Package 'psygenet2r'

October 12, 2016

Title psygenet2r - An R package for querying PsyGeNET and to perform comorbidity studies in psychiatric disorders

Version 1.4.0

Description Package to retrieve data from PsyGeNET database (www.psygenet.org) and to perform comorbidity studies with PsyGeNET's and user's data.

Depends R (>= 3.3)

License MIT + file LICENSE

LazyData true

Suggests testthat

Imports stringr, RCurl, igraph, ggplot2, reshape2, grid, parallel, biomaRt, BgeeDB, topGO

NeedsCompilation no

Author Alba Gutierrez-Sacristan [aut],
Carles Hernandez-Ferrer [cre]

Maintainer Alba Gutierrez-Sacristan <alba.gutierrez@upf.edu>

biocViews Software, BiomedicalInformatics, Genetics, Infrastructure,
DataImport, DataRepresentation

RoxygenNote 5.0.1

R topics documented:

DataGeNET.Psy-class   *Class DataGeNET.Psy*

Description

Class `DataGeNET.Psy` is the basic object use in `psygenet2r` package. It is the main data container to using the different functions to query PsyGeNET database and generate teir output. The constructors of this class are the functions `psygenetGene` and `psygenetDisease`.

Slots

type Character containing `'gene'` of `'disease'`. It is used to eprform the correct query to Psy-GeNET.

search Character containing `'single'` of `'list'`.It is used to eprform the correct query to Psy-GeNET.

database Character containing the name of the database that will be queried. It can take the values `'MODELS'` to use Comparative Toxigenomics Database, data from mouse and rat; `'GAD'` to use Genetic Association Database; `'CTD'` to use Comparative Toxigenomics Database, data from human; `'PsyCUR'` to use Psychiatric disorders Gene association manually curated; `'CURATED'` to use Human, manually curated databases (PsyCUR and CTD); or `'ALL'` to use all these databases.

term Charcter with the term(s) to search into the database(s).

qresult `data.frame` with the obtained result

See Also

psygenetGene, psygenetDisease, DataGeNET.Psy-methods

| enrichedPD | *Enrichment of a user's input (genes) in PsyGeNET's diseases.* |

Description

Test the enrichment of a given gene list on Psychiatric Disorders from PsyGeNET.

Usage

```
enrichedPD(gene, database = "ALL", verbose = FALSE, warnings = FALSE)
```

Arguments

gene
: Name or vector of names (that can be both code or uml) to specific genes from PsyGeNET.

database
: Name of the database that will be queried. It can take the values `'psycur15'` to use data validated by experts for first release of PsyGeNET; `'psycur16'` to use data validated by experts for second release of PsyGeNET; or `'ALL'` to use both databases. Default `'ALL'`.

verbose
: By default FALSE. Change it to TRUE to get a on-time log from the function.

warnings
: By default TRUE. Change it to FALSE to not see the warnings.

Value

A `data.frame` with the enricment at each Psychiatric Disorder

Examples

```
enrichedPD(c("ADCY2", "AKAP13", "ANK3"), "ALL")
```

| extract | *Raw data from* `DataGeNET.Psy` *and* `JaccardIndexPsy`. |

Description

Obtain the raw data from a PsyGeNET's query stored in a `DataGeNET.Psy` object or the raw data with all the Jaccard Index for the disease of interest of an `JaccardIndexPsy` object.

Usage

```
extract(object, ...)

## S4 method for signature 'DataGeNET.Psy'
extract(object)

## S4 method for signature 'JaccardIndexPsy'
extract(object, order.cl = "pval", ...)
```

Arguments

`object`	Object of class `DataGeNET.Psy` or `JaccardIndexPsy`
`...`	NO USED
`order.cl`	Order resulting `data.frame` by the name of this column.

Value

A `data.frame` containing the raw result from PsyGeNET or a `data.frame` with the result Jaccard Index for each disease.

A `data.frame` containing the raw result from PsyGeNET

A `data.frame` with the result Jaccard Index for each disease.

