Package ‘MultiDataSet’

April 26, 2017

Type Package

Title Implementation of the BRGE’s (Bioinformatic Research Group in Epidemiology from Center for Research in Environmental Epidemiology) MultiDataSet and MethylationSet

Version 1.4.0

Date 2017-02-21

Description Implementation of the BRGE’s (Bioinformatic Research Group in Epidemiology from Center for Research in Environmental Epidemiology) MultiDataSet and MethylationSet. MultiDataSet is designed for integrating multi omics data sets and MethylationSet to contain normalized methylation data. These package contains base classes for MEAL and rexposome packages.

License file LICENSE

LazyData TRUE

biocViews Software, DataRepresentation

Depends R (>= 3.3), Biobase

Imports BiocGenerics, GenomicRanges, IRanges, S4Vectors, SummarizedExperiment, methods, utils

RoxygenNote 6.0.1

Suggests MEALData, minfi, minfiData, knitr, rmarkdown, testthat, methylumi, omicade4, iClusterPlus, GEOquery, MultiAssayExperiment

VignetteBuilder knitr

NeedsCompilation no

Author Carlos Ruiz-Arenas [aut, cre], Carles Hernandez-Ferrer [aut], Juan R. Gonzalez [aut]

Maintainer Carlos Ruiz-Arenas <carlos.ruiz@isglobal.org>

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>add_eset</td>
<td>2</td>
</tr>
<tr>
<td>add_genexp</td>
<td>3</td>
</tr>
<tr>
<td>add_methy</td>
<td>3</td>
</tr>
<tr>
<td>add_maseq</td>
<td>4</td>
</tr>
</tbody>
</table>
add_eset

Method to add an eSet to MultiDataSet.

Description

This method adds or overwrites a slot of a MultiDataSet with the content of the given eSet.

Usage

```r
add_eset(object, set, dataset.type, dataset.name = NULL, warnings = TRUE, overwrite = FALSE, GRanges)
```

Arguments

- **object**: MultiDataSet that will be filled.
- **set**: Object derived from eSet to be used to fill the slot.
- **dataset.type**: Character with the type of data of the omic set (e.g. expression, methylation...).
- **dataset.name**: Character with the specific name for this set (NULL by default). It is useful when there are several sets of the same type (e.g. multiple expression assays).
- **warnings**: Logical to indicate if warnings will be displayed.
- **overwrite**: Logical to indicate if the set stored in the slot will be overwritten.
- **GRanges**: GenomicRanges to be included in rowRanges slot.

Value

A new MultiDataSet with a slot filled.

See Also

- `add_methy`
- `add_genexp`
- `add_rnaseq`
- `add_snps`
add_genexp

Examples

```r
multi <- createMultiDataSet()
eset <- new("ExpressionSet", exprs = matrix(runif(10), 5))
multi <- add_eset(multi, eset, "exampledata", GRanges = NA)
```

Description

This method adds or overwrites the slot "expression" of an MultiDataSet with the content of the given ExpressionSet. The fData of the ExpressionSet must contain the columns chromosome, start and end.

Usage

```r
add_genexp(object, gexpSet, ...)
```

Arguments

- `object` MultiDataSet that will be filled.
- `gexpSet` ExpressionSet to be used to fill the slot.
- `...` Arguments to be passed to add_eset.

Value

A new MultiDataSet with the slot "expression" filled.

Examples

```r
multi <- createMultiDataSet()
eset <- new("ExpressionSet", exprs = matrix(runif(4), 2))
fData(eset) <- data.frame(chromosome = c("chr1", "chr2"), start = c(12414, 1234321),
end = c(121241, 124124114), stringsAsFactors = FALSE)
multi <- add_genexp(multi, eset)
```

add_methy

Method to add a slot of methylation to MultiDataSet.

Description

This method adds or overwrites the slot "methylation" of an MultiDataSet with the content of the given MethylationSet or RatioSet. The fData of the input object must contain the columns chromosome and position.

