Package ‘methylGSA’

March 5, 2024

**Type** Package

**Title** Gene Set Analysis Using the Outcome of Differential Methylation

**Version** 1.20.0

**Description** The main functions for methylGSA are methylglm and methylRRA. methylGSA implements logistic regression adjusting number of probes as a covariate.
methylRRA adjusts multiple p-values of each gene by Robust Rank Aggregation.
For more detailed help information, please see the vignette.

**Encoding** UTF-8

**Imports** RobustRankAggreg, ggplot2, stringr, stats, clusterProfiler,
missMethyl, org.Hs.eg.db, reactome.db, BiocParallel, GO.db,
AnnotationDbi, shiny,
IlluminaHumanMethylation450kanno.ilmn12.hg19,
IlluminaHumanMethylationEPIanno.ilm10b4.hg19

**Depends** R (>= 3.5)

**Suggests** knitr, rmarkdown, testthat, enrichplot

**License** GPL-2

**URL** [https://github.com/reese3928/methylGSA](https://github.com/reese3928/methylGSA)

**BugReports** [https://github.com/reese3928/methylGSA/issues](https://github.com/reese3928/methylGSA/issues)

**RoxygenNote** 6.1.0

**VignetteBuilder** knitr

**biocViews** DNAMethylation,DifferentialMethylation,GeneSetEnrichment,Regression,
GeneRegulation,Pathways

**git_url** [https://git.bioconductor.org/packages/methylGSA](https://git.bioconductor.org/packages/methylGSA)

**git_branch** RELEASE_3_18

**git_last_commit** 8cb9bb1

**git_last_commit_date** 2023-10-24

**Repository** Bioconductor 3.18

**Date/Publication** 2024-03-04
Description

This function visualizes methylGSA analysis result by barplot.

Usage

barplot(res, xaxis = "Size", num = 5, colorby = "padj", title = "")

Arguments

res
A data frame which contains methylGSA analysis result.

xaxis
A string which specify the x-axis in the barplot. Either "Size" (number of genes in gene set) or "Count" (number of significant genes in gene set). Default is "Size". "Count" option is not available for methylglm and methylRRA(GSEA) result.

num
An integer. Number of gene sets to display on the barplot. Default is 5.

colorby
A string. Either "pvalue" or "padj". Default is "padj".

title
A string. Barplot title. Default is NULL.

Details

The implementation of the function is adapted from barplot function in enrichplot package.
cpg.pval

Value

ggplot object

References


Examples

```r
res = data.frame(ID = c("04144", "04510", "04740", "04810", "05200"),
                 Description = c("Endocytosis", "Focal adhesion",
                                "Olfactory transduction",
                                "Regulation of actin cytoskeleton", "Pathways in cancer"),
                 Size = c(201, 200, 388, 213, 326),
                 pvalue = c(0.481, 0.696, 1, 1, 1),
                 padj = 1
)
barplot(res)
```

Description

An example of user input cpg.pval

Usage

cpg.pval

Format

A named vector contains p-values of each probe tested

CpG2Gene

An example of user user-supplied mapping between CpGs and genes

Description

An example of user user-supplied mapping between CpGs and genes

Usage

CpG2Gene

Format

A data frame contains mapping between CpGs and genes
getAnnot  

Get CpG annotation

Description

This function gets CpG IDs and their corresponding gene symbols.

Usage

getAnnot(array.type, group = "all")

