Package 'mBPCR'

April 9, 2025

Version 1.61.0

Date 2020-01-11

Title Bayesian Piecewise Constant Regression for DNA copy number estimation

Author P.M.V. Rancoita <rancoita.paola@gmail.com>, with contributions from M. Hutter <marcus.hutter@anu.edu.au>

Maintainer P.M.V. Rancoita <rancoita.paola@gmail.com>

Description It contains functions for estimating the DNA copy number profile using mBPCR with the aim of detecting regions with copy number changes.

Depends oligoClasses, GWASTools

Imports Biobase, graphics, methods, utils, grDevices

Suggests xtable

License GPL (>= 2)

URL http://www.idsia.ch/~paola/mBPCR

LazyData yes

biocViews aCGH, SNP, Microarray, CopyNumberVariation

git_url https://git.bioconductor.org/packages/mBPCR

git_branch devel

git_last_commit 4db7f5f

git_last_commit_date 2024-10-29

Repository Bioconductor 3.21

Date/Publication 2025-04-09

Contents

centromere 2
computeMBPCR 3
estGlobParam 5

estProfileWithMBPCR 6
estProfileWithMBPCRforOligoSnpSet 8
importCNData 10
jekoChr11Array250Knsp 11
logAdd 12
mBPCR-internal 12
plotEstProfile 13
rec10k 14
writeEstProfile 15

Index **17**

`centromere` *Retrieve base positions of centromeres*

Description

Function to retrieve base positions of the centromere of a specific chromosome.

Usage

```
centromere(chr, hg='hg18')
```

Arguments

`chr`	chromosome of which you want to retrieve the centromere base positions.
`hg`	genome build used for retrieving the centromere base positions of the selected chromosome. Current available options are: `'hg18'`, `'hg19'` and `'hg38'`.

Value

The function returns the start and end base positions of the centromere of the selected chromosome, by using the specified genome build. The function is based on the annotation provided in the package `GWASTools`.

Examples

```
#centromere base positions of chromosome 1 in genome build hg18
centromere(1, hg='hg18')
```

computeMBPCR *Estimate the copy number profile*

Description

Function to estimate the copy number profile with a piecewise constant function using mBPCR. Eventually, it is possible to estimate the profile with a smoothing curve using either the Bayesian Regression Curve with K_2 (BRC with K_2) or the Bayesian Regression Curve Averaging over k (BRCAk). It is also possible to choose the estimator of the variance of the levels `rhoSquare` (i.e. either $\hat{\rho}_1^2$ or $\hat{\rho}^2$) and by default $\hat{\rho}_1^2$ is used.

Usage

```
computeMBPCR(y, kMax=50, nu=NULL, rhoSquare=NULL, sigmaSquare=NULL, typeEstRho=1,
             regr=NULL)
```

Arguments

`y`	array containing the log2ratio of the copy number data
`kMax`	maximum number of segments
`nu`	mean of the segment levels. If `nu=NULL`, then the algorithm estimates it on the sample.
`rhoSquare`	variance of the segment levels. If `rhoSquare=NULL`, then the algorithm estimates it on the sample.
`sigmaSquare`	variance of the noise. If `sigmaSquare=NULL`, then the algorithm estimates it on the sample.
`typeEstRho`	choice of the estimator of `rhoSquare`. If `typeEstRho=1`, then the algorithm estimates `rhoSquare` with $\hat{\rho}_1^2$, while if `typeEstRho=0`, it estimates `rhoSquare` with $\hat{\rho}^2$.
`regr`	choice of the computation of the regression curve. If `regr=NULL`, then the regression curve is not computed, if `regr="BRC"` the Bayesian Regression Curve with K_2 is computed (BRC with K_2), if `regr="BRCAk"` the Bayesian Regression Curve Averaging over k is computed (BRCAk).

Details

By default, the function estimates the copy number profile with mBPCR and estimating rhoSquare on the sample, using $\hat{\rho}_1^2$. It is also possible to use $\hat{\rho}^2$ as estimator of `rhoSquare`, by setting `typeEstRho=0`, or to directly set the value of the parameter.

