Package ‘M3D’

January 21, 2017

Type Package

Title Identifies differentially methylated regions across testing groups

Version 1.8.0

Date 2016-06-22

Author Tom Mayo

Maintainer Tom Mayo <t.mayo@ed.ac.uk>

Description This package identifies statistically significantly differentially methylated regions of CpGs. It uses kernel methods (the Maximum Mean Discrepancy) to measure differences in methylation profiles, and relates these to inter-replicate changes, whilst accounting for variation in coverage profiles.

License Artistic License 2.0

Imports parallel, Rcpp, BiocGenerics, S4Vectors, IRanges, GenomicRanges, SummarizedExperiment, BiSeq

LinkingTo Rcpp

Depends R (>= 3.3.0)

VignetteBuilder knitr

Suggests BiocStyle, knitr, testthat

biocViews DNA Methylation, Differential Methylation, Coverage, CpG Island

RoxygenNote 5.0.1

NeedsCompilation yes

R topics documented:

M3D-package ................................................. 2
CpGsDemo .................................................... 3
determineGroupComps ..................................... 3
findComps ..................................................... 4
M3D_Para .................................................... 5
M3D_Para_lite ............................................... 6
M3D_Single .................................................. 7
M3D_Wrapper ............................................... 8
M3D_Wrapper_lite ......................................... 9
medianFreq .................................................. 10
**M3D-package**

Non-parametric statistical testing

**Description**

This package identifies statistically significantly differentially methylated regions of CpGs. It uses kernel methods, specifically the Maximum Mean Discrepancy (Gretton et al. 2006), to measure differences in methylation profiles, and relates these to inter-replicate changes, whilst accounting for variation in coverage profiles.

**Details**

- **Package:** M3D
- **Type:** Package
- **Version:** 0.99.0
- **Date:** 2014-07-17
- **License:** Artistic License 2.0

This package works on RRBS data as processed by the BiSeq package. The starting point is an rrbs object, a class defined by the BiSeq package (Hebestreit et al. 2013), and a GRanges object outlining the regions to test. The maximum mean discrepancy (MMD) (Gretton et al. 2006) is calculated over each region for each pair of samples, once with respect to methylation levels and once respecting only coverage. These two values are subtracted to form a test-statistic and between-group values are compared to inter-replicate values to provide p-values. These reflect the empirical probability of observing the between-group methylation differences among the replicates.

**Function list:**

- `determineGroupComps`: returns a vector of the sample comparisons
- `findComps`: returns the indices of the M3D test-statistic that corresponding to particular samples
- `M3D_Single`: Computes the two components of the M3D test-statistic over 1 island for 1 sample pair.
- `M3D_Wrapper`: Computes the two components of the M3D test-statistic over all sample pairs over all islands.
- `medianFreq`: Returns the median of data summarised by unique values and the frequency with which they occur.
- `pvals`: Returns empirical p-values for the regions based on the M3D test-statistic.
Author(s)

Tom Mayo

Maintainer: Tom Mayo <t.mayo@ed.ac.uk>

References


Description

Toy data for the package - 1000 CpG regions to be tested in a GRanges object

Author(s)

Tom Mayo

determineGroupComps

Creates strings of sample pair comparisons

Description

Takes in a vector of strings of sample names and returns strings of all the comparisons, either within a testing group or between testing groups. This is not intended to be called directly by the user.

Usage

determineGroupComps(samples1, samples2 = NULL, type)

Arguments

samples1 A vector of sample names from one group
samples2 A vector of sample names from the other group, if we want to specify between-group comparisons
type 'within' or 'between'. 'within' returns all the sample pairs within samples1, 'between' returns all the sample pairs between samples1 and samples2

Value

A vector of sample pair comparisons of the form 'sample1 vs sample2' for use with the M3D functions
findComps

Finds columns in the M3D test-statistic matrix

Description

Returns the columns of the test-statistic matrix that refer to specific samples. This is not intended to be called directly by the user.

Usage

findComps(MMD, samples)

Arguments

- **MMD**
  - A matrix containing the M3D test-statistic, the difference the full and methylation blind metrics, for each region in the CpGs object. Each column is a comparison between two samples, which are described in the column names.

