Package 'NanoMethViz'

August 5, 2022

Type Package

Title Visualise methlation data from Oxford Nanopore sequencing

Version 2.3.0

Description NanoMethViz is a toolkit for visualising methylation data from Oxford Nanopore sequencing. It can be used to explore methylation patterns from reads derived from Oxford Nanopore direct DNA sequencing with methylation called by callers including nanopolish, f5c and megalodon. The plots in this package allow the visualisation of methylation profiles aggregated over experimental groups and across classes of genomic features.

biocViews Software, Visualization, DifferentialMethylation

URL https://github.com/shians/NanoMethViz

BugReports https://github.com/Shians/NanoMethViz/issues

Depends R (>= 4.0.0), methods, ggplot2

Imports cpp11 (>= 0.2.5), readr, S4Vectors, SummarizedExperiment, BiocSingular, bsseq, forcats, assertthat, AnnotationDbi, Rcpp, dplyr, data.table, e1071, fs, GenomicRanges, glue, limma (>= 3.44.0), patchwork, purrr, rlang, RSQLite, Rsamtools, scales, scico, stats, stringr, tibble, tidyr, utils, withr, zlibbioc

Suggests DSS, Mus.musculus, Homo.sapiens, knitr, rmarkdown, testthat (>= 3.0.0), covr

LinkingTo Rcpp

License Apache License (>= 2.0)

SystemRequirements C++11

VignetteBuilder knitr

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

Config/testthat/edition 3

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

git_branch master

git_last_commit 0569dc5

git_last_commit_date 2022-04-26

Date/Publication 2022-08-05

Author Shian Su [cre, aut]

Maintainer Shian Su <su.s@wehi.edu.au>

R topics documented:

`bsseq_to_edger` *Convert BSseq object to edgeR methylation matrix*

Description

Convert BSseq object to edgeR methylation matrix

Usage

```
bsseq_to_edger(bsseq, regions = NULL)
```

Arguments

bsseq	the BSseq object.
regions	the regions to calculate log-methylation ratios over. If left NULL, ratios will be calculated per site.

Value

a matrix compatible with the edgeR differential methylation pipeline

Examples

```
methy <- system.file("methy_subset.tsv.bgz", package = "NanoMethViz")
bsseq <- methy_to_bsseq(methy)
edger_mat <- bsseq_to_edger(bsseq)
```

---

bsseq_to_log_methy_ratio

Convert BSseq object to log-methylation-ratio matrix

Description

Creates a log-methylation-ratio matrix from a BSseq object that is useful for dimensionality reduction plots.

Usage

```
bsseq_to_log_methy_ratio(bsseq, regions = NULL, prior_count = 2)
```

Arguments

bsseq	the BSseq object.
regions	the regions to calculate log-methylation ratios over. If left NULL, ratios will be calculated per site.
prior_count	the prior count added to avoid taking log of 0.

Value

a matrix containing log-methylation-ratios.

Examples

```
nmr <- load_example_nanomethresult()
bsseq <- methy_to_bsseq(nmr)
regions <- exons_to_genes(NanoMethViz::exons(nmr))
log_m_ratio <- bsseq_to_log_methy_ratio(bsseq, regions)
```

cluster_regions *Cluster regions by K-means*

Description

Cluster regions by k-means based on their methylation profiles. In order to cluster using k-means the methylation profile of each region is interpolated and sampled at fixed points. The first 10 principal components are used for the k-means clustering. The clustering is best behaved in regions of similar width and CpG density.

Usage

```
cluster_regions(x, regions, centers = 2, grid_method = c("density", "uniform"))
```

Arguments

x	the NanoMethResult object.
regions	a table of regions containing at least columns chr, strand, start and end.
centers	number of centers for k-means, identical to the number of output clusters.
grid_method	the method for generating the sampling grid. The default option "density" attempts to create a grid with similar density as the data, "uniform" creates a grid of uniform density.

Value

the table of regions given by the 'regions' argument with the column 'cluster' added.