Usage

```r
add_methy(object, methySet, ...)
```
add_rnaseq

Method to add an expression RNA seq dataset to MultiDataSet.

Description

This method adds or overwrites the slot "rnaseq" of an MultiDataSet with the content of the given ExpressionSet. The fData of the ExpressionSet must contain the columns chromosome, start and end.

Usage

add_rnaseq(object, rnaSet, ...)

Arguments

object MultiDataSet that will be filled.
rnaSet ExpressionSet to be used to fill the slot.
... Arguments to be passed to add_eset.

Value

A new MultiDataSet with the slot "rnaseq" filled.

Examples

multi <- createMultiDataSet()
eset <- new("ExpressionSet", exprs = matrix(runif(4), 2))
fData(eset) <- data.frame(chromosome = c("chr1", "chr2"), start = c(12414, 1234321),
end = c(121241, 12122414), stringsAsFactors = FALSE)
multi <- add_genexp(multi, eset)
Description

This method adds or overwrites a slot of a MultiDataSet with the content of the given RangedSummarizedExperiment.

Usage

add_rse(object, set, dataset.type, dataset.name = NULL, warnings = TRUE, overwrite = FALSE)

Arguments

object MultiDataSet that will be filled.
set Object derived from RangedSummarizedExperiment to be used to fill the slot.
dataset.type Character with the type of data of the omic set (e.g. expression, methylation...)
dataset.name Character with the specific name for this set (NULL by default). It is useful when there are several sets of the same type (e.g. multiple expression assays)
warnings Logical to indicate if warnings will be displayed.
overwrite Logical to indicate if the set stored in the slot will be overwritten.

Value

A new MultiDataSet with a slot filled.

Examples

if (require(GenomicRanges) & require(SummarizedExperiment)){
  multi <- createMultiDataSet()
  counts <- matrix(runif(200 * 6, 1, 1e4), 200)
  rowRanges <- GRanges(rep(c("chr1", "chr2"), c(50, 150)),
    IRanges(floor(runif(200, 1e5, 1e6)), width=100),
    strand=sample(c("+", "-"), 200, TRUE),
    feature_id=sprintf("ID%03d", 1:200))
  colData <- DataFrame(Treatment=rep(c("ChIP", "Input"), 3),
    row.names=LETTERS[1:6], id = LETTERS[1:6])
  names(rowRanges) <- 1:200
  rse <- SummarizedExperiment(assays=SimpleList(counts=counts),
    rowRanges=rowRanges, colData=colData)
  multi <- add_rse(multi, rse, "rseEx")
}
**add_se**

Method to add a SummarizedExperiment to MultiDataSet.

**Description**

This method adds or overwrites a slot of a MultiDataSet with the content of the given SummarizedExperiment.

**Usage**

```r
add_se(object, set, dataset.type, dataset.name = NULL, warnings = TRUE, overwrite = FALSE, GRanges)
```

**Arguments**

- `object`: MultiDataSet that will be filled.
- `set`: Object derived from SummarizedExperiment to be used to fill the slot.
- `dataset.type`: Character with the type of data of the omic set (e.g. expression, methylation...)
- `dataset.name`: Character with the specific name for this set (NULL by default). It is useful when there are several sets of the same type (e.g. multiple expression assays)
- `warnings`: Logical to indicate if warnings will be displayed.
- `overwrite`: Logical to indicate if the set stored in the slot will be overwritten.
- `GRanges`: GenomicRanges to be included in rowRanges slot.

**Value**

A new MultiDataSet with a slot filled.

**Examples**

```r
multi <- createMultiDataSet()
se <- SummarizedExperiment::SummarizedExperiment(matrix(runif(10), 5))
multi <- add_se(multi, se, "exampledata", GRanges = NA)
```

**add_snps**

Method to add a slot of SNPs to MultiDataSet.