- **samples**
  - A vector of sample pairs of the form 'sample1 vs sample2' as returned from determineGroupComps

Value

Returns the indices of the M3D test-statistic components that contain the sample pair comparisons in 'samples'

Author(s)

Tom Mayo <t.mayo@ed.ac.uk>

Examples

```r
library(BiSeq)
data(rrbsDemo)
samples1 <- rownames(colData(rrbsDemo))[colData(rrbsDemo)[,]=='H1-hESC']
within1 <- determineGroupComps(samples1, type='within')
findComps(MMDlistDemo, within1)
```
M3D_Para

Description

Parallel implementation of M3D_Wrapper function. Returns the two components of the M3D test-statistic - the MMD (Gretton et al. 2006) for the full data and the coverage only data, respectively - for all regions and all samples pairs, as a matrix.

Usage

M3D_Para(rrbs, CpGs, overlaps, num.cores = NaN)

Arguments

rrbs An rrbs object containing methylation and coverage data as created using the BiSeq package

CpGs A GRanges object detailing the testing regions.

overlaps The overlaps between the list of testing regions and the methylation data. This is obtained using the function findOverlaps(CpGs,rrbs) for a GRanges object CpGs detailing the testing regions.

num.cores Integer giving the number of cores to use. Defaults to the maximum available

Value

This returns the two components of the M3D test-statistic for each region over all sample pairs as a matrix. Subtracting them gives the M3D test-statistic. This is processed with the function pvals.

Author(s)

Tom Mayo <t.mayo@ed.ac.uk>

References


Examples

library(BiSeq)
library('GenomicRanges')
data(rrbsDemo)
data(CpGsDemo)
data(MMDlistDemo)
CpGs1 <- CpGsDemo[1:100]
overlaps1 <- findOverlaps(CpGs1,rrbsDemo)
M3D_stat_lite <- M3D_Para(rrbsDemo,CpGs1,overlaps1,num.cores=4)
head(M3D_stat_lite[[1]] - M3D_stat_lite[[2]])
M3D_Para_lite

Computes the components of the M3D test-statistic over all regions for all sample-pairs.

Description

Parallel implementation of M3D_Wrapper_lite function. Returns the M3D test-statistic, without the components, for all regions with the cross group sample pairs averaged to save memory (in column one), as a matrix.

Usage

M3D_Para_lite(rrbs, CpGs, overlaps, group1 = NaN, group2 = NaN, num.cores = NaN)

Arguments

rrbs An rrbs object containing methylation and coverage data as created using the BiSeq package
CpGs A GRanges object detailing the testing regions.
overlaps The overlaps between the list of testing regions and the methylation data. This is obtained using the function findOverlaps(CpGs,rrbs) for a GRanges object CpGs detailing the testing regions.
group1 The name of the first group for the comparison. This is stored in colData(rrbs). Default finds first unique group in colData(rrbs).
group2 The name of the second group for the comparison. This is stored in colData(rrbs). Default finds second unique group in colData(rrbs).
num.cores Integer giving the number or cores to use. Defaults to the maximum available

Value

This returns the two components of the M3D test-statistic for each region over all sample pairs as a matrix. Subtracting them gives the M3D test-statistic. This is processed with the function pvals.

Author(s)

Tom Mayo <t.mayo@ed.ac.uk>

References


Examples

library(parallel)
library(BiSeq)
library('GenomicRanges')
data(rrbsDemo)
data(CpGsDemo)
data(MMDListDemo)
M3D_Single

Computes the components of the M3D test-statistic over one region for 2 samples

Description

Returns the two components of the M3D test-statistic - the MMD (Gretton et al. 2006) for the full data and the coverage only data, respectively. This is not intended to be called directly by the user.

Usage

M3D_Single(test_data, loc_mx, locInds, method = "MinusCovMMD")

Arguments

test_data  Contains the methylation data over two samples for a given region.
loc_mx  The matrix of distances between the CpG sites.
locInds  The indices of the non-zero entries of loc_mx.
method  This specifies whether to return the full MMD and the methylation blind MMD (focusing only on the coverage) or just the former. if method = 'MinusCovMMD' it is both, all other values return just the full MMD.

Value

This returns the value of the MMD for the region between the two samples as a numeric. If method is set to 'MinusCovMMD', a list is returned of the full MMD and the coverage only MMD. Subtracting them gives the M3D test-statistic.