Methods (by class)

- `DataGeNET.Psy`: Extract function for DataGeNET.Psy
- `JaccardIndexPsy`: Extract function for JacardIndexPsy

Examples

```
data(qr)
extract(qr)[1:2, ] # get internat data.frame
## Not run:
#Being x an JaccardIndexPsy
extract(x)

## End(Not run)
```

---

extractSentences   *Method to obtain the evidences from a* `DataGeNET.Psy` *object.*

Description

Internally, PsyGeNET uses a series of collected evidences from public literature. The internal table of a `DataGeNET.Psy` object can contains this information. The method `extractSentences` allows to extrat this information.

Usage

```
extractSentences(object, disorder, verbose)

## S4 method for signature 'DataGeNET.Psy'
extractSentences(object, disorder, verbose = FALSE)
```

Arguments

`object`	Object of class `DataGeNET.Psy`.
`disorder`	A disorder to check if any evidence exists.
`verbose`	If set to `TRUE` informative messages are show.

Value

A data frame showing the evidence.

Methods (by class)

- `DataGeNET.Psy`: Get sentences or evidences

Examples

```
data(qr)
extractSentences(qr, "Depression")
```

`geneAttrPlot`	*Ploting the relation between genes and disease-categories*

Description

Given a set of genes or a result of psygenetGene creates four types of plots showing the relation of the genes with the disease's category in psyGeNET.

Usage

```
geneAttrPlot(x, type = "pie", verbose = FALSE)
```

Arguments

`x`	Vector of genes of interest of `DataGeNET.Psy` resulting of `psyegnetDisease`.
`type`	(default `"pie"`) It can takes `"pie"`,
`verbose`	By default `FALSE`. Change it to `TRUE` to get a on-time log from the function.

Value

A plot for a `DataGeNET.Psy` in terms of the panther-class.

---

getUMLs   *Query PsyGeNET for given gene(s) and generates an* DataGeNET.Psy

---

Description

Given the name of one or multiple gene and retrives their information from PsyGeNET and creates an object of type DataGeNET.Psy.

Usage

```
getUMLs(word, database = "CURATED")
```

Arguments

word   Disese to convert to UMLS using PsyGeNET database.

database   Name of the database that will be queried. It can take the values 'MODELS'
                to use Comparative Toxigenomics Database, data from mouse and rat; 'GAD'
                to use Genetic Association Database; 'CTD' to use Comparative Toxigenomics
                Database, data from human; 'PsyCUR' to use Psychiatric disorders Gene associ-
                ation manually curated; 'CURATED' to use Human, manually curated databases
                (PsyCUR and CTD); or 'ALL' to use all these databases. Default 'CURATED'.

Value

The corresponding UMLs for the input disease/s

Examples

```
umls <- getUMLs( "Alcoholism", "CURATED" )
```

---

jaccardEstimation   *Calculation of the Jaccard Index between ideseases*

Description

This function is able to calculate the Jacard Index between: 1. muliple disases, 2. a set og genes and multiple diseases, 3. a set of genes and multiple main psychiatric disorders and 4. multiple diseases and multiple main psychiatric disorders.

Usage

```
jaccardEstimation(pDisease, sDisease, database = "CURATED", nboot = 100,
  ncores = 1, verbose = FALSE)
```

Arguments

`pDisease`	vector of diseases, vector of genes, vector of main psychiatric disorder.
`sDisease`	vector of diseases, vector of genes, vector of main psychiatric disorder. Only necessary when comparing genes vs. diseases, genes vs. main psychiatric disorders or diseases vs. main psychiatric disorders. To compare multiple diseases only use `pDisease`.
`database`	Name of the database that will be queried. It can take the values `'MODELS'` to use Comparative Toxigenomics Database, data from mouse and rat; `'GAD'` to use Genetic Association Database; `'CTD'` to use Comparative Toxigenomics Database, data from human; `'PsyCUR'` to use Psychiatric disorders Gene association manually curated; `'CURATED'` to use Human, manually curated databases (PsyCUR and CTD); or `'ALL'` to use all these databases. Default `'CURATED'`.
`nboot`	Number of iterations sued to compute the pvalue associated to the calculated Jaccard Index (default 100).
`ncores`	Number of cores used to calculate the pvalue associated to the computed Jaccard Index (default 1).
`verbose`	By default `FALSE`. Change it to `TRUE` to get a on-time log from the function.

