

Package ‘MetaboSignal’

October 18, 2017

Type Package

Title MetaboSignal: a network-based approach to overlay and explore metabolic and signaling KEGG pathways

Version 1.6.0

Date 2016-06-27

Author Andrea Rodriguez-Martinez

<andrea.rodriguez-martinez13@imperial.ac.uk>, Rafael Ayala

<r.ayala14@imperial.ac.uk>, Joram M. Posma

<j.posma11@imperial.ac.uk>, Ana L. Neves

<ana.luisa.neves14@imperial.ac.uk>, Marc-Emmanuel Dumas

<m.dumas@imperial.ac.uk>

Maintainer Andrea Rodriguez-Martinez

<andrea.rodriguez-martinez13@imperial.ac.uk>, Rafael Ayala

<r.ayala14@imperial.ac.uk>

Description MetaboSignal is an R package that allows merging, analyzing and customizing metabolic and signaling KEGG pathways. It is a network-based approach designed to explore the topological relationship between genes (signaling- or enzymatic-genes) and metabolites, representing a powerful tool to investigate the genetic landscape and regulatory networks of metabolic phenotypes.

License GPL-3

Depends R(>= 3.3)

Suggests RUnit, BiocGenerics, knitr, BiocStyle, rmarkdown

VignetteBuilder knitr

Imports KEGGgraph, mygene, hpar, org.Hs.eg.db, igraph, RCurl, biomaRt, KEGGREST, AnnotationDbi, MWASTools, stats, graphics, utils

biocViews GraphAndNetwork, GeneSignaling, GeneTarget, Network, Pathways, KEGG, Reactome, Software

NeedsCompilation no

LazyData true

Encoding UTF-8

R topics documented:

directionality_reactions	2
hpaNormalTissue	3
MetaboSignal_distances	3
MetaboSignal_matrix	5
MetaboSignal_NetworkCytoscape	7
MetaboSignal_table	9
MetaboSignal_tableUnfiltered	10
MS_ChangeNames	10
MS_FilterNetwork	11
MS_FindKEGG	12
MS_FindMappedNodes	13
MS_GetKEGG_GeneID	14
MS_GetShortestpaths	15
MS_interactionType	17
MS_NodeBW	18
MS_RemoveNode	18
MS_ReplaceNode	19
MS_ToCytoscape	20
neglected_genes	21
rno_pathways	21
subnetwork	22
Index	23

directionality_reactions

List of KEGG reactions with incorrect/inconsistent directionality

Description

This matrix contains a list of KEGG reactions with incorrect/inconsistent directionality. The directionality of these reactions has been corrected based on published literature. This matrix can be updated or edited by the user if required.

Usage

```
directionality_reactions
```

Format

Matrix

Value

Matrix

hpaNormalTissue

Expression profiles for proteins in human tissues

Description

This data frame contains tissue expression data of human proteins, based on the Human Protein Atlas project. This data frame was obtained from the hpar package, and it is used in MetaboSignal to filter signaling genes based on tissue expression.

Usage

```
data(hpaNormalTissue)
```

Format

Data.frame

Value

Data.frame

MetaboSignal_distances

Calculate gene-metabolite distance matrix

Description

This function generates a distance matrix containing the length of all shortest paths from a set of genes (or reactions) to a set of metabolites. The shortest path length between two nodes is defined as the minimum number of edges between these two nodes.

Usage

```
MetaboSignal_distances(network_table, organism_code, organism_name, mode = "SP",
                       source_genes = "all", target_metabolites = "all", names = FALSE)
```

Arguments

network_table two-column matrix where each row represents an edge between two nodes. See function "MetaboSignal_matrix ()".

organism_code character vector containing the KEGG code for the organism of interest. For example the KEGG code for the rat is "rno". See the function "MS_FindKEGG()".

organism_name character vector containing the common name of the organism of interest (e.g. "rat", "mouse", "human", "zebrafish") or taxonomy id. For more details, check: <http://docs.mygene.info/en/latest/doc/data.html#species>. This argument is only required when source_genes are gene symbols.

