CGHcall

April 19, 2010

`CGHcall-package` *Calling aberrations for array CGH tumor profiles.*

Description

Calls aberrations for array CGH data using a six state mixture model as well as several biological concepts that are ignored by existing algorithms. Visualization of profiles is also provided.

Details

Package:	CGHcall
Type:	Package
Version:	2.0.0
Date:	2007-03-14
License:	GPL

Author(s)

Sjoerd Vosse and Mark van de Wiel

Maintainer: Sjoerd Vosse <s.vosse@vumc.nl>

References

Mark A. van de Wiel, Kyung In Kim, Sjoerd J. Vosse, Wessel N. van Wieringen, Saskia M. Wilting and Bauke Ylstra. CGHcall: calling aberrations for array CGH tumor profiles. *Bioinformatics, 23*, 892-894.

`CGHcall` *Calling aberrations for array CGH tumor profiles.*

Description

Calls aberrations for array CGH data using a six state mixture model.

Usage

```
CGHcall(inputSegmented, prior = "auto", nclass = 3, organism = "human", robustsi
```

Arguments

inputSegmented	An object of class cghSeg
prior	Options are all, not all, or auto. See details for more information.
nclass	The number of levels to be used for calling. Either 3 (loss, normal, gain) or 4 (including amplifications).
organism	Either human or other. This is only used for chromosome arm information when prior is set to all or auto (and samplesize > 20).
robustsig	Options are yes or no. yes enforces a lower bound on the standard deviation of the normal segments
digits	Number of decimal digits to be saved in the resulting call object. Allows for saving storage space
nsegfit	Maximum number of segments used for fitting the mixture model. Posterior probabilities are computed for all segments
maxnumseg	Maximum number of segments per profile used for fitting the model
minlsforfit	Minimum length of the segment (in Mb) to be used for fitting the model

Details

Please read the article and the supplementary information for detailed information on the algorithm. The parameter prior states how the data is used to determine the prior probabilities. When set to all, the probabilities are determined using the entire genome of each sample. When set to not all probabilites are determined per chromosome for each sample when organism is set to other or per chromosome arm when organism is human. The chromosome arm information is taken from the March 2006 version of the UCSC database. When prior is set to auto, the way probabilities are determined depends on the sample size. The entire genome is used when the sample size is smaller than 20, otherwise chromosome (arm) information is used. Please note that CGHcall uses information from all input data to determine the aberration probabilities. When for example triploid or tetraploid tumors are observed, we advise to run CGHcall separately on those (groups of) samples. Note that robustsig = yes enforces the sd corresponding to the normal segments to be at least half times the pooled gain/loss sd. Use of "nsegfit" significantly lower computing time with respect to previous CGHcall versions without much accuracy loss. Moreover, "maxnumseg" decreases the impact on the results of profiles with inferior segmentation results. Finally, "minlsforfit" decreases the impact of very small aberations (potentially CNVs rather than CNAs) on the fit of the model. Note that always a result for all segments is produced.

Value

This function return a list with three components:

probabilities	A dataframe with 3 columns of probe information (name, chromosome and position), followed by k columns with aberration probabilities for each sample, where k is the number of levels used for calling (nclass).
calls	A dataframe with the calls for each sample. Values are −1 (loss), 0 (normal) or 1 (gain). If 4 levels were used for calling, a value of 2 represents an amplification.

segments — A matrix with the segments for each profile. The first column contains the sample number. The second column the level of the current segment and the third and fourth columns the start and end of the segment in probe number respectively.

Author(s)

Sjoerd Vosse & Mark van de Wiel

References

Mark A. van de Wiel, Kyung In Kim, Sjoerd J. Vosse, Wessel N. van Wieringen, Saskia M. Wilting and Bauke Ylstra. CGHcall: calling aberrations for array CGH tumor profiles. *Bioinformatics, 23*, 892-894.

