

Package ‘BOBaFIT’

November 14, 2024

Type Package

Title Refitting diploid region profiles using a clustering procedure

Version 1.10.0

Description This package provides a method to refit and correct the diploid region in copy number profiles. It uses a clustering algorithm to identify pathology-specific normal (diploid) chromosomes and then use their copy number signal to refit the whole profile. The package is composed by three functions: DRrefit (the main function), ComputeNormalChromosome and PlotCluster.

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

URL <https://github.com/andrea-poletti-unibo/BOBaFIT>

BugReports <https://github.com/andrea-poletti-unibo/BOBaFIT/issues>

Imports dplyr, NbClust, ggplot2, ggbio, grDevices, stats, tidyr, GenomicRanges, ggforce, stringr, plyranges, methods, utils, magrittr

Suggests rmarkdown, markdown, BiocStyle, knitr, testthat (>= 3.0.0), utils, testthat

Config/testthat/edition 3

biocViews CopyNumberVariation, Clustering, Visualization, Normalization, Software

Depends R (>= 2.10)

VignetteBuilder knitr

git_url <https://git.bioconductor.org/packages/BOBaFIT>

git_branch RELEASE_3_20

git_last_commit 558c8ab

git_last_commit_date 2024-10-29

Repository Bioconductor 3.20

Date/Publication 2024-11-14

Author Andrea Poletti [aut],
Gaia Mazzocchetti [aut, cre],
Vincenza Solli [aut]

Maintainer Gaia Mazzocchetti <bioinformatic.seragnoli@gmail.com>

Contents

computeNormalChromosomes	2
DRrefit	3
DRrefit_plot	3
PlotChrCluster	5
Popeye	6
TCGA_BRCA_CN_segments	6
%>%	7

Index	8
--------------	----------

computeNormalChromosomes
computeNormalChromosomes

Description

This function compute the DRrefits' input "chromosome list". It is a vector that contains the chromosomal arms considered "normal" in the cohort of samples tested (BED file), under a specific tolerance value

Usage

```
computeNormalChromosomes(
  segments,
  tolerance_val = 0.15,
  maxCN = 6,
  min_threshold = 1.6,
  max_threshold = 2.4,
  verbose = FALSE
)
```

Arguments

segments	data.frame formatted with correct column names
tolerance_val	decimal value of alteration frequency. By default is 0.15
maxCN	threshold of max copy number to consider. By default is 6
min_threshold	minimum threshold to define a normal CN. By default is 1.60
max_threshold	maximum threshold to define a normal CN. By default is 2.40
verbose	print information about the processes of the function. By default is FALSE

Value

vector with chromosome names and plot with the alteration rate of each chromosomal arms

Examples

```
data("TCGA_BRCA_CN_segments")
chr_list <- computeNormalChromosomes(segments = TCGA_BRCA_CN_segments)
```

DRrefit

DRrefit

Description

This function refits the diploid region of input copy number profiles (segments - BED file)

Usage

```
DRrefit(
  segments_chort,
  chrlist,
  maxCN = 6,
  clust_method = "ward.D2",
  verbose = FALSE
)
```

Arguments

`segments_chort` data.frame formatted with correct column names

`chrlist` list of normal chromosome arms (pathology-specific)

`maxCN` threshold of max copy number to consider. By default is 6

`clust_method` clustering method. By default is "ward.D2"

`verbose` print information about the processes of the function. By default is FALSE

Value

Return two data frames, one is the DRrefit-corrected segments and the other is the samples report. See the vignette for data frame descriptions.

Examples

```
data("TCGA_BRCA_CN_segments")

chr_list <- c("10q", "11p", "12p", "19q", "1p", "21q", "2q", "3p", "4p", "4q", "6p", "6q", "7p" )

results <- DRrefit(segments_chort = TCGA_BRCA_CN_segments,
  chrlist = chr_list)
```

DRrefit_plot

DRrefit_plot

Description

The function plot the copy number profile before and after DRrefit recalibration

Usage

```
DRrefit_plot(  
  corrected_segments,  
  DRrefit_report,  
  plot_viewer = F,  
  plot_save = F,  
  plot_format = "png",  
  plot_path  
)
```

Arguments

<code>corrected_segments</code>	DRrefit output dataframe.
<code>DRrefit_report</code>	DRrefit output dataframe.
<code>plot_viewer</code>	Logical parameter. When it is TRUE, the function print the output plot in the R viewer. By default is FALSE.
<code>plot_save</code>	Logical parameter. When it is TRUE, the function save the plot in the chosen path and format. By default is FALSE.
<code>plot_format</code>	File format for the output plots (accepts "png", "jpg", "pdf", "tiff"). By default is "png"
<code>plot_path</code>	Path to save output plots.