The function gives also the possibility to estimate the profile with a Bayesian regression curve: if `regr="BRC"` the Bayesian Regression Curve with K_2 is computed (BRC with K_2), if `regr="BRCAk"` the Bayesian Regression Curve Averaging over k is computed (BRCAk).

Value

A list containing:

estK	the estimated number of segments
estBoundaries	the estimated boundaries
estPC	the estimated profile with mBPCR
regrCurve	the estimated bayesian regression curve. It is returned only if regr!=NULL.
nu	
rhoSquare	
sigmaSquare	
postProbT	for each probe, the posterior probablity to be a breakpoint

References

Rancoita, P. M. V., Hutter, M., Bertoni, F., Kwee, I. (2009). Bayesian DNA copy number analysis. *BMC Bioinformatics* 10: 10. http://www.idsia.ch/~paola/mBPCR

See Also

estProfileWithMBPCR, plotEstProfile, writeEstProfile, estGlobParam

Examples

```
##import the 250K NSP data of chromosome 11 of cell line JEKO-1
data(jekoChr11Array250Knsp)


##first example
## we select a part of chromosome 11
y <- jekoChr11Array250Knsp$log2ratio[6400:6900]
p <- jekoChr11Array250Knsp$PhysicalPosition[6400:6900]
##we estimate the profile using the global parameters estimated on the whole genome
##the profile is estimated with mBPCR and with the Bayesian Regression Curve
results <- computeMBPCR(y, nu=-3.012772e-10, rhoSquare=0.0479, sigmaSquare=0.0699, regr="BRC")
plot(p, y)
points(p, results$estPC, type='l', col='red')
points(p, results$regrCurve,type='l', col='green')

###second example
### we select a part of chromosome 11
#y <- jekoChr11Array250Knsp$log2ratio[10600:11600]
#p <- jekoChr11Array250Knsp$PhysicalPosition[10600:11600]
###we estimate the profile using the global parameters estimated on the whole genome
###the profile is estimated with mBPCR and with the Bayesian Regression Curve Ak
#results <- computeMBPCR(y, nu=-3.012772e-10, rhoSquare=0.0479, sigmaSquare=0.0699, regr="BRCAk")
#plot(p,y)
#points(p, results$estPC, type='l', col='red')
#points(p, results$regrCurve, type='l', col='green')
```

`estGlobParam` *Estimate global parameters of copy number data*

Description

Function to estimate the global parameters of copy number data: the mean and the variance of the segment levels (called `nu` and `rhoSquare`, respectively), the variance of the noise (`sigmaSquare`). It is possible to choose the estimator of `rhoSquare` (i.e. either $\hat{\rho}_1^2$ or $\hat{\rho}^2$) and by default $\hat{\rho}_1^2$ is used.

Usage

```
estGlobParam(y, nu=NULL, rhoSquare=NULL, sigmaSquare=NULL, typeEstRho=1)
```

Arguments

`y`	array containing the log2ratio of the copy number data
`nu`	mean of the segment levels. If `nu=NULL`, then the algorithm estimates it on the sample.
`rhoSquare`	variance of the segment levels. If `rhoSquare=NULL`, then the algorithm estimates it on the sample.
`sigmaSquare`	variance of the noise. If `sigmaSquare=NULL`, then the algorithm estimates it on the sample.
`typeEstRho`	choice of the estimator of `rhoSquare`. If `typeEstRho=1`, then the algorithm estimates `rhoSquare` with $\hat{\rho}_1^2$, while if `typeEstRho=0`, it estimates `rhoSquare` with $\hat{\rho}^2$.