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>array.type</td>
<td>A string. Either &quot;450K&quot; or &quot;EPIC&quot;. Default is &quot;450K&quot;.</td>
</tr>
<tr>
<td>group</td>
<td>A string. &quot;all&quot;, &quot;body&quot;, &quot;promoter1&quot; or &quot;promoter2&quot;. Default is &quot;all&quot;.</td>
</tr>
<tr>
<td></td>
<td>If group = &quot;body&quot;, only CpGs on gene body will be pulled out. If group =</td>
</tr>
<tr>
<td></td>
<td>&quot;promoter1&quot; or group = &quot;promoter2&quot;, only CpGs on promoters will be pulled</td>
</tr>
<tr>
<td></td>
<td>out. Here is the definition of &quot;body&quot;, &quot;promoter1&quot; and &quot;promoter2&quot;</td>
</tr>
<tr>
<td></td>
<td>according to the annotation in IlluminaHumanMethylation450kanno.ilmm12.hg19</td>
</tr>
<tr>
<td></td>
<td>or IlluminaHumanMethylationEPICanno.ilm10b4.hg19.</td>
</tr>
<tr>
<td></td>
<td>• body: CpGs whose gene group correspond to &quot;Body&quot; or &quot;1stExon&quot;</td>
</tr>
<tr>
<td></td>
<td>• promoter1: CpGs whose gene group correspond to &quot;TSS1500&quot; or &quot;TSS200&quot;</td>
</tr>
<tr>
<td></td>
<td>• promoter2: CpGs whose gene group correspond to &quot;TSS1500&quot;, &quot;TSS200&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;1stExon&quot;, or &quot;5’UTR&quot;.</td>
</tr>
<tr>
<td></td>
<td>If group = &quot;all&quot;, all CpGs will be pulled out.</td>
</tr>
</tbody>
</table>

Details

The implementation of the function is modified from .flattenAnn function in missMethyl package.

Value

A data frame contains CpG IDs and gene symbols.

References


**getDescription**

*Get gene set description*

**Description**

This function gets description of gene sets.

**Usage**

```r
getDescription(GSids, GS.type)
```

**Arguments**

- **GSids**: A vector contains gene set IDs.
- **GS.type**: A string. "GO", "KEGG", or "Reactome".

**Value**

A vector contains gene sets description.

**References**


**Examples**

```r
GSids = c("GO:0007389", "GO:0000978", "GO:0043062")
Description = getDescription(GSids, "GO")
head(Description)
```

**getGS**

*Get Gene Sets*

**Description**

This function gets gene sets information.

**Usage**

```r
getGS(geneids, GS.type)
```
Arguments

geneids A vector contains all gene ids of interest. Gene ids should be gene symbol.
GS.type A string. "GO", "KEGG", or "Reactome".

Value

A list contains all gene sets of interest and their corresponding genes.

References


Examples

geneids = c("FKBP5", "NDUFA1", "STAT5B")
GO.list = getGS(geneids, "KEGG")
head(GO.list)

GS.list An example of user input gene sets

Description

An example of user input gene sets

Usage

GS.list

Format

A list contains user input gene set names and their corresponding genes
methylglm

Implement logistic regression adjusting for number of probes in enrichment analysis

Description

This function implements logistic regression adjusting for number of probes in enrichment analysis.

Usage

methylglm(cpg.pval, array.type = "450K", FullAnnot = NULL,
group = "all", GS.list = NULL, GS.idtype = "SYMBOL",
GS.type = "GO", minsize = 100, maxsize = 500, parallel = FALSE,
BPPARAM = bpparam())

Arguments

cpg.pval A named vector containing p-values of differential methylation test. Names should be CpG IDs.
array.type A string. Either "450K" or "EPIC". Default is "450K". This argument will be ignored if FullAnnot is provided.
FullAnnot A data frame provided by prepareAnnot function. Default is NULL.
group A string. "all", "body", "promoter1" or "promoter2". Default is "all". If group = "body", only CpGs on gene body will be considered in methylglm. If group = "promoter1" or group = "promoter2", only CpGs on promoters will be considered. Here is the definition of "body", "promoter1" and "promoter2" according to the annotation in IlluminaHumanMethylation450kanno.ilmn12.hg19 or IlluminaHumanMethylationEPICanno.ilm10b4.hg19.
  • body: CpGs whose gene group correspond to "Body" or "1stExon"
  • promoter1: CpGs whose gene group correspond to "TSS1500" or "TSS200"
  • promoter2: CpGs whose gene group correspond to "TSS1500", "TSS200", "1stExon", or "5'UTR".

If group = "all", all CpGs are considered regardless of their gene group.

