Package 'CopywriteR'

October 10, 2021

Type Package

Title Copy number information from targeted sequencing using off-target reads

Version 2.24.0

Date 2016-02-23

Author Thomas Kuilman

Maintainer Oscar Krijgsman <o.krijgsman@nki.nl>

Imports matrixStats, gtools, data.table, S4Vectors, chipseq, IRanges, Rsamtools, DNAcopy, GenomicAlignments, GenomicRanges, CopyhelpeR, GenomeInfoDb, futile.logger

Depends R(>= 3.2), BiocParallel

Suggests BiocStyle, SCLCBam, snow

URL https://github.com/PeeperLab/CopywriteR

Description CopywriteR extracts DNA copy number information from targeted sequencing by utiizing off-target reads. It allows for extracting uniformly distributed copy number information, can be used without reference, and can be applied to sequencing data obtained from various techniques including chromatin immunoprecipitation and target enrichment on small gene panels. Thereby, CopywriteR constitutes a widely applicable alternative to available copy number detection tools.

License GPL-2

biocViews ImmunoOncology, TargetedResequencing, ExomeSeq, CopyNumberVariation, Preprocessing, Visualization, Coverage

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

git_branch RELEASE_3_13

git_last_commit 5957876

git_last_commit_date 2021-05-19

Date/Publication 2021-10-10

R topics documented:

`CopywriteR` *CopywriteR: ENhanced COpy number Detection from Exome Reads*

Description

Generates DNA copy number profiles from targeted sequencing data using off-target reads

Usage

```
CopywriteR(sample.control, destination.folder, reference.folder, bp.param,
           capture.regions.file, keep.intermediary.files = FALSE)
```

Arguments

`sample.control`	a data.frame or matrix that contains the locations of sample and control BAM files, respectively, in columns. The file locations in a row represent a sample and its corresponding control, which is used for calling peaks.
`destination.folder`	the path to the folder to which output should be written. The path can be either absolute or relative.
`reference.folder`	the path to the folder with the helper files generated by `preCopywriteR()`. The helper files include the bin, mappability, GC-content, and blacklist, files in .bed format.
`bp.param`	a BiocParallelParam instance (see `BiocParallel` Bioconductor pacakage) that determines the settings used for parallel computing. Please refer to the vignette for more information.
`capture.regions.file`	optional; the path to the capture regions file, which should be in .bed format. Overlapping bait regions should be reduced into single regions. If included, statistics on the overlap of peaks called by MACS and the capture regions will be provided.
`keep.intermediary.files`	optional; logical that indicates whether intermediary .bam, .bai and peak regions files should be kept after the analysis is done. Defaults to FALSE.

Details

CopywriteR uses off-target sequence reads from targeted sequencing to create copy number profiles. First, it removes non-random off-target reads, and it subsequently calculates the depth of coverage for the bins that are provided in the helper files. It then performs GC-content and mappability corrections, and removes blacklisted regions. `plotCNA()` generates a DNA copy number profile from the output of the `CopywriteR()` function. Helper files required for CopywriteR analysis can be created using `preCopywriteR()`.

Value

`BamBaiPeaksFiles`
: a folder with the .bam, .bai and peak regions files that are created during the `CopywriteR()` run. The folder will only be created when the keep.intermediary.files argument is set to TRUE.

`input.Rdata`
: an R object that contains information for `plotCNA()`.

`log2_read_counts.igv`
: the file that contains the compensated corrected read counts after GC-content and mappability corrections, and after removal of data points in blacklisted regions. Counts are log2-transformed. The file is a tab-separated file formatted to be viewed in the IGV genome browser.

`CopywriteR.log`
: log file of CopywriteR.

`qc`
: a folder with quality control files. The .png files contain the plots and the loesses that are used for GC-content and mappability corrections. The fraction_of-bin .pdf files display the empirical cumulative distribution function for the fraction of bin (the bin size after removal of peak regions expressed as a fraction of the original size).

`read_counts.txt`
: the file that contains the raw and compensated read counts per bin.

Author(s)

Thomas Kuilman (t.kuilman@nki.nl)

References

CopywriteR: DNA copy number detection from off-target sequence data. Thomas Kuilman, Arno Velds, Kristel Kemper, Marco Ranzani, Lorenzo Bombardelli, Marlous Hoogstraat, Ekaterina Nevedomskaya, Guotai Xu, Julian de Ruiter, Martijn P. Lolkema, Bauke Ylstra, Jos Jonkers, Sven Rottenberg, Lodewyk F. Wessels, David J. Adams, Daniel S. Peeper, Oscar Krijgsman. Submitted for publication.

