

Package ‘SubpathwayMiner’

September 3, 2009

Version 2.0

Title Annotation and identification of the KEGG pathways.

Author Chunquan Li

Maintainer Chunquan Li <lcqbio@yahoo.com.cn>

Description SubpathwayMiner is an R-based software for flexible pathway identification. (1) SubpathwayMiner can provide users with sub-pathway annotation and identification of metabolic pathways based on enzyme commission (EC) numbers. (2) SubpathwayMiner can provide users with sub-pathway annotation and identification based on KEGG Orthology (KO) identifiers. (new!) (3) SubpathwayMiner can provide annotation and identification of entire pathways. (4) SubpathwayMiner can support most of organisms in the KEGG GENE database. (5) Data can be automatically updated on demand by the user.

Depends R (>= 2.8.0),RBGL(>= 1.18.0),fdrtool(>= 1.2.4)

Suggests Rgraphviz(>= 1.20.3),XML(>= 1.94)

Collate ann.R initialize.R subGraph.R id.R graph.R go2go.R visualize.R ko.R

License GPL (>= 2)

biocViews Statistics, Annotation

Repository CRAN

Date/Publication 2009-09-03 09:25:03

R topics documented:

background	2
cutoffAnn	3
dgraph	4
ec2gene	5
gene2ec	5
gene2KO	6
gene2path	6

getAexample	7
getAnn	7
getDefaultBackground	9
getDefaultGraph	10
getDefaultKOPathway	11
getDefaultKOUndirectedGraph	11
getDefaultMetabolicPathway	12
getDefaultUndirectedGraph	13
getEnzymeFromGene	13
getGeneFromEnzyme	14
getGeneFromKO	15
getGeneFromPathway	16
getKcsmpAnn	17
getKcSubGraph	18
getKOAnn	19
getKOFromGene	21
getMpAnn	22
getOrgAndIdType	23
getPathwayFromGene	24
getPathwayNameFromId	25
gotoKEGG	25
ke2g	26
Ke2gEnvironment	27
keggpathid2name	28
KO2gene	29
KOUgraph	29
kpidList	30
mpidList	30
path2gene	31
plotAnn	31
plotKOAnn	32
printAnn	33
ugraph	34
updateGraphs	35
updateKOGraphs	36
updateOrgAndIdType	37
Index	39

background

The Background Genes.

Description

A whole genome genes. They are often used as the statistical significance annalysis.

Format

A character vector

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

`cutoffAnn`*Get the statistically significantly enriched pathways*

Description

Get the statistically significantly enriched pathways according to the results returned from [getAnn](#) or [getKOAnn](#) function.

Usage

```
cutoffAnn(ann, type="pvalue", operate="<=", cutoff=0.01)
```

Arguments

<code>ann</code>	A list. The results returned from getAnn function.
<code>type</code>	A character string. Should be one of "pvalue", "qvalue".
<code>operate</code>	A character string. Should be one of "<", "<=", ">", ">=".
<code>cutoff</code>	A numeric. Detailed information is provided in getAnn .

Details

The function is used to identify the statistically significantly enriched pathways compared with a background distribution. If users don't input background gene list in the function [getAnn](#) or [getKOAnn](#), then background distribution will be obtained from the whole genome.

Value

A list of annotation results. Detailed information is provided in [getAnn](#) or [getKOAnn](#).

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

References

Strimmer, K. (2008). A unified approach to false discovery rate estimation.

See Also

[getAnn](#), [getKOAnn](#), [printAnn](#)

Examples

```
##get an example of gene sets
geneList<-getAexample(k=1000)

##get the annotated results
ann<-getAnn(geneList)

##get the statistically significantly enriched pathways according to pvalue<0.001
cutedAnn<-cutoffAnn(ann,"pvalue", "<", 0.001)

##print results to screen
printAnn(cutedAnn)
```

dgraph

Directed graph list based on enzyme commission (EC) numbers constructed from KEGG metabolic pathways

Description

Directed graph list based on enzyme commission (EC) numbers constructed from KEGG metabolic pathways.

Format

A list of graph

Details

Detailed information is provided in [uGraph](#).

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[uGraph](#)

ec2gene

A list that maps Enzyme Commission (EC) numbers to gene identifiers

Description

A list that maps Enzyme Commission (EC) numbers to gene identifiers.

Format

A list

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[gene2ec](#)

gene2ec

A list that maps gene identifiers to Enzyme Commission (EC) numbers

Description

A list that maps gene identifiers to Enzyme Commission (EC) numbers.

Format

A list

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[ec2gene](#)

gene2KO

A list that maps gene identifiers to KEGG Orthology (KO)

Description

A list that maps gene identifiers to KEGG Orthology (KO).

Format

A list

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[KO2gene](#)

gene2path

A list that maps gene identifiers to KEGG pathway identifiers

Description

A list that maps gene identifiers to KEGG pathway identifiers.

Format

A list

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[path2gene](#)

getAexample	<i>Get a set of genes</i>
-------------	---------------------------

Description

Get a set of genes.

Usage

```
getAexample(k=1000)
```

Arguments

k An integer. Set numbers of genes returned from the function.

Value

A vector of character.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

Examples

```
##get an example of gene sets
geneList<-getAexample(k=1000)
geneList[1:10]
```

getAnn	<i>Annotate a set of genes to pathways or sub-pathways based on enzyme commission (EC)</i>
--------	--------------------------------------------------------------------------------------------

Description

Annotate a set of genes to pathways or sub-pathways based on enzyme commission (EC).

