Package 'RBedMethyl'

May 5, 2026

Type Package

Title Disk-backed Representation of ONT bedMethyl Files

Version 1.0.0

Description Bioconductor-native infrastructure for handling large nanoporetech modkit bedMethyl pileup files from ONT data using HDF5Array and DelayedArray.

URL https://github.com/CMG-UA/RBedMethyl

BugReports https://github.com/CMG-UA/RBedMethyl/issues

License GPL (>= 2)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

VignetteBuilder knitr

biocViews DNAMethylation, DifferentialMethylation, Epigenetics, Infrastructure, DataImport, Software

LazyData false

Imports methods, HDF5Array, rhdf5, DelayedArray, DelayedMatrixStats, SummarizedExperiment, bsseq, GenomicRanges, S4Vectors, IRanges, data.table

Suggests BiocStyle, knitr, rmarkdown, testthat

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

git_branch RELEASE_3_23

git_last_commit 08623bb

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-05-04

Author Vasileios Lemonidis [aut, cre, cph] (ORCID: <https://orcid.org/0000-0002-6446-3536>), Center for Oncological Research, University of Antwerp [cph, fnd], Stichting Tegen Kanker [fnd]

Maintainer Vasileios Lemonidis <vasileios.lemonidis@uantwerpen.be>

Contents

bedMethylFields 2
beta,RBedMethyl,missing-method 2
filterByCoverage 3
RBedMethyl-class 4
readBedMethyl 4
subsetBy 5
subsetByChromosomes 6
subsetByRegion 6
subsetByRegion,RBedMethyl,GRanges,missing,missing-method 7
summarizeByRegion 8
[,RBedMethyl,missing,missing,missing-method 8

Index **10**

`bedMethylFields` *List retrievable bedMethyl fields*

Description

Returns a data.frame describing retrievable bedMethyl fields and their types.

Usage

```
bedMethylFields()
```

Value

A `data.frame` with columns `field`, `type`, and `description`.

Examples

```
bedMethylFields()
```

`beta,RBedMethyl,missing-method` *Per-site methylation fraction*

Description

Compute per-site methylation fraction for an `RBedMethyl` object. Requires the `mod_reads` assay to be loaded.

Usage

```
## S4 method for signature 'RBedMethyl,missing'
beta(a, b)
```

Arguments

`a`	An `RBedMethyl` object.
`b`	Unused, kept for `base::beta` compatibility.

Value

Numeric vector of per-site methylation fractions.

`filterByCoverage` *Filter by coverage*

Description

Filter an `RBedMethyl` object by minimum coverage.

Usage

```
filterByCoverage(x, min_cov)
```

Arguments

`x`	An `RBedMethyl` object.
`min_cov`	Minimum coverage threshold.

Value

A filtered `RBedMethyl` object.

Examples

```
lines <- c(
  paste("chr1", 0, 1, "m", 0, "+", 0, 1, 0, 10, 0.5, 5, 5, 0, 0, 0, 0, 0, sep = "\t"),
  paste("chr1", 10, 11, "m", 0, "+", 10, 11, 0, 20, 0.25, 5, 15, 0, 0, 0, 0, 0, sep = "\t")
)
tmp <- tempfile(fileext = ".bed")
writeLines(lines, tmp)
bm <- readBedMethyl(tmp, mod = "m", fields = c("coverage", "pct", "mod_reads"))
bm2 <- filterByCoverage(bm, min_cov = 15)
length(RBedMethyl::beta(bm2))
```

`RBedMethyl-class` *RBedMethyl class*

Description

Disk-backed representation of nanoporetech modkit bedMethyl data from ONT sequencing.

Slots

`assays` A `SimpleList` of assay arrays.

`chrom_levels` Character vector of chromosome names.

`strand_levels` Character vector of strand levels.

`chr_index` Matrix of chromosome row ranges (start/end).

`index` Integer vector of active row indices.

`mod` Modification code.

`readBedMethyl` *Read an ONT modkit bedMethyl file*

Description

Create an `RBedMethyl` object backed by HDF5Array from a nanoporetech modkit bedMethyl file (headerless).

