Package ‘cfTools’

October 24, 2025

Type Package

Title Informatics Tools for Cell-Free DNA Study

Version 1.9.0

Description The cfTools R package provides methods for cell-free DNA (cfDNA) methylation data analysis to facilitate cfDNA-based studies. Given the methylation sequencing data of a cfDNA sample, for each cancer marker or tissue marker, we deconvolve the tumor-derived or tissue-specific reads from all reads falling in the marker region. Our read-based deconvolution algorithm exploits the pervasiveness of DNA methylation for signal enhancement, therefore can sensitively identify a trace amount of tumor-specific or tissue-specific cfDNA in plasma. cfTools provides functions for (1) cancer detection: sensitively detect tumor-derived cfDNA and estimate the tumor-derived cfDNA fraction (tumor burden); (2) tissue deconvolution: infer the tissue type composition and the cfDNA fraction of multiple tissue types for a plasma cfDNA sample. These functions can serve as foundations for more advanced cfDNA-based studies, including cancer diagnosis and disease monitoring.

License file LICENSE

Encoding UTF-8

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

Config/testthat/edition 3

RoxygenNote 7.2.3

Imports Rcpp, utils, GenomicRanges, basilisk, R.utils, stats, cfToolsData, grDevices, graphics

StagedInstall no

biocViews Software, BiomedicalInformatics, Epigenetics, Sequencing, MethylSeq, DNAMethylation, DifferentialMethylation

VignetteBuilder knitr

LinkingTo Rcpp, BH

URL https://github.com/jasminezhoulab/cfTools

BugReports https://github.com/jasminezhoulab/cfTools/issues

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

git_branch devel

git_last_commit 970eb2d

git_last_commit_date 2025-07-15

Repository Bioconductor 3.23

Date/Publication 2025-10-24

Author Ran Hu [aut, cre] (ORCID: <https://orcid.org/0000-0002-0563-8957>),
Mary Louisa Stackpole [aut] (ORCID:
<https://orcid.org/0000-0003-4432-6718>),
Shuo Li [aut] (ORCID: <https://orcid.org/0000-0002-1960-6016>),
Xianghong Jasmine Zhou [aut] (ORCID:
<https://orcid.org/0000-0002-4522-7490>),
Wenyuan Li [aut] (ORCID: <https://orcid.org/0000-0002-5029-8525>)

Maintainer Ran Hu <huran@ucla.edu>

Contents

`beta_matrix` *Beta value matrix*

Description

A list of methylation levels (e.g., beta values), where each row is a sample and each column is a marker

Usage

```
data("beta_matrix")
```

Format

A tibble with 20 rows and 3 variables

marker1 Beta values of marker1 for all samples

marker2 Beta values of marker2 for all samples

marker3 Beta values of marker3 for all samples

Value

A tibble with 20 rows and 3 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`CancerDetector` *Cancer Detector*

Description

Detect tumor-derived cfDNA and estimate the tumor burden.

Usage

```
CancerDetector(
  readsBinningFile,
  tissueMarkersFile,
  lambda = 0.5,
  id = "sample"
)
```

Arguments

`readsBinningFile`	a file of the fragment-level methylation states of reads that mapped to the markers.
`tissueMarkersFile`	a file of paired shape parameters of beta distributions for markers.
`lambda`	a number controlling "confounding" markers' distance from average markers.
`id`	the sample ID.

Value

a list containing the cfDNA tumor burden and the normal cfDNA fraction.

Examples

```
## input files
demo.dir <- system.file("data", package="cfTools")
readsBinningFile <- file.path(demo.dir, "CancerDetector.reads.txt.gz")
tissueMarkersFile <- file.path(demo.dir, "CancerDetector.markers.txt.gz")
lambda <- 0.5
id <- "test"

CancerDetector(readsBinningFile, tissueMarkersFile, lambda, id)
```