Author(s)

Tom Mayo <t.mayo@ed.ac.uk>

References

M3D_Wrapper

**Description**

Returns the two components of the M3D test-statistic - the MMD (Gretton et al. 2006) for the full data and the coverage only data, respectively - for all regions and all sample pairs, as a matrix.

**Usage**

M3D_Wrapper(rrbs, overlaps, para = FALSE)

**Arguments**

- **rrbs**: An rrbs object containing methylation and coverage data as created using the BiSeq package
- **overlaps**: The overlaps between the list of testing regions and the methylation data. This is obtained using the function findOverlaps(CpGs, rrbs) for a GRanges object CpGs detailing the testing regions.
- **para**: Set to true if called via M3D_Para

**Value**

This returns the two components of the M3D test-statistic for each region over all sample pairs as a matrix. Subtracting them gives the M3D test-statistic. This is processed with the function pvals.

**Author(s)**

Tom Mayo <t.mayo@ed.ac.uk>

**References**


**Examples**

data(rrbsDemo)
data(CpGsDemo)
CpGsDemo <- CpGsDemo[1:5]
overlaps <- GenomicRanges::findOverlaps(CpGsDemo, rrbsDemo)
M3D_list <- M3D_Wrapper(rrbsDemo, overlaps)
head(M3D_list$Full - M3D_list$Coverage)
M3D_Wrapper_lite

Computes the components of the M3D test-statistic over all regions for all sample-pairs.

Description

Returns the M3D test-statistic, without the components, for all regions with the cross group sample pairs averaged to save memory (in column one), as a matrix.

Usage

M3D_Wrapper_lite(rrbs, overlaps, group1 = NaN, group2 = NaN, verbose = TRUE)

Arguments

rrbs An rrbs object containing methylation and coverage data as created using the BiSeq package
overlaps The overlaps between the list of testing regions and the methylation data. This is obtained using the function findOverlaps(CpGs,rrbs) for a GRanges object CpGs detailing the testing regions.
group1 The name of the first group for the comparison. This is stored in colData(rrbs). Default finds first unique group in colData(rrbs).
group2 The name of the second group for the comparison. This is stored in colData(rrbs). Default finds second unique group in colData(rrbs).
verbose Logical vector. If true, the function prints a progress bar.

Value

This returns the two components of the M3D test-statistic for each region over all sample pairs as a matrix. Subtracting them gives the M3D test-statistic. This is processed with the function pvals.

Author(s)

Tom Mayo <t.mayo@ed.ac.uk>

References


Examples

data(rrbsDemo)
data(CpGsDemo)
CpGsDemo <- CpGsDemo[1:5]
overlaps <- GenomicRanges::findOverlaps(CpGsDemo,rrbsDemo)
M3D_list <- M3D_Wrapper_lite(rrbsDemo,overlaps)
head(M3D_list)
medianFreq  Finds the median

Description
Returns the median of a list of values with corresponding frequencies. This is not intended to be called directly by the user.

Usage
medianFreq(values, freqs)

Arguments
values A vector of the unique values that occur
freqs A vector of the number of occurrences of each value

Value
Returns the median value of the data comprising each entry in values repeated the corresponding entry in freqs number of times, as a numeric.

Author(s)
Tom Mayo <t.mayo@ed.ac.uk>
### MMDlistDemo

**Author(s)**
Tom Mayo <t.mayo@ed.ac.uk>

**Examples**

```r
c(c(1,2,3), c(3,6,10))
```

**Description**

Toy data for the package - the output of the `M3D_Wrapper` function.

**Author(s)**
Tom Mayo

### PDemo

**Author(s)**
Tom Mayo

### plotCoverageSampleProfiles

**Description**

Plots coverage profiles over a specific region.

**Usage**

```r
plotCoverageSampleProfiles(rrbs, CpGs, group1, group2, CpGindex,
                          samples = NaN, plot_title = NaN)
```
plotMethProfile

Arguments

rrbs An rrbs object containing methylation and coverage data as created using the BiSeq package
CpGs A GRanges object with each row being a testing region
group1 The name of the first testing group
group2 The name of the second testing group
CpGindex The index within the CpGs object of the region we are plotting
samples Names of specific samples in the groups to plot, defaults to all samples in both groups
plot_title Optional title for the plot, overrides automatic title

Value

NULL, the function plots the profiles

Author(s)

Tom Mayo <t.mayo@ed.ac.uk>

Examples

# plot the 9th region in the Toy Data Set
data(rrbsDemo)
data(CpGsDemo)
plotMethProfile(rrbsDemo, CpGsDemo, 'H1-hESC', 'K562', 9)

Description

Plots a smoothed methylation profile for each of the two testing groups. Within each group, the mean of methylation level is taken, smoothed and plotted, along with the individual values.