Examples

```
nmr <- load_example_nanomethresult()
gene_anno <- exons_to_genes(NanoMethViz::exons(nmr))
# uniform grid due to low number of input features
gene_anno_clustered <- cluster_regions(nmr, gene_anno, centers = 2, grid_method = "uniform")
plot_agg_regions(nmr, gene_anno_clustered, group_col = "cluster")
```

create_tabix_file *Create a tabix file using methylation calls*

Description

Create a tabix file using methylation calls

Usage

```
create_tabix_file(
  input_files,
  output_file,
  samples = extract_file_names(input_files),
  verbose = TRUE
)
```

Arguments

input_files	the files to convert
output_file	the output file to write results to (must end in .bgz)
samples	the names of samples corresponding to each file
verbose	TRUE if progress messages are to be printed

Value

invisibly returns the output file path, creates a tabix file (.bgz) and its index (.bgz.tbi)

Examples

```
methy_calls <- system.file(package = "NanoMethViz",
    c("sample1_nanopolish.tsv.gz", "sample2_nanopolish.tsv.gz"))
temp_file <- paste0(tempfile(), ".tsv.bgz")

create_tabix_file(methy_calls, temp_file)
```

---

exons_to_genes *Convert exon annotation to genes*

Description

Convert exon annotation to genes

Usage

```
exons_to_genes(x)
```

Arguments

x	the exon level annotation containing columns "gene_id", "chr", "strand" and "symbol".

Value

the gene level annotation where each gene is taken to span the earliest start position and latest end position of its exons.

Examples

```
nmr <- load_example_nanomethresult()
exons_to_genes(NanoMethViz::exons(nmr))
```

`filter_methy` *Create filtered methylation file*

Description

Create a filtered methylation file from an existing one.

Usage

```
filter_methy(nmr, output_file, ...)
```

Arguments

`nmr`	the NanoMethResult object.
`output_file`	the output file to write results to (must end in .bgz).
`...`	filtering criteria given in dplyr syntax.

Value

invisibly returns 'output_file'

Examples

```
nmr <- load_example_nanomethresult()
output_file <- paste0(tempfile(), ".tsv.bgz")
filter_methy(nmr, output_file = output_file, chr == "chrX")
```

`get_example_exons_mus_musculus`
Get example exon annotations for mus musculus

Description

This is a small subset of the exons returned by `get_exons_mus_musculus()` for demonstrative purposes. It contains the exons for the genes Brca1, Brca2, Impact, Meg3, Peg3 and Xist.

Usage

```
get_example_exons_mus_musculus()
```

Value

data.frame containing exons

Examples

```
example_exons <- get_example_exons_mus_musculus()
```

---

get_exons_homo_sapiens

Get exon annotations for homo sapiens

Description

Get exon annotations for homo sapiens

Usage

```
get_exons_homo_sapiens()
```

Value

data.frame containing exons

Examples

```
h_sapiens_exons <- get_exons_homo_sapiens()
```

---

get_exons_mus_musculus

Get exon annotations for mus musculus

Description

Get exon annotations for mus musculus

Usage

```
get_exons_mus_musculus()
```

Value

data.frame containing exons

Examples

```
m_musculus_exons <- get_exons_mus_musculus()
```

load_example_nanomethresult *Load an example NanoMethResult object*

Description

Load an example NanoMethResult object

Usage

```
load_example_nanomethresult()
```

Value

a NanoMethResults object

Examples

```
nmr <- load_example_nanomethresult()
```

methy_col_names *Column names for methylation data*

Description

Column names for methylation data

Usage

```
methy_col_names()
```

Value

column names for methylation data

Examples

```
methy_col_names()
```

methy_to_bsseq *Create BSSeq object from methylation tabix file*

Description

Create BSSeq object from methylation tabix file

Usage

```
methy_to_bsseq(methy, out_folder = tempdir(), verbose = TRUE)
```

Arguments

methy	the path to the methylation tabix file.
out_folder	the folder to store intermediate files. One file is created for each sample and contains columns "chr", "pos", "total" and "methylated".
verbose	TRUE if progress messages are to be printed

Value

a BSSeq object.

Examples

```
nmr <- load_example_nanomethresult()
bsseq <- methy_to_bsseq(nmr)
```

methy_to_edger *Convert NanoMethResult object to edgeR methylation matrix*

Description

Convert NanoMethResult object to edgeR methylation matrix

Usage

```
methy_to_edger(methy, regions = NULL, out_folder = tempdir(), verbose = TRUE)
```

Arguments

methy	the path to the methylation tabix file.
regions	the regions to calculate log-methylation ratios over. If left NULL, ratios will be calculated per site.
out_folder	the folder to store intermediate files. One file is created for each sample and contains columns "chr", "pos", "total" and "methylated".
verbose	TRUE if progress messages are to be printed

Value

a matrix compatible with the edgeR differential methylation pipeline

Examples

```
nmr <- load_example_nanomethresult()
edger_mat <- methy_to_edger(nmr)
```

NanoMethResult-class *Nanopore Methylation Result*

Description

A NanoMethResult object stores data used for NanoMethViz visualisation. It contains stores a path to the methylation data, sample information and optional exon information. The object is constructed using the NanoMethResult() constructor function described in "Usage".

