mBPCR: A package for DNA copy number profile estimation

P. M. V. Rancoita[1,2,3] and M. Hutter[4]

[1]Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), Manno-Lugano, Switzerland
[2]Laboratory of Experimental Oncology, Oncology Institute of Southern Switzerland (IOSI), Bellinzona, Switzerland
[3]Dipartimento di Matematica, Università degli Studi di Milano, Milano, Italy
[4]RSISE @ ANU and SML @ NICTA, Canberra, ACT, 0200, Australia

paola@idsia.ch

Contents

1 Introduction

The algorithm mBPCR is a tool for estimating the profile of the $\log_2$ratio of copy number data. The procedure is a Bayesian piecewise constant regres-

sion and can be applied, generally, to estimate any piecewise constant function (like the $\log_2$ratio of the copy number data). The method is described in [3] and represents a significant improvement of the original algorithm BPCR, presented in [1] and [2].

This document shows several examples of how to use the package. The data used are principally: the Affymetrix GeneChip Mapping 10K Array data of cell line REC-1 [4] and the Affymetrix GeneChip Mapping 250K Array data of chromosome 11 of cell line JEKO-1 (unpublished).

2 Example 1: profile estimation

In this example we estimate the copy number profile of sample `rec10k`.

```
> library(mBPCR)
```

First, we import the 10K Array data of cell line REC-1.

```
> data(rec10k)
```

During the computation, the algorithm needs to create a vector of size (`maxProbeNumber`+1)(`maxProbeNumber`+2)/2, where `maxProbeNumber` is the maximum number of probes of a chromosome (or arm of a chromosome, for denser Array). Hence, before the estimation, we must verify if we have enough RAM to allocate such a vector. In case of the 10K Array data, we know that all chromosomes have less than 1000 probes, thus we verify if we can set `maxProbeNumber=1000`, with the following commands,

```
> maxProbeNumber <- 1000
```

```
> A <- array(1, dim = (maxProbeNumber + 1) * (maxProbeNumber +
+     2)/2)
```

If last command does not give any error regarding the memory allocation, then we can set `maxProbeNumber=1000` and remove `A` to save space.

```
> remove(A)
```

To estimate the profile of one or more chromosomes, we need to set the parameter `chrToBeAnalyzed` with the vector of the names of the chromosomes that we want to analyze (the names allowed are: `X`, `Y` and any integer from `1` to `22`). In the following example, we estimate the profile of chromosomes 3 and 5 of sample REC-1. Instead, to estimate the profile of the whole genome, we need to set `chrToBeAnalyzed = c(1:22,"X")`.

```
> results <- estProfileWithMBPCR(rec10k$SNPname, rec10k$Chromosome,
+     rec10k$PhysicalPosition, rec10k$log2ratio, chrToBeAnalyzed = c(3,
+         5), maxProbeNumber = 1000)
```

We can nicely write the results on tab delimited files in the working directory, by using the function `writeEstProfile` (Tables 1 and 2 show the first lines of the two tables created by the command below). Setting `sampleName="rec10k"`, the name of the files will contain the name of the sample `rec10k`. If `path=NULL`, the tables will not be written on files, but only returned by the function.

```
> writeEstProfile(path = "", sampleName = "rec10k", rec10k$SNPname,
+     rec10k$Chromosome, rec10k$PhysicalPosition, rec10k$log2ratio,
+     chrToBeWritten = c(3, 5), results$estPC, results$estBoundaries)
```

SNPname	chromosome	position	rawLog2ratio	mBPCRestimate
SNP_A-1511742	3	540961	0.367371066	-0.33118451
SNP_A-1515436	3	653347	-0.051399153	-0.33118451
SNP_A-1515061	3	1100383	-0.577766999	-0.33118451
SNP_A-1510244	3	1167829	-1.377069649	-0.33118451
SNP_A-1517422	3	1167988	-1.058893689	-0.33118451
SNP_A-1515258	3	1478475	-0.184424571	-0.33118451

Table 1: Example of table containing the profile estimated with mBPCR.