Value

Return the sample copy number profile before and after DRrefit recalibration. The function can output the figure in the R viewer on save it in a specific path.

Examples

```
data("TCGA_BRCA_CN_segments")  
  
chr_list <- c("10q", "11p", "12p", "19q", "1p", "21q", "2q", "3p", "4p", "4q", "6p", "6q", "7p" )  
  
results <- DRrefit(segments_chort = TCGA_BRCA_CN_segments, chrlist = chr_list)  
  
my_segments <- results$corrected_segments  
my_report <- results$report  
  
DRrefit_plot(corrected_segments = my_segments,  
             DRrefit_report = my_report,  
             plot_viewer= FALSE,  
             plot_save = FALSE)
```

PlotChrCluster	<i>PlotChrCluster</i>
----------------	-----------------------

Description

The function clusters chromosomes based on the copy number (CN) and returns a graph where it is possible to observe the different groups and two data frames (report and plot_table). See the vignette for the data frame descriptions.

Usage

```
PlotChrCluster(
  segs,
  clust_method = "ward.D2",
  plot_output = TRUE,
  plot_viewer = TRUE,
  plot_save = FALSE,
  plot_format = "png",
  plot_path,
  verbose = FALSE
)
```

Arguments

segs	data.frame with segments of samples. It must be formatted with correct column names (start, end, ID)
clust_method	clustering method. Default is "ward.D2"
plot_output	Whether to plot refitted profiles (logical)
plot_viewer	Logical parameter. When it is TRUE, the function print the output plot in the R viewer. By default is TRUE.
plot_save	Logical parameter. When it is TRUE, the function save the plot in the chosen path and format. By default is TRUE.
plot_format	File format for the output plots (accepts "png", "jpg", "pdf", "tiff"). By default is "png"
plot_path	Path to save output plots.
verbose	print information about the processes of the function. By default is FALSE

Value

Plot with chromosomes clustered

Examples

```
data(TCGA_BRCA_CN_segments)
Cluster <- PlotChrCluster(segs=TCGA_BRCA_CN_segments,
  clust_method= "ward.D2",
  plot_output=FALSE)
```

 Popeye

Popeye

Description

The function assign the chromosomal arm to each segment.

Usage

```
Popeye(segments)
```

Arguments

`segments` data.frame formatted with correct column names (see package vignette)

Value

Return a data frame containg segments with the arm annotation.

Examples

```
data("TCGA_BRCA_CN_segments")
data <- TCGA_BRCA_CN_segments[1:9] #as it already presents the arm column
data_annotated <- Popeye(segments = data)
```

TCGA_BRCA_CN_segments *Segments of 100 Breast Cancer samples, downloaded from TCGA-BRCA.*

Description

Segments of 100 Breast Cancer samples, downloaded from TCGA-BRCA.

Usage

```
TCGA_BRCA_CN_segments
```

Format

A data frame with 79,607 rows and 12 variables:

chr Chromosome which the segment belong
start Starting point of the segment, in Mb
end Ending point of the segment, in Mb
width Width of the segment, in Mb
strand Strand of the segment
ID Sample name
Num_Probes Probes involved
Segment_Mean LogR of the segments

Sample Barcode of tCGA-BRCA database
arm Arm information, p o q
chrarm Chromosomal arm which the segment belong
CN Segments Copy Number value obtained by the logR

Source

<https://portal.gdc.cancer.gov/projects/TCGA-BRCA>

%>% *Pipe operator*

Description

See `magrittr::%>%` for details.

Usage

lhs %>% rhs

Arguments

lhs A value or the magrittr placeholder.
rhs A function call using the magrittr semantics.

Value

The result of calling 'rhs(lhs)'.

Index

* **datasets**

TCGA_BRCA_CN_segments, [6](#)

* **internal**

%>%, [7](#)

%>%, [7](#), [7](#)

computeNormalChromosomes, [2](#)

DRrefit, [3](#)

DRrefit_plot, [3](#)

PlotChrCluster, [5](#)

Popeye, [6](#)

TCGA_BRCA_CN_segments, [6](#)