Package: BEAT (via r-universe)

November 6, 2026

Type Package

Version 1.50.0

Title BEAT - BS-Seq Epimutation Analysis Toolkit

Date 2014-02-14

Maintainer Kemal Akman <akman@mpipz.mpg.de>

Author Kemal Akman <akman@mpipz.mpg.de>

Depends R (>= 2.13.0)

Imports GenomicRanges, ShortRead, Biostrings, BSgenome

Description Model-based analysis of single-cell methylation data

License LGPL (>= 3.0)

biocViews ImmunoOncology, Genetics, MethylSeq, Software, DNAMethylation, Epigenetics

Config/pak/sysreqs make libbz2-dev libjpeg-dev liblzma-dev libpng-dev libxml2-dev libssl-dev xz-utils zlib1g-dev

Repository Bioconductor 3.23

Date/Publication 2026-04-28 12:38:33 UTC

RemoteUrl https://github.com/bioc/BEAT

RemoteRef RELEASE_3_23

RemoteSha bab1e9e06196584a36b42733f43e3f08d98022b8

Contents

BEAT 2
epimutation_calls 2
generate_results 4
makeParams 5
positions 6
positions.reference 6
positions_to_regions 7

Index **9**

`BEAT` *BEAT - BS-Seq Epimutation Analysis Toolkit*

Description

Modelling, data preparation and analysis of BS-Seq derived, region-based epimutation data

Details

Package:	BEAT - Rpackage
Type:	Package
Version:	0.99.1
Date:	2013-01-10
License:	LGPL version 3 or later
Maintainer:	Kemal Akman, akmank@mpipz.mpg.de

`epimutation_calls` *Returns epimutation rates and sites.*

Description

Returns epimutation rates per genome and per genomic feature, as well as individual genomic sites at which epimutations were called.

Usage

```
epimutation_calls(params, outputPath = getwd())
```

Arguments

`params`	BEAT parameter object.
`outputPath`	Path to which output files will be written, the default is the current working directory.

Format

Necessary function arguments are passed via a BEAT parameter object, which includes working path, sample names, reference sample name, model parameters and region sizes.

params Parameter object created by calling `makeParams`.

Value

The function `epimutation_calls` returns :

`resultsList` A list is returned consisting of the two data.frames methSites and demethSites. methSites contains all regions at which methylating epimutations were observed, while demethSites contains all sites at which demethylating epimutations were observed. Each data.frame describes the genomic regions covered by the given sample and the reference sample using the columns: 'chr' (chromosome), 'start' (starting position), 'stop' (last position), 'meth' (methylated counts), 'unmeth' (unmethylatd counts) and 'epimutation_call_test' (epimutation call, 1 for methylating epimutation and -1 for demethylating epimutation).

`methEpicalls` For each single-cell sample, a methEpicalls.RData object is saved in the working directory, which is a data.frame of all sites at which methylating epimutations were observed, consisting of the columns: chr, pos, endpos, meth, unmeth and methstate.

`demethEpicalls` For each single-cell sample, a demethEpicalls.RData object is saved in the working directory, which is a data.frame of all sites at which demethylating epimutations were observed, consisting of the columns: chr, pos, endpos, meth, unmeth and methstate.

Author(s)

Kemal Akman <akmank@mpipz.mpg.de>

See Also

See also `makeParams`.

Examples

```
# Local working directory
localpath <- system.file('extdata', package = 'BEAT')
# Names of samples, expected filenames are e.g. reference.positions.csv
sampNames <- c("reference", "sample")
# Empirical BS-conversion rates, e.g. estimated from non-CpG methylation
convrates <- c(0.8,0.5)
# Vector denoting reference vs. single-cell status of given samples
is.reference <- c(TRUE,FALSE)
params <- makeParams(localpath, sampNames, convrates, is.reference, pminus = 0.2, regionSize = 10000, min
# pool CG positions into regions
positions_to_regions(params)
# compute model statistics
generate_results(params)
# call epimutations
methDemethPlusMinus <- epimutation_calls(params)
```

`generate_results` *Computes model methylation states for genomic region counts of a list of samples.*

Description

Computes model methylation states for genomic region counts of a list of samples.

Usage

```
generate_results(params, outputPath = getwd())
```

Arguments

`params`	BEAT parameter object.
`outputPath`	Path to which output files will be written, the default is the current working directory.

Format

Necessary function arguments are passed via a BEAT parameter object, which includes working path, sample names, reference sample name, model parameters and region sizes.

params Parameter object created by calling `makeParams`.

Author(s)

Kemal Akman <akmank@mpipz.mpg.de>

See Also

See also `makeParams`.