GS.list A list. Default is NULL. If there is no input list, Gene Ontology is used. Entry names are gene sets names, and elements correpond to genes that gene sets contain.
GS.idtype A string. "SYMBOL", "ENSEMBL", "ENTREZID" or "REFSEQ". Default is "SYMBOL"
GS.type A string. "GO", "KEGG", or "Reactome". Default is "GO"
minsize An integer. If the number of genes in a gene set is less than this integer, this gene set is not tested. Default is 100.
maxsize An integer. If the number of genes in a gene set is greater than this integer, this gene set is not tested. Default is 500.
methylgometh

parallel  either TRUE or FALSE indicating whether parallel should be used. Default is FALSE
BPPARAM an argument provided to bplapply. See register for details.

Details
The implementation of this function is modified from goglm function in GOglm package.

Value
A data frame contains gene set tests results.

References

Examples
data(CpG2Genetoy)
data(cpgtoy)
data(GSlisttoy)
GS.list = GS.list[1:10]
FullAnnot = prepareAnnot(CpG2Gene)
res = methylglm(cpg.pval = cpg.pval, FullAnnot = FullAnnot,
GS.list = GS.list, GS.idtype = "SYMBOL")
head(res)

methylgometh Adjusting number of probes in gene set testing using gometh or gsameth in missMethyl

Description
This function calls gometh or gsameth function in missMethyl package to adjust number of probes in gene set testing

Usage
methylgometh(cpg.pval, sig.cut = 0.001, topDE = NULL,
array.type = "450K", GS.list = NULL, GS.idtype = "SYMBOL",
GS.type = "GO", minsize = 100, maxsize = 500)
**methylgometh**

**Arguments**

- **cpg.pval**
  A named vector containing p-values of differential methylation test. Names should be CpG IDs.

- **sig.cut**
  A numeric value indicating cut-off value for significant CpG. Default is 0.001. This argument will be ignored if topDE is provided.

- **topDE**
  An integer. The top number of CpGs to be declared as significant.

- **array.type**
  A string. Either "450K" or "EPIC". Default is "450K".

- **GS.list**
  A list. Default is NULL. If there is no input list, Gene Ontology is used. Entry names are gene sets names, and elements correspond to genes that gene sets contain.

- **GS.idtype**
  A string. "SYMBOL", "ENSEMBL", "ENTREZID" or "REFSEQ". Default is "SYMBOL".

- **GS.type**
  A string. "GO", "KEGG", or "Reactome".

- **minsize**
  An integer. If the number of genes in a gene set is less than this integer, this gene set is not tested. Default is 100.

- **maxsize**
  An integer. If the number of genes in a gene set is greater than this integer, this gene set is not tested. Default is 500.

**Value**

A data frame contains gene set tests results.

**References**


**Examples**

```r
## Not run:
library(IlluminaHumanMethylation450kanno.ilmn12.hg19)
data(cpgtoy)
res = methylgometh(cpg.pval = cpg.pval, sig.cut = 0.001, GS.type = "KEGG",
minsize = 200, maxsize = 205)
head(res)
## End(Not run)
```
**methylRRA**

*Enrichment analysis after adjusting multiple p-values of each gene by Robust Rank Aggregation*

**Description**

This function implements enrichment after adjusting multiple p-values of each gene by Robust Rank Aggregation.

**Usage**

```r
methylRRA(cpg.pval, array.type = "450K", FullAnnot = NULL,
          group = "all", method = "ORA", sig.cut = 0.05, topDE = NULL,
          GS.list = NULL, GS.idtype = "SYMBOL", GS.type = "GO",
          minsize = 100, maxsize = 500)
```