`CancerDetector.markers`

Cancer-specific marker parameter

Description

The paired shape parameters of beta distributions for cancer-specific markers

Usage

```
data("CancerDetector.markers")
```

Format

A tibble with 1266 rows and 3 variables

markerName Name of the marker

tumor Paired beta distribution shape parameters for tumor samples

normalPlasma Paired beta distribution shape parameters for normal plasma samples

Value

A tibble with 1266 rows and 3 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`CancerDetector.reads` *Fragment-level methylation state for cancer detection*

Description

The fragment-level methylation states of reads that mapped to the cancer-specific markers

Usage

```
data("CancerDetector.reads")
```

Format

A tibble with 9991 rows and 2 variables

markerName Name of the marker

methState Fragment-level methylation states, which are represented by a sequence of binary values (0 represents unmethylated CpG and 1 represents methylated CpG on the same fragment)

Value

A tibble with 9991 rows and 2 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`cfDeconvolve` *cfDNA methylation read deconvolution*

Description

Infer the tissue-type composition of plasma cfDNA.

Usage

```
cfDeconvolve(
  readsBinningFile,
  tissueMarkersFile,
  numTissues,
  emAlgorithmType = "em.global.unknown",
  likelihoodRatioThreshold = 2,
  emMaxIterations = 100,
  randomSeed = 0,
  id = "sample"
)
```

Arguments

`readsBinningFile`	a file of the fragment-level methylation states of reads that mapped to the markers. Either in plain text or compressed form.
`tissueMarkersFile`	a file of paired shape parameters of beta distributions for markers.
`numTissues`	a number of tissue types.
`emAlgorithmType`	a read-based tissue deconvolution EM algorithm type: em.global.unknown (default), em.global.known, em.local.unknown, em.local.known.
`likelihoodRatioThreshold`	a positive float number. Default is 2.
`emMaxIterations`	a number of EM algorithm maximum iteration. Default is 100.
`randomSeed`	a random seed that initialize the EM algorithm. Default is 0.
`id`	the sample ID.

Value

a list containing the cfDNA fractions of different tissue types and an unknown class.

Examples

```
## input files
demo.dir <- system.file("data", package="cfTools")
readsBinningFile <- file.path(demo.dir, "cfDeconvolve.reads.txt.gz")
tissueMarkersFile <- file.path(demo.dir, "cfDeconvolve.markers.txt.gz")
numTissues <- 7
emAlgorithmType <- "em.global.unknown"
likelihoodRatioThreshold <- 2
emMaxIterations <- 100
randomSeed <- 0
id <- "test"

cfDeconvolve(readsBinningFile, tissueMarkersFile, numTissues,
emAlgorithmType, likelihoodRatioThreshold, emMaxIterations, randomSeed, id)
```

`cfDeconvolve.markers` *Tissue-specific marker parameter*

Description

The paired shape parameters of beta distributions for tissue-specific markers

Usage

```
data("cfDeconvolve.markers")
```

Format

A tibble with 10 rows and 8 variables

markerName Name of the marker

tissue1 Paired beta distribution shape parameters for tissue1 samples

tissue2 Paired beta distribution shape parameters for tissue2 samples

tissue3 Paired beta distribution shape parameters for tissue3 samples

tissue4 Paired beta distribution shape parameters for tissue4 samples

tissue5 Paired beta distribution shape parameters for tissue5 samples

tissue6 Paired beta distribution shape parameters for tissue6 samples

tissue7 Paired beta distribution shape parameters for tissue7 samples

Value

A tibble with 10 rows and 8 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`cfDeconvolve.reads` *Fragment-level methylation state for tissue deconvolution*

Description

The fragment-level methylation states of reads that mapped to the tissue-specific markers

Usage

```
data("cfDeconvolve.reads")
```

Format

A tibble with 942 rows and 2 variables

markerName Name of the marker

methState Fragment-level methylation states, which are represented by a sequence of binary values (0 represents unmethylated CpG and 1 represents methylated CpG on the same fragment)

Value

A tibble with 942 rows and 2 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`cfSort` *cfSort: tissue deconvolution*

Description

Tissue deconvolution in cfDNA using DNN models.

Usage

```
cfSort(readsBinningFile, id = "sample")
```

Arguments

`readsBinningFile`	a file of the fragment-level methylation states of reads that mapped to the cfSort markers. In compressed form.
`id`	the sample ID.

