Package ‘NoRCE’

September 25, 2022

Type Package

Title NoRCE: Noncoding RNA Sets Cis Annotation and Enrichment

Version 1.8.0

Description While some non-coding RNAs (ncRNAs) are assigned critical regulatory roles, most remain functionally uncharacterized. This presents a challenge whenever an interesting set of ncRNAs needs to be analyzed in a functional context. Transcripts located close-by on the genome are often regulated together. This genomic proximity on the sequence can hint to a functional association. We present a tool, NoRCE, that performs cis enrichment analysis for a given set of ncRNAs. Enrichment is carried out using the functional annotations of the coding genes located proximal to the input ncRNAs. Other biologically relevant information such as topologically associating domain (TAD) boundaries, co-expression patterns, and miRNA target prediction information can be incorporated to conduct a richer enrichment analysis. To this end, NoRCE includes several relevant datasets as part of its data repository, including cell-line specific TAD boundaries, functional gene sets, and expression data for coding & ncRNAs specific to cancer. Additionally, the users can utilize custom data files in their investigation. Enrichment results can be retrieved in a tabular format or visualized in several different ways. NoRCE is currently available for the following species: human, mouse, rat, zebrafish, fruit fly, worm, and yeast.

License MIT + file LICENSE

Depends R (>= 4.0)

Imports KEGGREST,png,dplyr,graphics,RSQLite,DBI,tidy,grDevices,
S4Vectors,SummarizedExperiment,reactome.db,rWikiPathways,RCurl,
dplyr,utils,ggplot2,igraph,stats,reshape2,readr,
GO.db,zlibbioc,
biomaRt,rtracklayer,IRanges,GenomicRanges,GenomicFeatures,AnnotationDbi

Encoding UTF-8

RoxygenNote 7.1.1

Suggests knitr,
TxDb.Hsapiens.UCSC.hg38.knownGene,TxDB.Drerio.UCSC.danRer10.refGene,
TxDb.Mmuscus.UCSC.mm10.knownGene,TxDB.Dmelanogaster.UCSC.dm6.ensGene,
testthat,TxDB.Celegans.UCSC.ce11.ensGene,rmarkdown,
TxDb.Rnorvegicus.UCSC.m6.refGene,TxDB.Hsapiens.UCSC.hg19.knownGene,
org.Mm.eg.db,
R topics documented:

org.Rn.eg.db, org.Hs.eg.db, org.Dr.eg.db, BiocGenerics,
org.Sc.sgd.db, org.Ce.eg.db, org.Dm.eg.db, methods, markdown

**VignetteBuilder**  knitr

**biocViews**  BiologicalQuestion, DifferentialExpression,
GenomeAnnotation, GeneSetEnrichment, GeneTarget,
GenomeAssembly, GO

**LazyData**  true

**BugReports**  [https://github.com/guldenolgun/NoRCE/issues](https://github.com/guldenolgun/NoRCE/issues)

**git_url**  https://git.bioconductor.org/packages/NoRCE

**git_branch**  RELEASE_3_15

**git_last_commit**  ffeef02

**git_last_commit_date**  2022-04-26

**Date/Publication**  2022-09-25

**Author**  Gulden Olgun [aut, cre]

**Maintainer**  Gulden Olgun <gulden@cs.bilkent.edu.tr>

R topics documented:

- annGO .................................................. 3
- assembly ................................................. 4
- brain_disorder_ncRNA ............................. 4
- brain_mirna ........................................... 5
- breastmRNA ......................................... 5
- calculateCorr ........................................ 6
- convertGeneID ....................................... 7
- convertGMT .......................................... 8
- corrbased .......................................... 8
- corrbasedMrna ..................................... 9
- createNetwork ...................................... 10
- drawDotPlot ......................................... 11
- extractBiotype ...................................... 11
- filterBiotype ....................................... 12
- geneGOEnricher .................................... 12
- genePathwayEnricher ............................... 14
- geneRegionGOEnricher ......................... 16
- geneRegionPathwayEnricher ..................... 18
- getGoDag ............................................. 20
- getKeggDiagram .................................... 21
- getmiRNACount ..................................... 22
- getNearToExon ..................................... 22
- getNearToIntron ................................... 23
- getReactomeDiagram ............................... 24
- getTADOverlap ..................................... 25
- getUCSC ............................................. 26
annGO

Annotate the set of genes with the GO terms for a given species and assembly

Description

Annotate the set of genes with the GO terms for a given species and assembly

Usage

```r
annGO(
  genes,
  GOtype = c("BP", "CC", "MF"),
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3")
)
```

Arguments

- **genes**
  - List of mRNA genes. Supported format for genes is Hugo.
- **GOtype**
  - Hierarchical category of the GO ontology. Possible values are 'BP', 'CC', 'MF'.
- **org_assembly**
  - Genome assembly of interest. Possible assemblies are 'mm10' for mouse, 'dre10' for zebrafish, 'rn6' for rat, 'dm6' for fruit fly, 'ce11' for worm, 'hg19' and 'hg38' for human
Value

data frame of the GO term annotation of the genes

---

**assembly**

*Get the required information for the given assembly*

Description

Get the required information for the given assembly

Usage

```r
assembly(
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3")
)
```

Arguments

| org_assembly | Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human |

Value

setting required information

Examples

```r
assembly('hg19')
```

---

**brain_disorder_ncRNA**

*Differentially expressed non-coding gene*

Description

Differentially expressed non-coding gene

Usage

```r
brain_disorder_ncRNA
```

Format

Not Available
\textit{brain\_mirna}

\textbf{Source}

http://resource.psychencode.org/

\textbf{Examples}

\begin{verbatim}
data(brain_disorder_ncRNA)
\end{verbatim}

---

\textit{brain\_mirna} \hspace{1cm} \textit{Differentially expressed human brain data}

\textbf{Description}

Differentially expressed human brain data

\textbf{Usage}

\texttt{brain\_mirna}

\textbf{Format}

Not Available

\textbf{Source}

http://resource.psychencode.org/

\textbf{Examples}

\begin{verbatim}
data(brain_mirna)
\end{verbatim}

---

\textit{breast\_mRNA} \hspace{1cm} \textit{Protein coding genes that are differentially expressed in TCGA breast cancer RNAseq data.}

\textbf{Description}

Protein coding genes that are differentially expressed in TCGA breast cancer RNAseq data.

\textbf{Usage}

\texttt{breast\_mRNA}

\textbf{Format}

Not Available
calculateCorr

Source
https://portal.gdc.cancer.gov/

Examples

data(breastmRNA)

---

calculateCorr

Calculates the correlation coefficient values between two custom expression data.

Description
Calculates the correlation coefficient values between two custom expression data.

Usage

calculateCorr(
  exp1,
  exp2,
  label1 = "",
  label2 = "",
  corrMethod = "pearson",
  varCutoff = 0.0025,
  corCutoff = 0.3,
  pcut = 0.05,
  alternate = "greater",
  conf = 0.95
)

Arguments

- **exp1**: Custom expression data matrix or SummarizedExperiment data. Columns must be genes and rows must be patients.
- **exp2**: Custom expression data matrix or SummarizedExperiment data. Columns must be genes and rows must be patients.
- **label1**: Gene names of the custom exp1 expression data. If it is not provided, column name of the exp1 data will be taken.
- **label2**: Gene names of the custom exp2 expression data. If it is not provided, column name of the exp2 data will be taken.
- **corrMethod**: Correlation coefficient method that will be used for evaluation. Possible values are "pearson", "kendall", "spearman"
- **varCutoff**: Variance cut off that genes have less variance than this value will be trimmed.
- **corCutoff**: Correlation cut off values for the given correlation method.
- **pcut**: P-value cut off for the correlation values.
- **alternate**: Comparison alternative.
- **conf**: Confidence level for the correlation test.
convertGeneID

alternate
Holds the alternative hypothesis and "two.sided", "greater" or "less" are the possible values.

conf
Confidence level for the returned confidence interval. It is only used for the Pearson correlation coefficient if there are at least 4 complete pairs of observations.