**Description**

This method adds or overwrites the slot "snps" of an MultiDataSet with the content of the given SnpSet. The fData of the SnpSet must contain the columns chromosome and position.

**Usage**

```r
add_snps(object, snpSet, ...)
```
**add_table**

**Arguments**

- **object**: MultiDataSet that will be filled.
- **snpSet**: SnpSet to be used to fill the slot.
- **...**: Arguments to be passed to `add_eset`.

**Value**

A new MultiDataSet with the slot "snps" filled.

**Examples**

```r
multi <- createMultiDataSet()
geno <- matrix(c(3,1,2,1), ncol = 2)
colnames(geno) <- c("VAL0156", "VAL0372")
rownames(geno) <- c("rs3115860", "SNP1-1628854")
map <- AnnotatedDataFrame(data.frame(chromosome = c("chr1", "chr2"), position = c(12414, 1234321), stringsAsFactors = FALSE))
rownames(map) <- rownames(geno)
snpSet <- new("SnpSet", call = geno, featureData = map)
pheno <- data.frame(id = c("VAL0156", "VAL0372"))
rownames(pheno) <- c("VAL0156", "VAL0372")
pData(snpSet) <- pheno
multi <- add_snps(multi, snpSet)
```

---

**Description**

This method adds or overwrites a slot of a MultiDataSet with the content of the given matrix.

**Usage**

```r
add_table(object, set, dataset.type, dataset.name = NULL, warnings = TRUE, overwrite = FALSE)
```

**Arguments**

- **object**: MultiDataSet that will be filled.
- **set**: matrix used to fill the slot.
- **dataset.type**: Character with the type of data
- **dataset.name**: Character with the specific name for this set (NULL by default). It is useful when there are several sets of the same type.
- **warnings**: Logical to indicate if warnings will be displayed.
- **overwrite**: Logical to indicate if the set stored in the slot will be overwritten.

**Value**

A new MultiDataSet with a slot filled.
checkSamples

**Examples**

```r
multi <- createMultiDataSet()
mat <- matrix(runif(12), nrow = 3)
colnames(mat) <- paste0("S", 1:4)
rownames(mat) <- paste0("F", 1:3)
multi <- add_table(multi, mat, "exampledata")
```

**checkProbes**  
*Filter MethylationSet probes*

**Description**

This function selects probes present in the annotation matrix. Probes without annotation and annotation values without beta values are discarded.

**Usage**

```r
checkProbes(object)
```

**Arguments**

- **object**  
  MethylationSet

**Value**

MethylationSet containing the common samples.

**Examples**

```r
if(require(MEALData)){
data(mset)
  checkProbes(mset)
}
```

**checkSamples**  
*Modify a MethylationSet to only contain common samples*

**Description**

This function removes samples that have beta values but no phenotypes and vice versa. If snps object is present, only samples present in the three set are retained.

**Usage**

```r
checkSamples(object)
```

**Arguments**

- **object**  
  MethylationSet
chrNumToChar

Value

MethylationSet containing the common samples.

Examples

```r
if(require(MEALData)){
  data(mset)
  checkSamples(mset)
}
```

chrNumToChar

Convert chr numbers to chr strings

Description

Given a vector of number representing the chromosomes, convert them to string (e.g. 1 to chr1). 23 is considered chrX, 24 is chrY, 25 is chrXY (probes shared between chromosomes X and Y) and 26 is chrMT.

Usage

```r
chrNumToChar(vector)
```

Arguments

- `vector` The vector with the chromosome numbers

Value

A vector with the chromosomes in string format.

Examples

```r
chromosomes <- c(1, 3, 4, 23, 15)
stringChrs <- chrNumToChar(chromosomes)
stringChrs
```

commonIds

Get the name of the ids common to all datasets

Description

Get the name of the ids common to all datasets

Usage

```r
commonIds(object)
```

Arguments

- `object` MultiDataSet that will be filtered.
Value

Character vector with the common ids.