Details

Warning: The main psychiatric disorders are understood as a single set of genes composed by the genes of all the diseases that the main psychiatric disorder cotains.

Value

An object of class `JaccardIndexPsy` with the computed calculation of the JaccardIndex.

Examples

```
ji <- jaccardEstimation( c( "COMT", "CLOCK", "DRD3" ), "umls:C0005586", "ALL" )
```

JaccardIndexPsy-class *Class JaccardIndexPsy*

Description

Class `JaccardIndexPsy` is theresult of the process to look for a Jaccard Index between muliple diseases in `psygenet2r` package.

Slots

`nit` Number of iterations to calculate the estimated Jaccard index

`type` Slot to save type of query (disease-disease, gene-disease)

`table` `data.frame` containing the result table of Jaccard indexes

`i1` [internal use] vector with names of first component

`i2` [internal use] vector with names of second component

See Also

psygenetGene, psygenetDisease, JaccardIndexPsy-methods

Examples

```
ji <- jaccardEstimation( c( "COMT", "CLOCK", "DRD3" ), "umls:C0005586", "ALL" )
```

---

ndisease   *Getter from* `DataGeNET.Psy`.

Description

Obtain the number of unique diseases in a `DataGeNET.Psy`.

Usage

```
ndisease(object)

## S4 method for signature 'DataGeNET.Psy'
ndisease(object)
```

Arguments

`object`   Object of class `DataGeNET.Psy`.

Value

The number of unique diseases

Methods (by class)

  • `DataGeNET.Psy`: Get number of diseases

Examples

```
data(qr)
ndisease(qr)
```

ngene	*Getter from* `DataGeNET.Psy`.

Description

Obtain the number of unique genes in a `DataGeNET.Psy`.

Usage

```
ngene(object)

## S4 method for signature 'DataGeNET.Psy'
ngene(object)
```

Arguments

`object`	Object of class `DataGeNET.Psy`.

Value

The number of unique genes

The number of unique genes

Methods (by class)

- `DataGeNET.Psy`: Get number of genes

Examples

```
data(qr)
ngene(qr)
```

pantherGraphic	*Query PsyGeNET for given genes and creates a representation in base of their panther-class*

Description

Given a vector of genes of interest (or using a DataGeNET.Psy object), this function creates a representation of a the panther-class these genes belongs to.

Usage

```
pantherGraphic(x, database = "ALL", score, verbose = FALSE)
```

Arguments

x	Vector of genes of interest of `DataGeNET.Psy` resulting of `psyegnetDisease`.
database	Name of the database that will be queried. It can take the values `'psycur15'` to use data validated by experts for first release of PsyGeNET; `'psycur16'` to use data validated by experts for second release of PsyGeNET; or `'ALL'` to use both databases. Default `'ALL'`.
score	threshold to take into account a gene in the analysis
verbose	By default `FALSE`. Change it to `TRUE` to get a on-time log from the function.

Value

A plot for a `DataGeNET.Psy` in terms of the panther-class.

Examples

```
d.alch <- pantherGraphic( c( "COMT", "CLOCK", "DRD3" ), "ALL" )
```

`plot,DataGeNET.Psy,ANY-method`
                          Plots the content of a `DataGeNET.Psy` *or* `JaccardIndexPsy` *object.*

Description

This functions llows to create a variety of plots for `DataGeNEt.Psy` and `JaccardIndexPsy` objects.

Usage

```
## S4 method for signature 'DataGeNET.Psy,ANY'
plot(x, y,
  layout = igraph::layout.fruchterman.reingold, type = "disease",
  verbose = FALSE, ...)
```

Arguments

x	Object of class `DataGeNET.Psy`
y	NOT USED
layout	Function to design the location of the different nodes. By default `layout.fruchterman.reingold` from `igraph` is used.
type	Type of the drawn chart. By default it is `"disease"` but it also can be `"individual disease"`, `"disease"`, `"disease class"`, `"barplot"`, `"heatmapGenes"` or `"heatmap"`.
verbose	By default `FALSE`. If set to `TRUE` information on the drawing process will be shown.
...	Passed to inner functions for different plots.

Value

A plot for `DataGeNET.Psy`.