Value
Pairwise relations between gene-gene with corresponding correlation value and pvalue

Examples
#Assume that mirnanorce and mrnanorce are custom patient by gene data
a<-calculateCorr(exp1 = mirna, exp2 = mrna )

convertGeneID

Convert gene ids according to the gene type

Description
Convert gene ids according to the gene type

Usage
convertGeneID(
  genotype = c("Entrez", "mirna", "Ensembl_gene", "Ensembl_trans", "NCBI"),
  genelist,
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3")
)

Arguments

  genotype
    Type of the input gene list. Possible values are "Entrez", "mirna", "Ensembl_gene", "Ensembl_trans", "NCBI". For HUGO gene symbol "NCBI" value, for Entrez gene id "Entrez", for mirbase id "mirna" is used.

  genelist
    Input gene list

  org_assembly
    Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human

Value
GRange object of the given input
Examples

```r
convGene <- convertGeneID(genetype = "mirna",
                          genelist = brain_mirna[1:30,],
                          org_assembly = 'hg19')
```

---

**convertGMT**

*Convert gmt formatted pathway file to the Pathway ID, Entrez, symbol formatted data frame*

---

**Description**

Convert gmt formatted pathway file to the Pathway ID, Entrez, symbol formatted data frame

**Usage**

```r
convertGMT(gmtName, org_assembly, isSymbol = FALSE)
```

**Arguments**

- `gmtName`: Custom pathway gmt file
- `org_assembly`: Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human
- `isSymbol`: Boolean variable that hold the gene format of the gmt file. If it is set as TRUE, gene format of the gmt file should be symbol. Otherwise, gene format should be ENTREZ ID. By default, it is FALSE.

**Value**

*return data frame*

---

**corrbased**

*Pearson correlation coefficient value of the miRNA genes between miRNA:mRNA for a given correlation cut-off and cancer.*

---

**Description**

Pearson correlation coefficient value of the miRNA genes between miRNA:mRNA for a given correlation cut-off and cancer.

**Usage**

```r
corrbased(mirnagene, cancer, minAbsCor, databaseFile)
```
**Arguments**

- **mirnagene**  
  Data frame of the miRNA genes in mature format

- **cancer**  
  Name of the TCGA project code such as 'BRCA' that is analyzed for miRNA-mRNA correlation. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COAD, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRC, KIRP, LGG, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM

- **minAbsCor**  
  Cut-off value for the Pearson correlation coefficient of the miRNA-mRNA

- **databaseFile**  
  Path of the miRcancer.db file

**Value**

Data frame of the miRNA-mRNA correlation result

---

**corrbasedMrna**  
*Pearson correlation coefficient value of the mRNA genes between miRNA:mRNA for a given correlation cut-off and cancer.*

**Description**

Pearson correlation coefficient value of the mRNA genes between miRNA:mRNA for a given correlation cut-off and cancer.

**Usage**

`corrbasedMrna(mRNAgene, cancer, minAbsCor, databaseFile)`

**Arguments**

- **mRNAgene**  
  Data frame of the mRNA genes

- **cancer**  
  Name of the TCGA project code such as 'BRCA' that is analyzed for miRNA-mRNA correlation. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COAD, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRC, KIRP, LGG, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM

- **minAbsCor**  
  Cut-off value for the Pearson correlation coefficient of the miRNA-mRNA

- **databaseFile**  
  Path of miRcancer.db file

**Value**

Data frame of the miRNA-mRNA correlation result
createNetwork

Create interaction network for top n enriched GO term:coding RNA or GO-term:noncoding RNA interaction. Nodes are GO term and RNA, edges are interactions between them. Each GO-term is annotated and enriched with the mRNAs provided from the input list.

Description

Create interaction network for top n enriched GO term:coding RNA or GO-term:noncoding RNA interaction. Nodes are GO term and RNA, edges are interactions between them. Each GO-term is annotated and enriched with the mRNAs provided from the input list.

Usage

createNetwork(
  mrnaObject,
  type = "pvalue",
  n,
  isNonCode = FALSE,
  takeID = FALSE
)

Arguments

mrnaObject  Output of enrichment results
type        Sort in terms of p-values or FDR. Possible values "pvalue", "padjust"
n           Number of top enrichments
isNonCode   Boolean value that checks whether node of the network is GO-term\& coding or GO-term\& noncoding genes. By default, it is FALSE so node of the network is GO-term\& coding gene. Otherwise, nodes are GO-term\& noncoding genes.
takeID      Boolean value that checks the name decision of the GO/pathway node, GO-term/pathway-term or GO ID/pathway ID. If it is true, name of the GO/pathway node will be GO ID/pathway ID will be used, otherwise, name of the GO/pathway node is GO-term. By default, it is FALSE. It is suggested to used when the GO-term is two long or the GO-term is missing for the custom enrichment database.

Value

Network
**drawDotPlot**

*Draw dot plot of the enrichment object*

**Description**
Draw dot plot of the enrichment object

**Usage**
drawDotPlot(mrnaObject, type = "pAdjust", n)

**Arguments**
- mrnaObject: Object of the enrichment result
- type: Draw the dot plot according to the p-value or adjusted p-value ("pvalue", "pAdjust")
- n: Number of GO terms or pathways, that ordered by type and has least number of top p-value

**Value**
Dot plot of the top n enrichment results

**extractBiotype**

*Get the biotype of the non-coding genes. It is suitable for the GENCODE gtf files*

**Description**
Get the biotype of the non-coding genes. It is suitable for the GENCODE gtf files

**Usage**
extractBiotype(gtfFile)

**Arguments**
- gtfFile: Path of the input gtf file which contains biotype information. The gtf file must be provided from the Ensembl or Gencode site. For space efficiency, gtf files should be in a zip format.

**Value**
Tabular form of the gtf file with the required features such as gene id and biotypes
Examples

fileImport <- system.file("extdata", "temp.gtf", package = "NoRCE")
gtf <- extractBiotype(gtfFile = fileImport)

filterBiotype

Extract the genes that have user provided biotypes. This method is useful when input gene list is mixed or when research of the interest is only focused on specific group of genes.

Description

Extract the genes that have user provided biotypes. This method is useful when input gene list is mixed or when research of the interest is only focused on specific group of genes.

Usage

filterBiotype(gtfFile, biotypes)

Arguments

gtfFile Input gtf file for the genes provided by the extractBiotype function
biotypes Selected biotypes for the genes

Value

Table format of genes with a given biotypes

Examples

biotypes <- c("unprocessed_pseudogene","transcribed_unprocessed_pseudogene")
fileImport <- system.file("extdata", "temp.gtf", package = "NoRCE")
extrResult <- filterBiotype(fileImport, biotypes)

geneGOEnricher

Given genes that fall in a given upstream and downstream region of mRNAs of interest, GO term enrichment analysis is carried out.