Usage

plotMethProfile(rrbs, CpGs, group1, group2, CpGindex, plot_title = NaN)

Arguments

rrbs An rrbs object containing methylation and coverage data as created using the BiSeq package
CpGs A GRanges object with each row being a testing region
group1 The name of the first testing group
group2 The name of the second testing group
CpGindex The index within the CpGs object of the region we are plotting
plot_title Optional title for the plot, overrides automatic title
**plotMethSampleProfiles**

**Value**

NULL, the function plots the profiles

**Author(s)**

Tom Mayo <t.mayo@ed.ac.uk>

**Examples**

```r
# plot the 9th region in the Toy Data Set
data(rrbsDemo)
data(CpGsDemo)
plotMethProfile(rrbsDemo, CpGsDemo, 'H1-hESC', 'K562', 9)
```

---

**Description**

Plots a smoothed methylation profile for each of the two testing groups. Within each group, the mean of methylation level is taken, smoothed and plotted, along with the individual values.

**Usage**

```r
plotMethSampleProfiles(rrbs, CpGs, group1, group2, CpGindex, samples = NaN, plot_title = NaN)
```

**Arguments**

- `rrbs`: An rrbs object containing methylation and coverage data as created using the BiSeq package
- `CpGs`: A GRanges object with each row being a testing region
- `group1`: The name of the first testing group
- `group2`: The name of the second testing group
- `CpGindex`: The index within the CpGs object of the region we are plotting
- `samples`: Names of specific samples in the groups to plot, defaults to all samples in both groups
- `plot_title`: Optional title for the plot, overrides automatic title

**Value**

NULL, the function plots the profiles

**Author(s)**

Tom Mayo <t.mayo@ed.ac.uk>
Examples

# plot the 9th region in the Toy Data Set
data(rrbsDemo)
data(CpGsDemo)
plotMethProfile(rrbsDemo, CpGsDemo, 'H1-hESC', 'K562', 9)

pvals Computes p-values

Description

Returns p-values for each region reflecting the probability of observing the mean test-statistic of the between group comparisons among the inter-replicate comparisons.

Usage

pvals(rrbs, CpGs, MMD, group1, group2, smaller = FALSE,
     comparison = "allReps", method = "empirical", outlier_test = FALSE,
     thresh = NA, cut_off = 0.975, sds = 8, closePara = 0.005)

Arguments

rrbs An rrbs object containing methylation and coverage data as created using the BiSeq package
CpGs A GRanges object with each row being a testing region
MMD A matrix containing the M3D test-statistic, the difference the full and methylation blind metrics, for each region in the CpGs object. Each column is a comparison between two samples, which are described in the column names.
group1 The name of the first group for the comparison. This is stored in colData(rrbs)
group2 The name of the second group for the comparison. This is stored in colData(rrbs)
smaller Determines whether the p-value is computed whether the test-statistic is greater or lesser than inter-replicate values. For our purposes, it should be set to FALSE.
comparison Details which groups we are using to define our empirical testing distribution. The default is to use all of them, however, should the user find one group contains unusually high variability, then that group can be selected. Values are either 'allReps', 'Group1' or 'Group2'.
method Determines which method is used to calculate p-values. 'empirical' uses the empirical distribution directly, without modelling. This is the default. 'model', fits an exponential distribution to the tail of our null distribution.
outlier_test Logical, indicating whether to screen for outliers in null distribution. Use only when comparison is to one group. Default is FALSE
thresh The threshold for cutting off regions as highly variable. This is only to be used if results are being standardised across multiple tests. Default is NA and is calculated using cut_off and sds parameters.
cut_off In the outlier test, we require any outlier to be in this quartile, as a minimum, set to 0.975 by default
sds In the outlier test, we require any outlier in the null to be greater than the mean of the null plus this many standard deviations, set to 8 by default
pvals_lite

closePara

Sets a threshold for how close the exponential curve should fit the empirical distribution in the 'model' method. If the method produces errors, consider raising this parameter.