Value

the tissue composition of the cfDNA sample.

Examples

```
## input files
demo.dir <- system.file("data", package="cfTools")
readsBinningFile <- file.path(demo.dir, "cfsort_reads.txt.gz")
id <- "test"

cfSort(readsBinningFile, id)
```

`cfsort_markers` *cfSort markers*

Description

Marker information for the cfSort function, where each row is the information about a marker

Usage

```
data("cfsort_markers")
```

Format

A tibble with 51035 rows and 4 variables

marker_index The marker index used in cfSort method

alpha_threshold The alpha threshold for each marker

pair The pair of tissues used for identifying the marker

group The group number for each marker

Value

A tibble with 51035 rows and 4 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`cfsort_reads` *Fragment-level methylation state for cfSort tissue deconvolution*

Description

The fragment-level methylation states of reads that mapped to the cfSort markers

Usage

```
data("cfsort_reads")
```

Format

A tibble with 99999 rows and 2 variables

markerName Name of the cfSort marker

methState Fragment-level methylation states, which are represented by a sequence of binary values (0 represents unmethylated CpG and 1 represents methylated CpG on the same fragment)

Value

A tibble with 99999 rows and 2 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`cfTools` *cfTools: a versatile package for analyzing cell-free DNA data*

Description

Given the methylation sequencing data of a cell-free DNA (cfDNA) sample, for each cancer marker or tissue marker, we deconvolve the tumor-derived or tissue-specific reads from all reads falling in the marker region. Our read-based deconvolution algorithm exploits the pervasiveness of DNA methylation for signal enhancement, therefore can sensitively identify a trace amount of tumor-specific or tissue-specific cfDNA in plasma.

Details

Specifically, cfTools can deconvolve different sources of cfDNA fragments (or reads) in two contexts:

1. Cancer detection: separate cfDNA fragments into tumor-derived fragments and background normal fragments (2 classes), and estimate the tumor-derived cfDNA fraction.

2. Tissue deconvolution: separate cfDNA fragments from different tissues (> 2 classes), and estimate the cfDNA fraction of different tissue types (including an unknown type) for a plasma cfDNA sample.

These functions can serve as foundations for more advanced cfDNA-based studies, including cancer diagnosis and disease monitoring.

For an overview of the functionality provided by the package, please see the vignette: `vignette(package="cfTools")`

Author(s)

Ran Hu <`huran@ucla.edu`>, Mary Louisa Stackpole, Shuo Li, Xianghong Jasmine Zhou <`XJZhou@mednet.ucla.edu`>, Wenyuan Li <`WenyuanLi@mednet.ucla.edu`>

See Also

`CancerDetector`, `cfDeconvolve`, `cfSort`, `MergeCpGs`, `MergePEReads`, `GenerateFragMeth`, `GenerateMarkerParam`, `PlotFractionPie`

`CpG_OB_demo` *Methylation information for CpG on the original bottom strand (OB)*

Description

Methylation information for CpG on the original bottom strand (OB), which is one of the outputs from 'bismark methylation extractor'

Usage

```
data("CpG_OB_demo")
```

Format

A tibble with 2224 rows and 5 variables

sequence ID ID of the sequence

methylation state Methylated or unmethylated CpG site

chromosome name Chromosome name

chromosome start Chromosome start position

methylation call Methylation call

Value

A tibble with 2224 rows and 5 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`CpG_OT_demo` *Methylation information for CpG on the original top strand (OT)*

Description

Methylation information for CpG on the original top strand (OT), which is one of the outputs from ’bismark methylation extractor’

Usage

`data("CpG_OT_demo")`

Format

A tibble with 2556 rows and 5 variables

sequence ID ID of the sequence

methylation state Methylated or unmethylated CpG site

chromosome name Chromosome name

chromosome start Chromosome start position

methylation call Methylation call

Value

A tibble with 2556 rows and 5 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`demo.fragment_level.meth.bed` *Fragment-level methylation information*