Description

Given genes that fall in a given upstream and downstream region of mRNAs of interest, GO term enrichment analysis is carried out.
geneGOEnricher

Usage
geneGOEnricher(
  gene,
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
  genetype = c("Entrez", "mirna", "Ensembl_gene", "Ensembl_trans", "NCBI"),
  backG = "",
  backGType = "pc_gene",
  near = FALSE,
  isTADSearch = FALSE,
  TAD = c(tad_hg19, tad_dmel, tad_hg38, tad_mm10),
  express = FALSE,
  isCustomExp = FALSE,
  cancer,
  exp1,
  exp2,
  label1 = "",
  label2 = "",
  isUnionCorGene = FALSE,
  databaseFile
)

Arguments
gene  Input genes other than miRNA
org_assembly  Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human
genetype  Type of the input gene list. Possible values are "Entrez", "mirna", "Ensembl_gene", "Ensembl_trans", "NCBI". For HUGO gene symbol "NCBI" value, for Entrez gene id "Entrez" is used.
backG  The set of genes that tested against to the input (background gene)
backGType  Type of the background gene. If miRNA gene set is used for background gene, backGType should be set to the 'mirna'
near  Boolean value presents whether cis-neighbourhood should be considered in the analysis
isTADSearch  Boolean value that shows whether TAD analysis is performed. This value has to be TRUE for TAD analysis.
TAD  TAD genomic regions for the species. Predefined TAD regions or any new TAD regions can be used for the analysis. TAD regions must be formatted as GRanges object. Predefined TAD regions are 'tad_hg19', 'tad_hg38', 'tad_mm10', 'tad_dmel' for hg19, hg38, mm9 and dm6 assembly, respectively.
express  Boolean variable whether co-expression analysis is performed. If this option is set to TRUE, co-expression analysis will be performed.
isCustomExp  Boolean variable whether co-expression analysis with custom data will be performed. When this option is set, exp1 and exp2 parameters must be defined.
cancer

Defines the name of the TCGA project code such as 'BRCA' for correlation analysis. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COAD, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRC, KIRP, LGG, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM

exp1

Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.

exp2

Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.

label1

Gene names of the custom exp1 expression data. If it is not provided, column name of the exp1 data will be taken.

label2

Gene names of the custom exp2 expression data. If it is not provided, column name of the exp2 data will be taken.

isUnionCorGene

Boolean value that shows whether union of the output of the co-expression analysis and the other analysis should be considered

databaseFile

Path of miRcancer.db file

Value

GO term enrichment object for the given input

Examples

```r
cGO<-geneGOEnricher(gene = brain_disorder_ncRNA, org_assembly='hg19', near=TRUE, genetype = 'Ensembl_gene')
```

genePathwayEnricher

Given genes that fall in the given upstream and downstream region of mRNAs of interest, pathway enrichment analysis is carried out

Description

Given genes that fall in the given upstream and downstream region of mRNAs of interest, pathway enrichment analysis is carried out

Usage

genePathwayEnricher(
    gene,
    org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
    genetype = c("Entrez", "mirna", "Ensembl_gene", "Ensembl_trans", "NCBI"),
    near = TRUE,
    isTADSearch = FALSE,
)
genePathwayEnricher

TAD = tad_hg19,
gmtName = "",
express = FALSE,
isCustomExp = FALSE,
cancer,
exp1,
exp2,
label1 = "",
label2 = "",
isUnionCorGene = FALSE,
databaseFile,
isGeneEnrich = FALSE
)

Arguments

gene Input noncoding genes other than miRNA
org_assembly Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human

genetype Type of the input gene list. Possible values are "Entrez", "mirna", "Ensembl_gene", "Ensembl_trans", "NCBI". For HUGO gene symbol "NCBI" value, for Entrez gene id "Entrez", for mirbase id "mirna" is used.
near Boolean value presents whether cis-neighbourhood should be considered in the analysis
isTADSearch Boolean value that shows whether TAD analysis is performed. This value has to be TRUE for TAD analysis.
TAD TAD genomic regions for the species. Predefined TAD regions or any new TAD regions can be used for the analysis. TAD regions must be formatted as GRanges object. Predefined TAD regions are 'tad_hg19', 'tad_hg38', 'tad_mm10', 'tad_dmel' for hg19, hg38, mm9 and dm6 assembly, respectively.
gmtName Custom pathway gmt file
express Boolean variable whether co-expression analysis is performed. If this option is set to TRUE, co-expression analysis will be performed.
isCustomExp Boolean variable whether co-expression analysis with custom data will be performed. When this option is set, exp1 and exp2 parameters must be defined.
cancer Defines the name of the TCGA project code such as 'BRCA' for correlation analysis. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRC, KIRP, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM, LGG
exp1 Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.
exp2  Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.

label1  Gene names of the custom exp1 expression data. If it is not provided, column name of the exp1 data will be taken.

label2  Gene names of the custom exp2 expression data. If it is not provided, column name of the exp2 data will be taken.

isUnionCorGene  Boolean value that shows whether union of the output of the co-expression analysis and the other analysis should be considered.

databaseFile  Path of miRcancer.db file.

isGeneEnrich  Boolean value whether gene enrichment should be performed.

Value

Pathway enrichment object for the given input.

Examples

#Pathway enrichment based on the gen sets that falls into the TAD regions
cRNAPathway<-genePathwayEnricher(gene = brain_disorder_ncRNA,
   org_assembly = "hg19",
   isTADSearch = TRUE,
   TAD = tad_hg19,
   genotype = "Ensembl_gene")

geneRegionGOEnricher  Given gene regions that fall in the given upstream and downstream region of mRNAs of interest, GO term enrichment analysis is carried out.

Description

Given gene regions that fall in the given upstream and downstream region of mRNAs of interest, GO term enrichment analysis is carried out.

Usage

geneRegionGOEnricher(
   region,
   org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
   near = TRUE,
   backG = "",
   backGType = "pc_gene",
   isTADSearch = FALSE,
   TAD = c(tad_hg19, tad_dmel, tad_hg38, tad_mm10),
)
geneRegionGOEnricher

```r
express = FALSE,
isCustomExp = FALSE,
cancer,
exp1,
exp2,
label1 = "",
label2 = "",
isUnionCorGene = FALSE,
databaseFile
)`

**Arguments**

- **region**
  - Bed format of the input gene regions other than miRNA
- **org_assembly**
  - Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human
- **near**
  - Boolean value presents whether cis-neighbourhood should be considered in the analysis
- **backG**
  - The set of genes that tested against to the input (background gene)
- **backGType**
  - Type of the background gene. If miRNA gene set is used for background gene, backGType should be set to the 'mirna'
- **isTADSearch**
  - Boolean value that shows whether TAD analysis is performed. This value has to be TRUE for TAD analysis.
- **TAD**
  - TAD genomic regions for the species. Predefined TAD regions or any new TAD regions can be used for the analysis. TAD regions must be formatted as GRanges object. Predefined TAD regions are 'tad_hg19', 'tad_hg38', 'tad_mm10', 'tad_dmel' for hg19, hg38, mm9 and dm6 assembly, respectively.
- **express**
  - Boolean variable whether co-expression analysis is performed. If this option is set to TRUE, co-expression analysis will be performed.
- **isCustomExp**
  - Boolean variable whether co-expression analysis with custom data will be performed. When this option is set, exp1 and exp2 parameters must be defined.
- **cancer**
  - Defines the name of the TCGA project code such as 'BRCA' for correlation analysis. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COAD, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRC, KIRP, LGG, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM
- **exp1**
  - Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.
- **exp2**
  - Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.
- **label1**
  - Gene names of the custom exp1 expression data. If it is not provided, column name of the exp1 data will be taken.
label2: Gene names of the custom exp2 expression data. If it is not provided, column name of the exp2 data will be taken.

isUnionCorGene: Boolean value that shows whether union of the output of the co-expression analysis and the other analysis should be considered.

databaseFile: Path of miRcancer.db file

Value

GO term enrichment object for the given input

Examples

```
regions <- system.file("extdata", "ncRegion.txt", package = "NoRCE")
regionNC <- rtracklayer::import(regions, format = "BED")
regionGO <- geneRegionGOEnricher(region = regionNC, org_assembly = "hg19",
                                near = TRUE)
```

Description

Given gene regions that fall in the given upstream and downstream region of mRNAs of interest, pathway enrichment analysis is carried out.

Usage