Value

Returns a list P, with 2 entries. 'FDRmean' is the Benjamini-Hochberg adjusted p-values. The unadjusted p-values are stored in 'Pmean'. If outlier_test is TRUE, we also have the highly variable regions as a third entry.

Author(s)

Tom Mayo <t.mayo@ed.ac.uk>

Examples

data(rrbsDemo)
data(CpGsDemo)
data(M3DlistDemo)
library(GenomicRanges)
library(BiSeq)
M3Dstat <- M3DlistDemo$Full-M3DlistDemo$Coverage
group1 <- unique(colData(rrbsDemo)$group)[1]
group2 <- unique(colData(rrbsDemo)$group)[2]
PDemo <- pvals(rrbsDemo, CpGsDemo, M3Dstat, 
group1, group2, smaller=FALSE,comparison='allReps')
head(PDemo)

pvals_lite

Computes p-values

Description

Returns p-values for each region reflecting the probability of observing the mean test-statistic of the between group comparisons among the inter-replicate comparisons. The 'lite' version works with the outcome of M3D_Wrapper_lite and the 'lite' parallel version.

Usage

pvals_lite(rrbs, CpGs, M3D_stat_lite, group1, group2, smaller = FALSE, 
comparison = "allReps", method = "empirical", outlier_test = FALSE, 
thresh = NA, cut_off = 0.975, sds = 8, closePara = 0.005)

Arguments

rrbs
An rrbs object containing methylation and coverage data as created using the BiSeq package.

CpGs
A GRanges object with each row being a testing region.

M3D_stat_lite
A matrix containing the M3D test-statistic, as returned from the M3D_Wrapper_lite function. This has the mean of the MMDs for the comparisons between groups in the first column, and the within groups MMDs in the others, for each region in the CpGs object.
group1 The name of the first group for the comparison. This is stored in colData(rrbs)
group2 The name of the second group for the comparison. This is stored in colData(rrbs)
smaller Determines whether the p-value is computed whether the test-statistic is greater or lesser than inter-replicate values. For our purposes, it should be set to FALSE.
comparison Details which groups we are using to define our empirical testing distribution. The default is to use all of them, however, should the user find one group contains unusually high variability, then that group can be selected. Values are either ‘allReps’, ‘Group1’ or ‘Group2’.
method Determines which method is used to calculate p-values. ‘empirical’ uses the empirical distribution directly, without modelling. This is the default. ‘model’, fits an exponential distribution to the tail of our null distribution.
outlier_test Logical, indicating whether to screen for outliers in null distribution. Use only when comparison is to one group. Default is FALSE
thresh The threshold for cutting off regions as highly variable. This is only to be used if results are being standardised across multiple tests. Default is NA and is calculated using cut_off and sds parameters.
cut_off In the outlier test, we require any outlier to be in this quartile, as a minimum, set to 0.975 by default
sds In the outlier test, we require any outlier in the null to be greater than the mean of the null plus this many standard deviations, set to 8 by default
closePara Sets a threshold for how close the exponential curve should fit the empirical distribution in the ‘model’ method. If the method produces errors, consider raising this parameter.

Value
Returns a list P, with 2 entries. ‘FDRmean’ is the Benjamini-Hochberg adjusted p-values. The unadjusted p-values are stored in ‘Pmean’. If we are test for outliers, we also have the highly variable regions as a third entry

Author(s)
Tom Mayo <t.mayo@ed.ac.uk>

Examples

data(rrbsDemo)
data(CpGsDemo)
library(GenomicRanges)
library(BiSeq)
CpGsDemo <- CpGsDemo[1:5]
overlaps <- GenomicRanges::findOverlaps(CpGsDemo,rrbsDemo)
M3D_list <- M3D_Wrapper_lite(rrbsDemo,overlaps)
group1 <- unique(colData(rrbsDemo)$group)[1]
group2 <- unique(colData(rrbsDemo)$group)[2]
PDemo <- pvals_lite(rrbsDemo, CpGsDemo, M3D_list, group1, group2, smaller=FALSE,comparison='allReps')
head(PDemo)
Description

Reads in RRBS data in bed file format from various styles and outputs an rrbs data structure. Adapted from readBismark in the BiSeq package.