Usage

```
NanoMethResult(methy, samples, exons = NULL)

## S4 method for signature 'NanoMethResult'
methy(object)

## S4 replacement method for signature 'NanoMethResult'
methy(object) <- value

## S4 method for signature 'NanoMethResult'
samples(object)

## S4 replacement method for signature 'NanoMethResult,data.frame'
samples(object) <- value

## S4 method for signature 'NanoMethResult'
exons(object)

## S4 replacement method for signature 'NanoMethResult,data.frame'
exons(object) <- value
```

Arguments

methy	the path to the methylation tabix file.
samples	the data.frame of sample annotation containg at least columns sample and group.
exons	(optional) the data.frame of exon information containing at least columns gene_id, chr, strand, start, end, transcript_id and symbol.
object	the NanoMethResult object.
value	the exon annotation.

Value

a NanoMethResult object to be used with plotting functions

the path to the methylation data.

the sample annotation.

the exon annotation.

Functions

- `NanoMethResult`: Constructor
- `methy,NanoMethResult-method`: methylation data path getter.
- `methy<-,NanoMethResult-method`: methylation data path setter.
- `samples,NanoMethResult-method`: sample annotation getter.
- `samples<-,NanoMethResult,data.frame-method`: sample annotation setter.
- `exons,NanoMethResult-method`: exon annotation getter.
- `exons<-,NanoMethResult,data.frame-method`: exon annotation getter.

Slots

`methy` the path to the methylation tabix file.

`samples` the data.frame of sample annotation containing at least columns sample and group.

`exons` the data.frame of exon information containing at least columns gene_id, chr, strand, start, end, transcript_id and symbol.

Examples

```
methy <- system.file(package = "NanoMethViz", "methy_subset.tsv.bgz")
sample <- c(
    "B6Cast_Prom_1_bl6",
    "B6Cast_Prom_1_cast",
    "B6Cast_Prom_2_bl6",
    "B6Cast_Prom_2_cast",
    "B6Cast_Prom_3_bl6",
    "B6Cast_Prom_3_cast"
)
group <- c(
    "bl6",
    "cast",
    "bl6",
    "cast",
    "bl6",
    "cast"
)
sample_anno <- data.frame(sample, group, stringsAsFactors = FALSE)
exon_tibble <- get_example_exons_mus_musculus()
NanoMethResult(methy, sample_anno, exon_tibble)

x <- load_example_nanomethresult()
```

```
methy(x)
```

`plot_agg_regions` *Plot aggregate regions*

Description

Plot aggregate regions

Usage

```
plot_agg_regions(
  x,
  regions,
  binary_threshold = 0.5,
  group_col = NULL,
  flank = 2000,
  stranded = TRUE,
  span = 0.05,
  palette = ggplot2::scale_colour_brewer(palette = "Set1")
)
```

Arguments

`x`	the NanoMethResult object.
`regions`	a table of regions containing at least columns chr, strand, start and end. Any additional columns can be used for grouping.
`binary_threshold`	the modification probability such that calls with modification probability above the threshold are considered methylated, and those with probability equal or below are considered unmethylated.
`group_col`	the column to group aggregated trends by. This column can be in from the regions table or samples(x).
`flank`	the number of flanking bases to add to each side of each region.
`stranded`	TRUE if negative strand features should have coordinates flipped to reflect features like transcription start sites.
`span`	the span for loess smoothing.
`palette`	the ggplot colour palette used for groups.

Value

a ggplot object containing the aggregate methylation trend.