Examples

```r
multi <- createMultiDataSet()
eset <- new("ExpressionSet", exprs = matrix(runif(9), ncol = 3))
fData(eset) <- data.frame(chromosome = c("chr1", "chr1", "chr1"),
                      start = c(1, 5, 10), end = c(4, 6, 14),
                      stringsAsFactors = FALSE)
sampleNames(eset) <- c("S1", "S2", "S3")
pData(eset) <- data.frame(id = c("S1", "S2", "S3"))
rownames(pData(eset)) <- c("S1", "S2", "S3")
multi <- add_genexp(multi, eset, dataset.name = "g1")
eset <- new("ExpressionSet", exprs = matrix(runif(8), ncol = 2))
fData(eset) <- data.frame(chromosome = c("chr1", "chr1", "chr1", "chr1"),
                      start = c(1, 14, 25, 104), end = c(11, 16, 28, 115),
                      stringsAsFactors = FALSE)
sampleNames(eset) <- c("S1", "G2")
pData(eset) <- data.frame(id = c("S1", "G2"))
rownames(pData(eset)) <- c("S1", "G2")
multi <- add_genexp(multi, eset, dataset.name="g2")
commonIds(multi)
```

**commonSamples**

Method to select samples that are present in all datasets.

Description

This method subsets the datasets to only contain the samples that are in all datasets. All sets will have the samples in the same order, taking into account that there can be duplicates.

Usage

```r
commonSamples(object, unify.names = FALSE)
```

Arguments

- **object**
  - MultiDataSet that will be filtered.
- **unify.names**
  - Logical indicating if sample names of the sets should be unified.

Details

If `unify.names` is `TRUE`, the sample names of the sets will be unified using the id column of phenodata. This option is only possible when there are no duplicated ids.

Value

A new MultiDataSet with only the common samples.
**Examples**

```r
multi <- createMultiDataSet()
eset <- new("ExpressionSet", exprs = matrix(runif(9), ncol = 3))
fData(eset) <- data.frame(chromosome = c("chr1", "chr1", "chr1"),
                          start = c(1, 5, 10), end = c(4, 6, 14),
                          stringsAsFactors = FALSE)
sampleNames(eset) <- c("S1", "S2", "S3")
pData(eset) <- data.frame(id = c("S1", "S2", "S3"))
rownames(pData(eset)) <- c("S1", "S2", "S3")
multi <- add_genexp(multi, eset, dataset.name = "g1")
eset <- new("ExpressionSet", exprs = matrix(runif(8), ncol = 2))
fData(eset) <- data.frame(chromosome = c("chr1", "chr1", "chr1", "chr1"),
                          start = c(1, 14, 25, 104), end = c(11, 16, 28, 115),
                          stringsAsFactors = FALSE)
sampleNames(eset) <- c("S1", "G2")
pData(eset) <- data.frame(id = c("S1", "G2"))
rownames(pData(eset)) <- c("S1", "G2")
multi <- add_genexp(multi, eset, dataset.name="g2")
commonSamples(multi)
```

**getMs**

*Transforms beta values to M-values*

**Description**

Given a MethylationSet or a AnalysisResults returns the matrix of M values using a logit2 transformation. Betas equal to 0 will be transformed to threshold and betas equal to 1, to 1 - threshold.

**Usage**

```r
getMs(object, threshold = 1e-04)
```

**Arguments**

- `object` MethylationSet or AnalysisResults
- `threshold` Numeric with the threshold to avoid 0s and 1s.

**Value**

Matrix with the M values.

**Examples**

```r
if(require(MEALData)){
  data(mset)
  Ms <- getMs(mset)
}
```
### mae2mds

*Convert a MultiAssayExperiment to a MultiDataSet*

**Description**

This function creates a MultiDataSet using the data of a MultiAssayExperiment.