Description

A BED file of fragment-level methylation information

Usage

```
data("demo.fragment_level.meth.bed")
```

Format

A tibble with 552 rows and 9 variables

chr Chromosome

start Chromosome start

end Chromosome end

name ID of the sequence

fragmentLength Fragment length

strand Strand

cpgNumber Number of CpG sites on the fragment

cpgPosition Postions of CpG sites on the fragment

methState A string of methylation states of CpG sites on the fragment

Value

A tibble with 552 rows and 9 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`demo.refo_frag.bed` *Fragment-level information*

Description

A BED file of fragment-level information

Usage

```
data("demo.refo_frag.bed")
```

Format

A tibble with 559 rows and 6 variables

chr Chromosome

start Chromosome start

end Chromosome end

fragmentLength Fragment length

strand Strand

name ID of the sequence

Value

A tibble with 559 rows and 6 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`demo.refo_meth.bed` *Methylation information on fragments*

Description

A BED file of methylation information on fragments

Usage

```
data("demo.refo_meth.bed")
```

Format

A tibble with 552 rows and 8 variables

chr Chromosome
cpgStart Start postion of first CpG on the fragment
cpgEnd End postion of first CpG on the fragment
strand Strand
cpgNumber Number of CpG sites on the fragment
cpgPosition Postions of CpG sites on the fragment
methState A string of methylation states of CpG sites on the fragment
name ID of the sequence

Value

A tibble with 552 rows and 8 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`demo.sorted.bed` *Paired-end sequencing reads*

Description

Paired-end sequencing reads information

Usage

```
data("demo.sorted.bed")
```

Format

A tibble with 1117 rows and 6 variables

chr Chromosome name
start Chromosome start
end Chromosome end
name Sequence ID
score Mapping quality score
strand Strand

Value

A tibble with 1117 rows and 6 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`GenerateFragMeth` *Generate fragment-level information about methylation states*

Description

Join two lists containing the fragment information and the methylation states on each fragment into one list.

Usage

```
GenerateFragMeth(frag_bed, meth_bed, output.dir = "", id = "")
```

Arguments

`frag_bed`	a BED file containing information for every fragment, which is the output of MergePEReads().
`meth_bed`	a BED file containing methylation states on every fragment, which is the output of MergeCpGs().
`output.dir`	a path to the output directory. Default is "", which means the output will not be written into a file.
`id`	an ID name for the input data. Default is "", which means the output will not be written into a file.

Value

a list in BED file format and/or written to an output BED file.

Examples

```
## input files
demo.dir <- system.file("data", package="cfTools")
frag_bed <- read.delim(file.path(demo.dir, "demo.refo_frag.bed.txt.gz"),
colClasses = "character")
meth_bed <- read.delim(file.path(demo.dir, "demo.refo_meth.bed.txt.gz"),
colClasses = "character")

output <- GenerateFragMeth(frag_bed, meth_bed)
```

`GenerateMarkerParam` *Generate the methylation pattern of markers*

Description

Output paired shape parameters of beta distributions for methylation markers.

Usage

```
GenerateMarkerParam(x, sample.types, marker.names, output.file = "")
```

Arguments

`x`	a list of methylation levels (e.g., beta values), where each row is a sample and each column is a marker.
`sample.types`	a vector of sample types (e.g., tumor or normal, tissue types) corresponding to the rows of the list.
`marker.names`	a vector of marker names corresponding to the columns of the list.
`output.file`	a character string naming the output file. Default is "", which means the output will not be written into a file.

Value

a list containing the paired shape parameters of beta distributions for markers and/or written to an output file.

Examples

```
## input files
demo.dir <- system.file("data", package="cfTools")
methLevel <- read.table(file.path(demo.dir, "beta_matrix.txt.gz"),
row.names=1, header = TRUE)
sampleTypes <- read.table(file.path(demo.dir, "sample_type.txt.gz"),
row.names=1, header = TRUE)$sampleType
markerNames <- read.table(file.path(demo.dir, "marker_index.txt.gz"),
row.names=1, header = TRUE)$markerIndex

output <- GenerateMarkerParam(methLevel, sampleTypes, markerNames)
```

`markers.bed` *Genomic postions of markers*

Description

A BED file of genomic regions of markers

Usage

```
data("markers.bed")
```

Format

A tibble with 3 rows and 4 variables

chr Chromosome

start Chromosome start

end Chromosome end

markerName Marker name

Value

A tibble with 3 rows and 4 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

`marker_index` *Marker name*

Description

A vector of marker names corresponding to the columns of the list of methylation levels.