```
geneRegionPathwayEnricher(
  region,
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
  near = FALSE,
  isTADSearch = FALSE,
  TAD = tad_hg19,
  gmtName = "",
  express = FALSE,
  isCustomExp = FALSE,
  cancer,
  exp1,
  exp2,
  label1 = "",
  label2 = "",
  isUnionCorGene = FALSE,
  databaseFile,
  isGeneEnrich = FALSE
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>region</td>
<td>Bed format of input gene regions other than miRNA. Input must be Granges object.</td>
</tr>
<tr>
<td>org_assembly</td>
<td>Genome assembly of interest for the analysis. Possible assemblies are &quot;mm10&quot; for mouse, &quot;dre10&quot; for zebrafish, &quot;rn6&quot; for rat, &quot;dm6&quot; for fruit fly, &quot;ce11&quot; for worm, &quot;sc3&quot; for yeast, &quot;hg19&quot; and &quot;hg38&quot; for human</td>
</tr>
<tr>
<td>near</td>
<td>Boolean value presents whether cis-neighbourhood should be considered in the analysis</td>
</tr>
<tr>
<td>isTADSearch</td>
<td>Boolean value that shows whether TAD analysis is performed. This value has to be TRUE for TAD analysis.</td>
</tr>
<tr>
<td>TAD</td>
<td>TAD genomic regions for the species. Predefined TAD regions or any new TAD regions can be used for the analysis. TAD regions must be formatted as GRanges object. Predefined TAD regions are 'tad_hg19', 'tad_hg38', 'tad_mm10', 'tad_dmel' for hg19, hg38, mm9 and dm6 assembly, respectively.</td>
</tr>
<tr>
<td>gmtName</td>
<td>Custom pathway gmt file</td>
</tr>
<tr>
<td>express</td>
<td>Boolean variable whether co-expression analysis is performed. If this option is set to TRUE, co-expression analysis will be performed.</td>
</tr>
<tr>
<td>isCustomExp</td>
<td>Boolean variable whether co-expression analysis with custom data will be performed. When this option is set, exp1 and exp2 parameters must be defined.</td>
</tr>
<tr>
<td>cancer</td>
<td>Defines the name of the TCGA project code such as 'BRCA' for correlation analysis. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COAD, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRC, KIRP, LGG, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM</td>
</tr>
<tr>
<td>exp1</td>
<td>Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.</td>
</tr>
<tr>
<td>exp2</td>
<td>Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.</td>
</tr>
<tr>
<td>label1</td>
<td>Gene names of the custom exp1 expression data. If it is not provided, column name of the exp1 data will be taken.</td>
</tr>
<tr>
<td>label2</td>
<td>Gene names of the custom exp2 expression data. If it is not provided, column name of the exp2 data will be taken.</td>
</tr>
<tr>
<td>isUnionCorGene</td>
<td>Boolean value that shows whether union of the output of the co-expression analysis and the other analysis should be considered</td>
</tr>
<tr>
<td>databaseFile</td>
<td>Path of miRcancer.db file</td>
</tr>
<tr>
<td>isGeneEnrich</td>
<td>Boolean value whether gene enrichment should be performed</td>
</tr>
</tbody>
</table>

Value

Pathway enrichment object of the given input
getGoDag

**Examples**

```r
regions <- system.file("extdata", "ncRegion.txt", package = "NoRCE")
regionNC <- rtracklayer::import(regions, format = "BED")
ncPath <- geneRegionPathwayEnricher(region = regionNC,
  org_assembly = "hg19",
  near = TRUE)
```

---

**Description**

Plot and save the GO term DAG of the top n enrichments in terms of p-values or adjusted p-values with an user provided format

**Usage**

```r
getGoDag(
  mrnaObject,
  type,
  n,
  filename,
  imageFormat,
  p_range = seq(0, 0.05, by = 0.001)
)
```

**Arguments**

- `mrnaObject`: Output of enrichment results
- `type`: Sort in terms of p-values or FDR. possible values "pvalue", "padjust"
- `n`: Number of top enrichments
- `filename`: Name of the DAG file
- `imageFormat`: Image format of the DAG. possible values "png" or "svg"
- `p_range`: Break points for the p-values or FDR. By default [0.05, 0.001, 0.0005, 0.0001, 0.00005,0.00001,0] is used

**Value**

Saves image file in a given format

**Examples**

```r
ncRNAPathway <- mirnaPathwayEnricher(gene = brain_mirna,
  org_assembly = 'hg19', near = TRUE)
```
getKeggDiagram

Display the enriched KEGG diagram of the KEGG pathway. This function is specific to only one KEGG pathway id and identifies the enriched genes in the diagram.

Description

Display the enriched KEGG diagram of the KEGG pathway. This function is specific to only one KEGG pathway id and identifies the enriched genes in the diagram.

Usage

getKeggDiagram(
  mrnaObject,
  pathway,
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3")
)

Arguments

mrnaObject  Output of enrichment results
pathway     Kegg pathway term such as 'hsa04010'
org_assembly Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human

Value

Shows kegg diagram marked with an enriched genes in a browser

Examples

ncRNAPathway<-mirnaPathwayEnricher(gene = brain_mirna,
  org_assembly = 'hg19',near = TRUE)

getKeggDiagram(mrnaObject = ncRNAPathway, org_assembly = 'hg19',
  pathway = ncRNAPathway@ID[1])
getmiRNACount

*Get TCGA miRNAseq expression of miRNA genes for the given cancer*

**Description**

Get TCGA miRNAseq expression of miRNA genes for the given cancer

**Usage**

```
getmiRNACount(mirnagene, cancer, databaseFile)
```

**Arguments**

- **mirnagene**: Data frame of the mature format
- **cancer**: Name of the TCGA project code such as 'BRCA' that is analyzed for miRNA-mRNA correlation
- **databaseFile**: Path of miRcancer.db file

**Value**

Data frame of the raw read count of the given miRNA genes for different patients

getNearToExon

*Get only those neighbouring genes that fall within exon region*

**Description**

Get only those neighbouring genes that fall within exon region

**Usage**

```
getNearToExon(
    bedfile,
    upstream,
    downstream,
    org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3")
)
```

**Arguments**

- **bedfile**: Input bed formatted file
- **upstream**: Maximum upstream distance from the TSS position
- **downstream**: Maximum downstream distance from the TES position
- **org_assembly**: Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human
getNearToIntron

Value

genes

Examples

regions <- system.file("extdata", "ncRegion.txt", package = "NoRCE")
regionNC <- rtracklayer::import(regions, format = "BED")

r<-getNearToExon(bedfile = regionNC,
    upstream = 1000,
    downstream = 2000,
    org_assembly = 'hg19')

getNearToIntron

Get only those neighbouring genes that fall within intron region

Description

Get only those neighbouring genes that fall within intron region

Usage

getNearToIntron(
    bedfile,
    upstream,
    downstream,
    org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3")
)

Arguments

bedfile Bed file
upstream upstream distance
downstream downstream distance
org_assembly genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human

Value

genes
getReactomeDiagram

Display the enriched Reactome diagram of the given Reactome pathway id. This function is specific to only one pathway id and identifies the enriched genes in the diagram.

**Usage**

```r
getReactomeDiagram(mrnaObject, pathway, imageFormat)
```

**Arguments**

- `mrnaObject`: Output of enrichment results
- `pathway`: Reactome pathway term
- `imageFormat`: Image format of the diagram. Possible image formats are 'png', 'svg'

**Value**

Shows reactome diagram marked with an enriched genes in a browser

**Examples**

```r
br_enr<-reactomeEnrichment(genes = breastmRNA, org_assembly='hg19')

getReactomeDiagram(mrnaObject = br_enr, pathway = br_enr@ID[1],
                   imageFormat = 'png')
```
getTADOverlap

For given region of interest, overlapped genes in the TAD regions are found. Results can be filtered according to the available cell lines.