SNPname(start)	SNPname(end)	chromosome	position(start)	position(end)	nProbes	mBPCRestimate
SNP_A-1511742	SNP_A-1517209	3	540961	3814711	23	-0.33118451
SNP_A-1512404	SNP_A-1508199	3	3887946	6473283	28	0.26189600
SNP_A-1519522	SNP_A-1509746	3	6482290	141372655	528	-0.05735106
SNP_A-1516670	SNP_A-1518807	3	141372855	141373169	4	0.04208212
SNP_A-1511225	SNP_A-1516851	3	141494264	141925969	3	0.11634699
SNP_A-1517017	SNP_A-1517017	3	142426479	142426479	1	-0.07473727

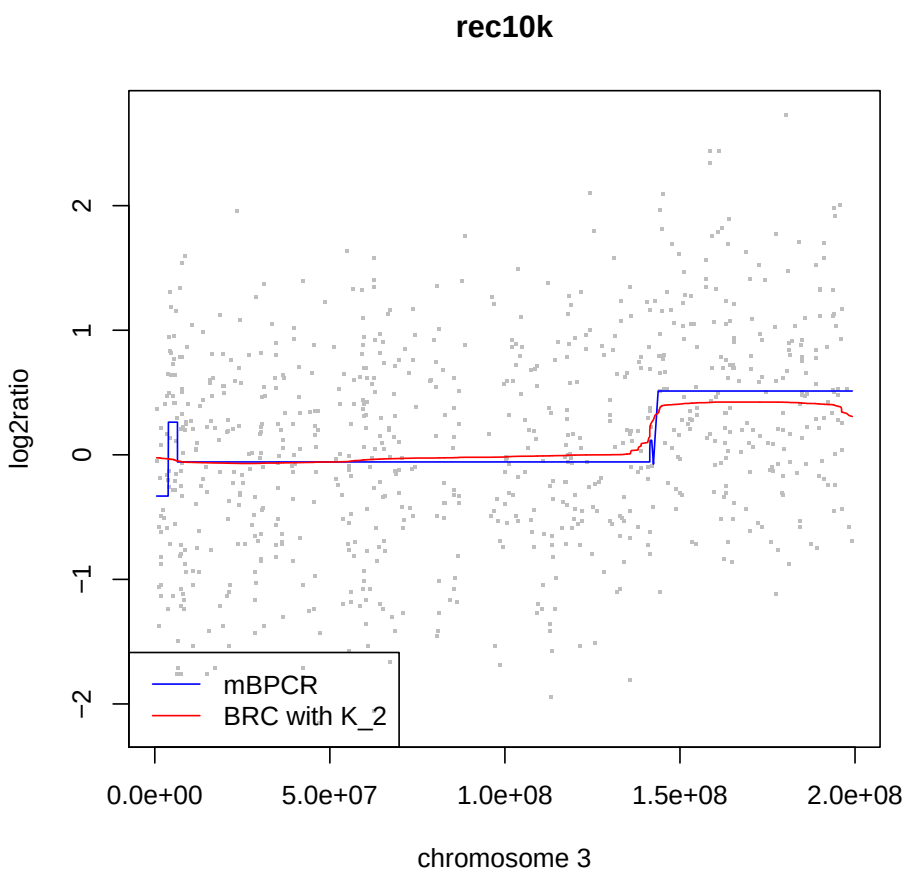
Table 2: Example of table containing a summary of the breakpoints estimated with mBPCR.

We can also estimate the profile with a Bayesian regression curve [3]. For example, with the following command we estimate the profile of chromosome 3 using both mBPCR and the Bayesian Regression Curve with $\hat{K}_2$.

```
> results <- estProfileWithMBPCR(rec10k$SNPname, rec10k$Chromosome,
+     rec10k$PhysicalPosition, rec10k$log2ratio, chrToBeAnalyzed = 3,
+     regr = "BRC", maxProbeNumber = 1000)
```

After the estimation, we can plot the profiles using the function `plotEstProfile`. For example, the following command plots the profile of chromosome 3 estimated with both methods.

```
> plotEstProfile(sampleName = "rec10k", rec10k$Chromosome, rec10k$PhysicalPosition,
+     rec10k$log2ratio, chrToBePlotted = 3, results$estPC, maxProbeNumber = 2000,
+     regrCurve = results$regrCurve, regr = "BRC")
```



As second example, we estimate the profile of chromosome 11 of sample JEKO-1. Notice that we need to set `maxProbeNumber <- 9000` (because

both arms of chromosome 11 contain less than 9000 probes) and, if this is possible on your machine, the computation can be long. Moreover, for the estimation, we use the estimates of the parameters computed on the whole genome to achieve a better profile (for the estimation of the global parameters, see the use of function `estGlobParam` in Section 3).

```
> data(jekoChr11Array250Knsp)

> maxProbeNumber <- 9000

> A <- array(1, dim = (maxProbeNumber + 1) * (maxProbeNumber +
+     2)/2)

> remove(A)

> results <- estProfileWithMBPCR(jekoChr11Array250Knsp$SNPname,
+     jekoChr11Array250Knsp$Chromosome, jekoChr11Array250Knsp$PhysicalPosition,
+     jekoChr11Array250Knsp$log2ratio, chrToBeAnalyzed = 11, maxProbeNumber = 9000,
+     rhoSquare = 0.0479, nu = -3.012772e-10, sigmaSquare = 0.0699)

> plotEstProfile(sampleName = "jeko250Knsp", jekoChr11Array250Knsp$Chromosome,
+     jekoChr11Array250Knsp$PhysicalPosition, jekoChr11Array250Knsp$log2ratio,
+     chrToBePlotted = 11, results$estPC, maxProbeNumber = 9000)
```