Usage

```
getAnn(geneList, background=getDefaultBackground(),
       order="pvalue", decreasing=FALSE, graphList=getDefaultGraph())
```

Arguments

<code>geneList</code>	A character vector of genes.
<code>background</code>	A character vector of genes used to identify the statistically significantly enriched pathways.
<code>order</code>	A character string. Should be one of "pvalue", "qvalue".
<code>decreasing</code>	A logical. Should the sort order be increasing or decreasing?
<code>graphList</code>	A list. its elements may be a <code>graph-class</code> or a vector of character.

Details

The function can annotate a set of genes to pathways or sub-pathways and identify the statistically significantly enriched pathways.

The default value of the argument `graphList` is obtained from the function [getDefaultGraph](#). It means that genes will be annotated to all pathway. If users hope to annotate genes to metabolic pathways or sub-pathways of metabolic pathways based on enzyme commission (EC), the value of `graphList` should be changed with the function [getDefaultUndirectedGraph](#) or [getKcSubGraph](#).

Before you use the function, had better use the function [getOrgAndIdType](#) to get the type of current organism and gene identifiers from the environment variable. If the value is different from the type of organism and gene identifier in your current study, you must change them by using the function [updateOrgAndIdType](#), [data](#) or [loadKe2g](#).

If users don't set the argument `background`, the background distribution will be obtained from the whole-genome genes. Detailed information is provided in the function [getDefaultBackground](#).

Value

A list. Each element of the list is another list. It includes eight elements: 'pathwayName', 'annGeneList', 'annGeneNumber', 'annBgNumber', 'geneNumber', 'bgNumber', 'pvalue', 'qvalue'. They correspond to pathway name, the submitted genes annotated to the pathway, numbers of submitted genes annotated to the pathway, numbers of background genes annotated to the pathway, numbers of submitted genes, numbers of background genes, p-value, and FDR-corrected q-value.

To visualize and save the results, the `list` can be converted to the `data.frame` by the function [printAnn](#). But, note that, compared with `data.frame`, the `list` provides more information, e.g., the annotated genes are saved in the `list`, yet not in the `data.frame`.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getKOAnn](#), [getMpAnn](#), [getKcsmpAnn](#), [cutoffAnn](#), [printAnn](#)

Examples

```

##get a set of genes
geneList<-getAexample(k=1000)
##get the annotated results
ann<-getAnn(geneList)
##print the annotation results to screen
printAnn(ann)

##get the details of the results
#get the informations of the first pathway
ann[[1]]
#get the informations of the pathway "path:00010"
ann[["path:00010"]]
#get the genes annotated to the pathway "path:00010"
ann[["path:00010"]]$"annGeneList"

##write the annotation results to tab delimited file.
#note that the argument col.names=NA is essential.
geneList<-getAexample(k=1000)
ann<-getAnn(geneList)
result<-printAnn(ann)
write.table(result,file="result",col.names=NA,sep="\t")

##annotate the genes to the KEGG metabolic pathways based on enzyme commission (EC)
geneList<-getAexample(k=1000)
ann<-getAnn(geneList,graphList=getDefaultUndirectedGraph())
printAnn(ann)[2:5]

##annotate the genes to the sub-pathways of metabolic pathways based on enzyme commission (E
geneList<-getAexample(k=1000)
subGraphList<-getKcSubGraph(k=4,graphList=getDefaultUndirectedGraph())
ann<-getAnn(geneList,graphList=subGraphList)
printAnn(ann)[2:5]

```

```
getDefaultBackground
```

Get the default background genes

Description

Get the default background genes.

Usage

```
getDefaultBackground()
```

Details

The default background genes are obtained from the variable `background` in the environment variable `ke2g`.

Value

A character vector of background genes.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

Examples

```
getDefaultBackground()
```

```
getDefaultGraph
```

Get relationship between KEGG pathways and genes

Description

Get relationship between KEGG pathways and genes

Usage

```
getDefaultGraph()
```

Details

Get relationship between KEGG pathways and genes. The relationship is obtained from the variable [path2gene](#) in the environment variable.

Note that the "graph" is a list of character rather than the `graph-class`.

Value

A list of character.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

Examples

```
getDefaultGraph()
```

```
getDefaultKOPathway
```

Get an identifier list of the default pathways based on KEGG Orthology (KO)

Description

Get an identifier list of the default pathways based on KEGG Orthology (KO).

Usage

```
getDefaultKOPathway ()
```

Details

The identifier list of pathways based on KEGG Orthology (KO) is obtained from the variable `kpIdList` in the environment variable.

Value

A character vector of pathway identifiers.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

Examples

```
getDefaultKOPathway ()
```

```
getDefaultKOUndirectedGraph
```

Get the default undirected graph based on KEGG Orthology

Description

Get the default undirected graph based on KEGG Orthology.

Usage

```
getDefaultKOUndirectedGraph
```

Details

The default undirected graph based on KEGG Orthology is obtained from the variable `KOuGraph` in the environment variable.

Value

A list of `graph-class`.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

Examples

```
getDefaultKOUndirectedGraph()
```

```
getDefaultMetabolicPathway
```

Get an identifier list of the default metabolic pathways

Description

Get an identifier list of the default metabolic pathways.

Usage

```
getDefaultMetabolicPathway()
```

Details

The identifier list of the default metabolic pathways is obtained from the variable `mpidList` in the environment variable.

Value

A character vector of pathway identifiers.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

Examples

```
getDefaultMetabolicPathway()
```

```
getDefaultUndirectedGraph
    Get the default undirected graph based onenzyme commission (EC)
    numbers
```

Description

Get the default undirected graph based onenzyme commission (EC) numbers.