mode	character constant indicating whether a directed or an undirected network will be considered. "all" indicates that all the edges of the network will be considered as undirected. "out" indicates that all the edges of the network will be considered as directed. "SP" indicates that all network will be considered as directed except the edges linked to target metabolite, which will be considered as undirected. The difference between the "out" and the "SP" options, is that the latter aids reaching target metabolites that are substrates of irreversible reactions. By default, mode = "SP".
source_genes	character vector containing the genes from which the shortest paths will be calculated. All input genes need to have the same ID format. Possible ID formats are: entrez IDs, official gene symbols, or gene nodes of the network (i.e. KEGG orthology IDs or KEGG gene IDs). The latter option allows reducing the time required to compute this function. Entrez IDs or gene symbols can be transformed into KEGG IDs using the function "MS_GetKEGG_GeneID()". By default, genes = "all" indicating that all genes or reactions of the network will be used.
target_metabolites	character vector containing the KEGG IDs of the metabolites to which the shortest paths will be calculated. Compound KEGG IDs can be obtained using the function "MS_FindKEGG()". By default, metabolites = "all", indicating that all metabolites of the network will be used.
names	logical scalar indicating whether the metabolite IDs or gene KEGG IDs will be transformed into common metabolite names or gene symbols. Reaction IDs remain unchanged. By default, names = FALSE.

Value

A matrix containing the shortest path length from the genes or reactions (in the rows) to the metabolites (in the columns). For unreachabe metabolites Inf is included.

References

Csardi, G. & Nepusz, T. (2006). The igraph software package for complex network research. *Inter-Journal, Complex Systems*, 1695.

Examples

```
data(MetaboSignal_table)

# Distances from Ship2 (65038) and Ppp2r5b (309179) to D-glucose ("cpd:C00031")
distances_targets <- MetaboSignal_distances(MetaboSignal_table, organism_code = "rno",
                                           source_genes = c("65038", "309179"),
                                           target_metabolites = "cpd:C00031",
                                           names = TRUE)

# Distances from all genes to all metabolites of the network
distances_all <- MetaboSignal_distances(MetaboSignal_table, organism_code = "rno")
```

MetaboSignal_matrix *Build MetaboSignal network-table*

Description

This function generates a directed network-table (i.e. two-column matrix), where each row represents an edge connecting two nodes (from node in column 1 to node in column 2). Nodes represent four different molecular entities: metabolic-genes (i.e. genes encoding enzymes that catalyze metabolic reactions), signaling-genes (e.g. kinases), reactions or metabolites. It is possible to build a tissue-specific network-table that excludes signaling genes that are not expressed in a given tissue. Tissue expression data is obtained using the hpar package, which is based on the The Human Protein Atlas database. The genes "non detected" in the target tissue (reliability = supportive) are neglected.

The network-table generated with this function can be customized based on several criteria. For instance, undesired nodes can be removed or replaced using the functions "MS_RemoveNode()" or "MS_ReplaceNode()" respectively. Also, the network can be filtered according to different topological parameters (e.g. node betweenness) using the function "MS_FilterNetwork()".

Usage

```
MetaboSignal_matrix(metabo_paths, signaling_paths, organism_name, tissue = "all",
                    expand_genes = FALSE)
```

Arguments

metabo_paths	character vector containing the KEGG IDs of the metabolic pathways of interest (organism-specific). For example, the KEGG ID for the pathway "glycolysis/gluconeogenesis" in the rat is "rno00010". See the function "MS_FindKEGG()".
signaling_paths	character vector containing the KEGG IDs for the signaling pathways of interest (organism-specific). For example, the KEGG ID for the pathway "insulin signaling pathway" in the rat is "rno04910".
organism_name	character vector containing the common name of the organism of interest (e.g. "rat", "mouse", "human", "zebrafish") or taxonomy id. For more details, check: http://docs.mygene.info/en/latest/doc/data.html#species . This argument is only required when filtering genes by tissue expression.
tissue	character vector containing the name(s) of the target tissue(s). By default, tissue = "all" indicating that signaling gene nodes will not be filtered by tissue expression. Otherwise, possible tissues are those included in the dataset hpaNormalTissue (see levels(hpaNormalTissue[, "Tissue"])).
expand_genes	logical scalar indicating whether the gene nodes will represent orthology IDs (FALSE) or organism-specific gene IDs (TRUE). By default, expand_genes = FALSE.