3 Example 2: use of function `estGlobParam`

In general, even if we are not interested in the analysis of the whole genome, the global parameters should be estimated on the entire sample, using the function `estGlobParam`. Here, we estimate the global parameters of sample REC-1 (in the following, the variance of the segment ρ^2 is estimated with $\hat{\rho}_1^2$).

```
> data(rec10k)

> estGlobParam(rec10k$log2ratio)

$nu
[1] -0.02403854

$rhoSquare
[1] 0.0889637

$sigmaSquare
[1] 0.5971426
```

4 Example 3: use of function `computeMBPCR`

If we are interested in estimating only a part of a chromosome or a simulated sample, we should not use the function `estProfileWithMBPCR`, but use the function `computeMBPCR` which estimates the profile directly. In the following example, we estimates the profile of a part of chromosome 11 of sample JEKO-1.

```
> data(jekoChr11Array250Knsp)
```

We select a part of chromosome 11.

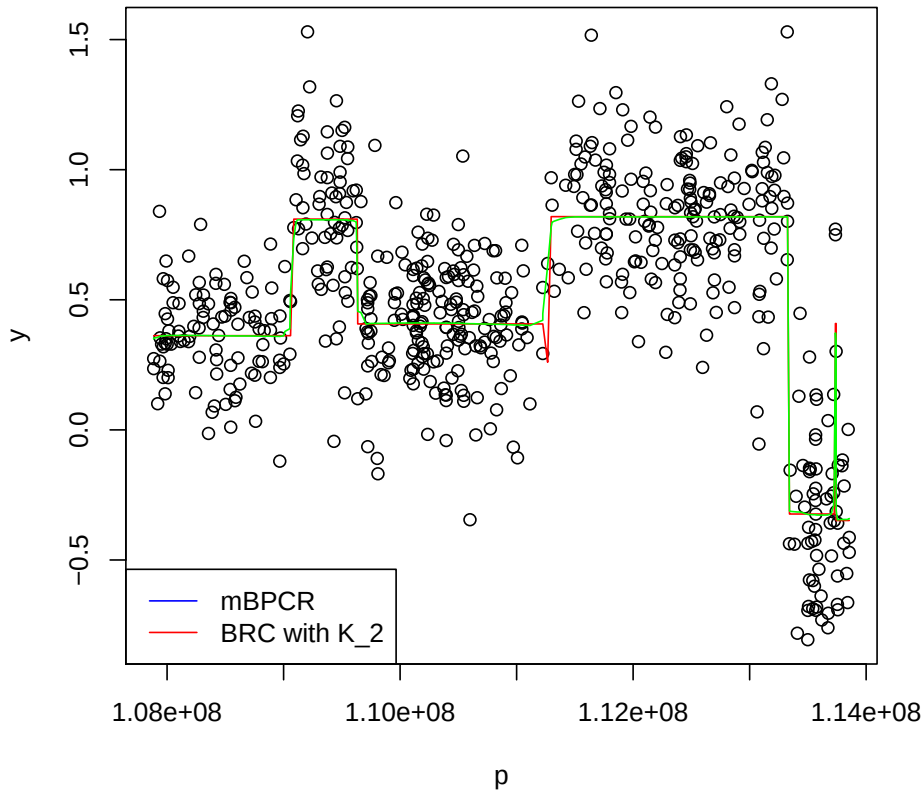
```
> y <- jekoChr11Array250Knsp$log2ratio[10600:11200]

> p <- jekoChr11Array250Knsp$PhysicalPosition[10600:11200]
```

We estimate the profile with mBPCR and BRC with $\hat{K}_2$, using the global parameters estimated on the whole genome.

```
> results <- computeMBPCR(y, nu = -3.012772e-10, rhoSquare = 0.0479,
+     sigmaSquare = 0.0699, regr = "BRC")
```

Finally, we plot the results.

```
> plot(p, y)
> points(p, results$estPC, type = "l", col = "red")
> points(p, results$regrCurve, type = "l", col = "green")
> legend(x = "bottomleft", legend = c("mBPCR", "BRC with K_2"),
+     lty = c(1, 1), col = c(4, 2))
```



5 Example 4: `importCNData`, an easy function to import data

There is also the possibility to easily import external data, by using `importCNData`. The data should be in a tab delimited file and the data table should have at least four columns representing, respectively, the probe names, the chromosome to which each probe belongs, the physical positions of the probes inside the chromosome and the copy number data (an example of table can be found in Table 3). The allowed names of the chromosomes are: `X`, `Y` and any integer from `1` to `22`). In the following example, we import data of sample REC-1.