Value

A list containing:

`nu`

`rhoSquare`

`sigmaSquare`

References

Rancoita, P. M. V., Hutter, M., Bertoni, F., Kwee, I. (2009). Bayesian DNA copy number analysis. *BMC Bioinformatics* 10: 10. http://www.idsia.ch/~paola/mBPCR

Examples

```
##import the 10K data of cell line REC
data(rec10k)
##estimation of all the global parameters (the variance of the segment is
##estimated with \eqn{\hat{\rho}^2_1})
estGlobParam(rec10k$log2ratio)
```

`estProfileWithMBPCR` *Estimate and print the copy number profile of some chromosomes of a sample*

Description

Function to estimate the copy number profile with a piecewise constant function using mBPCR. Eventually, it is possible to estimate the profile with a smoothing curve, using either the Bayesian Regression Curve with K_2 (BRC with K_2) or the Bayesian Regression Curve Averaging over k (BRCAk). It is also possible to choose the estimator of the variance of the levels `rhoSquare` (i.e. either $\hat{\rho}_1^2$ or $\hat{\rho}^2$) and by default $\hat{\rho}_1^2$ is used.

Usage

```
estProfileWithMBPCR(snpName, chr, position, logratio, chrToBeAnalyzed, maxProbeNumber,
              rhoSquare=NULL, kMax=50, nu=NULL, sigmaSquare=NULL, typeEstRho=1,
                    regr=NULL, hg='hg18')
```

Arguments

`snpName`	array containing the name of each probe
`chr`	array containing the name of the chromosome to which each of the probes belongs. The possible values of the elements of `chr` are: the integers from 1 to 22, 'X' and 'Y'.
`position`	array containing the physical position of each probe
`logratio`	array containing the log2ratio of the raw copy number data
`chrToBeAnalyzed`	array containing the name of the chromosomes that the user wants to analyze. The possible values of the chromosomes are: the integers from 1 to 22, 'X' and 'Y'.
`maxProbeNumber`	maximum number of probes that a chromosome (or arm of a chromosome) can have to be analyzed. The procedure of profile estimation needs the computation of an array of length $(length(chromosome) + 1) * (length(chromosome) + 2)/2$. To be sure to have set this parameter correctly, try to create the array `A <- array(1, dim=(maxProbeNumber+1)*(maxProbeNumber+2)/2)`, before starting with the estimation procedure.
`rhoSquare`	variance of the segment levels. If `rhoSquare=NULL`, then the algorithm estimates it on the sample.
`kMax`	maximum number of segments
`nu`	mean of the segment levels. If `nu=NULL`, then the algorithm estimates it on the sample.
`sigmaSquare`	variance of the noise. If `sigmaSquare=NULL`, then the algorithm estimates it on the sample.

typeEstRho	choice of the estimator of rhoSquare. If typeEstRho=1, then the algorithm estimates rhoSquare with $\hat{\rho}_1^2$, while if typeEstRho=0, it estimates rhoSquare with $\hat{\rho}^2$.
regr	choice of the computation of the regression curve. If regr=NULL, then the regression curve is not computed, if regr="BRC" the Bayesian Regression Curve is computed (BRC with K_2), if regr="BRCAk" the Bayesian Regression Curve Averaging over k is computed (BRCAk).
hg	genome build used for retrieving the base positions of the centromeres in case the chromosomes need to be divided into two parts for the estimation (see explanation of maxProbeNumber). Current available options are: 'hg18', 'hg19' and 'hg38'.

Details

By default, the function estimates the copy number profile with mBPCR and estimating rhoSquare on the sample, using $\hat{\rho}_1^2$. It is also possible to use $\hat{\rho}^2$ as estimator of rhoSquare, by setting typeEstRho=0, or to directly set the value of the parameter.

The function gives also the possibility to estimate the profile with a Bayesian regression curve: if regr="BRC" the Bayesian Regression Curve with K_2 is computed (BRC with K_2), if regr="BRCAk" the Bayesian Regression Curve Averaging over k is computed (BRCAk).

See function writeEstProfile, to have the results in nicer tables or to write them on files.

Value

A list containing:

estPC	an array containing the estimated profile with mBPCR
estBoundaries	the list of estimated breakpoints for each of the analyzed chomosomes
postProbT	the list of the posterior probablity to be a breakpoint for each estimated breakpoint of the analyzed chomosomes
regrCurve	an array containing the estimated bayesian regression curve

estPC and regrCurve have the same length of logratio, hence their components, corresponding to the not analyzed chromosomes, are equal to NA.

