:

# beta(CGHo) = 0.5
# multikmax = getmultikmax(CGHd, CGHo)

## Modification of uniKmax

# uniKmax = getuniKmax(CGHd, CGHo)
# uniKmax[[1]] = 80
# multikmax = getmultikmax(CGHd, CGHo, uniKmax)

## Modification of uniKmax and multikmax
## and check that the value is correct wrt options

# uniKmax[[1]] = 80
# multikmax = getmultikmax(CGHd, CGHo, uniKmax, multikmax = 10)

gosegprofiles

Description

Extract segmented profiles from a CGHresults object

Usage

gosegprofiles(.Object)

Arguments

.Object an object of the CGHresults class

Value

segprofiles list of segmented profiles

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin
getuniKmax

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

Examples

```r
#set.seed(1)
#simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
#CGHd = new("CGHdata",Y=simul$Y)
#CGHo = new("CGHoptions")
#calling(CGHo) = FALSE
#CGHr = multiseg(CGHd,CGHo)
#segprofiles = getsegprofiles(CGHr)

#calling(CGHo) = TRUE
#CGHr = multiseg(CGHd,CGHo)
#segprofiles = getsegprofiles(CGHr)
```

getuniKmax  
*Maximum number of segments per profile*

Description

Maximum number of segments per profile according to a given proportion of the length of the data using a univariate pre-segmentation.

Usage

```r
getuniKmax(.Object,CGHo,uniKmax=NULL)
```

Arguments

- `.Object`  
an object of the CGHdata class
- `CGHo`  
an object of the CGHoption class
- `uniKmax`  
NULL if no value is proposed, list of Kmax for each profile

Details

`CGHo["alpha"]` sets the proportion of length of each profile that will be used for pre-segmentation. If `ni` is the length of profile `i` then the pre-segmentation is done using a `Kmax=max(200,alpha n_i)`. Pre-segmentation is done to propose a number of segments `Khi` for each profile `i`. Then the maximum number of segments authorized per profile will be `2*Khi`. The speed of execution of the method decreases with `Kmax`. Each `Kmax` must fulfill some conditions wrt `CGHo`:

\[
uniKmax[i] > nlevels
\]

\[
uniKmax[i] < n_i
\]

Tuning `Khi` for each profile is a way to reduce the complexity of the multisample segmentation.
itermax <-

Value

uniKmax list containing the maximum number of segments per profile. If !is.null(uniKmax) in the entry of the method, the method checks its validity.

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

Examples

```r
#set.seed(1)
simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
CGHd = new("CGHdata",Y=simul$Y)
CGHo = new("CGHoptions")

## Default initialization

#uniKmax = getuniKmax(CGHd,CGHo)

## Modification of all fields:
## to tune uniKmax using to 50% of the length of each profile :
## this may provide the same result since a model selection is performed
## for tuning.
#alpha(CGHo) = 0.5
#uniKmax = getuniKmax(CGHd,CGHo)

## Modification of one field for one profile
## and check that the value is correct wrt options

#uniKmax[[1]] = 80
#uniKmax = getuniKmax(CGHd,CGHo,uniKmax)
```

Description

changes slot itermax of a CGHoptions object

Arguments

.Object an object of the CGHoptions class

Details

CGHo["itermax"] sets the number of iterations when using wave correction
multiseg

Author(s)
F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References
Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

Examples

```r
set.seed(1)
simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
CGHd = new("CGHdata",Y=simul$Y)
CGHo = new("CGHoptions")
itermax(CGHo) = 10
```

Description
Joint segmentation of multiple CGH profiles

Usage

```r
multiseg(.Object,CGHo,uniKmax=NULL,multiKmax=NULL)
```

Arguments

- `.Object` an object of the CGHdata class
- `CGHo` an object of the CGHoption class
- `uniKmax` list containing the maximum number of segments per profile. If !is.null(uniKmax) in the entry of the method, the method checks its validity.
- `multiKmax` total number of segments for joint segmentation

Value

- `CGHr` object of class CGHresult

Author(s)
F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References
Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)
Examples

```r
#set.seed(1)
#simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
#CGHd = new("CGHdata",Y=simul$Y)
#CGHo = new("CGHoptions")
#CGHr = multiseg(CGHd,CGHo)
```

Description

changes slot `nblevels` of a CGHoptions object

Arguments

- `.Object` an object of the CGHoptions class

Details

`CGHo["nblevels"]` determines the total number of levels when calling is performed.