Usage

```
getDefaultUndirectedGraph()
```

Details

The default undirected graph based onenzyme commission (EC) numbers is obtained from the variable `uGraph` in the environment variable.

Value

A list of `graph-class`.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

Examples

```
getDefaultUndirectedGraph()
```

```
getEnzymeFromGene Get a list of enzymes from a list of genes
```

Description

Get a list of enzymes from a list of genes.

Usage

```
getEnzymeFromGene(geneList)
```

Arguments

`geneList` A character vector of genes.

Details

Note that the result is the union of sets of enzymes.

Value

A character vector of enzymes.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getGeneFromEnzyme](#)

Examples

```
## get the list of enzymes of two genes.  
getEnzymeFromGene(c("5232", "5224"))
```

`getGeneFromEnzyme` *Get a list of genes from a list of enzymes*

Description

Get a list of genes from a list of enzymes.

Usage

```
getGeneFromEnzyme(enzymeList)
```

Arguments

`enzymeList` A character vector of genes.

Details

Note that the result is the union of sets of genes.

Value

A character vector of genes.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getEnzymeFromGene](#)

Examples

```
## get the list of gene of two enzymes.  
getGeneFromEnzyme(c("ec:2.7.2.3", "ec:5.4.2.4"))
```

getGeneFromKO	<i>Get a list of genes from a list of KOs</i>
---------------	-----------------------------------------------

Description

Get a list of genes from a list of KOs.

Usage

```
getGeneFromKO(KOList)
```

Arguments

KOList A character vector of genes.

Details

Note that the result is the union of sets of genes.

Value

A character vector of genes.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getKOfFromGene](#)

Examples

```
## get the list of gene of two KOs.  
getGeneFromKO(c("ko:K00622", "ko:K01488"))
```

getGeneFromPathway *Get genes from pathways*

Description

Get genes from pathways.

Usage

```
getGeneFromPathway (pathwayList)
```

Arguments

pathwayList A character vector of pathways.

Details

Note that the result is the union of sets of genes.

Value

A character vector of genes.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getEnzymeFromGene](#)

Examples

```
## get the list of genes of two pathways.  
getGeneFromPathway(c("path:00010", "path:00020"))
```

getKcsmpAnn	<i>Get the sub-pathways annotation and identification of metabolic network</i>
-------------	--------------------------------------------------------------------------------

Description

Annotate a set of genes to sub-pathways of metabolic pathways and identify the statistically significantly enriched sub-pathways.

Usage

```
getKcsmpAnn(geneList, background=getDefaultBackground(), k=4,  
            order="pvalue", decreasing=FALSE)
```

Arguments

geneList	A character vector of genes.
background	A character vector of genes used to identify the statistically significantly enriched sub-pathways.
k	An integer. A distance similarity parameter.
order	A character string. Should be one of "pvalue", "qvalue".
decreasing	A logical. Should the sort order be increasing or decreasing?

Details

The function can implement the sub-pathways annotation of metabolic pathways. It is the special form of the function [getAnn](#) where the argument `graphList` is the return value of the function [getKcSubGraph](#). Detailed information is provided in [getAnn](#).

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getMpAnn](#), [getAnn](#), [cutoffAnn](#), [printAnn](#)

Examples

```
##Annotate a set of genes to sub-pathways of metabolic pathways  
geneList<-getAexample(k=1000)  
ann<-getKcsmpAnn(geneList, k=4)  
printAnn(ann) [2:5]
```

getKcSubGraph *Mine sub-pathways*

Description

Mine sub-pathways by using the k-clique concept in social network analysis.

Usage

```
getKcSubGraph(k=4, graphList=getDefaultUndirectedGraph())
```

Arguments

`k` An integer. A distance similarity parameter.
`graphList` An undirected graph list. Detailed information is provided in [uGraph](#) or [KOUGraph](#).

Details

The function uses the k-clique concept in social network analysis to mine sub-pathways. In social network analysis, a k-clique in a graph is a subgraph where the distance between any two nodes is no greater than k.

The default value of the argument `graphList` is obtained from the function [getDefaultUndirectedGraph](#). It means to mine sub-pathways based on enzyme commission (EC) numbers from the version of metabolic pathways. If users hope to mine sub-pathways based on KEGG Orthology (KO), the value of `graphList` should be changed with the function [getDefaultKOUndirectedGraph](#).

Value

A list of graphs.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

References

Wasserman,S. and Faust,K. (1994) Social network analysis: methods and applications. Cambridge University Press., New York, America.
Huber,W., Carey,V.J., Long,L., Falcon,S. and Gentleman,R. (2007) Graphs in molecular biology. BMC Bioinformatics., 8, s8.

See Also

[getAnn](#),[getKOAnn](#),[getKcsmPAnn](#)

Examples

```
##get graph representation of metabolic pathways
graphList<-getDefaultUndirectedGraph()
#get all 4-clique subgraphs
subGraphList<-getKcSubGraph(k=4,graphList)
#display first subGraph
library(Rgraphviz)
plot(subGraphList[[1]])

##annotate a set of genes to the sub-pathways of metabolic pathways by using enzyme commissi
geneList<-getAexample(k=1000)
subGraphList<-getKcSubGraph(k=4)
ann<-getAnn(geneList,graphList=subGraphList)
printAnn(ann)

##annotate the genes to sub-pathway by using KEGG Orthology (KO).
geneList<-getAexample(k=100)
subGraphList<-getKcSubGraph(k=4,graphList=getDefaultKOUndirectedGraph())
ann<-getKOAnn(geneList,graphList=subGraphList)
printAnn(ann)[2:5]
```

getKOAnn	<i>Annotate a set of genes to sub-pathways by using KEGG Orthology (KO)</i>
----------	-----------------------------------------------------------------------------

Description

Annotate a set of genes to sub-pathways by using KEGG Orthology (KO).