Examples

```
nmr <- load_example_nanomethresult()
gene_anno <- exons_to_genes(NanoMethViz::exons(nmr))
plot_agg_regions(nmr, gene_anno)
plot_agg_regions(nmr, gene_anno, group_col = "sample")
plot_agg_regions(nmr, gene_anno, group_col = "group")
```

---

plot_gene *Plot gene*

Description

Plot gene

Usage

```
plot_gene(x, gene, ...)

## S4 method for signature 'NanoMethResult,character'
plot_gene(
  x,
  gene,
  window_prop = 0.3,
  anno_regions = NULL,
  binary_threshold = NULL,
  avg_method = c("mean", "median"),
  spaghetti = FALSE,
  heatmap = FALSE,
  span = NULL,
  gene_anno = TRUE,
  palette = ggplot2::scale_colour_brewer(palette = "Set1")
)
```

Arguments

x	the NanoMethResult object.
gene	the gene symbol for the gene to plot.
...	additional arguments
window_prop	the size of flanking region to plot. Can be a vector of two values for left and right window size. Values indicate proportion of gene length.
anno_regions	the data.frame of regions to annotate.
binary_threshold	the modification probability such that calls with modification probability above the threshold are set to 1 and probabilities equal to or below the threshold are set to 0.

avg_method	the average method for pre-smoothing at each genomic position. Data is pre-smoothed at each genomic position before the smoothed aggregate line is generated for performance reasons. The default is "mean" which corresponds to the average methylation fraction. The alternative "median" option is closer to an average within the more common methylation state.
spaghetti	whether or not individual reads should be shown.
heatmap	whether or not read-methylation heatmap should be shown.
span	the span for loess smoothing.
gene_anno	whether or not gene annotation tracks are plotted.
palette	the ggplot colour palette used for groups.

Value

a patchwork plot containing the methylation profile in the specified region.

a patchwork plot containing the methylation profile in the specified region.

Examples

```
nmr <- load_example_nanomethresult()
plot_gene(nmr, "Peg3")
```


```
nmr <- load_example_nanomethresult()
plot_gene(nmr, "Peg3")
```


---

plot_gene_heatmap *Plot gene methylation heatmap*

Description

Plot gene methylation heatmap

Usage

```
plot_gene_heatmap(x, gene, ...)

## S4 method for signature 'NanoMethResult,character'
plot_gene_heatmap(
  x,
  gene,
  window_prop = 0.3,
  pos_style = c("to_scale", "compact")
)
```

Arguments

x	the NanoMethResult object.
gene	the gene symbol for the gene to plot.
...	additional arguments
window_prop	the size of flanking region to plot. Can be a vector of two values for left and right window size. Values indicate proportion of gene length.
pos_style	the style for plotting the base positions along the x-axis. Defaults to "to_scale", plotting (potentially) overlapping squares along the genomic position to scale. The "compact" options plots only the positions with measured modification.

Value

a ggplot object of the heatmap

a ggplot plot containing the heatmap.

Examples

```
nmr <- load_example_nanomethresult()
plot_gene_heatmap(nmr, "Peg3")

nmr <- load_example_nanomethresult()
plot_gene_heatmap(nmr, "Peg3")
```

---

plot_grange *Plot GRanges*

Description

Plot GRanges

Usage

```
plot_grange(
  x,
  grange,
  anno_regions = NULL,
  binary_threshold = NULL,
  avg_method = c("mean", "median"),
  spaghetti = FALSE,
  heatmap = FALSE,
  span = NULL,
  window_prop = 0,
  palette = ggplot2::scale_colour_brewer(palette = "Set1")
)
```

Arguments

x	the NanoMethResult object.
grange	the GRanges object with one entry.
anno_regions	the data.frame of regions to be annotated.
binary_threshold	the modification probability such that calls with modification probability above the threshold are set to 1 and probabilities equal to or below the threshold are set to 0.
avg_method	the average method for pre-smoothing at each genomic position. Data is pre-smoothed at each genomic position before the smoothed aggregate line is generated for performance reasons. The default is "mean" which corresponds to the average methylation fraction. The alternative "median" option is closer to an average within the more common methylation state.
spaghetti	whether or not individual reads should be shown.
heatmap	whether or not read-methylation heatmap should be shown.
span	the span for loess smoothing.
window_prop	the size of flanking region to plot. Can be a vector of two values for left and right window size. Values indicate proportion of gene length.
palette	the ggplot colour palette used for groups.

Value

a patchwork plot containing the methylation profile in the specified region.

a patchwork plot containing the methylation profile in the specified region.