Examples

```
data(Wilting)
## Convert to cghRaw object
cgh <- make_cghRaw(Wilting)
print(cgh)
## First preprocess the data
raw.data <- preprocess(cgh)
## Simple global median normalization for samples with 75% tumor cells
perc.tumor <- rep(0.75, 3)
normalized.data <- normalize(raw.data, cellularity=perc.tumor)
## Segmentation with slightly relaxed significance level to accept change-points.
## Note that segmentation can take a long time.
## Not run: segmented.data <- segmentData(normalized.data, alpha=0.02)
## Not run: postsegnormalized.data <- postsegnormalize(segmented.data)
## Call aberrations
## Not run: result <- CGHcall(postsegnormalized.data)
```

---

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

---

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

```
new('cghCall', phenoData = [AnnotatedDataFrame], experimentData = [MIAME],
annotation = [character], copynumber = [matrix], segmented = [matrix],
calls = [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 represening 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. 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(-5,5): limits of the y-axis

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

See eSet for derived methods.

Author(s)

Sjoerd Vosse

See Also

eSet-class, cghRaw-class, cghSeg-class

Examples

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

# create an instance of cghCall through CGHcall
## Not run:
   data(Wilting)
   rawcgh <- make_cghSeg(Wilting)
   normalized <- normalize(rawcgh)
   segmented <- segmentData(normalized)
   called <- CGHcall(segmented)

   # plot the first sample. Default only every 10th log2-ratio is plotted (dotres=10). Ad
   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 = [AnnotatedDataFra
...)
```

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 represening 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)
```

---

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

```
new('cghSeg', phenoData = [AnnotatedDataFrame], experimentData = [MIAME],
annotation = [character], copynumber = [matrix], segmented = [matrix],
featureData = [AnnotatedDataFrame], ...)
```

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 nrow(phenoData). assayData must contain matrices copynumber and segmented with rows represening 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(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 `eSet` for derived methods.

Author(s)

Sjoerd Vosse

See Also

`eSet-class`, `cghRaw-class`, `cghCall-class`

Examples

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

# create an instance of cghSeg through '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). Ad
   plot(segmented[,1])
   # first three chromosomes
   plot(segmented[chromosomes(segmented)<=3,1])

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

   # get the names of the samples
   sampleNames(segmented)

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

## End(Not run)
```

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

```
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

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

Description

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

Usage

```
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

```
data(Wilting)
## Convert to 'cghRaw' object
cgh <- make_cghRaw(Wilting)
```

---

normalize *Normalization and cellularity adjustment for arrayCGH data.*

Description

This function normalizes arrayCGH data using the global mode or median. It can also adjust for the cellularity of your data.

Usage

```
normalize(input, method = "median", cellularity = 1, smoothOutliers = TRUE, ...)
```

Arguments

`input`	Object of class `cghRaw`.
`method`	Normalization method, either 'median', 'mode', or 'none'.
`cellularity`	A vector of cellularities ranging from 0 to 1 to define the contamination of your sample with healthy cells (1 = no contamination). See details for more information.
`smoothOutliers`	Logical. Indicates whether outliers should be smoothed using the `smooth.CNA` function.
`...`	Arguments for `smooth.CNA`.

Details

The cellularity parameter should be a vector of length n where n is the number of samples in your dataset. The vector is recycled if there are not enough values in it, or truncated if there are too many. For more information on the correction we refer to section 1.6 of the supplementary information for van de Wiel et al. 2006.

Value

This function returns a dataframe in the same format as the input with normalized and/or cellularity adjusted log2 ratios.

Author(s)

Sjoerd Vosse & Mark van de Wiel

Examples

```
data(WiltingData)
## Convert to 'cghRaw' object
cgh <- cghRaw(WiltingData)
## First preprocess the data
raw.data <- preprocess(cgh)
## Simple global median normalization for samples with 75% tumor cells
perc.tumor <- rep(0.75, 3)
normalized.data <- normalize(raw.data, cellularity=perc.tumor)
```

---

plot.summary *Visualization of aCGH profiles.*

Description

This function creates a summary plot for aCGH profiles.

Usage

```
plot.summary(x, y, ...)
```

Arguments

x	An object of class cghCall.
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 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. When 4 levels have been used for calling, amplifications are indicated with a blue tickmark at the top of the plot.

Value

This function creates a plot.

Author(s)

Sjoerd Vosse & Mark van de Wiel

References

Mark A. van de Wiel, Kyung In Kim, Sjoerd J. Vosse, Wessel N. van Wieringen, Saskia M. Wilting and Bauke Ylstra. CGHcall: calling aberrations for array CGH tumor profiles. *Bioinformatics, 23*, 892-894.