As first step, we need to set a variable with the path of the file containing

SNPname	Chromosome	PhysicalPosition	log2ratio
SNP_A-1509443	1	2882121	-0.184424571
SNP_A-1518557	1	3985402	0.097610797
SNP_A-1517286	1	4804829	0.443606651
SNP_A-1516024	1	4982250	-1.089267338
SNP_A-1514538	1	5468765	-0.862496476
SNP_A-1516403	1	5596686	1.097610797
⋮	⋮	⋮	⋮

Table 3: Example of data table.

the data. To import our data, we set `path` as the path of REC-1 data in the folder of package `mBPCR`,

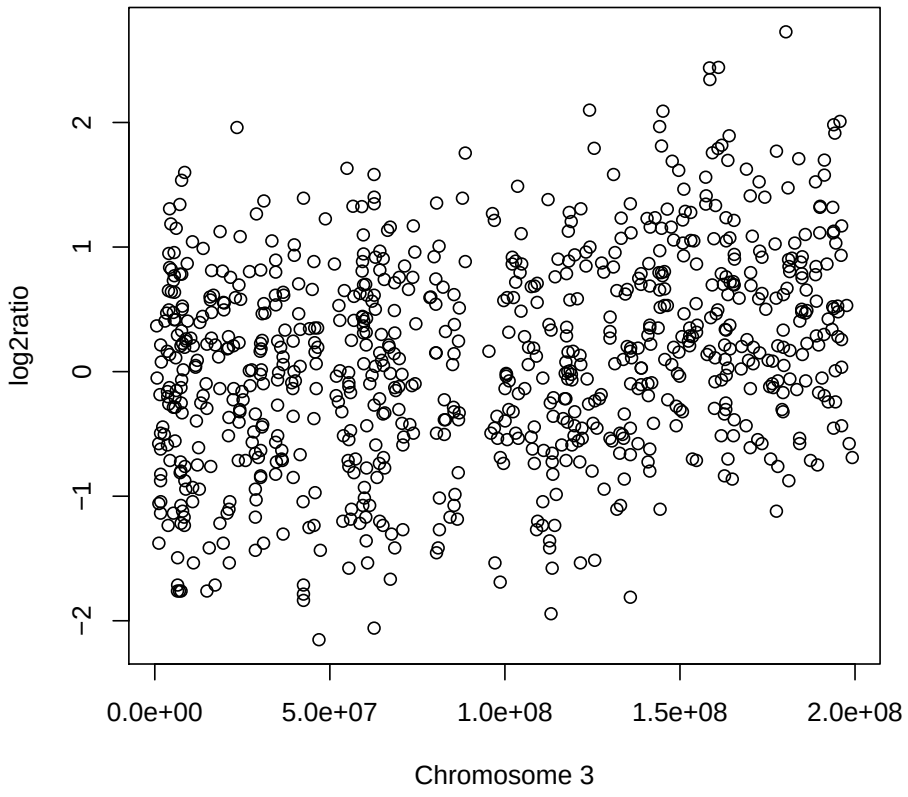
```
> path <- system.file("extdata", "rec10k.txt", package = "mBPCR")
```

Then, we use function `importCNData`. The parameter `NRowSkip` denotes how many rows there are before the table (notice that the name of the columns must be skipped). If the copy number data are not in $\log_2$ratio scale, the parameter `ifLogRatio` should be put as zero.

```
> rec10k <- importCNData(path, NRowSkip = 1)
```

Now, the SNP name are in the variable `SNPname`, the chromosomes of the probes are in `chr`, the physical positions in `position` and the raw $\log_2$ratio data in `logratio`. Here, we plot the raw data of chromosome 3.

```
> plot(rec10k$position[rec10k$chr == 3], rec10k$logratio[rec10k$chr ==
+     3], xlab = "Chromosome 3", ylab = "log2ratio")
```



6 Example 5: estimation of samples in an `oligoSnpSet` object

In this example, we estimate the profile of chromosome 8 of two samples that are contained in an `oligoSnpSet` object. We load the data that are contained in the package `SNPchip`.

```
> library(SNPchip)
```

```
> data(sample.snpset)
```

