CGHnormaliter

April 19, 2010

`CGHnormaliter-package`

This package implements the CGHnormaliter algorithm which is a strategy for improved normalization of array Comparative Genomic Hybridization (aCGH) data.

Description

CGHnormaliter is an iterative algorithm for normalization of aCGH data displaying imbalanced aberrations. First, provisory balanced copy numbers are identified and subsequently used for normalization based on LOWESS. These two steps are then iterated to refine the normalization. The assumption here is that the temporary exclusion of aberrations allows for a more appropriate calculation of the LOWESS regression curve. As a result, after normalization, the log2 intensity ratios of the normals will generally be closer to zero and better reflect the biological reality.

Details

Package:	CGHnormaliter
Type:	Package
Version:	1.1.0
Date:	2009-06-25
License:	GPLv3 (http://www.gnu.org/copyleft/gpl.html)

The package contains two public functions. The function `CGHnormaliter` performs the iterative normalization of aCGH data, while the function `CGHnormaliter.write.table` prints normalized aCGH data to a file. See function documentation for details.

Author(s)

Thomas W. Binsl, Bart P.P. van Houte, Hannes Hettling

References

Bart P.P. van Houte, Thomas W. Binsl, Hannes Hettling, Walter Pirovano and Jaap Heringa. CGHnormaliter: an iterative strategy to enhance normalization of array CGH data with imbalanced aberrations. BMC Genomics, 10:401, 2009.

See Also

CGHcall, DNAcopy

CGHnormaliter *Iterative normalization of aCGH data*

Description

Implementation of an iterative algorithm for normalization of aCGH data displaying imbalanced aberrations.

Usage

```
CGHnormaliter(data, nchrom = 24, stop_threshold = 0.01, max_iterations = 5)
```

Arguments

data	Either a dataframe or character string containing a filename. See section Details for the format.
nchrom	Number of chromosomes.
stop_threshold	Threshold value for the mean difference between the LOWESS regression curves from two consecutive iterations. The iteration is terminated if this difference is below the stop_threshold for all samples.
max_iterations	Maximum number of iterations.

Details

The input should be either a data.frame or the file name of a tabseparated text file (text files 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 position columns must contain numbers only. Following these are two columns with the raw test and reference intensities for each of your samples. These intensities must be numeric as well. If the input type is a text file, missing values should be represented as 'NA' or an empty field. There is a CGHnormaliter.write.table method that prints the results in a tabular format.

Value

This function returns a matrix of objects of class cghCall with dimension (number of clones) * (number of samples). Each object contains the following components (See section Examples on how to access them):

normalized data	A matrix with the normalized log2 intensity ratios for each profile.
segments	A matrix with the segments for each profile.
calls	A data.frame with the calls for each profile. Values are −1 (loss), 0 (normal) or 1 (gain).
probabilities	A data.frame with 3 columns of probe information (name, chromosome and position), followed by 3 columns with aberration probabilities for each sample.

Author(s)

Thomas W. Binsl, Bart P.P. van Houte, Hannes Hettling

References

Bart P.P. van Houte, Thomas W. Binsl, Hannes Hettling, Walter Pirovano and Jaap Heringa. CGHnormaliter: an iterative strategy to enhance normalization of array CGH data with imbalanced aberrations. BMC Genomics, 10:401, 2009.

Examples

```
data(Leukemia)
## Normalize the raw intensity values of the first 3 chromosomes.
result <- CGHnormaliter(Leukemia, nchrom=3)
## Get the normalized log2 intensity ratios, segments and calls
normalized.data <- copynumber(result)
segmented.data <- segmented(result)
called.data <- calls(result)
## Plot the normalization result of sample 2
plot(result[,2])
## Write the normalized log2 intensity ratios to file
CGHnormaliter.write.table(result)
```

CGHnormaliter.write.table

Print normalized data to a file

Description

This function stores the results of a CGHnormaliter normalization into a plain text file.

Usage

```
CGHnormaliter.write.table(result, file="normalized.txt")
```

Arguments

result	Result object of a CGHnormaliter normalization.
file	Filename to store the data in.

Details

This function prints the normalized log2 intensity ratios to a tabseparated file with the specified file name.

Author(s)

Thomas W. Binsl, Bart P.P. van Houte, Hannes Hettling

References

Bart P.P. van Houte, Thomas W. Binsl, Hannes Hettling, Walter Pirovano and Jaap Heringa. CGHnormaliter: an iterative strategy to enhance normalization of array CGH data with imbalanced aberrations. BMC Genomics, 10:401, 2009.

See Also

CGHnormaliter

Examples

```
data(Leukemia)
## Normalize the intensity ratios
## Not run: result <- CGHnormaliter(Leukemia)
## Write the result to a file
## Not run: CGHnormaliter.write.table(result, "normalization_results.txt")
```

Leukemia

Array CGH experiment data on childhood acute lymphoblastic leukemia (ALL) in humans.

Description

A dataframe containing 30180 rows and 10 columns, representing the array CGH data of 3 ALL samples.

Usage

```
Leukemia
```

Format

A dataframe containing the following 10 columns:

CloneID The unique identifiers of array elements.

Chromosome Chromosome number of each array element.

Start Chromosomal start position in bp of each array element.

End Chromosomal end position in bp of each array element.

Case1.test Background corrected test intensity values for sample 1.

Case1.ref Background corrected reference intensity values for sample 1.

Case2.test Background corrected test intensity values for sample 2.

Case2.ref Background corrected reference intensity values for sample 2.

Case3.test Background corrected test intensity values for sample 3.

Case3.ref Background corrected reference intensity values for sample 3.

Source

Provided by the authors (see references).

References

Paulsson K, Heidenblad M, Morse H, Borg A, Fioretos T, Johansson B: Identication of cryptic aberrations and characterization of translocation breakpoints using array CGH in high hyperdiploid childhood acute lymphoblastic leukemia. Leukemia 2006, 20:2002-2007.

Index