Examples

```
## Not run:
  data(Wilting)
  rawcgh <- make_cghSeg(Wilting)
  normalized <- normalize(rawcgh)
  segmented <- segmentData(normalized)
  called <- CGHcall(segmented)
  plot.summary(called)

## End(Not run)
```

---

postsegnormalize *Post-segmentation normalization*

Description

This function normalizes arrayCGH data after segmentation in order to find a better 0-level.

Usage

```
postsegnormalize(segmentData, inter=c(-0.1,0.1))
```

Arguments

segmentData	Object of class cghSeg.
inter	Interval in which the function should search for the normal level.

Details

This function recursively searches for the interval containing the most segmented data, decreasing the interval length in each recursion.

The recursive search makes the post-segmentation normalization robust against local maxima. This function is particularly useful for profiles for which, after segmentation, the 0-level does not coincide with many segments. It is more or less harmless to other profiles. We advise to keep the search interval (inter) small, in particular at the positive (gain) side to avoid that the 0-level is set to a common gain level.

Value

This function returns a cghSeg object in the same format as the input with post-segmentation-normalized adjusted log2 ratios and segmented values.

Author(s)

Mark van de Wiel

Examples

```
data(Wilting)
## Convert to cghRaw object
cgh <- make_cghRaw(Wilting)
## First preprocess the data
raw.data <- preprocess(cgh)
## Simple global median normalization for samples with 75% tumor cells
perc.tumor <- rep(0.75, 3)
normalized.data <- normalize(raw.data, cellularity=perc.tumor)
## Segmentation with slightly relaxed significance level to accept change-points.
## Note that segmentation can take a long time.
## Not run: segmented.data <- segmentData(normalized.data, alpha=0.02)
## Not run: postsegnormalized.data <- postsegnormalize(segmented.data, inter=c(-0.1,0.1
```

---

preprocess *Preprocess arrayCGH data*

Description

This function preprocesses your aCGH data so it can be processed by other functions without errors.

Usage

```
preprocess(input, maxmiss = 30, nchrom = 23, ...)
```

Arguments

input	Object of class cghRaw.
maxmiss	Maximum percentage of missing values per row.
nchrom	Number of chromosomes.
...	Arguments for impute.knn from the impute package.

Details

This function performs the following actions on arrayCGH data:

- Filter out data with missing position information.
- Remove data on chromosomes larger than nchrom.
- Remove rows with more than maxmiss percentage missing values.
- Imputes missing values using the impute.knn function from the impute package.

Value

This function returns a dataframe in the same format as the input with missing values imputed.

Author(s)

Sjoerd Vosse & Mark van de Wiel

References

Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B. Altman (2001). Missing value estimation methods for DNA microarrays. *Bioinformatics, 17*, 520-525.

Examples

```
data(WiltingRaw)
preprocessed <- preprocess(WiltingRaw, nchrom = 22)
```

`probloss` *Retrieve call probabilities from a cghCall object.*

Description

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

Usage

```
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.

Value

`probloss` returns matrix of call probabilities;

Author(s)

Sjoerd Vosse

See Also

`cghCall-class`

segmentData *Breakpoint detection for arrayCGH data.*

Description

A wrapper function to run existing breakpoint detection algorithms on arrayCGH data. Currently only DNAcopy is implemented.

Usage

```
segmentData(input, method = "DNAcopy", ...)
```

Arguments

input	Object of class cghRaw.
method	The method to be used for breakpoint detection. Currently only 'DNAcopy' is supported, which will run the segment function.
...	Arguments for segment.

Details

See segment for details on the algorithm.

Value

This function returns a dataframe in the same format as the input with segmented arrayCGH data.

Author(s)

Sjoerd Vosse & Mark van de Wiel

References

Venkatraman, A.S., Olshen, A.B. (2007). A faster circulary binary segmentation algorithm for the analysis of array CGH data. *Bioinformatics, 23*, 657-663.

Examples

```
data(WiltingNorm)
## Not run: segmented.data <- segmentData(WiltingNorm, alpha=0.02)
```

`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.

SCC39 Raw log2 ratios for cervical cancer sample SCC39.

Source

Wilting, S.M., Snijders, P.J., Meijer, G.A., Ylstra, B., van den IJssel, P.R., Snijders, A.M., Albertson, D.G., Coffa, J., Schouten, J.P., van de Wiel, M.A., Meijer, C.J., & Steenbergen, R.D. (2006). 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