Examples

```
nmr <- load_example_nanomethresult()
plot_grange(nmr, GRanges("chr7:6703892-6730431"))
```

---

plot_grange_heatmap *Plot GRanges heatmap*

Description

Plot GRanges heatmap

Usage

```
plot_grange_heatmap(
  x,
  grange,
  pos_style = c("to_scale", "compact"),
  window_prop = 0
)
```

Arguments

x	the NanoMethResult object.
grange	the GRanges object with one entry.
pos_style	the style for plotting the base positions along the x-axis. Defaults to "to_scale", plotting (potentially) overlapping squares along the genomic position to scale. The "compact" options plots only the positions with measured modification.
window_prop	the size of flanking region to plot. Can be a vector of two values for left and right window size. Values indicate proportion of gene length.

Value

a ggplot plot containing the heatmap.

Examples

```
nmr <- load_example_nanomethresult()
gr <- GenomicRanges::GRanges(data.frame(chr = "chr7", start = 6703892, end = 6730431))
plot_grange_heatmap(nmr, gr[1, ])
```

---

plot_mds *Plot MDS*

Description

Plot multi-dimensional scaling plot using algorithm of limma::plotMDS(). It is recommended this be done with the log-methylation-ratio matrix generated by bsseq_to_log_methy_ratio().

Usage

```
plot_mds(
  x,
  top = 500,
  plot_dims = c(1, 2),
  labels = colnames(x),
  groups = NULL
)
```

Arguments

x	the log-methylation-ratio matrix.
top	the number of top genes used to calculate pairwise distances.
plot_dims	the numeric vector of the two dimensions to be plotted.
labels	the character vector of labels for data points. By default uses column names of x, set to NULL to plot points.
groups	the character vector of groups the data points will be coloured by.

Value

ggplot object of the MDS plot.

Examples

```
nmr <- load_example_nanomethresult()
bss <- methy_to_bsseq(nmr)
lmr <- bsseq_to_log_methy_ratio(bss)
plot_mds(lmr)
```

`plot_pca` *Plot PCA*

Description

Plot multi-dimensional scaling plot using algorithm of BiocSingular::runPCA(). It is recommended this be done with the log-methylation-ratio matrix generated by bsseq_to_log_methy_ratio().

Usage

```
plot_pca(x, plot_dims = c(1, 2), labels = colnames(x), groups = NULL)
```

Arguments

`x`	the log-methylation-ratio matrix.
`plot_dims`	the numeric vector of the two dimensions to be plotted.
`labels`	the character vector of labels for data points. By default uses column names of x, set to NULL to plot points.
`groups`	the character vector of groups the data points will be coloured by.

Value

ggplot object of the MDS plot.

Examples

```
nmr <- load_example_nanomethresult()
bss <- methy_to_bsseq(nmr)
lmr <- bsseq_to_log_methy_ratio(bss)
plot_pca(lmr)
```

plot_region *Plot region*

Description

Plot region

Usage

```
plot_region(x, chr, start, end, ...)

## S4 method for signature 'NanoMethResult,character,numeric,numeric'
plot_region(
  x,
  chr,
  start,
  end,
  anno_regions = NULL,
  binary_threshold = NULL,
  avg_method = c("mean", "median"),
  spaghetti = FALSE,
  heatmap = FALSE,
  span = NULL,
  window_prop = 0,
  palette = ggplot2::scale_colour_brewer(palette = "Set1")
)

## S4 method for signature 'NanoMethResult,factor,numeric,numeric'
plot_region(
  x,
  chr,
  start,
  end,
  anno_regions = NULL,
  binary_threshold = NULL,
  avg_method = c("mean", "median"),
  spaghetti = FALSE,
  heatmap = FALSE,
  span = NULL,
  window_prop = 0,
  palette = ggplot2::scale_colour_brewer(palette = "Set1")
)
```

Arguments

x	the NanoMethResult object.
chr	the chromosome to plot.

start	the start of the plotting region.
end	the end of the plotting region.
...	additional arguments.
anno_regions	the data.frame of regions to be annotated.
binary_threshold	the modification probability such that calls with modification probability above the threshold are set to 1 and probabilities equal to or below the threshold are set to 0.
avg_method	the average method for pre-smoothing at each genomic position. Data is pre-smoothed at each genomic position before the smoothed aggregate line is generated for performance reasons. The default is "mean" which corresponds to the average methylation fraction. The alternative "median" option is closer to an average within the more common methylation state.
spaghetti	whether or not individual reads should be shown.
heatmap	whether or not read-methylation heatmap should be shown.
span	the span for loess smoothing.
window_prop	the size of flanking region to plot. Can be a vector of two values for left and right window size. Values indicate proportion of gene length.
palette	the ggplot colour palette used for groups.