References

Rancoita, P. M. V., Hutter, M., Bertoni, F., Kwee, I. (2009). Bayesian DNA copy number analysis. *BMC Bioinformatics* 10: 10. http://www.idsia.ch/~paola/mBPCR

See Also

plotEstProfile, writeEstProfile, computeMBPCR

Examples

```
##import the 10K data of cell line REC
data(rec10k)
##estimation of the profile of chromosome 5
results <- estProfileWithMBPCR(rec10k$SNPname, rec10k$Chromosome, rec10k$PhysicalPosition,
rec10k$log2ratio, chrToBeAnalyzed=5, maxProbeNumber=2000)
##plot the estimated profile of chromosome 5
y <- rec10k$log2ratio[rec10k$Chromosome == 5]
p <- rec10k$PhysicalPosition[rec10k$Chromosome == 5]
plot(p, y)
points(p, results$estPC[rec10k$Chromosome == 5], type='l', col='red')

###for the estimation of the profile of all chromosomes
#results <- estProfileWithMBPCR(rec10k$SNPname, rec10k$Chromosome, rec10k$PhysicalPosition,
#rec10k$log2ratio, chrToBeAnalyzed=c(1:22,'X'), maxProbeNumber=2000)
```

---

estProfileWithMBPCRforOligoSnpSet

Estimate and print the copy number profile of some chromosomes of samples in an oligoSnpSet object

---

Description

Function to estimate the copy number profile with a piecewise constant function using mBPCR. Eventually, it is possible to estimate the profile with a smoothing curve, using either the Bayesian Regression Curve with K_2 (BRC with K_2) or the Bayesian Regression Curve Averaging over k (BRCAk). It is also possible to choose the estimator of the variance of the levels `rhoSquare` (i.e. either $\hat{\rho}_1^2$ or $\hat{\rho}^2$) and by default $\hat{\rho}_1^2$ is used.

Usage

```
estProfileWithMBPCRforOligoSnpSet(sampleData, sampleToBeAnalyzed, chrToBeAnalyzed,
                maxProbeNumber, ifLogRatio=1, rhoSquare=NULL, kMax=50, nu=NULL,
                     sigmaSquare=NULL, typeEstRho=1, regr=NULL, hg='hg18')
```

Arguments

`sampleData`	object of type oligoSnpSet. The following fields must not be empty: `assayData(sampleData)$copyNumb` (it contains the raw copy number values with scale log2 multiplied by 100 and transformed as integers), `featureNames(featureData(sampleData))` (it contains the names of the SNPs), `featureData(sampleData)$chromosome` (it contains the names of the chromosomes to which each of the SNPs belongs), `featureData(sampleData)$position` (it contains the physical positions of the SNPs).

`sampleToBeAnalyzed`
: vector containing the number of the columns corresponding to the samples the user wants to analyze.

`chrToBeAnalyzed`
: array containing the name of the chromosomes that the user wants to analyze. The possible values of the chromosomes are: the integers from 1 to 22, 'X' and 'Y'.

`maxProbeNumber`
: maximum number of probes that a chromosome (or arm of a chromosome) can have to be analyzed. The procedure of profile estimation needs the computation of an array of length $(length(chromosome) + 1) * (length(chromosome) + 2)/2$. To be sure to have set this parameter correctly, try to create the array `A <- array(1, dim=(maxProbeNumber+1)*(maxProbeNumber+2)/2)`, before starting with the estimation procedure.

`ifLogRatio`
: denotes whether the original log2 data were centered at zero (i.e. they were in log2ratio scale) or not. By default, they are considered as derived by log2ratio data (`ifLogRatio=1`), otherwise (`ifLogRatio=0`) they are transformed in order to be derived from log2ratio data.

`rhoSquare`
: variance of the segment levels. If `rhoSquare=NULL`, then the algorithm estimates it on the sample.

`kMax`
: maximum number of segments

`nu`
: mean of the segment levels. If `nu=NULL`, then the algorithm estimates it on the sample.