Value

A two-column matrix where each row represents an edge between two nodes.

 MetaboSignal_NetworkCytoscape

Build shortest-path subnetwork

Description

This function allows calculating the shortest paths from a set of genes to a set of metabolites, and representing them as a network-table (i.e. two-column matrix). By default, the function exports a network file ("CytoscapeNetwork.txt") and two attribute files ("CytoscapeAttributesType.txt", "CytoscapeAttributesTarget.txt"), which can be imported into cytoscape to visualize the network. The first attribute file allows customizing the nodes of the network based on the molecular entity they represent: metabolic-genes, signaling-genes, or metabolites. The second attribute file allows discriminating the source_genes and the target_metabolites ("target") from any other node ("untarget") of the network.

The network-table generated with this function can be further customized based on different criteria. For instance, undesired nodes can be removed or replaced using the functions "MS_RemoveNode()" or "MS_ReplaceNode()" respectively. The final version of the network-table can be used to generate new cytoscape files using the function "MS_ToCytoscape()".

Usage

```
MetaboSignal_NetworkCytoscape(network_table, organism_code, organism_name,
                               source_genes,target_metabolites, mode = "SP",
                               type = "first", interaction_type = NULL,
                               distance_th = Inf, collapse_genes = FALSE,
                               names = TRUE, export_cytoscape = TRUE,
                               file_name = "Cytoscape")
```

Arguments

network_table	two-column matrix where each row represents an edge between two nodes. See function "MetaboSignal_matrix ()".
organism_code	character vector containing the KEGG code for the organism of interest. For example the KEGG code for the rat is "rno". See the function "MS_FindKEGG()".
organism_name	character vector containing the common name of the organism of interest (e.g. "rat", "mouse", "human", "zebrafish") or taxonomy id. For more details, check: http://docs.mygene.info/en/latest/doc/data.html#species . This argument is only required when source_genes are gene symbols.
source_genes	character vector containing the genes from which the shortest paths will be calculated. All input genes need to have the same ID format. Possible ID formats are: entrez IDs, official gene symbols, or gene nodes of the network (i.e. KEGG orthology IDs or KEGG gene IDs). The latter option allows reducing the time required to compute this function. Entrez IDs or gene symbols can be transformed into KEGG IDs using the function "MS_GetKEGG_GeneID()".
target_metabolites	character vector containing the KEGG IDs of the metabolites to which the shortest paths will be calculated. Compound KEGG IDs can be obtained using the function "MS_FindKEGG()".

mode	character constant indicating whether a directed or an undirected network will be considered. "all" indicates that all the edges of the network will be considered as undirected. "out" indicates that all the edges of the network will be considered as directed. "SP" indicates that all network will be considered as directed except the edges linked to target metabolite, which will be considered as undirected. The difference between the "out" and the "SP" options, is that the latter aids reaching target metabolites that are substrates of irreversible reactions. By default, mode = "SP".
type	character constant indicating whether all shortest paths or a single shortest path will be considered when there are several shortest paths between a source_gene and a target_metabolite. If type = "all", all shortest paths will be considered. If type = "first" a single path will be considered. If type = "bw" the path with the highest betweenness score will be considered. The betweenness score is calculated as the average betweenness of the gene nodes of the path. Note that using type = "bw" increases the time required to compute this function. By default, type = "first".
interaction_type	optional three-column matrix generated using the function "MS_interactionType()".
distance_th	establishes a shortest path length threshold. Only shortest paths with length below this threshold will be included in the network. By default, distance_th = Inf.
collapse_genes	logical scalar indicating whether KEGG gene IDs will be transformed into orthology IDs. Since several gene isoforms are associated with the same orthology ID, this options leads to a dramatic decrease in the dimensionality of the network. This argument is ignored if the gene nodes of the network_table already represent orthology IDs. By default, collapse_genes = FALSE.
names	logical scalar indicating whether the metabolite or gene KEGG IDs will be transformed into common metabolite names or gene symbols. Reaction IDs remain unchanged. By default, names = TRUE.
export_cytoscape	logical scalar indicating whether network and attribute cytoscape files will be generated and exported. By default, export_cytoscape = TRUE.
file_name	character vector that allows customizing the name of the exported files. By default, file_name = "Cytoscape".