**Usage**

```r
mae2mds(MAE, warnings = TRUE)
```

**Arguments**

- **MAE**
  - a `MultiAssayExperiment`
- **warnings**
  - Logical to indicate if warnings will be displayed.

**Value**

A `MultiDataSet` with the data of the incoming `MultiAssayExperiment`.

### mds2mae

*Convert a MultiDataSet to a MultiAssayExperiment*

**Description**

This function creates a MultiAssayExperiment using the data of a MultiDataSet.

**Usage**

```r
mds2mae(MDS)
```

**Arguments**

- **MDS**
  - a `MultiDataSet`

**Value**

A `MultiAssayExperiment` with the data of the incoming `MultiDataSet`.
MethylationSet

MethylationSet instances

Description
Container with the data needed to perform methylation analysis. MethylationSet inherits from eSet and contains meth matrix as assay data member.

Usage
methylationSet(betas, phenotypes, annotationDataFrame, annoString = "custom")

## S4 method for signature 'MethylationSet'
betas(object)

## S4 method for signature 'MethylationSet'
getMs(object, threshold = 1e-04)

## S4 method for signature 'MethylationSet'
checkProbes(object)

## S4 method for signature 'MethylationSet'
checkSamples(object)

Arguments

- betas: Matrix of beta values
- phenotypes: Data.frame or AnnotatedDataFrame with the phenotypes
- annotationDataFrame: Data.frame or AnnotatedDataFrame with the annotation of the methylation sites.
- annoString: Character with the name of the annotation used.
- object: MethylationSet
- threshold: Numeric with the threshold to avoid 0s and 1s.

Details
FeatureData, which contains annotation data, is required to perform any of the analysis.

Value
MethylationSet

Methods (by generic)

- betas: Get beta matrix
- getMs: Get Ms values
- checkProbes: Filter probes with annotation
- checkSamples: Modify a MethylationSet to only contain common samples
Slots

- `assayData` Contains matrices with equal dimensions, and with column number equal to nrow(phenodata).
  assayData must contain a matrix meth with rows representing features (e.g., methylation probes sets) and columns representing samples.

- `phenodata` See eSet

- `annotation` See eSet

- `featureData` See eSet. fData should contain at least chromosome and positions columns.

Examples

```r
test <- showClass("MethylationSet")
```

## MultiDataSet: Implementation of the BRGE’s basic classes

Description

Implementation of the BRGE’s (Bioinformatic Research Group in Epidemiology from Center for Research in Environmental Epidemiology) MultiDataSet and MethylationSet. MultiDataSet is designed for integrating multi-omics data sets and MethylationSet to contain normalized methylation data. MultiDataSet for integrating multi-omics data sets

See Also

- MultiDataSet

## MultiDataSet instances

Description

The class MultiDataSet is a superior class to store multiple datasets in form of triplets (assayData-phenodata-featureData). The datasets must be eSet or SummarizedExperiment.

Usage

```r
## S4 method for signature 'MultiDataSet, eSet'
add_eset(object, set, dataset.type,
         dataset.name = NULL, warnings = TRUE, overwrite = FALSE, GRanges)

## S4 method for signature 'MultiDataSet, ExpressionSet'
add_genexp(object, gexpSet, ...)

## S4 method for signature 'MultiDataSet, ExpressionSet'
add_rnaseq(object, rnaSet, ...)

## S4 method for signature 'MultiDataSet, MethylationSet'
add_methy(object, methySet, ...)
```
## S4 method for signature 'MultiDataSet,RatioSet'
add_methy(object, methySet, ...)