`sigmaSquare`
: variance of the noise. If `sigmaSquare=NULL`, then the algorithm estimates it on the sample.

`typeEstRho`
: choice of the estimator of `rhoSquare`. If `typeEstRho=1`, then the algorithm estimates `rhoSquare` with $\hat{\rho}_1^2$, while if `typeEstRho=0`, it estimates `rhoSquare` with $\hat{\rho}^2$.

`regr`
: choice of the computation of the regression curve. If `regr=NULL`, then the regression curve is not computed, if `regr="BRC"` the Bayesian Regression Curve is computed (BRC with K_2), if `regr="BRCAk"` the Bayesian Regression Curve Averaging over k is computed (BRCAk).

`hg`
: genome build used for retrieving the base positions of the centromeres in case the chromosomes need to be divided into two parts for the estimation (see explanation of `maxProbeNumber`). Current available options are: `'hg18'`, `'hg19'` and `'hg38'`.

Details

By default, the function estimates the copy number profile with mBPCR and estimating rhoSquare on the sample, using $\hat{\rho}_1^2$. It is also possible to use $\hat{\rho}^2$ as estimator of `rhoSquare`, by setting `typeEstRho=0`, or to directly set the value of the parameter.

The function gives also the possibility to estimate the profile with a Bayesian regression curve: if `regr="BRC"` the Bayesian Regression Curve with K_2 is computed (BRC with K_2), if `regr="BRCAk"` the Bayesian Regression Curve Averaging over k is computed (BRCAk).

Value

A list containing:

estPC	an oligoSnpSet equal to sampleData apart from the field assayData(estPC)$copyNumber, which contains the estimated profile with mBPCR in scale log2ratio multiplied by 100
regrCurve	an oligoSnpSet equal to sampleData apart from the field assayData(regrCurve)$copyNumber, which contains the estimated bayesian regression curve in scale log2ratio multiplied by 100. This object is returned only if regr!=NULL.

The matrices assayData(estPC)$copyNumber and assayData(regrCurve)$copyNumber have the same dimension of assayData(sampleData)$copyNumber, hence their elements, corresponding to the not analyzed chromosomes and samples, are equal to NA.

References

Rancoita, P. M. V., Hutter, M., Bertoni, F., Kwee, I. (2009). Bayesian DNA copy number analysis. *BMC Bioinformatics* 10: 10. http://www.idsia.ch/~paola/mBPCR

See Also

estProfileWithMBPCR, computeMBPCR

Examples

```
###import an example of oligoSnpSet data
#data(oligoSetExample, package="oligoClasses")
##estimation of chromosome 2 in sample 1
#r <-estProfileWithMBPCRforOligoSnpSet(oligoSet, sampleToBeAnalyzed=1, chrToBeAnalyzed=2,
#maxProbeNumber=1000, ifLogRatio=0, rhoSquare=0.0889637)
##plot of the estimated chromosome
#library(SNPchip)
#cc <- r$estPC
#cc1 <- cc[chromosome(cc) == "2",1]
#par(las=1)
#plot(position(cc1), copyNumber(cc1)/100, ylim=c(-0.23, 0.1), ylab="copy number",
#xlab="base position")
```

---

importCNData *Import the copy number data*

Description

Function to import the raw copy number data from a tab delimited file.

Usage

```
importCNData(path, NRowSkip, ifLogRatio=1)
```

Arguments

path	path of the tab delimited file containing the copy number data. The file must contain a table, where in the first column there are the names of the probes (snpName), in the second one, the chromosome to which each probe belongs (the possible values of the chromosomes are: the integers from 1 to 22, 'X' and 'Y'), in the third one, the phisical positions of the probes and in the forth one, the copy number data.
NRowSkip	number of row to skip in the file, before the table. The names of the columns are to be skipped.
ifLogRatio	denotes if the data are either the log2ratio of raw copy number data or raw copy number data. By default, they are considered as log2ratio data, otherwise (ifLogRatio=0) they are transformed in log2ratio data.