Examples

```
# Local working directory
localpath <- system.file('extdata', package = 'BEAT')
# Names of samples, expected filenames are e.g. reference.positions.csv
sampNames <- c("reference", "sample")
# Empirical BS-conversion rates, e.g. estimated from non-CpG methylation
convrates <- c(0.8,0.5)
# Vector denoting reference vs. single-cell status of given samples
is.reference <- c(TRUE,FALSE)
params <- makeParams(localpath, sampNames, convrates, is.reference, pminus = 0.2, regionSize = 10000, min
# pool CG positions into regions
positions_to_regions(params)
# compute model statistics
generate_results(params)
```

`makeParams` *Creates a parameter object of arguments to be used with other BEAT functions.*

Description

Creates a parameter object of arguments to be used with other BEAT functions.

Usage

```
makeParams(localpath = getwd(), sampNames, convrates, is.reference, pminus = 0.2, regionSize =
```

Arguments

`localpath`	Full path to working directory from which files are read and where results are saved.
`sampNames`	Vector of sample names to be analyzed.
`convrates`	Vector of empirically determined bisulfite conversion rates per sample. Determines p+, the model parameter for incomplete conversion (false negative rates).
`is.reference`	Vector of reference (TRUE) vs. single-cell (FALSE) status per sample.
`pminus`	Model parameter for false conversion (false positive rate).
`regionSize`	Region size in nucleotides into which genomic sites are grouped.
`minCounts`	Minimum counts necessary for each region to be included in epimutation modeling and analysis.
`verbose`	Shows more verbose console output during computation steps.
`computeRegions`	If set to TRUE, regions will be recomputed from individual positions and saved as cpgregions.RData objects for each sample.
`computeMatrices`	If set to TRUE, model parameters will be recomputed and saved as results.RData objects for each sample.
`writeEpicallMatrix`	If set to TRUE, epimutation calls will be written as RData object.

Value

The function `makeParams` returns :

`params`	Parameter object to be used in other BEAT functions.

Author(s)

Kemal Akman <akmank@mpipz.mpg.de>

Examples

```
# Local working directory
localpath <- system.file('extdata', package = 'BEAT')
# Names of samples, expected filenames are e.g. reference.positions.csv
sampNames <- c("reference", "sample")
# Empirical BS-conversion rates, e.g. estimated from non-CpG methylation
convrates <- c(0.8,0.5)
# Vector denoting reference vs. single-cell status of given samples
is.reference <- c(TRUE,FALSE)
params <- makeParams(localpath, sampNames, convrates, is.reference, pminus = 0.2, regionSize = 10000, min
# Example usage of the params object
positions_to_regions(params)
```

---

positions *Sample dataset of CpG positions for a single cell BS-seq sample*

Description

Sample dataset of CpG positions for a single cell sequencing sample

Author(s)

Kemal Akman <akmank@mpipz.mpg.de>

---

positions.reference *Sample dataset of CpG positions for a reference BS-Seq sample*

Description

Sample dataset of CpG positions for a single cell sequencing sample

Author(s)

Kemal Akman <akmank@mpipz.mpg.de>

positions_to_regions *Converts methylation counts for single genomic positions to counts for genomic regions*

Description

Converts methylation counts of a data.frame of single genomic positions into a data.frame of counts for genomic regions.

Usage

```
positions_to_regions(params, outputPath = getwd())
```

Arguments

params	BEAT parameter object.
outputPath	Path to which output files will be written, the default is the current working directory.

Format

Necessary function arguments are passed via a BEAT parameter object, which includes working path, sample names, reference sample name, model parameters and region sizes.

params Parameter object created by calling makeParams.

sample.positions.csv For each sample referenced by the params argument sampleNames, a corresponding csv must be present in the working directory. The csv input must contain one row per genomic position for each CG site. Columns must be chr, pos, meth, unmeth (chromosome, position, methylated counts, unmethylated counts).

Author(s)

Kemal Akman <akmank@mpipz.mpg.de>

See Also

See also makeParams.

Examples

```
# Local working directory
localpath <- system.file('extdata', package = 'BEAT')
# Names of samples, expected filenames are e.g. reference.positions.csv
sampNames <- c("reference", "sample")
# Empirical BS-conversion rates, e.g. estimated from non-CpG methylation
convrates <- c(0.8,0.5)
# Vector denoting reference vs. single-cell status of given samples
is.reference <- c(TRUE,FALSE)
params <- makeParams(localpath, sampNames, convrates, is.reference, pminus = 0.2, regionSize = 10000, min
```

```
# Pool CG positions into regions
positions_to_regions(params)
```

Index

∗ **BEAT**
epimutation_calls, 2
generate_results, 4
makeParams, 5
positions, 6
positions.reference, 6
positions_to_regions, 7
∗ **package**
BEAT, 2

BEAT, 2
BEAT-package *(BEAT)*, 2

epimutation_calls, 2

generate_results, 4

makeParams, *2–4*, 5, 7

positions, 6
positions.reference, 6
positions_to_regions, 7