Value

A matrix where each row represents an edge between two nodes. By default, the function also generates a network file ("CytoscapeNetwork.txt") and two attribute files ("CytoscapeAttributesType.txt", "CytoscapeAttributesTarget.txt"), which can be imported into cytoscape to visualize the network.

Note

The network-table generated with this function can be also visualized in R using the igraph package. The network-table can be transformed into an igraph object using the function "graph.data.frame()" from igraph.

References

Csardi, G. & Nepusz, T. (2006). The igraph software package for complex network research. *Inter-Journal, Complex Systems*, 1695.

Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N. & Ideker, B.S.T. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Research*, 13, 2498-2504.

Examples

```
data(MetaboSignal_table)

# Shortest-path subnetwork from Foxo1 (84482), Ldha (24533) to alpha D-glucose
#("cpd:C00267") and lactate ("cpd:C00186"). Different source_gene formats are valid:

# 1) Source_genes as network IDs (in this case orthology IDs): fastest option.
# To get gene KEGG IDs use "MS_GetKEGG_GeneID( )", as shown below:

MS_GetKEGG_GeneID(c("foxo1", "ldha"), organism_code = "rno", organism_name = "rat")

subnet_KEGG <- MetaboSignal_NetworkCytoscape(MetaboSignal_table, organism_code="rno",
                                             source_genes = c("K07201", "K00016"),
                                             target_metabolites = c("cpd:C00267",
                                                                    "cpd:C00186"),
                                             names = FALSE)

# 2) Source_genes as entrez IDs

subnet_Entrez <- MetaboSignal_NetworkCytoscape(MetaboSignal_table, organism_code="rno",
                                                source_genes = c("84482", "24533"),
                                                target_metabolites = c("cpd:C00267",
                                                                       "cpd:C00186"),
                                                names = FALSE)

# 3) Source_genes as symbols

subnet_Symbol <- MetaboSignal_NetworkCytoscape(MetaboSignal_table,
                                                organism_code="rno", organism_name="rat",
                                                source_genes = c("foxo1", "ldha"),
                                                target_metabolites = c("cpd:C00267",
                                                                       "cpd:C00186"),
                                                names = FALSE)
```

MetaboSignal_table *Example of MetaboSignal network-table filtered by tissue expression*

Description

This network-table was generated using two metabo_paths ("rno00010", "rno00562") and two signaling_paths ("rno04910", "rno04151"). Signaling genes were filtered by adipose tissue. Notice that due to KEGG/hpar updates, this network might be different to the one generated when running the vignette.

Usage

```
data(MetaboSignal_table)
```

Format

Matrix

Value

Matrix

 MetaboSignal_tableUnfiltered

Example of MetaboSignal network-table unfiltered by tissue expression

Description

This network-table was generated using two metabo_paths ("rno00010", "rno00562") and two signaling_paths ("rno04910", "rno04151"). Signaling genes were not filtered by tissue expression. Notice that due to updates in KEGG, this network might be different to the one generated when running the vignette.

Usage

```
data(MetaboSignal_tableUnfiltered)
```

Format

Matrix

Value

Matrix

 MS_ChangeNames

Transform KEGG IDs into common names

Description

This function allows transforming KEGG IDs of genes or metabolites into their corresponding common names (for metabolites) or symbols (for genes).