Usage

```r
readBedFiles(files, colData, bed_type = "Encode", eData = NaN)
```

Arguments

- `files` A character pointing the the rrbs files downloads from the ENCODE database.
- `colData` Samples' names plus additional sample information as character, data.frame or DataFrame.
- `bed_type` Character string representing the style of the bed file. Options are
- `eData` Experiment data to describe the work. This is used to create the BSraw object as in the BiSeq package.

Value

Returns a BSraw object storing methylation and coverage data - the underlying structure for this package.

Author(s)

Tom Mayo <t.mayo@ed.ac.uk>

Examples

```r
# download the files and change the working directory
# to that location
files <- c("wgEncodeHaibMethylRrbsH1hescHaibSitesRep1.bed.gz",
            "wgEncodeHaibMethylRrbsH1hescHaibSitesRep2.bed.gz",
            "wgEncodeHaibMethylRrbsK562HaibSitesRep1.bed.gz",
            "wgEncodeHaibMethylRrbsK562HaibSitesRep2.bed.gz")
group <- factor(c("H1-hESC", "H1-hESC", "K562", "K562"))
samples <- c("H1-hESC1", "H1-hESC2", "K562-1", "K562-2")
colData <- DataFrame(group, row.names = samples)
rrbs <- readBedFiles(files, colData)
```
readBedRaw

**Description**

Reads in RRBS data in bed file format from a variety of formats and outputs a list that is turned into an rrbs data structure in the wrapper function. Adapted from readBismark in the BiSeq package.

**Usage**

```r
readBedRaw(files, colData, bed_type = "Encode")
```

**Arguments**

- `files` A character pointing the the rrbs files downloads from the ENCODE database.
- `colData` Samples’ names plus additional sample information as character, data.frame or DataFrame.
- `bed_type` Character string representing the style of the bed file. Options are

**Value**

Returns a list with all the data that will be turned into a BSraw object - the underlying structure for this package.

**Author(s)**

Tom Mayo <t.mayo@ed.ac.uk>

**Examples**

```r
# download the files and change the working directory
# to that location
files <- c("wgEncodeHaibMethyl1RrbsH1hescHaibSitesRep1.bed.gz",
            "wgEncodeHaibMethyl1RrbsH1hescHaibSitesRep2.bed.gz",
            "wgEncodeHaibMethyl1RrbsK562HaibSitesRep1.bed.gz",
            "wgEncodeHaibMethyl1RrbsK562HaibSitesRep2.bed.gz")
group <- factor(c("H1-HESC","H1-hESC","K562","K562"))
samples <- c("H1-HESC1","H1-hESC2","K562-1","K562-2")
colData <- DataFrame(group,row.names= samples)
rrbs <- readBedRaw(files,colData)
```
**rrbsDemo**

Toy data for the package - methylation data for cytosines sites within the testing regions only, in an rrbs object.

**Description**

Toy data for the package - methylation data for cytosines sites within the testing regions only, in an rrbs object.

**Author(s)**

Tom Mayo

---

**test_colData_structure**

Tests that the files are specified in a workable order for M3D to run

**Description**

M3D requires that the files be specified group by group (all of condition 1, then all of condition 2 etc.). This function tests this, and is used before loading so that the error is caught before time is wasted.

**Usage**

```
test_colData_structure(colData)
```

**Arguments**

- `colData`: Samples’ names plus additional sample information as character, data.frame or DataFrame.

**Value**

Returns TRUE if the colData is structured correctly for downstream analysis, FALSE otherwise

**Author(s)**

Tom Mayo <t.mayo@ed.ac.uk>
Index

*Topic data

- CpGsDemo, 3
- MMDlistDemo, 11
- PDemo, 11
- rrbsDemo, 19

*Topic package

- M3D-package, 2
- CpGsDemo, 3
- determineGroupComps, 3
- findComps, 4
- M3D (M3D-package), 2
- M3D-package, 2
- M3D_Para, 5
- M3D_Para_lite, 6
- M3D_Single, 7
- M3D_Wrapper, 8
- M3D_Wrapper_lite, 9
- median_freq, 10
- medianFreq, 10
- MMDlistDemo, 11
- PDemo, 11
- plotCoverageSampleProfiles, 11
- plotMethProfile, 12
- plotMethSampleProfiles, 13
- pvals, 14
- pvals_lite, 15
- readBedFiles, 17
- readBedRaw, 18
- rrbsDemo, 19
- test_colData_structure, 19