## S4 method for signature 'MultiDataSet,RangedSummarizedExperiment'
add_rse(object, set,
         dataset.type, dataset.name = NULL, warnings = TRUE, overwrite = FALSE)

## S4 method for signature 'MultiDataSet,SummarizedExperiment'
add_se(object, set, dataset.type,
        dataset.name = NULL, warnings = TRUE, overwrite = FALSE, GRanges)

## S4 method for signature 'MultiDataSet,SnpSet'
add_snps(object, snpSet, ...)

## S4 method for signature 'MultiDataSet,matrix'
add_table(object, set, dataset.type,
           dataset.name = NULL, warnings = TRUE, overwrite = FALSE)

## S4 method for signature 'MultiDataSet'
as.list(x)

## S4 method for signature 'MultiDataSet'
commonIds(object)

## S4 method for signature 'MultiDataSet'
commonSamples(object, unify.names = FALSE)

createMultiDataSet()

## S4 method for signature 'MultiDataSet'
dims(object)

## S4 method for signature 'MultiDataSet'
w_iclusterplus(object, commonSamples = TRUE, ...)

## S4 method for signature 'MultiDataSet'
length(x)

## S4 method for signature 'MultiDataSet'
w_mcia(object, ...)

## S4 method for signature 'MultiDataSet'
names(x)

## S4 method for signature 'MultiDataSet'
ncols(object)

## S4 method for signature 'MultiDataSet'
nrows(object)

## S4 method for signature 'MultiDataSet'
rowRangesElements(object)

## S4 method for signature 'MultiDataSet'
rowRangesElements(object)

sampleNames(object)

## S4 method for signature 'MultiDataSet'
sampleNames(object)

assayData(object)

## S4 method for signature 'MultiDataSet'
assayData(object)

fData(object)

## S4 method for signature 'MultiDataSet'
fData(object)

featureData(object)

## S4 method for signature 'MultiDataSet'
featureData(object)

pData(object)

## S4 method for signature 'MultiDataSet'
pData(object)

phenoData(object)

## S4 method for signature 'MultiDataSet'
phenoData(object)

rowRanges(x)

## S4 method for signature 'MultiDataSet'
rowRanges(x)

x[i]

## S4 method for signature 'MultiDataSet'
x[i]

x[i, j, k, ..., drop = FALSE]

## S4 method for signature 'MultiDataSet'
x[i, j, k, ..., drop = FALSE]

subset(x, feat, phe, warnings = TRUE, keep = TRUE)

Arguments

object MultiDataSet

set Object derived from eSet to be used to fill the slot.

dataset.type Character with the type of data of the omic set (e.g. expression, methylation...)

dataset.name Character with the specific name for this set (NULL by default). It is useful when there

warnings Logical to indicate if warnings will be displayed.

overwrite Logical to indicate if the set stored in the slot will be overwritten.

GRanges GenomicRanges to be included in rowRanges slot.

gexpSet ExpressionSet to be used to fill the slot.

... Further arguments passed to add_eset.

rnaSet ExpressionSet to be used to fill the slot.

methySet MethylationSet to be used to fill the slot.

snpSet SnpSet to be used to fill the slot.

x MultiDataSet

unify.names Logical indicating if sample names of the sets should be unified.
MultiDataSet-class

commonSamples Logical to indicate if common samples are selected
i Character corresponding to selected sample names. They should match the id column of phenoData.
j Character with the name of the selected tables.
k GenomicRange used to filter the features.
drop If TRUE, sets with no samples or features will be discarded
feat Logical expression indicating features to keep
phe Logical expression indicating the phenotype of the samples to keep
keep If FALSE, sets where the expression cannot be evaluated will be discarded.

Details
The names of the three lists (assayData, phenoData and featureData) must be the same.