Usage

```
getKOAnn(geneList,background=getDefaultBackground(),
order="pvalue",decreasing=FALSE,graphList)
```

Arguments

geneList	A character vector of genes.
background	A character vector of genes used to identify the statistically significantly enriched pathways.
order	A character string. Should be one of "pvalue", "qvalue".
decreasing	A logical. Should the sort order be increasing or decreasing?
graphList	A list. its elements may be a graph-class or a vector of character.

Details

The function can annotate a set of genes to sub-pathways and identify the statistically significantly enriched pathways.

The value of `graphList` should be changed with the function `getKcSubGraph` by setting arguments `graphList=getDefaultKOUndirectedGraph`.

Before you use the function, had better use the function `getOrgAndIdType` to get the type of current organism and gene identifiers from the environment variable. If the value is different from the type of organism and gene identifier in your current study, you must change them by using the function `updateOrgAndIdType`, `data` or `loadKe2g`.

If users don't set the argument `background`, the background distribution will be obtained from the whole-genome genes. Detailed information is provided in the function `getDefaultBackground`.

Value

A list. Each element of the list is another list. It includes eight elements: 'pathwayName', 'annGeneList', 'annGeneNumber', 'annBgNumber', 'geneNumber', 'bgNumber', 'pvalue', 'qvalue'. They correspond to pathway name, the submitted genes annotated to the pathway, numbers of submitted genes annotated to the pathway, numbers of background genes annotated to the pathway, numbers of submitted genes, numbers of background genes, p-value, and FDR-corrected q-value.

To visualize and save the results, the `list` can be converted to the `data.frame` by the function `printAnn`. But, note that, compared with `data.frame`, the `list` provides more information, e.g., the annotated genes are saved in the `list`, yet not in the `data.frame`.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

`getKcSubGraph`, `getDefaultKOUndirectedGraph`, `getAnn`, `cutoffAnn`, `printAnn`, `plotKOAnn`

Examples

```
##annotate the genes to sub-pathway by using KEGG Orthology (KO).
##get a set of genes
geneList<-getAexample(k=100)
#
subGraphList<-getKcSubGraph(k=4,graphList=getDefaultKOUndirectedGraph())
#get the annotated results
ann<-getKOAnn(geneList,graphList=subGraphList)
#print the annotation results to screen
result<-printAnn(ann)
result[2:5]

##write the annotation results to tab delimited file.
#note that the argument col.names=NA is essential.
write.table(result,file="result",col.names=NA,sep="\t")
```

getKOFromGene	<i>Get a list of KOs from a list of genes</i>
---------------	-----------------------------------------------

Description

Get a list of KOs from a list of genes.

Usage

```
getKOFromGene(geneList)
```

Arguments

geneList A character vector of genes.

Details

Note that the result is the union of sets of KOs.

Value

A character vector of KOs.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getGeneFromKO](#)

Examples

```
## get the list of KOs of two genes.  
getKOFromGene(c("10", "9", "100"))
```

`getMpAnn`*get annotation informations of metabolic pathways*

Description

Annotate a set of genes to metabolic pathways based on EC.

Usage

```
getMpAnn(geneList, background=getDefaultBackground(),
         order="pvalue", decreasing=FALSE)
```

Arguments

<code>geneList</code>	A character vector of genes.
<code>background</code>	A character vector of genes used to identify the statistically significantly enriched sub-pathways.
<code>order</code>	A character string. Should be one of "pvalue", "qvalue".
<code>decreasing</code>	A logical. Should the sort order be increasing or decreasing?

Details

The function can implement the annotation and identification of metabolic pathways. It is the special form of the function `getAnn` where the argument `graphList` is the return value of the function `getDefaultUndirectedGraph`. Detailed information is provided in `getAnn`.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

`getKcsmpAnn`, `getAnn`, `cutoffAnn`, `printAnn`

Examples

```
##annotate a set of genes to metabolic pathways
geneList<-getAexample(k=1000)
ann<-getMpAnn(geneList)
printAnn(ann)
```

getOrgAndIdType	<i>Get the type of organism and identifier in the current environment variable</i>
-----------------	------------------------------------------------------------------------------------

Description

Get the type of organism and identifier in the current environment variable.

Usage

```
getOrgAndIdType ()
```

Details

Users should ensure that the type of organism and gene identifiers accord with the return value of the function [getOrgAndIdType](#). The function can help you check the type of organism and identifiers in the current system.

If the return values are different from the type of your genes, you need to change them with some methods. Detailed information is provided in [loadKe2g](#), [updateOrgAndIdType](#).

Value

A character vector.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[updateOrgAndIdType](#)

Examples

```
getOrgAndIdType ()
```

getPathwayFromGene *Get pathwaysfrom genes*

Description

Get pathwaysfrom genes.

Usage

```
getPathwayFromGene (geneList)
```

Arguments

geneList A character vector of genes.

Details

Note that the result is the union of sets of pathways.

Value

A character vector of pathways.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getGeneFromPathway](#)

Examples

```
## get the list of pathways of two genes.  
getPathwayFromGene(c("5224", "8802"))
```

```
getPathwayNameFromId
```

Get pathway name from identifier

Description

Get pathway name from identifier.