Author(s)

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

Description

changes slot `nbprocs` of a CGHoptions object

Arguments

- `.Object` an object of the CGHoptions class

Details

`CGHo["nbprocs"]` sets the number of computer processes to use for parallel computations.
select<- 

Author(s) 
F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin 

References 
Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011) 

Examples 

cgho = new("CGHoptions")
nbprocs(cgho) = 2 # if 2 processes are available

select<- 
Replace slot select 

Description 
changes slot select of a CGHoptions object 

Arguments 
.Object an object of the CGHoptions class 

Details 

cgho["select"] sets the model selection criterion to be used in the procedure. The default criterion is based on the mBIC proposed by Zhang & Siegmund (2007). If cgho["select"]="none" then the algorithm is run for the maximum number of segments. 

Author(s) 
F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin 

References 
Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011) 

See Also 
alpha, beta, getmultiKmax, getuniKmax
Examples

```r
# set.seed(1)
# simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
# CGHd = new("CGHdata",Y=simul$Y)
# CGHo = new("CGHoptions")
# select(CGHo) = "mBIC"
# CGHr = multiseg(CGHd,CGHo)

# select(CGHo) = "none"
# CGHr = multiseg(CGHd,CGHo)
```

---

**Description**

Simulates multiple CGH profiles

**Usage**

`simulprofiles(M, n,k.mean,SNR,lambda)`

**Arguments**

- `M` Number of profiles
- `n` length of profiles
- `k.mean` average number of segments per profile
- `SNR` Signal to Noise Ratio between the mean and the measurement noise. The mean of segments can take 3 values -1;0;0.58.
- `lambda` Variance ratio between measurement noise and background noise

**Value**

- `simul` list containing Y (dataframe of simulated CGH signals), Ki (the true number of segments per profile), mu0 (the true signal), theta0 (the true background intensity)

**Author(s)**

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

**References**

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)
uniseg

Segmentation of univariate CGH profiles

Usage
uniseg(.Object, CGHo, uniKmax=NULL)

Arguments

.Object  an object of the CGHdata class
CGHo     an object of the CGHoption class
uniKmax  list containing the maximum number of segments per profile. If !is.null(uniKmax)
in the entry of the method, the method checks its validity.

Value
cghr     object of class CGHresult

Author(s)
F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

References

Examples

#set.seed(1)
#simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
#CGHd = new("CGHdata",Y=simul$Y)
#CGHo = new("CGHoptions")
#CGHr = uniseg(CGHd, CGHo)
Replace slot \texttt{wavenorm}

\textbf{Description}

changes slot \texttt{wavenorm} of a \texttt{CGHoptions} object

\textbf{Arguments}

\begin{itemize}
  \item \texttt{object} \quad an object of the \texttt{CGHoptions} class
\end{itemize}

\textbf{Details}

\texttt{cgho["wavenorm"]} determines the method to remove the wave-effect: none, position, spline.

\textbf{Author(s)}

F. Picard, E. Lebarbier, M. Hoebeke, G. Rigaill, B. Thiam, S. Robin

\textbf{References}

Joint segmentation, calling and normalization of multiple CGH profiles, Biostatistics (2011)

\textbf{Examples}

\begin{verbatim}
#set.seed(1)
#simul = simulprofiles(M=5,n=100,k.mean=2,SNR=5,lambda=1)
#CGHd = new("CGHdata",Y=simul$Y)
#CGHo = new("CGHoptions")
#wavenorm(CGHo) = "spline"
#CGHr = multiseg(CGHd,CGHo)
\end{verbatim}
Index