Value

A list containing:

snpName	an array containing the names of the probes
chr	an array containing the name of the chromosome to which each probe belongs
position	an array containing the physical position of each probe
logratio	an array containing the log2ratio of the raw copy number data

Examples

```
###import the 10K data of cell line REC
path <- system.file("extdata", "rec10k.txt", package = "mBPCR")
rec10k <- importCNData(path, NRowSkip=1)
plot(rec10k$position[rec10k$chr == 3], rec10k$logratio[rec10k$chr == 3])
```

---

jekoChr11Array250Knsp *Affymetrix GeneChip Mapping 250K NSP Array data of JEKO-1 cell line (chr. 11)*

Description

Affymetrix GeneChip Mapping 250K NSP Array data of JEKO-1 cell line.

Usage

```
data(jekoChr11Array250Knsp)
```

Format

A data frame containing four variables: first is SNP name ('SNPname'), second is probe chromosome ('Chromosome'), third is probe position ('PhysicalPosition') and fourth is probe raw log2ratio ('log2ratio').

Source

Poretti, G. Rancoita, P.M.V. Kwee, I. Bertoni, F., unpublished

---

logAdd *Overflow-safe computation of the logarithm of a sum*

Description

Function to compute the logarithm of a sum of small numbers, avoiding overflow.

Usage

```
logAdd(x)
```

Arguments

x	array or matrix containing the logarithm of the terms of the sum. If x is a matrix, the function return the results by column.

Value

If x is an array, the function returns $log(sum_i(e^x[i]))$, otherwise it returns an array containing the results by column.

Examples

```
x <- log(c(0.0001, 0.0003, 0.000006))
y <- logAdd(x)
##verification that the computation is correct
z <- sum(c(0.0001, 0.0003, 0.000006))
z
exp(y)
```

---

mBPCR-internal *Internal mBPCR functions*

Description

Internal functions of package mBPCR.

Usage

```
computeA10(i, j, y, nu, rhoSquare, sigmaSquare)
computeLA0Vect(y, nu, rhoSquare, sigmaSquare)
computePCReg(y, lA0, lL, lR, nu, rhoSquare, sigmaSquare, kMax=50, regr=NULL)
computeRecursions(lA0, n, kMax=50)
computeRegrCurve(y, typeRegr="BRC", n, kMax=50, lL, lR, lA0, nu, rhoSquare,
                 sigmaSquare, option)
indexLA0(r, c, n)
```

Details

These functions are not to be called directly by the user

`plotEstProfile` *Plot the estimated profile of copy number data*

Description

Function to plot the estimated profiles of copy number data.

Usage

```
plotEstProfile(sampleName='', chr, position, logratio, chrToBePlotted, estPC,
               maxProbeNumber, legendPosition='bottomleft', regrCurve=NULL,
               regr=NULL, hg='hg18')
```

Arguments

`sampleName`	name of the sample, if the user wants to put it in the title of the graph
`chr`	array containing the name of the chromosome to which each probe belongs. The possible values of the elements of `chr` are: the integers from 1 to 22, 'X' and 'Y'.
`position`	array containing the physical position of each probe
`logratio`	array containing the log2ratio of the raw copy number data
`chrToBePlotted`	array containing the name of the estimated chromosomes, that the user wants to plot. The possible values of the chromosomes are: the integers from 1 to 22, 'X' and 'Y'.
`estPC`	array containing the estimated copy number profile as a piecewise constant function. If `estPC=NULL`, only the estimated Bayesian regression curve is plotted.
`maxProbeNumber`	maximum number of probes that a chromosome (or arm of a chromosome) can have to be analyzed. The procedure of profile estimation needs the computation of an array of length $(length(chromosome) + 1) * (length(chromosome) + 2)/2$. To be sure to have set this parameter correctly, try to create the array `A <- array(1, dim=(maxProbeNumber+1)*(maxProbeNumber+2)/2)`, before starting with the estimation procedure.
`legendPosition`	string containing the position of the legend in the plot. The possible values are the same used in the function `plot`.
`regrCurve`	array containing the estimated regression curve. If `regrCurve=NULL`, then the estimated Bayesian regression curve is not plotted. If `regrCurve!=NULL` and also `estPC!=NULL` both estimated profiles are plotted on the same graph.