Usage

```
data("marker_index")
```

Format

A tibble with 3 rows and 1 variables

markerIndex Marker name

Value

A tibble with 3 rows and 1 variables

Author(s)

Ran Hu <`huran@ucla.edu`>

`MergeCpGs` *Generate fragment-level methylation states of CpGs*

Description

Merge the methylation states of all CpGs corresponding to the same fragment onto one line in output.

Usage

```
MergeCpGs(CpG_OT, CpG_OB, output.dir = "", id = "")
```

Arguments

`CpG_OT`	a file of methylation information for CpG on the original top strand (OT), which is one of the outputs from 'bismark methylation extractor'.
`CpG_OB`	a file of methylation information for CpG on the original bottom strand (OB), which is one of the outputs from 'bismark methylation extractor'.
`output.dir`	a path to the output directory. Default is "", which means the output will not be written into a file.
`id`	an ID name for the input data. Default is "", which means the output will not be written into a file.

Value

a list in BED file format and/or written to an output BED file.

Examples

```
## input files
demo.dir <- system.file("data", package="cfTools")
CpG_OT <- file.path(demo.dir, "CpG_OT_demo.txt.gz")
CpG_OB <- file.path(demo.dir, "CpG_OB_demo.txt.gz")

output <- MergeCpGs(CpG_OT, CpG_OB)
```

`MergePEReads` *Generate fragment-level information for paired-end sequencing reads*

Description

Merge BED file (the output of ‘bedtools bamtobed‘) to fragment-level for paired-end sequencing reads.

Usage

```
MergePEReads(bed_file, output.dir = "", id = "")
```

Arguments

`bed_file`	a (sorted) BED file of paired-end reads.
`output.dir`	a path to the output directory. Default is "", which means the output will not be written into a file.
`id`	an ID name for the input data. Default is "", which means the output will not be written into a file.

Value

a list in BED file format and/or written to an output BED file.

Examples

```
## input files
demo.dir <- system.file("data", package="cfTools")
PEReads <- file.path(demo.dir, "demo.sorted.bed.txt.gz")

output <- MergePEReads(PEReads)
```

`PlotFractionPie` *Plot Pie Chart*

Description

Generate a pie chart for a vector of class fractions (e.g., tissue composition or cfDNA fractions). Automatically filters small values into an "Other" group, and allows for custom colors and font size control.

Usage

```
PlotFractionPie(
  fraction_vector,
  title = "Composition",
  threshold = 0.01,
  class_colors = NULL,
  font_size = 1
)
```

Arguments

`fraction_vector`	a named numeric vector or one-row data.frame, where each value represents a class proportion.
`title`	the title of the plot.
`threshold`	a numeric value. Classes with fraction values below this threshold will be grouped into "Other".
`class_colors`	a named character vector assigning colors to specific class names (e.g., c("tumor" = "red")).
`font_size`	numeric, font scaling factor (default is 1.0).

Value

A pie chart is plotted to the current device.

Examples

```
df <- data.frame(
  WBC = 0.93,
  Liver = 0.04,
  Lung = 0.02,
  Muscle = 1.2345e-4,
  Stomach = 9.87655e-03
)
PlotFractionPie(df, title = "cfDNA Composition", font_size = 1.2)
```

`sample_type` *Sample type*

Description

A vector of sample types (e.g., tumor or normal, tissue types) corresponding to the rows of the list of methylation levels.

Usage

```
data("sample_type")
```

Format

A tibble with 20 rows and 1 variables

sampleType Sample type

Value

A tibble with 20 rows and 1 variables

Author(s)

Ran Hu `<huran@ucla.edu>`

Index