+Topic classes
  CGHdata-class, 5
  CGHOptions-class, 6
  CGHresults-class, 7

+Topic methods
  alpha<-, 2
  beta<-, 3
  calling<-, 4
  GCnrm<-, 9
  getbgoutliers, 9
  getbp, 10
  getlevels, 11
  getmultiKmax, 12
  getsegprofiles, 13
  getuniKmax, 14
  itermax<-, 15
  nblevels<-, 17
  nbprocs<-, 17
  select<-, 18
  uniseg, 20
  wavenorm<-, 21

+Topic ts
  alpha<-, 2
  beta<-, 3
  calling<-, 4
  GCnrm<-, 9
  getbp, 10
  getmultiKmax, 12
  getuniKmax, 14
  itermax<-, 15
  nblevels<-, 17
  nbprocs<-, 17
  select<-, 18
  wavenorm<-, 21

[], CGHdata-method (CGHdata-class), 5
[], CGHOptions-method (CGHOptions-class), 6
[], CGHresults-method (CGHresults-class), 7

alpha<-, 2
alpha<-, CGHOptions-method
  (CGHOptions-class), 6
alpha<--methods (alpha<--), 2

beta<-, 3
beta<-, CGHOptions-method
  (CGHOptions-class), 6
beta<--methods (beta<--), 3

calling<-, 4
calling<-, CGHOptions-method
  (CGHOptions-class), 6
calling<--methods (calling<--), 4
CGHdata-class, 5
CGHOptions-class, 6
CGHresults-class, 7

Gcnorm<-, 9
Gcnorm<-, CGHOptions-method
  (CGHOptions-class), 6
Gcnorm<--methods (Gcnorm<--), 9
getbgoutliers, 9
getbgoutliers, CGHresults-method
  (CGHresults-class), 7
getbgoutliers-methods (getbgoutliers), 9
getbp, 10
getbp, CGHResults-method
  (CGHResults-class), 7
getbp-methods (getbp), 10
getlevels, 11
getlevels, CGHresults-method
  (CGHresults-class), 7
getlevels-methods (getlevels), 11
getmultiKmax, 12
getmultiKmax, CGHdata-method
  (CGHdata-class), 5
gtmultiKmax-methods (getmultiKmax), 12
getprobebias, CGHresults-method (CGHresults-class), 7
getsegprofiles, 13
getsegprofiles, CGHresults-method (CGHresults-class), 7
getsegprofiles-methods (getsegprofiles), 13
getunikmax, 14
getunikmax, CGHdata-method (CGHdata-class), 5
getunikmax-methods (getunikmax), 14
initialize, CGHdata-method (CGHdata-class), 5
initialize, CGHresults-method (CGHresults-class), 7
itermax<-, 15
itermax<-, CGHOptions-method (CGHOptions-class), 6
itermax<--methods (itermax<-), 15
multiseg, 16
multiseg, CGHdata-method (CGHdata-class), 5
multiseg-methods (multiseg), 16
nlevels<-, 17
nlevels<-, CGHOptions-method (CGHOptions-class), 6
nlevels<--methods (nlevels<--), 17
nbprocs<-, 17
nbprocs<-, CGHOptions-method (CGHOptions-class), 6
nbprocs<--methods (nbprocs<--), 17
print, CGHdata-method (CGHdata-class), 5
print, CGHOptions-method (CGHOptions-class), 6
select<-, 18
select<-, CGHOptions-method (CGHOptions-class), 6
select<--methods (select<--), 18
show, CGHdata-method (CGHdata-class), 5
show, CGHOptions-method (CGHOptions-class), 6
simulprofiles, 19
summary, CGHdata-method (CGHdata-class), 5
summary, CGHresults-method (CGHresults-class), 7
uniseg, 20
uniseg, CGHdata-method (CGHdata-class), 5
uniseg-methods (uniseg), 20
wavenorm<-, 21
wavenorm<-, CGHOptions-method (CGHOptions-class), 6
wavenorm<--methods (wavenorm<--), 21