regr	choice of the computation of the regression curve. If regr=NULL, then the regression curve was not computed (then the estimated Bayesian regression curve is not plotted), if regr="BRC" the Bayesian Regression Curve was computed (mBRC with K_2), if regr="BRCAk" the Bayesian Regression Curve Averaging over k was computed (BRCAk).
hg	genome build used for retrieving the base positions of the centromeres in case the chromosomes need to be divided into two parts for the estimation (see explanation of maxProbeNumber). Current available options are: 'hg18', 'hg19' and 'hg38'.

Details

The function plots the estimated profiles of the chromosomes of chrToBePlotted, separately.

Examples

```
##import the 10K data of cell line REC
data(rec10k)
##estimation of chromosomes 3 and 5
results <- estProfileWithMBPCR(rec10k$SNPname, rec10k$Chromosome, rec10k$PhysicalPosition,
rec10k$log2ratio, chrToBeAnalyzed=c(3,5), maxProbeNumber=2000)
##plot the corresponding estimated profiles
plotEstProfile(sampleName='rec10k', rec10k$Chromosome, rec10k$PhysicalPosition, rec10k$log2ratio,
chrToBePlotted=c(3,5), results$estPC, maxProbeNumber=2000)
```

---

rec10k *Affymetrix GeneChip Mapping 10K Array data of REC-1 cell line*

Description

Affymetrix GeneChip Mapping 10K Array data of REC-1 cell line taken from the reference below.

Usage

```
data(rec10k)
```

Format

A data frame containing five variables: first is SNP name ('SNPname'), second is probe chromosome ('Chromosome'), third is probe position ('PhysicalPosition'), fourth is probe raw log2ratio ('log2ratio') and fifth are is probe genotype ('call').

Source

Rinaldi et al. (2006), Genomic and expression profiling identifies the B-cell associated tyrosine kinase Syk as a possible therapeutic target in mantle cell lymphoma, *British Journal of Haematology*, 132, 303-316

writeEstProfile *Write the estimated profile of copy number data*

Description

Function to write nicely the results of the copy number profile estimation. The function either writes the tables directly on a tab delimited file or returns the corresponding tables.

Usage

```
writeEstProfile(path='', sampleName='', snpName, chr, position, logratio,
                chrToBeWritten, estPC, estBoundaries=NULL, postProbT=NULL,
                regrCurve=NULL, regr=NULL)
```

Arguments

path	path of the folder where the user wants to write the results of the estimation (it must end with '\' in windows, or '//' in linux). If path='', they will be written in the working directory. If path=NULL, the tables will not be written on a file, but only returned by the function.
sampleName	name of the sample. If the name of the sample if provided, it is used to named the files.
snpName	array containing the name of each probe
chr	array containing the name of the chromosome to which each probe belongs. The possible values of the elements of chr are: the integers from 1 to 22, 'X' and 'Y'.
position	array containing the physical position of each probe
logratio	array containing the log2ratio of the raw copy number data
chrToBeWritten	array containing the name of the estimated chromosomes, of which the user wants to write the results. The possible values of the chromosomes are: the integers from 1 to 22, 'X' and 'Y'.
estPC	array containing the estimated copy number profile as a piecewise constant function
estBoundaries	list containing the vectors of the estimated breakpoints, for each of the chromosomes mentioned in chrToBeWritten. If estBoundaries=NULL, then this information is not written.
postProbT	list containing the vectors of the posterior probabilities to be a breakpoint of the estimated breakpoints, for each of the chromosomes mentioned in chrToBeWritten. If postProbT=NULL, then this information is not written in the file containing the estimated breakpoints.
regrCurve	array containing the estimated regression curve. If regrCurve=NULL, then the file containing this information is not written.

regr — choice of the computation of the regression curve. If regr=NULL, then the regression curve was not computed (then the file containing this information is not written), if regr="BRC" the Bayesian Regression Curve with K_2 was computed (BRC with K_2), if regr="BRCAk" the Bayesian Regression Curve Averaging over k was computed (BRCAk).