Value
MultiDataSet
MultiDataSet

Methods (by generic)

• add_eset: Method to add an eSet to MultiDataSet.
• add_genexp: Method to add a slot of expression to MultiDataSet.
• add_rnaseq: Method to add a slot of (RNASeq) expression to MultiDataSet.
• add_methy: Method to add a slot of methylation to MultiDataSet.
• add_methy: Method to add a slot of methylation to MultiDataSet.
• add_rse: Method to add a RangedSummarizedExperiment to MultiDataSet.
• add_se: Method to add a SummarizedExperiment to MultiDataSet.
• add_snps: Method to add a slot of SNPs to MultiDataSet.
• add_table: Method to add a matrix to MultiDataSet.
• as.list: Returns a list with the first matrix of each dataset.
• commonIds: Get the name of the ids common to all datasets
• commonSamples: Get a MultiDataSet only with the samples present in all the tables
• dims: Returns the dimensions of the sets
• w_iclusterplus: Apply iClusterPlus clustering method to a MultiDataSet object
• length: Returns the number of sets into the object.
• w_mcia: Apply mcia integration method to a MultiDataSet object
• names: Get the names of the slots.
• ncol: Get number of samples of each set
• nrow: Get number of features of each set
• rowRangesElements: Get the name of the datasets that have rowRanges
• sampleNames: Get sample names
• assayData: Retrieve all assay data blocks.
• fData: Retrieve information on features.
• **featureData**: Retrieve information on features.
• **pData**: Retrieve information on experimental phenotypes
• **phenoData**: Retrieve information on experimental phenotypes
• **rowRanges**: Retrieve information on feature ranges.
• **\[\]**: Get a set from a slot
• **\[\]**: Subset a MultiDataSet
• **subset**: Filter a subset using feature ids or phenotypes

**Slots**

- **assayData**: List of assayData elements.
- **phenoData**: List of AnnotatedDataFrame containing the phenoData of each assayData.
- **featureData**: List of AnnotatedDataFrame containing the featureData of each assayData.
- **rowRanges**: List of GenomicRanges containing the rowRanges of each assayData.
- **return_method**: List of functions used to create the original object.

**See Also**

- `add_eset`, `add_rse`

**Examples**

```r
createMultiDataSet()
```

---

**rowRangesElements**

*Get the name of the datasets that have rowRanges*

**Description**

Get the name of the datasets that have rowRanges

**Usage**

```r
rowRangesElements(object)
```

**Arguments**

- **object**: MultiDataSet

**Value**

Character vector with the slots that have rowRanges.
Examples

```r
multi <- createMultiDataSet()
eset <- new("ExpressionSet", exprs = matrix(runif(10), 5))
eset2 <- new("ExpressionSet", exprs = matrix(runif(8), ncol = 2))
fData(eset2) <- data.frame(chromosome = c("chr1", "chr1", "chr1", "chr1"),
                          start = c(1, 14, 25, 104), end = c(11, 16, 28, 115),
                          stringsAsFactors = FALSE)
multi <- add_eset(multi, eset, "exampledata", GRanges = NA)
multi <- add_genexp(multi, eset2)
rowRangesElements(multi)
```

### w_iclusterplus

Apply iClusterPlus clustering method to a MultiDataSet object

#### Description
Method iClusterPlus is applied on a MultiDataSet object after getting the common samples along all the contained datasets.

#### Usage

```r
w_iclusterplus(object, commonSamples = TRUE, ...)
```

#### Arguments
- **object**: MultiDataSet
- **commonSamples**: Logical to indicate if common samples are selected
- **...**: Arguments passed to function iClusterPlus

#### Value
A list of results from iClusterPlus

#### Note
Argument type for iClusterPlus is filled within the method.

### w_mcia

Apply mcia integration method to a MultiDataSet object

#### Description
Method mcia is applied on a MultiDataSet object after getting the common samples along all the contained datasets.