The object `sample.snpset` contains the data of five HapMap samples and we want to analyze the third and the fourth samples. Since the samples

should not have many copy number changes, we use `rhoSquare` equal to the one of the sample REC-1. Moreover, the data are not in $\log_2$ratio scale, thus we set `ifLogRatio=0`.

```
> library(mBPCR)
```

```
> r <- estProfileWithMBPCRforOligoSnpSet(sample.snpset, sampleToBeAnalyzed = 3:4,
+     chrToBeAnalyzed = 8, maxProbeNumber = 1000, ifLogRatio = 0,
+     rhoSquare = 0.0889637)
```

After the estimation, we can plot the profiles using a function of the package `SNPchip`.

```
> cc <- r$estPC
```

```
> cc1 <- cc[chromosome(cc) == 8, 3:4]
```

```
> plotSnp(cc1)
```

7 Suggestions

For an optimal use of mBPCR, especially in case of samples coming from patients, we suggest to take care to the following issues:

- even if the goal is to estimate the profile of only a part of the genome, the global parameters should be estimated on the whole genome;
- if the goal is to estimate the profile of one or more patients, it is better to estimate the variance of the segment levels (ρ^2) on a cell line, or on a sample with many aberrations, and use this value in the profile estimation of all patients. In fact, we need many aberrations to estimate well ρ^2.

References

[1] M. Hutter. Exact Bayesian regression of piecewise constant functions. *Bayesian Analysis*, 2(4): 635–664, 2007.

[2] M. Hutter. Bayesian Regression of Piecewise Constant Functions. In J.M. Bernardo, M.J. Bayarri, J.O. Berger, A.P. David, D. Heckerman, A.F.M. Smith, and M. West, editors, *Bayesian Statistics: Proceedings of the Eighth Valencia International Meeting*. Universitat de València and International Society for Bayesian Analysis, 2007.

[3] P.M.V. Rancoita, M. Hutter, F. Bertoni, and I. Kwee. Bayesian DNA copy number analysis. *BMC Bioinformatics*, 10(10), 2009.

[4] A. Rinaldi, I. Kwee, M. Taborelli, C. Largo, S. Uccella, V. Martin, G. Poretti, G. Gaidano, G. Calabrese, G. Martinelli, *et al.*. 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, 2006.

mBPCR

October 24, 2009

`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=
```

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="
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=
#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
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, maxProb
                    rhoSquare=NULL, kMax=50, nu=NULL, sigmaSquare=NULL, typeEs
```

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).

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$PhysicalPositio
##plot the estimated profile of chromosome 5
y <- rec10k$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$PhysicalPositic
```

`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, chrToBeAnaly
                    rhoSquare=NULL, kMax=50, nu=NULL, sigmaSquare=NULL, typeEs
```

Arguments

`sampleData` object of type oligoSnpSet. The following fields must not be empty: `assayData(sampleData)$` (it contains the raw copy number values), `featureNames(featureData(sampleData))` (it contains the names of the SNPs), `featureData(sampleData)$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 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.

`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).

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

`regrCurve` an oligoSnpSet equal to sampleData apart from the field `assayData(regrCurve)$copyNumbe` which contains the estimated bayesian regression curve. 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(sample.snpset)
##estimation of chromosome 8 in samples 3 and 4
#r <- estProfileWithMBPCRforOligoSnpSet(sample.snpset, sampleToBeAnalyzed=3:4, chrToBeAna
##plot of the estimated chromosomes
#cc <- r$estPC
#cc1 <- cc[chromosome(cc) == 8,3:4]
#plotSnp(cc1)
```

---

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)
```

`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,
               legendPosition='bottomleft', regrCurve=NULL, regr=NULL)
```

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).

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$PhysicalPositio
##plot the corresponding estimated profiles
plotEstProfile(sampleName='rec10k', rec10k$Chromosome, rec10k$PhysicalPosition, rec10k$lo
```

---

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$PhysicalPositio
##write the estimated profile of chromosome 5 in a file in  the working directory
writeEstProfile(path='', sampleName='rec10k', rec10k$SNPname, rec10k$Chromosome, rec10k$P


#### the same result can be obtained in the following way, by using the function computeM
#
##estimation of the global parameters
#param <- estGlobParam(rec10k$log2ratio)
##estimation of chromosome 5
#results <- computeMBPCR(rec10k$log2ratio[rec10k$Chromosome == 5], nu=param$nu, rhoSquare
##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$Chromosome, rec10k$
```

Index