Description

For given region of interest, overlapped genes in the TAD regions are found. Results can be filtered according to the available cell lines.

Usage

getTADOverlap(
    bedfile,
    org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
    tad = c(tad_hg19, tad_dmel, tad_hg38, tad_mm10),
    near = FALSE,
    upstream = 10000,
    downstream = 10000,
    cellline = "all"
)

Arguments

bedfile Region of interest
org_assembly Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human
tad TAD genomic regions for the species. Predefined TAD regions or any new TAD regions can be used for the analysis. TAD regions must be formatted as GRanges object. Predefined TAD regions are 'tad_hg19', 'tad_hg38', 'tad_mm10', 'tad_dmel' for hg19, hg38, mm9 and dm6 assembly, respectively.
near Boolean value presents whether cis-neighbourhood should be considered in the analysis
upstream Holds upstream distance from the transcription start position
downstream Holds downstream distance from the transcription end position
cellline Cell lines for TAD regions.

Value

List of protein coding genes that falls into the TAD regions
Examples

```r
regions<-system.file("extdata", "ncRegion.txt", package = "NoRCE")
regionNC <- rtracklayer::import(regions, format = "BED")

r<-getTADOverlap(bedfile = regionNC,
                 tad = tad_hg19,
                 org_assembly = 'hg19',
                 cellline = 'HUVEC')
```

getUCSC

Get nearest genes for the window of the upstream/downstream region.

Description

When downstream = 0 / upstream = 0, function converts bed formatted regions to HUGO genes

Usage

```r
getUCSC(  
  bedfile, 
  upstream, 
  downstream, 
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3")
)
```

Arguments

- **bedfile**: Bed formatted input gene regions
- **upstream**: Maximum upstream distance from the transcription start region of the input gene
- **downstream**: Maximum downstream distance from the transcription end region of the input gene
- **org_assembly**: genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human

Value

- **genes**: genes

Examples

```r
regions<-system.file("extdata", "ncRegion.txt", package = "NoRCE")
regionNC <- rtracklayer::import(regions, format = "BED")

neighbour <- getUCSC(bedfile = regionNC,
                      upstream = 1000,
                      downstream = 1000)
```
### goEnrichment

Perform enrichment analysis of the given genes

#### Usage

```r
goEnrichment(
  genes,
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
  GOtype = c("BP", "CC", "MF"),
  pCut = 0.05,
  pAdjCut = 0.05,
  pAdjust = c("holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"),
  min = 5,
  backG = "",
  backGType = "pc_gene",
  enrichTest = c("hyper", "binom", "fisher", "chi")
)
```

#### Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>genes</td>
<td>Set of input genes. Supported format HUGO.</td>
</tr>
<tr>
<td>org_assembly</td>
<td>Genome assembly of interest for the analysis. Possible assemblies are &quot;mm10&quot; for mouse, &quot;dre10&quot; for zebrafish, &quot;rn6&quot; for rat, &quot;dm6&quot; for fruit fly, &quot;ce11&quot; for worm, &quot;sc3&quot; for yeast, &quot;hg19&quot; and &quot;hg38&quot; for human</td>
</tr>
<tr>
<td>GOtype</td>
<td>Hierarchical category of the GO ontology. Possible values are &quot;BP&quot;(default), &quot;CC&quot;, &quot;MF&quot;.</td>
</tr>
<tr>
<td>pCut</td>
<td>Threshold value for the pvalue. Default value is 0.05</td>
</tr>
<tr>
<td>pAdjCut</td>
<td>Cutoff value for the adjusted p-values using one of given method. Default value is 0.05.</td>
</tr>
<tr>
<td>pAdjust</td>
<td>Methods of the adjusted p-values. Possible methods are &quot;bonferroni&quot;, &quot;holm&quot;, &quot;BH&quot;(default)</td>
</tr>
<tr>
<td>min</td>
<td>Minimum number of gene that are required for enrichment. By default, it is set to 5</td>
</tr>
<tr>
<td>backG</td>
<td>The set of genes that tested against to the input (background gene)</td>
</tr>
<tr>
<td>backGType</td>
<td>Type of the background gene. If miRNA gene set is used for background gene, backGType should be set to the 'mirna'</td>
</tr>
<tr>
<td>enrichTest</td>
<td>Types of enrichment methods to perform enrichment analysis. Possible values are &quot;hyper&quot;(default), &quot;binom&quot;, &quot;fisher&quot;, &quot;chi&quot;.</td>
</tr>
</tbody>
</table>
Value

GO enrichment results

Examples

```r
subsetGene <- breastmRNA[1:30,]
breastEnr <- goEnrichment(genes = subsetGene,
                        org_assembly = 'hg19',
                        GOtype = 'MF',
                        min = 2)
```

KeggEnrichment

### KEGG pathway enrichment

**Description**

KEGG pathway enrichment

**Usage**

```r
KeggEnrichment(
  genes,
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
  pCut = 0.05,
  pAdjCut = 0.05,
  pAdjust = c("holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"),
  min = 5,
  gmtFile = "",
  isSymbol = "",
  isGeneEnrich = ""
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>genes</td>
<td>Input genes</td>
</tr>
<tr>
<td>org_assembly</td>
<td>Genome assembly of interest for the analysis. Possible assemblies are &quot;mm10&quot; for mouse, &quot;dre10&quot; for zebrafish, &quot;rn6&quot; for rat, &quot;dm6&quot; for fruit fly, &quot;ce11&quot; for worm, &quot;sc3&quot; for yeast, &quot;hg19&quot; and &quot;hg38&quot; for human</td>
</tr>
<tr>
<td>pCut</td>
<td>Threshold value for the pvalue. Default value is 0.05</td>
</tr>
<tr>
<td>pAdjCut</td>
<td>Cutoff value for the adjusted p-values using one of given method. Default value is 0.05.</td>
</tr>
<tr>
<td>pAdjust</td>
<td>Methods of the adjusted p-values. Possible methods are &quot;holm&quot;, &quot;hochberg&quot;, &quot;hommel&quot;, &quot;bonferroni&quot;, &quot;BH&quot;, &quot;BY&quot;, &quot;fdr&quot;, &quot;none&quot;</td>
</tr>
<tr>
<td>min</td>
<td>Minimum number of genes that are required for enrichment. By default, it is set to 5.</td>
</tr>
</tbody>
</table>
gmtFile  File path of the gmt file
isSymbol  Boolean value that controls the gene formats. If it is TRUE, gene format of the
          gmt file should be symbol. Otherwise, gene format must be ENTREZ ID.
isGeneEnrich  Boolean value whether gene enrichment should be performed

Value

KEGG pathway enrichment results

Examples

```r
subsetGene <- breastmRNA[1:30,]
br_enr<-KeggEnrichment(genes = subsetGene,
                        org_assembly='hg19')
```

listTAD  List cell line of the given topological domain regions

Description

List cell line of the given topological domain regions

Usage

`listTAD(TADName)`

Arguments

  TADName  input TAD regions

Value

  cell line of the input tad data

Examples

`listTAD(TADName = tad_hg19)`
mirna

*Brain miRNA expression retrieved from the TCGA*

**Description**

Brain miRNA expression retrieved from the TCGA

**Usage**

mirna

**Format**

Not Available

**Source**

https://www.gencodegenes.org/

**Examples**

```
data(mirna)
```

---

mirnaGOEnricher

*GO term enrichments of the microRNA genes with mRNAs that fall in the given upstream/downstream regions of the microRNA genes*

**Description**

GO term enrichments of the microRNA genes with mRNAs that fall in the given upstream/downstream regions of the microRNA genes

**Usage**