Examples

```
## Not run:
setwd("/PATH/TO/BAMFILES/")
samples <- list.files(pattern = ".bam$", full.names = TRUE)
## Use the first .bam file as a control for every sample
# controls <- samples[rep(1, length(samples))]
```

```
## Use every sample as its own control (i.e., peaks are called on sample itself)
controls <- samples
sample.control <- data.frame(samples, controls)

CopywriteR(sample.control = sample.control, destination.folder =
        "/PATH/TO/DESTINATIONFOLDER/", reference.folder =
        "/PATH/TO/REFERENCEFOLDER", ncpu = nrow(sample.control),
        capture.regions.file <- "/PATH/TO/CAPTUREREGIONSFILE")

## End(Not run)
```

---

plotCNA *CopywriteR: ENhanced COpy number Detection from Exome Reads*

Description

plotCNA analyses CopywriteR output using the segmentation algorithm CBS and creates both whole-genome and per-chromosome plots.

Usage

```
plotCNA(destination.folder, smoothed = TRUE, sample.plot, y.min, y.max, ...)
```

Arguments

`destination.folder`	the path to the output folder of CopywriteR.
`smoothed`	logical that determines whether DNAcopy smoothens copy number information before segmentation. Defaults to TRUE.
`sample.plot`	optional; a data.frame or matrix that contains the locations of sample and control BAM files. The column names should be "samples" and "controls", respectively. The file locations in a row represent a sample and its corresponding control, which is used for plotting of relative copy number values. Every sample is plotted both without a reference and with a reference. Defaults to sample.controls as used for running CopywriteR.
`y.min`	determines lower boundary of the plotting range.
`y.max`	determines upper boundary of the plotting range.
`...`	takes additional arguments that will be passed on to DNAcopy.

Value

`segment.Rdata`	an R object of the DNAcopy class, which contains the segmentation values for all analysed samples.
`plots`	a folder that contains all copy number profiles in portable document file (pdf) file format. Both profiles per chromosome, as well as a genome-wide profile are provided for every sample.

Author(s)

Thomas Kuilman (t.kuilman@nki.nl)

References

CopywriteR: DNA copy number detection from off-target sequence data. Thomas Kuilman, Arno Velds, Kristel Kemper, Marco Ranzani, Lorenzo Bombardelli, Marlous Hoogstraat, Ekaterina Nevedomskaya, Guotai Xu, Julian de Ruiter, Martijn P. Lolkema, Bauke Ylstra, Jos Jonkers, Sven Rottenberg, Lodewyk F. Wessels, David J. Adams, Daniel S. Peeper, Oscar Krijgsman. Submitted for publication.

See Also

`CopywriteR()`

Examples

```
## Not run:
setwd("/PATH/TO/BAMFILES/")
samples <- list.files(pattern = ".bam$", full.names = TRUE)
## Use the first .bam file as a reference for plotting every sample
controls <- samples[rep(1, length(samples))]
sample.plot <- data.frame(samples, controls)

plotCNA("./PATH/TO/DESTINATIONFOLDER/", sample.plot = sample.plot, y.min = -3,
        y.max = 3)

## End(Not run)
```

`preCopywriteR` *CopywriteR: DNA copy number detection from off-target sequence data*

Description

preCopywriteR is used to generate helper files (blacklist, bin region, GC-content, and mappability .bed files) for the desired bin size from pre-assembled 1 kb helper files. These binary 1 kb helper files are available at https://github.com/PeeperLab/CopywriteR for hg18, hg19, hg38, mm9 and mm10 reference genomes.

Usage

```
preCopywriteR(output.folder, bin.size, ref.genome, prefix = "")
```

Arguments

output.folder	the path to the output folder.
bin.size	desired bin length (in bp) for which helper files should be generated.
ref.genome	the reference genome for which helper files should be generated, e.g. hg18, hg19, hg38, mm9 or mm10.
prefix	the prefix that is used for chromosome notation. Standard notation is "1", "2", ..., "X", "Y". For "chr1", ... notation, use prefix = "chr".

Details

Currently helper files can be generated for human (hg18, hg19 and hg38) and mouse (mm9 and mm10) reference genomes.

Helper files can only be generated for a bin.size that is a multiple of 1000 bp.

Value

blacklist.rda	an R data file containing a GRanges object with blacklisted regions of known CNVs.
GC_mappability.rda	an R data file containing a GRanges object with the mappability and GC-content for bins of the specified size.

Author(s)

Oscar krijgsman (o.krijgsman@nki.nl)
Thomas Kuilman (t.kuilman@nki.nl)

References

CopywriteR: DNA copy number detection from off-target sequence data. Thomas Kuilman, Arno Velds, Kristel Kemper, Marco Ranzani, Lorenzo Bombardelli, Marlous Hoogstraat, Ekaterina Nevedomskaya, Guotai Xu, Julian de Ruiter, Martijn P. Lolkema, Bauke Ylstra, Jos Jonkers, Sven Rottenberg, Lodewyk F. Wessels, David J. Adams, Daniel S. Peeper, Oscar Krijgsman. Submitted for publication.

Examples

```
## Not run:
preCopywriteR("/PATH/TO/HG19_1KB_FOLDER/", "./", 20000, "hg19")

## End(Not run)
```

Index