Examples

```
data(qr)
plot(qr) # for all-disease plot
plot(qr, type = 'disease class') # for MPI plot
```

`plot,JaccardIndexPsy,ANY-method`
Plot the content of a `JaccardIndexPsy` *object.*

Description

This functions llows to create a variety of plots for `DataGeNEt.Psy` and `JaccardIndexPsy` objects.

Usage

```
## S4 method for signature 'JaccardIndexPsy,ANY'
plot(x, y, cutOff, zero.remove = TRUE,
  noTitle = FALSE, verbose = FALSE, ...)
```

Arguments

x	Object of class `JaccardIndexPsy`.
y	NOT USED
cutOff	Number to filter the shown results.
zero.remove	By deffault `TRUE`. It removes those relations with a Jaccard Index of 0.
noTitle	By default `FALSE`. If set to true no title will be added to the plot.
verbose	By default `FALSE`. If set to `TRUE` information on the drawing process will be shown.
...	NOT USED

Value

A plot for `JaccardIndexPsy`.

Examples

```
## Not run:
#Being x an JaccardIndexPsy
qr <- plot(x)

## End(Not run)
```

---

psygenet2r	*psygenet2r: Package to query PsyGeNET database and to perform comorbidity studies*

Description

psygenet2r has two categories of functions: querying functions and analysis and plotting functions.

querying functions

The functions to retrieve data from PsyGeNET are psygenetDisease and psygenetGene. There are some other support functions like psygenetGeneSentences.

analysis and plotting functions

The functions extract and extractSentences allows to retrieve the row data obtained from on-line resources. The functions plot and pantherGraphic draws a variety of charts to illustrate the obtained results. The function enrichedPD was built to perform enrichment studies on PsyGeNET data. Finally the function jaccardEstimation computes a Jaccard Index from a given input on PsyGeNET data.

---

psygenetDisease	*Query PsyGeNET for given disease(s) and generates an* DataGeNET.Psy

---

Description

Given the name of one or multiple diseases and retrives their information from PsyGeNET and creates an object of type DataGeNET.Psy.

Usage

```
psygenetDisease(disease, database = "ALL", score = c(">", 0),
  verbose = FALSE, warnings = TRUE)
```

Arguments

disease
: Name or vector of names (that can be both code or uml) to specific diseases from PsyGeNET. The diseases non existing in PsyGeNET will be removed from the output.

database
: Name of the database that will be queried. It can take the values 'psycur15' to use data validated by experts for first release of PsyGeNET; 'psycur16' to use data validated by experts for second release of PsyGeNET; or 'ALL' to use both databases. Default 'ALL'.

score	A vector with two elements: 1) character with greather '>' or with lower '<' meaing greather or equal and lower or equal; 2) the evidence index cut-off to be compared. By default: c('>', 0).
verbose	By default FALSE. Change it to TRUE to get a on-time log from the function.
warnings	By default TRUE. Change it to FALSE to don't see the warnings.

Value

An object of class DataGeNET.Psy

Examples

```
d.sch <- psygenetDisease( "schizophrenia", "ALL" )
```

---

psygenetDiseaseSentences

Query PsyGeNET for given disease(s) and extract the pmids sentences that report a gene-disease asssociation.

Description

Given a disease or a disease list, retrives the pmids and sentences for each gene-disease association from PsyGeNET and creates an object of type DataGeNET.Psy.

Usage

```
psygenetDiseaseSentences(diseaseList, database = "ALL", verbose = FALSE)
```

Arguments

diseaseList	Name or vector of names (that can be both code or uml) to specific diseases from PsyGeNET. The diseases non existing in PsyGeNET will be removed from the output.
database	Name of the database that will be queried. It can take the values 'psycur15' to use data validated by experts for first release of PsyGeNET; 'psycur16' to use data validated by experts for second release of PsyGeNET; or 'ALL' to use both databases. Default 'ALL'.
verbose	By default FALSE. Change it to TRUE to get a on-time log from the function.

Value

An object of class DataGeNET.Psy

Examples

```
diseasesOfInterest <- c( "Bipolar Disorder","Depressive Disorder, Major" )
psyDisSen <- psygenetDiseaseSentences( diseaseList = diseasesOfInterest,
                                       database = "ALL" )
```

psygenetGene	*Query PsyGeNET for given gene(s) and generates an* `DataGeNET.Psy`

Description

Given the name of one or multiple gene and retrives their information from PsyGeNET and creates an object of type `DataGeNET.Psy`.