Value

a patchwork plot containing the methylation profile in the specified region.

a patchwork plot containing the methylation profile in the specified region.

Examples

```
nmr <- load_example_nanomethresult()
plot_region(nmr, "chr7", 6703892, 6730431)

nmr <- load_example_nanomethresult()
plot_region(nmr, "chr7", 6703892, 6730431)
```

---

plot_region_heatmap *Plot region methylation heatmap*

Description

Plot region methylation heatmap

Usage

```
plot_region_heatmap(x, chr, start, end, ...)

## S4 method for signature 'NanoMethResult,character,numeric,numeric'
plot_region_heatmap(
  x,
  chr,
  start,
  end,
  pos_style = c("to_scale", "compact"),
  window_prop = 0.3
)

## S4 method for signature 'NanoMethResult,factor,numeric,numeric'
plot_region_heatmap(
  x,
  chr,
  start,
  end,
  pos_style = c("to_scale", "compact"),
  window_prop = 0.3
)
```

Arguments

x	the NanoMethResult object.
chr	the chromosome to plot.
start	the start of the plotting region.
end	the end of the plotting region.
...	additional arguments.
pos_style	the style for plotting the base positions along the x-axis. Defaults to "to_scale", plotting (potentially) overlapping squares along the genomic position to scale. The "compact" options plots only the positions with measured modification.
window_prop	the size of flanking region to plot. Can be a vector of two values for left and right window size. Values indicate proportion of gene length.

Value

a ggplot object of the heatmap.

a ggplot plot containing the heatmap.

Examples

```
nmr <- load_example_nanomethresult()
plot_region_heatmap(nmr, "chr7", 6703892, 6730431)

nmr <- load_example_nanomethresult()
```

```
plot_region_heatmap(nmr, "chr7", 6703892, 6730431)
```

`query_exons` *Query exons*

Description

Query a data.frame of exons for a subset.

Usage

```
query_exons_region(exons, chr, start, end)

query_exons_gene_id(exons, gene_id)

query_exons_symbol(exons, symbol)
```

Arguments

`exons`	the data.frame of exons.
`chr`	the chromosome to query.
`start`	the start of the query region.
`end`	the end of the query region.
`gene_id`	the gene_id to query.
`symbol`	the gene_id to query.

Value

data.frame of queried exons.

Functions

- `query_exons_region`: Query region.
- `query_exons_gene_id`: Query gene ID.
- `query_exons_symbol`: Query gene symbol.

`query_methy` *Query methylation data*

Description

Query methylation data

Usage

```
query_methy(x, chr, start, end, simplify = TRUE, force = FALSE)
```

Arguments

`x`	the path to the methylation data (tabix-bgzipped)
`chr`	the vector of chromosomes
`start`	the vector of start positions
`end`	the vector of end positions
`simplify`	whether returned results should be row-concatenated
`force`	whether to force empty output when query region 'chr' does not appear in data. Without 'force', an empty result indicates that the requested 'chr' appears in the data but no data overlaps with requested region, and an invalid 'chr' will cause an error.

Value

A table containing the data within the queried regions. If simplify is TRUE (default) then all data is contained within one table, otherwise it is a list of tables where each element is the data from one region.

Examples

```
nmr <- load_example_nanomethresult()
query_methy(methy(nmr), "chr7", 6703892, 6730431)
```

`region_methy_stats` *Calculate region methylation statistics*

Description

Calculate the average methylation probability and prevalence based on specified probability threshold.

Usage

```
region_methy_stats(nmr, regions, threshold = 0.5)
```

Arguments

nmr	the NanoMethResult object.
regions	the table of regions to query statistics for.
threshold	the threshold to use for determining methylation calls for the calculation of prevalence.

Value

table of regions with additional columns of methylation summary statistics.

Examples

```
nmr <- load_example_nanomethresult()
gene_anno <- exons_to_genes(NanoMethViz::exons(nmr))
region_methy_stats(nmr, gene_anno)
```

Index