#### Usage

```r
w_mcia(object, ...)
```
Arguments

object MultiDataSet
... Arguments passed to function mcia

Value

A list of results from mcia
Index

[ (MultiDataSet-class), 14
[ ,MultiDataSet,ANY,ANY,ANY-method (MultiDataSet-class), 14
[ [,MultiDataSet,ANY,ANY-method (MultiDataSet-class), 14

add_eset, 2, 18
add_eset,MultiDataSet,eSet-method (MultiDataSet-class), 14
add_genexp, 2, 3
add_genexp,MultiDataSet,ExpressionSet-method (MultiDataSet-class), 14
add_methy, 2, 3
add_methy,MultiDataSet,MethylationSet-method (MultiDataSet-class), 14
add_methy,MultiDataSet,RatioSet-method (MultiDataSet-class), 14
add_rnaseq, 2, 4
add_rnaseq,MultiDataSet,ExpressionSet-method (MultiDataSet-class), 14
add_rse, 5, 18
add_rse,MultiDataSet,RangedSummarizedExperiment-method (MultiDataSet-class), 14
add_se, 6
add_se,MultiDataSet,SummarizedExperiment-method (MultiDataSet-class), 14
add_snps, 2, 6
add_snps,MultiDataSet,SnpSet-method (MultiDataSet-class), 14
add_table, 7
add_table,MultiDataSet,matrix-method (MultiDataSet-class), 14
as.list (MultiDataSet-class), 14
as.list,MultiDataSet-method (MultiDataSet-class), 14
assayData (MultiDataSet-class), 14
assayData,MultiDataSet-method (MultiDataSet-class), 14

betas (MethylationSet), 13
betas,MethylationSet-method (MethylationSet), 13

checkProbes, 8
checkProbes,MethylationSet-method (MethylationSet), 13
checkSamples, 8
checkSamples,MethylationSet-method (MethylationSet), 13
chrNumToChar, 9
commonIds, 9
commonIds,MethylationSet-method (MethylationSet), 13
commonSamples, 10
commonSamples,MethylationSet-method (MethylationSet), 13
createMultiDataSet (MultiDataSet-class), 14
dims (MultiDataSet-class), 14
dims,MethylationSet-method (MethylationSet), 13
eSet, 14
fData (MultiDataSet-class), 14
fData,MultiDataSet-method (MultiDataSet-class), 14
featureData (MultiDataSet-class), 14
featureData,MultiDataSet-method (MultiDataSet-class), 14
getMs, 11
getMs,MethylationSet-method (MethylationSet), 13
iClusterPlus, 19
length (MultiDataSet-class), 14
length,MultiDataSet-method (MultiDataSet-class), 14
mae2mds, 12
mcia, 19, 20
mds2mae, 12
MethylationSet, 13
methylationSet (MethylationSet), 13
MethylationSet-class (MethylationSet), 13
MethylationSet-methods
   (MethylationSet), 13
MultiDataSet, 14, 14, 19
MultiDataSet-class, 14
MultiDataSet-methods
   (MultiDataSet-class), 14
MultiDataSet-methods,
   (MultiDataSet-class), 14
MultiDataSet-package (MultiDataSet), 14
names (MultiDataSet-class), 14
names, MultiDataSet-method
   (MultiDataSet-class), 14
ncols (MultiDataSet-class), 14
ncols, MultiDataSet-method
   (MultiDataSet-class), 14
nrows (MultiDataSet-class), 14
nrows, MultiDataSet-method
   (MultiDataSet-class), 14
pData (MultiDataSet-class), 14
pData, MultiDataSet-method
   (MultiDataSet-class), 14
phenoData (MultiDataSet-class), 14
phenoData, MultiDataSet-method
   (MultiDataSet-class), 14
rowRanges (MultiDataSet-class), 14
rowRanges, MultiDataSet-method
   (MultiDataSet-class), 14
rowRangesElements, 18
rowRangesElements, MultiDataSet-method
   (MultiDataSet-class), 14
sampleNames, MultiDataSet-method
   (MultiDataSet-class), 14
subset, MultiDataSet-method
   (MultiDataSet-class), 14
w_iclusterplus, 19
w_iclusterplus, MultiDataSet-method
   (MultiDataSet-class), 14
w_mcia, 19
w_mcia, MultiDataSet-method
   (MultiDataSet-class), 14