Usage

```
readBedMethyl(
  bedmethyl,
  mod = "m",
  chunk_size = 5e+06,
  h5file = NULL,
  check_sorted = TRUE,
  fields = c("coverage", "mod_reads")
)
```

Arguments

`bedmethyl`	Path to a nanoporetech modkit bedMethyl file (optionally gzipped).
`mod`	Modification code to retain (`"m"` or `"h"`).
`chunk_size`	Reserved for future use.
`h5file`	Path to the HDF5 file to create. Defaults to a deterministic path in `tempdir()` derived from the input `bedmethyl` filename, so subsequent calls reuse the same file.
`check_sorted`	Logical, check that records are sorted by chrom and chromStart.
`fields`	Character vector of numeric fields to load. Defaults to `c("coverage", "mod_reads")`.

Value

An `RBedMethyl` object.

Examples

```
lines <- c(
  paste("chr1", 0, 1, "m", 0, "+", 0, 1, 0, 10, 0.5, 5, 5, 0, 0, 0, 0, 0, sep = "\t"),
  paste("chr1", 10, 11, "m", 0, "+", 10, 11, 0, 20, 0.25, 5, 15, 0, 0, 0, 0, 0, sep = "\t")
)
tmp <- tempfile(fileext = ".bed")
writeLines(lines, tmp)
bm <- readBedMethyl(tmp, mod = "m", fields = c("coverage", "pct", "mod_reads"))
bm
```

`subsetBy` *Subset by assay predicate*

Description

Subset an `RBedMethyl` object using a predicate over an assay.

Usage

```
subsetBy(x, column, FUN)
```

Arguments

`x`	An `RBedMethyl` object.
`column`	Assay name to filter on (must be loaded).
`FUN`	Predicate function returning a logical vector.

Value

A filtered `RBedMethyl` object.

Examples

```
lines <- c(
  paste("chr1", 0, 1, "m", 0, "+", 0, 1, 0, 10, 0.5, 5, 5, 0, 0, 0, 0, 0, sep = "\t"),
  paste("chr1", 10, 11, "m", 0, "+", 10, 11, 0, 20, 0.25, 5, 15, 0, 0, 0, 0, 0, sep = "\t")
)
tmp <- tempfile(fileext = ".bed")
writeLines(lines, tmp)
bm <- readBedMethyl(tmp, mod = "m", fields = c("coverage", "pct", "mod_reads"))
bm2 <- subsetBy(bm, "coverage", function(v) v >= 15)
length(RBedMethyl::beta(bm2))
```

`subsetByChromosomes` *Subset by chromosomes*

Description

Subset an `RBedMethyl` object by one or more chromosomes.

Usage

```
subsetByChromosomes(x, chr)
```

Arguments

`x`	An `RBedMethyl` object.
`chr`	Character vector of chromosome names.

Value

A filtered `RBedMethyl` object.

Examples

```
lines <- c(
  paste("chr1", 0, 1, "m", 0, "+", 0, 1, 0, 10, 0.5, 5, 5, 0, 0, 0, 0, 0, sep = "\t"),
  paste("chr2", 10, 11, "m", 0, "+", 10, 11, 0, 20, 0.25, 5, 15, 0, 0, 0, 0, 0, sep = "\t")
)
tmp <- tempfile(fileext = ".bed")
writeLines(lines, tmp)
bm <- readBedMethyl(tmp, mod = "m", fields = c("coverage", "pct", "mod_reads"))
bm2 <- subsetByChromosomes(bm, c("chr1"))
length(RBedMethyl::beta(bm2))
```

`subsetByRegion` *Subset by region*

Description

Subset an `RBedMethyl` object by genomic interval.

Usage

```
subsetByRegion(x, chr, start, end)
```

Arguments

`x`	An `RBedMethyl` object.
`chr`	Chromosome name.
`start`	Region start (0-based, half-open).
`end`	Region end.

Value

A filtered `RBedMethyl` object.