**Arguments**

- `cpg.pval` - A named vector containing p-values of differential methylation test. Names should be CpG IDs.
- `array.type` - A string. Either "450K" or "EPIC". Default is "450K". This argument will be ignored if FullAnnot is provided.
- `FullAnnot` - A data frame provided by prepareAnnot function. Default is NULL.
- `group` - A string. "all", "body", "promoter1" or "promoter2". Default is "all". If group = "body", only CpGs on gene body will be considered in methylRRA. If group = "promoter1" or group = "promoter2", only CpGs on promoters will be considered. Here is the definition of "body", "promoter1" and "promoter2" according to the annotation in IlluminaHumanMethylation450kanno.ilmn12.hg19 or IlluminaHumanMethylationEPICanno.ilm10b4.hg19.
  - body: CpGs whose gene group correspond to "Body" or "1stExon"
  - promoter1: CpGs whose gene group correspond to "TSS1500" or "TSS200"
  - promoter2: CpGs whose gene group correspond to "TSS1500", "TSS200", "1stExon", or "5'UTR".
- `method` - A string. "ORA" or "GSEA". Default is "ORA".
- `sig.cut` - A numeric value indicating FDR cut-off for significant gene in ORA. Default is 0.05. This argument will be ignored if topDE is provided or method = "GSEA" is used.
- `topDE` - An integer. The top number of genes to be declared as significant after robust rank aggregation. This argument will be ignored if method = "GSEA" is used.
- `GS.list` - A list. Default is NULL. If there is no input list, Gene Ontology is used. Entry names are gene sets names, and elements correspond to genes that gene sets contain.
prepareAnnot

GS.idtype  A string. "SYMBOL", "ENSEMBL", "ENTREZID" or "REFSEQ". Default is "SYMBOL".
GS.type  A string. "GO", "KEGG", or "Reactome". Default is "GO"
minsize  An integer. If the number of genes in a gene set is less than this integer, this gene set is not tested. Default is 100.
maxsize  An integer. If the number of genes in a gene set is greater than this integer, this gene set is not tested. Default is 500.

Value

A data frame contains gene set tests results.

References


Examples

data(CpG2Genetoy)
data(cpgtoy)
data(GSlisttoy)
GS.list = GS.list[1:10]
FullAnnot = prepareAnnot(CpG2Gene)
res1 = methylRRA(cpg.pval = cpg.pval, FullAnnot = FullAnnot, method = "ORA", GS.list = GS.list)
head(res1)

prepareAnnot

Prepare user-supplied mapping between CpGs and genes.

Description

This function prepares CpG to gene mapping which will be used by methylRRA and methylglm.

Usage

prepareAnnot(CpG2Gene, geneidtype = "SYMBOL")
Arguments

CpG2Gene  A matrix, or a data frame or a list contains CpG to gene mapping. For a matrix or data frame, 1st column should be CpG ID and 2nd column should be gene name. For a list, entry names should be gene names, and elements correspond to CpG IDs.

geneidtype  A string. "SYMBOL", "ENSEMBL", "ENTREZID" or "REFSEQ". Default is "SYMBOL".

Value

A data frame contains ready to use CpG to gene mapping.

References


Examples

data(CpG2Genetoy)
FullAnnot = prepareAnnot(CpG2Gene)
head(FullAnnot)

Description

This is an interface for Bioconductor package methylGSA.

Usage

runExample(run = TRUE)

Arguments

run  Run the app or not. Default is TRUE

Value

The shiny app will be opened in a web browser.

Note

In order to run the app, the following R/Bioconductor packages needs to be installed properly: shinyCSSloaders, DT, ggplot2, IlluminaHumanMethylation450anno.ilmn12.hg19 (if analyzing 450K array) IlluminaHumanMethylationEPICan.ilm10b4.hg19 (if analyzing EPIC array)
Examples

```r
## Please note: in this example, the argument run is set to be FALSE in
## order to pass R CMD check. However, when using the app, users are
## expected to launch the app by runExample()
runExample(FALSE)
```
Index

* datasets
  cpg.pval, 3
  CpG2Gene, 3
  GS.list, 6

barplot, 2
bplapply, 8

cpg.pval, 3
CpG2Gene, 3

getAnnot, 4
getDescription, 5
getGS, 5
GS.list, 6

methylglm, 7
methylgomeh, 8
methylRRA, 10

prepareAnnot, 11

register, 8
runExample, 12