Usage

```
psygenetGene(gene, database = "ALL", score = c(">", 0), verbose = FALSE,
  warnings = TRUE)
```

Arguments

gene   Name or vector of names (that can be both code or symbol) to specific genes
            from PsyGeNET. The genes non existing in PsyGeNET will be removed from
            the output.

database   Name of the database that will be queried. It can take the values `'psycur15'` to
            use data validated by experts for first release of PsyGeNET; `'psycur16'` to use
            data validated by experts for second release of PsyGeNET; or `'ALL'` to use both
            databases. Default `'ALL'`.

score   A vector with two elements: 1) character with greather `'>'` or with lower `'<'`
            meaing greather or equal and lower or equal; 2) the evidence index cut-off to be
            compared. By default: `c('>', 0)`.

verbose   By default `FALSE`. Change it to `TRUE` to get a on-time log from the function.

warnings   By default `TRUE`. Change it to `FALSE` to not see the warnings.

Value

An object of class `DataGeNET.Psy`

Examples

```
d.alch <- psygenetGene( "ALDH2", "ALL" )
```

---

psygenetGeneSentences	*Query PsyGeNET for given gene(s) and extract the pmids sentences that report a gene-disease asssociation.*

Description

Given a gene or a gene list, retrieves the pmids and sentences for each gene-disease association from PsyGeNET and creates an object of type `DataGeNET.Psy`.

Usage

```
psygenetGeneSentences(geneList, database = "ALL", verbose = FALSE)
```

Arguments

geneList	Name or vector of names (that can be both code or symbol) to specific genes from PsyGeNET. The genes non existing in PsyGeNET will be removed from the output.
database	Name of the database that will be queried. It can take the values `'psycur15'` to use data validated by experts for first release of PsyGeNET; `'psycur16'` to use data validated by experts for second release of PsyGeNET; or `'ALL'` to use both databases. Default `'ALL'`.
verbose	By default FALSE. Change it to TRUE to get a on-time log from the function.

Value

An object of class `DataGeNET.Psy`

Examples

```
genesOfInterest <- c("PECR", "ADH1C", "CAST", "ERAP1", "PPP2R2B",
                     "ESR1", "GATA4", "CDH13")
psyGeneSen <- psygenetGeneSentences( geneList = genesOfInterest,
                                     database = "ALL")
```

---

qr	`DataGeNET.Psy` *obtained from quering PsyGeNET for gene '4852'.*

Description

A dataset obtained from PsyGeNET after being queried with `psygenetGene` usig the term '4852' on `"ALL"` databse.

Usage

```
data("qr")
```

Format

The format is: Formal class 'DataGeNET.Psy' [package "psygenet2r"] with 5 slots .. type : chr "gene" .. search : chr "" .. database: chr "ALL" .. term : chr "4852" .. qresult :'data.frame'

Value

A `DataGeNET.Psy` object.

Source

<http://psygenet.org>

Examples

```
ngene(qr)
ndisease(qr)
```

---

topAnatEnrichment   *Enrichment of a user's input (genes) in anatomical terms (TopAnat).*

Description

Test the enrichment of a given gene list on Psychiatric Disorders from PsyGeNET.

Usage

```
topAnatEnrichment(gene, dataType = "rna_seq", statistic = "fisher",
  cutOff = 1, verbose = FALSE, warnings = FALSE)
```

Arguments

gene	Name or vector of names (that can be both code or uml) to specific genes from PsyGeNET.
dataType	It can take the values `'rna_seq'` or `'affymetrix'`. Default `'rna_seq'`.
statistic	Default `'fisher'`.
cutOff	Default 1.
verbose	By default FALSE. Change it to TRUE to get a on-time log from the function.
warnings	By default TRUE. Change it to FALSE to not see the warnings.

Value

A `data.frame` with the enrichment results

Examples

```
## Not run:
topAnatEnrichment(c("ADCY2", "AKAP13", "ANK3"), "ALL")

## End(Not run)
```

Index