Usage

```
getPathwayNameFromId(pid)
```

Arguments

`pid` A character vector of pathway identifiers.

Value

A character vector.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

Examples

```
getPathwayNameFromId("00010")
```

```
gotoKEGG
```

Visualize pathways or sub-pathways through linking to the KEGG web site

Description

Visualize pathways or sub-pathways through linking to the KEGG web site.

Usage

```
gotoKEGG(pathway, ann)
```

Arguments

`pathway` A character string of pathway identifier, e.g., "path:00010".
`ann` A list, e.g., the return value of the function [getAnn](#).

Details

On these pictures, The red nodes represent the enzymes (or KOs) that include the submitted genes.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[plotAnn](#), [getAnn](#), [printAnn](#)

Examples

```
##visualize pathways through linking to the KEGG web site.
library(Rgraphviz)
geneList<-getAexample(k=100)
ann<-getAnn(geneList)
#gotoKEGG("path:00010",ann)

##visualize sub-pathways of metabolic pathways based on enzyme commission (EC) numbers throu
geneList<-getAexample(k=1000)
subGraphList<-getKcSubGraph(k=4)
ann<-getAnn(geneList,graphList=subGraphList)
#gotoKEGG("path:00010_1",ann)

##visualize sub-pathways sub-pathways based on KEGG Orthology (KO) identifiers through linki
geneList<-getAexample(k=100)
subGraphList<-getKcSubGraph(k=4,graphList=getDefaultKOUndirectedGraph())
ann<-getKOAnn(geneList,graphList=subGraphList)
#gotoKEGG("path:00010_1",ann)
```

ke2g

The environment variable of the system

Description

The environment variable of the system.

Format

An environment variable

Details

The environment variable includes the variable `uGraph`, `dGraph`, `KOuGraph`, `gene2path`, `path2gene`, `gene2ec`, `ec2gene`, `gene2KO`, `KO2gene`, `mpidList`, `background`, `keggpathid2name`, `kid2oid`, `oid2kid`.

The system provides the environment variable of some organisms for biologists. The `ath` is the abbreviation of *Arabidopsis thaliana*. The `osa` is the abbreviation of *Oryza sativa japonica*. The `dme` is the abbreviation of *Drosophila melanogaster*. The `eco` is the abbreviation of *Escherichia coli* K-12 MG1655. The `hsa` is the abbreviation of *Homo sapiens*. The `sce` is the abbreviation of *Saccharomyces cerevisiae*. The `mmu` is the abbreviation of *Mus musculus*. `Cel` is the abbreviation of *Caenorhabditis elegans*.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[loadKe2g](#), [saveKe2g](#)

Ke2gEnvironment *ke2g environment function*

Description

Load and save the environment variable `ke2g` of the system.

Usage

```
saveKe2g(file="ke2g.rda")
loadKe2g(file="ke2g.rda")
```

Arguments

`file` A character string.

Details

The functions are used to load or save the environment variable `ke2g` of the system. If one has changed the environment variable with some functions, e.g., `updateOrgAndIdType` or `updateGraphs` or `updateKOGraphs` and hope to use the setting in the future, then the functions can solve the problem. For example, one uses the function `saveKe2g` to save the environment variable. When one needs to use the setting next time, one can use the function `loadKe2g` to load the last environment variable.

The functions implement the localization of the system. Therefore, the system not only implements the most up-to-date annotations but also the localized annotations. The functions are very important for the user to frequently annotate genes with the different genomes and the type of identifiers. It

can increase largely the running speed because one can update the data one time only, and then repeatedly use it.

Note that if and only if the function `loadKe2g` runs after loading the `SubpathwayMiner` package, it will be in effect.

In addition, the environment variables of organisms with well annotated genomes are provided by the system and users can use the function `data` to load them.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[updateOrgAndIdType](#), [updateGraphs](#), [updateKOGraphs](#)

Examples

```
##for example
#save two ke2g environment about yeast.
#Note that the data is saved to the working directory
#and its name is sce_ncbi-geneid.rda.
# updateOrgAndIdType("sce", "ncbi-geneid")
# saveKe2g("sce_ncbi-geneid.rda")

#shut down the R system
#start up the R system
#library(SubpathwayMiner)

#when the user annotate yeast genes. Note that you need to
#working directory to the directory of the data file.
#loadKe2g("sce_ncbi-geneid.rda")

#The code below is used to load the environment variables provided by the system.
#the type of gene identifier is ncbi-geneid.
#data("sce_ncbi-geneid")
```

keggpathid2name *The data that maps KEGG pathway identifiers to names*

Description

The data that maps KEGG pathway identifiers to names.

Format

A vector of character

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

KO2gene

A list that maps KEGG Orthology (KO) to gene identifiers

Description

A list that maps KEGG Orthology (KO) to gene identifiers.

Format

A list

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[gene2KO](#)

KOugraph

The undirected graph list based on KEGG Orthology (KO) identifiers constructed from KEGG pathways

Description

Undirected graph list based on KEGG Orthology (KO) identifiers constructed from KEGG pathways.