Examples

```
lines <- c(
  paste("chr1", 0, 1, "m", 0, "+", 0, 1, 0, 10, 0.5, 5, 5, 0, 0, 0, 0, 0, sep = "\t"),
  paste("chr1", 10, 11, "m", 0, "+", 10, 11, 0, 20, 0.25, 5, 15, 0, 0, 0, 0, 0, sep = "\t")
)
tmp <- tempfile(fileext = ".bed")
writeLines(lines, tmp)
bm <- readBedMethyl(tmp, mod = "m", fields = c("coverage", "pct", "mod_reads"))
bm2 <- subsetByRegion(bm, "chr1", 0, 5)
length(RBedMethyl::beta(bm2))
```

`subsetByRegion,RBedMethyl,GRanges,missing,missing-method`
Subset by GRanges

Description

Subset an `RBedMethyl` object by overlaps with a `GRanges`.

Usage

```
## S4 method for signature 'RBedMethyl,GRanges,missing,missing'
subsetByRegion(x, chr, start, end)
```

Arguments

`x`	An `RBedMethyl` object.
`chr`	A `GRanges` object of regions.
`start`	Unused (for signature compatibility).
`end`	Unused (for signature compatibility).

Value

A filtered `RBedMethyl` object.

`summarizeByRegion` *Summarize by regions*

Description

Summarize methylation by a set of regions.

Usage

```
summarizeByRegion(x, regions)
```

Arguments

`x`	An `RBedMethyl` object.
`regions`	A `GRanges` of regions.

Value

A `DataFrame` with coverage, mod_reads, beta, and n_sites.

Examples

```
lines <- c(
  paste("chr1", 0, 1, "m", 0, "+", 0, 1, 0, 10, 0.5, 5, 5, 0, 0, 0, 0, 0, sep = "\t"),
  paste("chr1", 10, 11, "m", 0, "+", 10, 11, 0, 20, 0.25, 5, 15, 0, 0, 0, 0, 0, sep = "\t")
)
tmp <- tempfile(fileext = ".bed")
writeLines(lines, tmp)
bm <- readBedMethyl(tmp, mod = "m", fields = c("coverage", "pct", "mod_reads"))
regions <- GenomicRanges::GRanges(
  seqnames = "chr1",
  ranges = IRanges::IRanges(start = 1, end = 12)
)
summarizeByRegion(bm, regions)
```

`[,RBedMethyl,missing,missing,missing-method`
Subset rows

Description

Subset an `RBedMethyl` object by integer, logical, or `GRanges` index.

Usage

```
## S4 method for signature 'RBedMethyl,missing,missing,missing'
x[i, j, ..., drop = TRUE]
```

Arguments

`x`	An `RBedMethyl` object.
`i`	Integer, logical, or `GRanges` index.
`j`	Unused.
`...`	Unused.
`drop`	Unused.

Value

A filtered `RBedMethyl` object.

Index

[,RBedMethyl,ANY,missing,missing-method
    ([,RBedMethyl,missing,missing,missing-method*)*, 8
[,RBedMethyl,missing,missing,missing-method, 8
[,RBedMethyl-method
    ([,RBedMethyl,missing,missing,missing-method*)*, 8

bedMethylFields, 2
beta,RBedMethyl,missing-method, 2
beta,RBedMethyl-method
    *(*beta,RBedMethyl,missing-method*)*, 2

filterByCoverage, 3
filterByCoverage,RBedMethyl-method
    *(*filterByCoverage*)*, 3

RBedMethyl-class, 4
readBedMethyl, 4

subsetBy, 5
subsetBy,RBedMethyl-method *(*subsetBy*)*, 5
subsetByChromosomes, 6
subsetByChromosomes,RBedMethyl-method
    *(*subsetByChromosomes*)*, 6
subsetByRegion, 6
subsetByRegion,RBedMethyl,character,numeric,numeric-method
    *(*subsetByRegion*)*, 6
subsetByRegion,RBedMethyl,GRanges,missing,missing-method, 7
subsetByRegion,RBedMethyl,GRanges-method
    *(*subsetByRegion,RBedMethyl,GRanges,missing,missing-method*)*, 7
subsetByRegion,RBedMethyl-method
    *(*subsetByRegion*)*, 6
summarizeByRegion, 8
summarizeByRegion,RBedMethyl-method
    *(*summarizeByRegion*)*, 8