```
mirnaGOEnricher(  
gene,  
org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),  
near = FALSE,  
target = FALSE,  
backGenes = "",  
backGType = "pc_gene",  
isNewSearch = FALSE,  
TAD = c(tad_hg19, tad_dmel, tad_hg38, tad_mm10),  
express = FALSE,  
isCustomExp = FALSE,  
cancer,
```
exp1,
exp2,
label1 = "",
label2 = "",
isUnionCorGene = FALSE,
databaseFile = ""
)

Arguments

gene Input microRNA gene. It supports both pre-miRNA and mature miRNA, however, when target prediction is performed (target= TRUE), miRNA genes should be mature.

org.assembly Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human

near Boolean value presents whether cis-neighbourhood should be considered in the analysis

target Boolean value shows whether miRNA target prediction should be performed

backGenes The set of genes that tested against to the input

backGType Type of the background gene. If miRNA gene set is used for background gene, backGType should be set to the 'mirna'

isTADSearch Boolean value that shows whether TAD analysis is performed. This value has to be TRUE for TAD analysis.

TAD TAD genomic regions for the species. Predefined TAD regions or any new TAD regions can be used for the analysis. TAD regions must be formatted as GRanges object. Predefined TAD regions are 'tad_hg19', 'tad_hg38', 'tad_mm10', 'tad_dmel' for hg19, hg38, mm9 and dm6 assembly, respectively.

express Boolean variable whether co-expression analysis is performed. If this option is set to TRUE, co-expression analysis will be performed.

isCustomExp Boolean variable whether co-expression analysis with custom data will be performed. When this option is set, exp1 and exp2 parameters must be defined.

cancer Defines the name of the TCGA project code such as 'BRCA' for correlation analysis. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COAD, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRC, KIRP, LGG, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM

exp1 Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.

exp2 Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.

label1 Gene names of the custom exp1 expression data. If it is not provided, column name of the exp1 data will be taken.
mirnaPathwayEnricher

**Value**

MiRNA GO term enrichment object for the given input

**Examples**

```r
subsetGene <- brain_mirna[1:30,]

miGO <-mirnaGOEnricher(gene=subsetGene,
                         org_assembly='hg19',
                         near = TRUE,
                         target = FALSE)
```

**Description**

Pathway enrichments of the microRNA genes with mRNAs that fall in the given upstream/downstream regions of the microRNA genes

**Usage**

```r
mirnaPathwayEnricher(
  gene,
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
  near = FALSE,
  target = FALSE,
  isTADSearch = FALSE,
  TAD = c(tad_hg19, tad_dmel, tad_hg38, tad_mm10),
  gmtName = "",
  express = FALSE,
  isCustomExp = FALSE,
  cancer,
  exp1,
  exp2,
  label1 = "",
  label2 = "",
  isUnionCorGene = FALSE,
  databaseFile,
  isGeneEnrich = FALSE
)
```
Arguments

- **gene**: Input microRNA gene. It supports both pre-miRNA and mature miRNA, however, when target prediction is performed (target= TRUE), miRNA genes should be mature.

- **org_assembly**: Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human

- **near**: Boolean value presents whether cis-neighbourhood should be considered in the analysis

- **target**: Boolean value shows whether miRNA target prediction should be performed

- **isTADSearch**: Boolean value that shows whether TAD analysis is performed. This value has to be TRUE for TAD analysis.

- **TAD**: TAD genomic regions for the species. Predefined TAD regions or any new TAD regions can be used for the analysis. TAD regions must be formatted as GRanges object. Predefined TAD regions are ‘tad_hg19’, ‘tad_hg38’, ‘tad_mm10’, ‘tad_dmel’ for hg19, hg38, mm9 and dm6 assembly, respectively.

- **gmtName**: Custom pathway gmt file

- **express**: Boolean variable whether co-expression analysis is performed. If this option is set to TRUE, co-expression analysis will be performed.

- **isCustomExp**: Boolean variable whether co-expression analysis with custom data will be performed. When this option is set, exp1 and exp2 parameters must be defined.

- **cancer**: Defines the name of the TCGA project code such as ‘BRCA’ for correlation analysis. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COAD, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRC, KIRP, LGG, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM

- **exp1**: Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.

- **exp2**: Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.

- **label1**: Gene names of the custom exp1 expression data. If it is not provided, column name of the exp1 data will be taken.

- **label2**: Gene names of the custom exp2 expression data. If it is not provided, column name of the exp2 data will be taken.

- **isUnionCorGene**: Boolean value that shows whether union of the output of the co-expression analysis and the other analysis should be considered

- **databaseFile**: Path of miRcancer.db file

- **isGeneEnrich**: Boolean value whether gene enrichment should be performed

Value

MiRNA pathway enrichment object for the given input
Examples

```r
miPath <- mirnaPathwayEnricher(gene = brain_mirna,
    org_assembly = 'hg19',
    near = TRUE)
```

**mirnaRegionGOEnricher**

*GO enrichments of the microRNA regions with mRNAs that fall in the given upstream/downstream regions of the microRNA genes*

**Description**

GO enrichments of the microRNA regions with mRNAs that fall in the given upstream/downstream regions of the microRNA genes

**Usage**

```r
mirnaRegionGOEnricher(
    region,
    org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
    near = FALSE,
    target = FALSE,
    backG = "",
    backGType = "pc-genes",
    isTADSearch = FALSE,
    TAD = c(tad_hg19, tad_dmel, tad_hg38, tad_mm10),
    express = FALSE,
    isCustomExp = FALSE,
    cancer,
    exp1,
    exp2,
    label1 = "",
    label2 = "",
    isUnionCorGene = FALSE,
    databaseFile
)
```

**Arguments**

- **region**: MiRNA region in a bed format
- **org_assembly**: Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human
- **near**: Boolean value presents whether cis-neighbourhood should be considered in the analysis
- **target**: Boolean value shows whether miRNA target prediction should be performed
mirnaRegionGOEnricher

backG  The set of genes that tested against the input

backGType  Type of the background gene. If miRNA gene set is used for background gene, backGType should be set to the 'mirna'

isTADSearch  Boolean value that shows whether TAD analysis is performed. This value has to be TRUE for TAD analysis.

TAD  TAD genomic regions for the species. Predefined TAD regions or any new TAD regions can be used for the analysis. TAD regions must be formatted as GRanges object. Predefined TAD regions are 'tad_hg19', 'tad_hg38', 'tad_mm10', 'tad_dmel' for hg19, hg38, mm9 and dm6 assembly, respectively.

express  Boolean variable whether co-expression analysis is performed. If this option is set to TRUE, co-expression analysis will be performed.

isCustomExp  Boolean variable whether co-expression analysis with custom data will be performed. When this option is set, exp1 and exp2 parameters must be defined.

cancer  Defines the name of the TCGA project code such as 'BRCA' for correlation analysis. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COAD, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRC, KIRP, LGG, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM

exp1  Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.

exp2  Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.

label1  Gene names of the custom exp1 expression data. If it is not provided, column name of the exp1 data will be taken.

label2  Gene names of the custom exp2 expression data. If it is not provided, column name of the exp2 data will be taken.

isUnionCorGene  Boolean value that shows whether union of the output of the co-expression analysis and the other analysis should be considered

databaseFile  Path of miRcancer.db file

Value

MiRNA GO enrichment object for the given input

Examples