Format

A list of graph

Details

For metabolic pathways, two KO identifiers are connected by an edge if there is a common compound in the KO identifiers corresponding reactions. For regulatory pathways, two KOs are connected by an edge if there are relationships between the two KOs, which can get from relation element of the XML file. The relation element specifies relationship between two KOs, which is indicated by an arrow or a line connecting two nodes in the KEGG pathways. Through the above methods to take out the relationship, all pathways are converted to an undirected graph with enzymes as nodes, which is considered as the corresponding simplification version of KEGG pathway.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

References

Ogata,H., Fujibuchi,W., Goto,S. and Kanehisa,M. (2000) A heuristic graph comparison algorithm and its application to detect functionally related enzyme clusters. *Nucleic Acids Res.*, 20, 4021-4028.

See Also

[uGraph](#)

kpIdList

pathway identifier list based on KEGG Orthology (KO)

Description

The pathway identifier list. Detailed information is provided in <http://www.genome.jp/kegg/pathway.html>.

Format

A list

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

mpidList

metabolic pathway identifier list

Description

The metabolic pathway identifier list. Detailed information is provided in <http://www.genome.jp/kegg/pathway.html>.

Format

A list

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

path2gene	<i>A list that maps KEGG pathway identifiers to gene identifiers</i>
-----------	----------------------------------------------------------------------

Description

A list that maps KEGG pathway identifiers to gene identifiers.

Format

A list

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[gene2path](#)

plotAnn	<i>Visualize pathways or sub-pathways based on enzyme commission (EC) numbers</i>
---------	-----------------------------------------------------------------------------------

Description

Visualize pathways or sub-pathways based on enzyme commission (EC) numbers.

Usage

```
plotAnn(pathway, graphList, ann, gotoKEGG=FALSE)
```

Arguments

pathway	An character string of pathway identifier, e.g., "path:00010".
graphList	An graph list.
ann	An list, e.g., the return value of the function getAnn .
gotoKEGG	An logical. Detailed informations is provided in the function gotoKEGG .

Details

The function can visualize the pathways or sub-pathways of metabolic pathways based on enzyme commission (EC) numbers. The red nodes in the result graph represent the enzymes which include the submitted genes.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[gotoKEGG](#), [getAnn](#), [printAnn](#)

Examples

```
##visualize metabolic pathways based on enzyme commission (EC) numbers
library(Rgraphviz)
geneList<-getAexample(k=1000)
ann<-getAnn(geneList,graphList=getDefaultUndirectedGraph())
plotAnn("path:00010",getDefaultUndirectedGraph(),ann)

##visualize sub-pathways of metabolic pathways based on enzyme commission (EC) numbers
library(Rgraphviz)
geneList<-getAexample(k=1000)
subGraphList<-getKcSubGraph(k=4)
ann<-getAnn(geneList,graphList=subGraphList)
plotAnn("path:00010_1",subGraphList,ann)
#go to KEGG
#plotAnn("path:00010_1",subGraphList,ann,gotoKEGG=TRUE)
```

plotKOAnn

Visualize pathways or sub-pathways based on KEGG Orthology (KO)

Description

Visualize pathways or sub-pathways based on KEGG Orthology (KO).

Usage

```
plotKOAnn(pathway,graphList,ann,gotoKEGG=FALSE)
```

Arguments

pathway	An character string of pathway identifier, e.g., "path:00010".
graphList	An graph list.
ann	An list, e.g., the return value of the function getKOAnn .
gotoKEGG	An logical. Detailed informations is provided in the function gotoKEGG .

Details

The function can visualize the pathways or sub-pathways based on KEGG Orthology (KO). The red nodes in the result graph represent the enzymes which include the submitted genes.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[gotoKEGG](#), [getKOAnn](#), [printAnn](#)

Examples

```
##visualize sub-pathways based on KEGG Orthology (KO)
library(Rgraphviz)
geneList<-getAexample(k=100)
subGraphList<-getKcSubGraph(k=4, graphList=getDefaultKOUndirectedGraph())
ann<-getKOAnn(geneList, graphList=subGraphList)
plotKOAnn("path:00010_1", subGraphList, ann)
#go to KEGG
#plotKOAnn("path:00010_1", subGraphList, ann, gotoKEGG=TRUE)
```

printAnn

Print the results of pathway annotation and identification

Description

Print the results of pathway annotation and identification.

Usage

```
printAnn(ann)
```

Arguments

ann A list. The results returned from the function [getAnn](#) or [getKOAnn](#).

Value

A data.frame of the annotation results. Its row names are pathway identifiers, e.g. path:00010. Columns include (pathwayName, annGeneRatio, annBgRatio, pvalue, qvalue).

The `annGeneRatio` is the ratio of the annotated genes ,e.g.,30/1000 means that 30 genes in 1000 genes are annotated. The `qvalue` is the FDR-corrected q-value.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getAnn](#), [getKOAnn](#), [cutoffAnn](#)

Examples

```
##get an example of gene list
geneList<-getAexample(k=1000)

##get annotation results
ann<-getAnn(geneList)

##print results to screen
printAnn(ann)

##print subset of columns to visilize well
result<-printAnn(ann)
result[,2:5]

##print subset of rows to visilize well
result[1:10,]

##change print order of columns
result[,c(4,3,2,5)]

##write the annotation results to tab delimited file.
# Notices that the argument col.names=NA is essential.
result<-printAnn(ann)
write.table(result,file="result",col.names=NA,sep="\t")
```

 ugraph

The undirected graph list based on enzyme commission (EC) numbers constructed from KEGG metabolic pathways

Description

Undirected graph list based on enzyme commission (EC) numbers constructed from KEGG pathways.

Format

A list of graph

Details

Generally, A metabolic pathway can be considered as a graph with chemical compounds as nodes and enzymes as edges.