Details

The function writes or returns at maximum three tables:

-one containing the estimated profile with mBPCR (the columns are: 'SNPname', 'chromosome', 'position', 'rawLog2ratio', 'mBPCRestimate')

-one containing a summary about the estimated profile with mBPCR (the columns are: 'SNPname(start)', 'SNPname(end)', 'chromosome', 'position(start)', 'position(end)', 'nProbes', 'mBPCRestimate' and, eventually, 'breakpointPostProb'). This table is not created if estBoundaries=NULL.

-one containing the estimated profile with a regression curve (the columns are: 'SNPname', 'chromosome', 'position', 'rawLog2ratio' and the name of the regression curve used). This table is not created if regrCurve=NULL.

Examples

```
##import the 10K data of cell line REC
data(rec10k)
##estimation of chromosome 5
results <- estProfileWithMBPCR(rec10k$SNPname, rec10k$Chromosome, rec10k$PhysicalPosition,
rec10k$log2ratio, chrToBeAnalyzed=5, maxProbeNumber=2000)
##write the estimated profile of chromosome 5 in a file in  the working directory
writeEstProfile(path='', sampleName='rec10k', rec10k$SNPname, rec10k$Chromosome,
rec10k$PhysicalPosition, rec10k$log2ratio, chrToBeWritten=5, results$estPC, results$estBoundaries,
results$postProbT)


#### the same result can be obtained in the following way, by using the function computeMBPCR
#### for the estimation
##estimation of the global parameters
#param <- estGlobParam(rec10k$log2ratio)
##estimation of chromosome 5
#results <- computeMBPCR(rec10k$log2ratio[rec10k$Chromosome == 5], nu=param$nu,
#rhoSquare=param$rhoSquare, sigmaSquare=param$sigmaSquare)
##write the estimated profile of chromosome 5 in a file in the working directory
#estPC <- array(dim=length(rec10k$SNPname))
#estBoundaries <- list(dim=1)
#postProbT <- list(dim=1)
#estPC[rec10k$Chromosome == 5] <- results$estPC
#estBoundaries[[1]] <- results$estBoundaries
#postProbT[[1]] <- c(results$postProbT[results$estBoundaries[-results$estK]],1)
#writeEstProfile(path='', sampleName='rec10k', rec10k$SNPname, rec10k$Chromosome,
#rec10k$PhysicalPosition, rec10k$log2ratio, chrToBeWritten=5, estPC, estBoundaries, postProbT)
```

Index

∗ **datasets**
  jekoChr11Array250Knsp, 11
  rec10k, 14
∗ **file**
  importCNData, 10
  writeEstProfile, 15
∗ **hplot**
  plotEstProfile, 13
∗ **internal**
  mBPCR-internal, 12
∗ **manip**
  centromere, 2
∗ **math**
  logAdd, 12
∗ **print**
  plotEstProfile, 13
∗ **regression**
  computeMBPCR, 3
  estProfileWithMBPCR, 6
  estProfileWithMBPCRforOligoSnpSet, 8
∗ **smooth**
  computeMBPCR, 3
  estProfileWithMBPCR, 6
  estProfileWithMBPCRforOligoSnpSet, 8
∗ **univar**
  estGlobParam, 5

centromere, 2
computeA10 *(*mBPCR-internal*)*, 12
computeLA0Vect *(*mBPCR-internal*)*, 12
computeMBPCR, 3, *7*, *10*
computePCReg *(*mBPCR-internal*)*, 12
computeRecursions *(*mBPCR-internal*)*, 12
computeRegrCurve *(*mBPCR-internal*)*, 12

estGlobParam, *4*, 5
estProfileWithMBPCR, *4*, 6, *10*
estProfileWithMBPCRforOligoSnpSet, 8

importCNData, 10
indexLA0 *(*mBPCR-internal*)*, 12

jekoChr11Array250Knsp, 11

logAdd, 12

mBPCR-internal, 12

plotEstProfile, *4*, *7*, 13

rec10k, 14

writeEstProfile, *4*, *7*, 15