```r
regions<-'system.file("extdata", "ncRegion.txt", package = "NoRCE")
regionNC <- rtracklayer::import(regions, format = "BED")

a<- mirnaRegionGOEnricher(region = regionNC, 
                          org_assembly = 'hg19', 
                          near = TRUE)
```
mirnaRegionPathwayEnricher

Pathway enrichments of the microRNA regions with mRNAs that fall in the given upstream/downstream regions of the microRNA genes

Description

Pathway enrichments of the microRNA regions with mRNAs that fall in the given upstream/downstream regions of the microRNA genes

Usage

```r
mirnaRegionPathwayEnricher(
  region,
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
  near = FALSE,
  target = FALSE,
  isTADSearch = FALSE,
  TAD = c(tad_hg19, tad_dmel, tad_hg38, tad_mm10),
  gmtName = "",
  express = FALSE,
  isCustomExp = FALSE,
  cancer,
  exp1,
  exp2,
  label1 = "",
  label2 = "",
  isUnionCorGene = FALSE,
  databaseFile,
  isGeneEnrich = FALSE
)
```

Arguments

- **region**: MiRNA region in a bed format
- **org_assembly**: Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human
- **near**: Boolean value presents whether cis-neighbourhood should be considered in the analysis
- **target**: Boolean value shows whether miRNA target prediction should be performed
- **isTADSearch**: Boolean value that shows whether TAD analysis is performed. This value has to be TRUE for TAD analysis.
- **TAD**: TAD genomic regions for the species. Predefined TAD regions or any new TAD regions can be used for the analysis. TAD regions must be formated as GRanges
mirnaRegionPathwayEnricher

object. Predefined TAD regions are 'tad_hg19', 'tad_hg38', 'tad_mm10', 'tad_dmel' for hg19, hg38, mm9 and dm6 assembly, respectively.

gmtName Custom pathway gmt file
express Boolean variable whether co-expression analysis is performed. If this option is set to TRUE, co-expression analysis will be performed.
isCustomExp Boolean variable whether co-expression analysis with custom data will be performed. When this option is set, exp1 and exp2 parameters must be defined.
cancer Defines the name of the TCGA project code such as 'BRCA' for correlation analysis. Possible cancer types ACC, BLCA, BRCA, CESC, CHOL, COAD, COADREAD, DLBC, ESCA, GBMLGG, HNSC, KICH, KIPAN, KIRP, LGG, LIHC, LUAD, LUSC, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, STES, TGCT, THCA, THYM, UCEC, UCS, UVM
exp1 Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.
exp2 Custom expression data matrix. Columns must be genes and rows must be patients. If gene names are provided as header, no need to redefine the headers(labels) of the expression data.
label1 Gene names of the custom exp1 expression data. If it is not provided, column name of the exp1 data will be taken.
label2 Gene names of the custom exp2 expression data. If it is not provided, column name of the exp2 data will be taken.
isUnionCorGene Boolean value that shows whether union of the output of the co-expression analysis and the other analysis should be considered
databaseFile Path of miRcancer.db file
isGeneEnrich Boolean value whether gene enrichment should be performed

Value

miRNA pathway enrichment object for the given input

Examples

regions<-system.file("extdata", "ncRegion.txt", package = "NoRCE")
regionNC <- rtracklayer::import(regions, format = "BED")

a<- mirnaRegionPathwayEnricher(region = regionNC,
org_assembly = 'hg19')
**mRNA**  
**Description**  
Brain mRNA expression retrieved from the TCGA  
**Usage**  
mRNA  
**Format**  
Not Available  
**Source**  
https://www.gencodegenes.org/  
**Examples**  
data(mRNA)  

**ncRegion**  
**Description**  
Differentially expressed non-coding gene regions  
**Usage**  
ncRegion  
**Format**  
Not Available  
**Source**  
http://resource.psychencode.org/  
**Examples**  
data(ncRegion)
**NoRCE-class**

*An S4 class to represent enrichment*

**Description**

An S4 class to represent enrichment

**Slots**

- ID factor
- Term factor
- geneList factor
- ncGeneList factor
- pvalue factor
- pAdj factor
- GeneRatio factor
- BckRatio factor

**packageCheck**

*Check the package availability for the given assembly*

**Description**

Check the package availability for the given assembly

**Usage**

```r
packageCheck(pkg)
```

**Arguments**

- `pkg` Required packages

**Value**

- return install packages
pathwayEnrichment

For a given gmt file of a specific pathway database, pathway enrichment can be performed. Function supports Entrez ID and symbol based gmt file.

Description

For a given gmt file of a specific pathway database, pathway enrichment can be performed. Function supports Entrez ID and symbol based gmt file.

Usage

pathwayEnrichment(
  genes,
  gmtFile,
  org_assembly = c("hg19", "hg38", "mm10", "dre10", "rn6", "dm6", "ce11", "sc3"),
  pCut = 0.05,
  pAdjCut = 0.05,
  pAdjust = c("holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"),
  isSymbol,  
  min = 5,
  isGeneEnrich = FALSE
)

Arguments

genes Input genes

gmtFile File path of the gmt file

org_assembly Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human

pCut Threshold value for the pvalue. Default value is 0.05

pAdjCut Cutoff value for the adjusted p-values using one of given method. Default value is 0.05.

pAdjust Methods of the adjusted p-values. Possible methods are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"

isSymbol Boolean value that controls the gene formats. If it is TRUE, gene format of the gmt file should be symbol. Otherwise, gene format must be ENTREZ ID.

min Minimum number of genes that are required for enrichment. By default, it is set to 5.

isGeneEnrich Boolean value whether gene enrichment should be performed

Value

Pathway Enrichment
predictmiTargets  

**Description**  
Predict the miRNA targets for the miRNA or mRNA genes, which is specified with type parameter.

**Usage**  
predictmiTargets(gene, type, org_assembly)

**Arguments**
- gene: Data frame of miRNA or mRNA gene. Formats should be NCBI gene name, ENSEMBL gene or transcript id, and mirna.
- type: Format of the gene, it should be "NCBI" for NCBI gene name, "Ensembl_gene" for ENSEMBL gene id, "Ensembl_trans" for Ensembl transcript id and "mirna" for miRNA gene.
- org_assembly: Analyzed genome assembly. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "hg19" and "hg38" for human.

**Value**
miRNA:mRNA target sets of the given genes.

**Examples**
```r
a<- predictmiTargets(gene = brain_mirna[1:100,], org_assembly = 'hg19', type = "mirna")
```

---

reactomeEnrichment  

**Description**  
Reactome pathway enrichment

**Usage**

**Arguments**

**Value**

**Examples**

---
**reactomeEnrichment**

**Usage**

```r
default_args <- list(genes = c(), org_assembly = c("mm10", "rn6", "dm6", "cell", "sc3"), pCut = 0.05, pAdjCut = 0.05, pAdjust = c("holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"), min = 5, gmtFile = "", isSymbol = "", isGeneEnrich = ""
```  

**Arguments**

- **genes**: Input genes
- **org_assembly**: Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "cell" for worm, "sc3" for yeast, "hg19" and "hg38" for human
- **pCut**: Threshold value for the pvalue. Default value is 0.05
- **pAdjCut**: Cutoff value for the adjusted p-values using one of given method. Default value is 0.05.
- **pAdjust**: Methods of the adjusted p-values. Possible methods are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"
- **min**: Minimum number of genes that are required for enrichment. By default, it is set to 5.
- **gmtFile**: File path of the gmt file
- **isSymbol**: Boolean value that controls the gene formats. If it is TRUE, gene format of the gmt file should be symbol. Otherwise, gene format must be ENTREZ ID.
- **isGeneEnrich**: Boolean value whether gene enrichment should be performed

**Value**

Reactome pathway enrichment results

**Examples**