We simplify metabolic pathways. Each metabolic pathway is converted to an undirected graph with enzymes as nodes. Two enzymes are connected by an edge if their corresponding reactions have a common compound. Chemical compounds are then omitted from graphs. .

If we consider the direction of reaction. The pathway will be a directed graph.

Note that each graph in `uGraph` or `dGraph` is constructed from metabolic pathways and one pathway can only build not more than one graph.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

References

Ogata,H., Fujibuchi,W., Goto,S. and Kanehisa,M. (2000) A heuristic graph comparison algorithm and its application to detect functionally related enzyme clusters. *Nucleic Acids Res.*, 20, 4021-4028.

See Also

[dGraph](#),[KOUGraph](#)

updateGraphs

Update graphs constructed from KEGG metabolic pathways

Description

Update graphs constructed from KEGG metabolic pathways.

Usage

```
updateGraphs (pathwayList=getDefaultMetabolicPathway() ,  
              path="ftp://ftp.genome.jp/pub/kegg/release/archive/kgml/KGML_v0.6.1/
```

Arguments

`pathwayList` A character vector of the KEGG metabolic pathway identifiers.
`path` A character string.
`verbose` A logical. If TRUE, the additional diagnostics are printed.

Details

The function is able to update the graph variable `uGraph` and `dGraph` in the environment variable of the system.

We construct and update the simplification version of metabolic pathways by extracting relations from the XML data of metabolic pathways. The XML data is available from the FTP site. ftp://ftp.genome.jp/pub/kegg/release/archive/kgml/KGML_v0.6.1/map/

The argument `pathwayList` is a character vector of the KEGG metabolic pathway identifiers. The default value is provided by the return value of `getDefaultMetabolicPathway`. The user can change the value for constructing the subset of the graphs of the KEGG pathways.

Note that if one needs to use the updated graphs in the future, one should run the function `saveKe2g` to save the setting. Detailed information is provided in [saveKe2g](#) and [loadKe2g](#).

Note that the programming is likely to be time consuming.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[uGraph](#), [dGraph](#), [getDefaultMetabolicPathway](#), [updateOrgAndIdType](#)

Examples

```
##construct the graphs list of Nucleotide Metabolism which include two pathways.
#It is considered as the subset of pathways of interest.
#updateGraphs (pathwayList=c("path:00230", "path:00240"))
```

updateKOGraphs	<i>Update graphs based on based on KEGG Orthology (KO) constructed from KEGG pathways</i>
----------------	-------------------------------------------------------------------------------------------

Description

Update graphs based on based on KEGG Orthology (KO) constructed from KEGG metabolic pathways.

Usage

```
updateKOGraphs (pathwayList=getDefaultKOPathway(),
                path="ftp://ftp.genome.jp/pub/kegg/xml/ko/", verbose=TRUE)
```

Arguments

pathwayList	A character vector of the KEGG pathway identifiers.
path	A character string.
verbose	A logical. If TRUE, the additional diagnostics are printed.

Details

The function is able to update the graph variable `KOuGraph` in the environment variable of the system.

We construct and update the simplification version by extracting relations from the XML data of pathways. The XML data is available from the FTP site. <ftp://ftp.genome.jp/pub/kegg/xml/ko/>

The argument `pathwayList` is a character vector of the KEGG metabolic pathway identifiers. The default value is provided by the return value of `getDefaultKOPathway`. The user can change the value for constructing the subset of the graphs of the KEGG pathways.

Note that if one needs to use the updated graphs in the future, one should run the function `saveKe2g` to save the setting. Detailed information is provided in [saveKe2g](#) and [loadKe2g](#).

Note that the programming is likely to be time consuming.

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[KOUGraph](#), [updateOrgAndIdType](#)

Examples

```
##construct the graphs list of Nucleotide Metabolism which include two pathways.
#It is considered as the subset of pathways of interest.
#updateKOGraphs (pathwayList=c("path:00230", "path:00240"))
```

updateOrgAndIdType *Update the organism and the type of gene identifier*

Description

Update the organism and the type of gene identifier.

Usage

```
updateOrgAndIdType (org="hsa", idType="ncbi-geneid",
                    path="ftp://ftp.genome.jp/pub/kegg/genes/organisms", verbose=TRUE)
```

Arguments

org	A character string. The abbreviation of a genome name.
idType	A character string. The type of gene identifier.
path	A character string.
verbose	A logical. If TRUE, the additional diagnostics are printed.

Details

The existing tools mainly use DBMS (data base management system) to store all data relative to analysis of pathways and the update process of the data is transparent to users, which means that the annotation results users get from these tools may become outdated. We don't use DBMS to store data. We present a new method that enables users to update data by themselves. Users are firstly required to set organism and type of gene identifier before annotating genes to the pathways. According to the setting, the system can download all data relative to analysis of pathways in the certain organism, and then treat and store them in an environment variable in R. Through the method the system can synchronize the data with the KEGG databases and support almost all organisms and cross reference identifiers in the KEGG GENE database.

The function is able to update the variable [gene2ec](#) , [ec2gene](#) , [gene2KO](#) , [KO2gene](#) , [gene2path](#) and [path2gene](#) and [background](#) in the environment variable.

Note that if the user don't run the function `updateOrgAndIdType,loadKe2g` after starting up R-system and loading the package of the system, then the default value of the argument `org` and `idType` is "hsa" (human) and "ncbi-geneid" (Entrez gene identifiers). The user can get the information from the return value of the function `getOrgAndIdType`.

The argument `org` must be the abbreviation of a genome name. For example, the hsa, mms, eco, sce and dme is the abbreviation of human, mouse, E.coli, yeast and fruit fly. Detailed information is provided in http://www.genome.jp/kegg/catalog/org_list.html.

The argument `idType` is a character string of the type of identifier. The system supports most KEGG cross-reference identifiers such as Entrez gene IDs (`idType="ncbi-geneid"`), NCBI gi numbers (`idType="ncbi-gi"`), UniProt accession numbers (`idType="uniprot"`), etc. Detailed information is provided in <ftp://ftp.genome.jp/pub/kegg/genes/organisms>. For example, because a file name in "hsa" file directory is "hsa_ensembl-hsa.list", `idType="ensembl-hsa"` is available as the input identifier type. Note that the `idType` is relative to the genome. Different genomes may support different `idType`. For example, "sgd-sce" is supported by yeast genome. However, it is not supported by human genome.

The argument `path` is the path of file directory of the organism cross-reference identifiers. The default value is "ftp://ftp.genome.jp/pub/kegg/genes/organisms". The setting ensures that the user is able to obtain the updated data from the KEGG FTP site. Of course, the user can also download the organisms data of interest from FTP site and change path to the data file for implementing the local update.

Note that the programming is time consuming. For solving the problem, see [saveKe2g](#) and [loadKe2g](#).

Author(s)

Chunquan Li <lcqbio@yahoo.com.cn>

See Also

[getAnn](#), [updateGraphs](#), [getOrgAndIdType](#)

Examples

```
##update organism and the type of gene identifiers
#getOrgAndIdType()
#updateOrgAndIdType("sce", "sgd-sce")
#getOrgAndIdType()
```

Index

*Topic **datasets**

- background, 2
- ec2gene, 4
- gene2ec, 4
- gene2KO, 5
- gene2path, 5
- KO2gene, 28
- kpidList, 29
- mpidList, 29
- path2gene, 30

*Topic **file**

- cutoffAnn, 2
- getAexample, 6
- getAnn, 6
- getDefaultBackground, 8
- getDefaultGraph, 9
- getDefaultKOPathway, 10
- getDefaultKOUndirectedGraph, 10
- getDefaultMetabolicPathway, 11
- getDefaultUndirectedGraph, 12
- getEnzymeFromGene, 12
- getGeneFromEnzyme, 13
- getGeneFromKO, 14
- getGeneFromPathway, 15
- getKcsmpAnn, 16
- getKcSubGraph, 17
- getKOAnn, 18
- getKOFromGene, 20
- getMpAnn, 21
- getOrgAndIdType, 22
- getPathwayFromGene, 23
- getPathwayNameFromId, 24
- gotoKEGG, 24
- ke2g, 25
- Ke2gEnvironment, 26
- keggpathid2name, 27
- plotAnn, 30

- plotKOAnn, 31
- printAnn, 32
- updateGraphs, 34
- updateKOGraphs, 35
- updateOrgAndIdType, 36

*Topic **graphs**

- dgraph, 3
- KOUgraph, 28
- ugraph, 33

- background, 2, 8, 26, 36

- cutoffAnn, 2, 7, 16, 19, 21, 32

- data, 7, 19, 27
- dGraph, 26, 34, 35
- dGraph (*dgraph*), 3
- dgraph, 3

- ec2gene, 4, 4, 26, 36

- gene2ec, 4, 4, 26, 36
- gene2KO, 5, 26, 28, 36
- gene2path, 5, 26, 30, 36
- getAexample, 6
- getAnn, 2, 3, 6, 16, 17, 19, 21, 24, 25, 30–32, 37

- getDefaultBackground, 7, 8, 19
- getDefaultGraph, 7, 9
- getDefaultKOPathway, 10, 35
- getDefaultKOUndirectedGraph, 10, 17, 19

- getDefaultMetabolicPathway, 11, 34, 35

- getDefaultUndirectedGraph, 7, 12, 17, 21

- getEnzymeFromGene, 12, 14, 15
- getGeneFromEnzyme, 13, 13
- getGeneFromKO, 14, 20
- getGeneFromPathway, 15, 23
- getKcsmpAnn, 7, 16, 17, 21

getKcSubGraph, 7, 16, 17, 19
getKOAnn, 2, 3, 7, 17, 18, 31, 32
getKOfFromGene, 14, 20
getMpAnn, 7, 16, 21
getOrgAndIdType, 7, 19, 22, 22, 37
getPathwayFromGene, 23
getPathwayNameFromId, 24
gotoKEGG, 24, 30–32

ke2g, 8, 25, 26
Ke2gEnvironment, 26
keggpathid2name, 26, 27
KO2gene, 5, 26, 28, 36
KOUGraph, 11, 17, 26, 34–36
KOUGraph (*KOUgraph*), 28
KOUgraph, 28
kpidList, 10, 29

loadKe2g, 7, 19, 22, 26, 34, 35, 37
loadKe2g (*Ke2gEnvironment*), 26

mpidList, 11, 26, 29

path2gene, 5, 9, 26, 30, 36
plotAnn, 25, 30
plotKOAnn, 19, 31
printAnn, 3, 7, 16, 19, 21, 25, 31, 32, 32

saveKe2g, 26, 34, 35, 37
saveKe2g (*Ke2gEnvironment*), 26

uGraph, 3, 12, 17, 26, 29, 34, 35
uGraph (*ugraph*), 33
ugraph, 33
updateGraphs, 26, 27, 34, 37
updateKOUgraphs, 26, 27, 35
updateOrgAndIdType, 7, 19, 22, 26, 27,
35, 36, 36, 37