```r
br_enr <- reactomeEnrichment(genes = breastmRNA, org_assembly = 'hg19')
```
setParameters

Description

Parameters: upstream: Upstream distance from the transcription start position downstream: Downstream distance from the transcription end position searchRegion: Search space of the cis-region. Possible values are "all", "exon", "intron" GOtype: Hierarchical category of the GO ontology. Possible values are "BP", "CC", "MF" pCut: Threshold value for the pvalue. Default value is 0.05 pAdjCut: Cutoff value for the adjusted p-values using one of given method. Default value is 0.05. pAdjust: Methods of the adjusted p-values. Possible methods are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none" min: Minimum number of genes that are required for enrichment. By default, this value is set to 5. cellline: Cell lines for TAD regions. corrMethod Correlation coefficient method that will be used for evaluation. Possible values are "pearson", "kendall", "spearman" varCutoff: Variance cut off that genes have less variance than this value will be trimmed pcut: P-value cut off for the correlation values alternate: Holds the alternative hypothesis and "two.sided", "greater" or "less" are the possible values. conf: Confidence level for the returned confidence interval. It is only used for the Pearson correlation coefficient if there are at least 4 complete pairs of observations. minAbsCor: Cut-off value for the Pearson correlation coefficient of the miRNA-mRNA pathwayType: Pathway database for enrichment. Possible values are 'reactome' for Reactome, 'kegg' for KEGG, 'wiki' for WikiPathways, 'other' for custom database enrichTest: Types of enrichment methods to perform enrichment analysis. Possible values are "hyper"(default), "binom", "fisher", "chi". isSymbol: Boolean variable that hold the gene format of the gmt file. If it is set as TRUE, gene format of the gmt file should be symbol. Otherwise, gene format should be ENTREZ ID. By default, it is FALSE.

Usage

setParameters(type, value)

Arguments

type List of parameter names
value New values for the parameters. Value and the parameter names must be in the same order.

Value

changed parameters

Examples

type <- c('downstream', 'upstream')
value <- c(2000, 30000)
setParameters(type, value)
tad_dmel  

**Description**

TAD regions for the fly

**Usage**

tad_dmel

**Format**

Not Available

**Source**

http://chorogenome.ie-freiburg.mpg.de/data_sources.html#hi-c_datasets

**Examples**

data(tad_dmel)

---

tad_hg19  

**Description**

TAD regions for human hg19 assembly

**Usage**

tad_hg19

**Format**

Not Available

**Source**

http://promoter.bx.psu.edu/hi-c/publications.html

**Examples**

data(tad_hg19)
tad_hg38

TAD regions for human hg38 assembly

Description
TAD regions for human hg38 assembly

Usage
tad_hg38

Format
Not Available

Source
http://promoter.bx.psu.edu/hi-c/publications.html

Examples
data(tad_hg38)

tad_mm10

TAD regions for mouse

Description
TAD regions for mouse

Usage
tad_mm10

Format
Not Available

Source
http://promoter.bx.psu.edu/hi-c/publications.html

Examples
data(tad_mm10)
topEnrichment  

Number of top enrichment results of the pathway or GO terms for the given object and the order type - p-value or adjusted p-value.

Description

Number of top enrichment results of the pathway or GO terms for the given object and the order type - p-value or adjusted p-value.

Usage

topEnrichment(mrnaObject, type, n)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mrnaObject</td>
<td>Object of the enrichment result</td>
</tr>
<tr>
<td>type</td>
<td>Draw the dot plot according to the p-value or adjusted p-value (&quot;pvalue&quot;,&quot;pAdjust&quot;)</td>
</tr>
<tr>
<td>n</td>
<td>Number of GO terms or pathways, that ordered by type and has least number of top p-value</td>
</tr>
</tbody>
</table>

Value

Give top n enrichment results

WikiEnrichment

WikiPathways Enrichment

Description

WikiPathways Enrichment

Usage

WikiEnrichment(
genes,
org_assembly = c("hg19", "hg38", "mm10", "drel0", "rn6", "dm6", "ce11", "sc3"),
pCut = 0.05,
pAdjCut = 0.05,
pAdjust = c("holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"),
min = 5,
gmtFile = "",
isSymbol = "",
isGeneEnrich = ""
)
Arguments

genes: Input genes
org_assembly: Genome assembly of interest for the analysis. Possible assemblies are "mm10" for mouse, "dre10" for zebrafish, "rn6" for rat, "dm6" for fruit fly, "ce11" for worm, "sc3" for yeast, "hg19" and "hg38" for human
pCut: Threshold value for the p-value. Default value is 0.05
pAdjCut: Cutoff value for the adjusted p-values using one of given method. Default value is 0.05.
pAdjust: Methods of the adjusted p-values. Possible methods are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"
min: Minimum number of genes that are required for enrichment. By default, it is set to 5.
gmtFile: File path of the gmt file
isSymbol: Boolean value that controls the gene formats. If it is TRUE, gene format of the gmt file should be symbol. Otherwise, gene format must be ENTREZ ID.
isGeneEnrich: Boolean value whether gene enrichment should be performed

Value

Wiki Pathway Enrichment

writeEnrichment

Write the tabular form of the pathway or GO term enrichment results

Description

Write the tabular form of the pathway or GO term enrichment results

Usage

writeEnrichment(mrnaObject, fileName, sept = "\t", type = "pAdjust", n)

Arguments

mrnaObject: Object of the enrichment result
fileName: File name of the txt file
sept: File separator, by default, it is tab("	")
type: Draw the dot plot according to the p-value or adjusted p-value ("pvalue", "pAdjust"). Default value is "pAdjust".
n: Number of GO terms or pathways, that ordered by type and has least number of top p-value
Value

Text file of the enrichment results in a tabular format

Examples

```
ncGO<-geneGOEnricher(gene = brain_disorder_ncRNA, org_assembly='hg19',
near=TRUE, genotype = 'Ensembl_gene')

writeEnrichment(mRNAObject = ncGO,fileName = "a.txt",sept = '\t')
```
## Index

* **datasets**
  - brain_disorder_ncRNA, 4
  - brain_mirna, 5
  - breastmRNA, 5
  - mirna, 30
  - mrna, 38
  - ncRegion, 38
  - tad_dmel, 44
  - tad_hg19, 44
  - tad_hg38, 45
  - tad_mm10, 45

  - annGO, 3
  - assembly, 4

  - brain_disorder_ncRNA, 4
  - brain_mirna, 5
  - breastmRNA, 5

  - calculateCorr, 6
  - convertGeneID, 7
  - convertGMT, 8
  - corrbased, 8
  - corrbasedMrna, 9
  - createNetwork, 10

  - drawDotPlot, 11

  - extractBiotype, 11

  - filterBiotype, 12

  - geneGOEnricher, 12
  - genePathwayEnricher, 14
  - geneRegionGOEnricher, 16
  - geneRegionPathwayEnricher, 18

  - getGoDag, 20
  - getKeggDiagram, 21

  - getmiRNACount, 22
  - getNearToExon, 22
  - getNearToIntron, 23

  - getReactomeDiagram, 24
  - getTADOverlap, 25
  - getUCSC, 26
  - goEnrichment, 27

  - KeggEnrichment, 28

  - listTAD, 29

  - mirna, 30
  - mirnaGOEnricher, 30
  - mirnaPathwayEnricher, 32
  - mirnaRegionGOEnricher, 34
  - mirnaRegionPathwayEnricher, 36

  - mrna, 38

  - ncRegion, 38

  - NoRCE-class, 39

  - packageCheck, 39

  - pathwayEnrichment, 40

  - predictmiTargets, 41

  - reactomeEnrichment, 41

  - setParameters, 43

  - tad_dmel, 44
  - tad_hg19, 44
  - tad_hg38, 45

  - tad_mm10, 45

  - topEnrichment, 46

  - WikiEnrichment, 46

  - writeEnrichment, 47