Usage

```
MS_ChangeNames(nodes, organism_code)
```

Arguments

nodes character vector or matrix containing the KEGG IDs of either metabolites, genes (organism-specific or orthology), or reactions.

organism_code character vector containing the KEGG code for the organism of interest. For example the KEGG code for the rat is "rno". See the function "MS_FindKEGG()". This argument is ignored when nodes are metabolites.

Value

A character string or a matrix containing the common metabolite names or gene symbols corresponding to the input KEGG IDs. Reaction IDs remain unchanged.

References

<http://www.kegg.jp/kegg/docs/keggapi.html>

Examples

```
MS_ChangeNames("K01082", organism_code = "rno")
```

```
MS_ChangeNames(c("rno:84482", "K01084", "cpd:C00267"), "rno")
```

MS_FilterNetwork

Filter network based on distances or betweenness

Description

This function allows reducing the dimensionality of a network, by removing nodes that do not meet the established distance and/or node betweenness criteria.

Usage

```
MS_FilterNetwork(network_table, mode = "all", type, target_node, distance_th, bw_th)
```

Arguments

- | | |
|---------------|--|
| network_table | two-column matrix where each row represents an edge between two nodes. See function "MetaboSignal_matrix ()". |
| mode | character constant indicating whether a directed ("out") or undirected ("all") network will be considered. By default, mode = "all". |
| type | character constant used to establish the criteria for filtering the network. "bw" indicates that edges (i.e. rows of the network_table) containing at least one node with betweenness below bw_th will be neglected. "distance" indicates edges containing at least one node with shortest path length to the target_node above distance_th will be neglected. "all" indicates that edges containing at least one node with either betweenness below bw_th or distance above distance_th, will be neglected. |
| target_node | character vector containing the ID of the node to which the distances will be calculated. |
| distance_th | numeric value corresponding to the distance threshold. Nodes with shortest path length to the target_node above this threshold will be removed from the network-table. |
| bw_th | numeric value corresponding to the normalized-betweenness threshold. Nodes with betweenness below this threshold will be removed from the network-table. See also "MS_NodeBW()". |

Value

A two-column matrix where each row represents an edge between two nodes.

References

Csardi, G. & Nepusz, T. (2006). The igraph software package for complex network research. *Inter-Journal, Complex Systems*, 1695.

Examples

```
data(MetaboSignal_table)
# Remove edges containing nodes with distance to D-glucose ("cpd:C00031") > 2

network_filtered1 <- MS_FilterNetwork(MetaboSignal_table, type = "distance",
                                     target_node = "cpd:C00031",
                                     distance_th = 2)

# Remove edges containing nodes with distance to D-glucose ("cpd:C00031") > 2 or
# normalized-betweenness < 0.00005

network_filtered2 <- MS_FilterNetwork(MetaboSignal_table, type = "all",
                                     target_node = "cpd:C00031",
                                     distance_th = 2, bw_th = 0.00005)

# Note below that network_filtered1 has one edge more than network_filtered2. This is
# because "cpd:C00031" has betweenness = 0, and therefore it is removed in network_filtered2:

setdiff(as.vector(network_filtered1), as.vector(network_filtered2))
```

MS_FindKEGG

Get KEGG IDs for compounds, organisms or pathways

Description

This function returns a list of entries corresponding to one of the following KEGG databases: "compound", "organism", "pathway". It can also find entries with matching query keywords in a given database.

Usage

```
MS_FindKEGG(KEGG_database, match = NULL, organism_code)
```

Arguments

KEGG_database	character vector containing the name of the KEGG database of interest: "compound", "organism", "pathway".
match	character vector containing one or more elements (i.e. key words) to be matched as compound names.
organism_code	character vector containing the KEGG code for the organism of interest. For example the KEGG code for the rat is "rno". This argument is only required for KEGG_database = "pathway".

Value

By default, a matrix where each row contains the KEGG entries of the database of interest. When using the option "match" a list is returned, each list element containing information of matched entries.

Examples

```
MS_FindKEGG(KEGG_database = "compound", match = "acetoacetic acid")

MS_FindKEGG(KEGG_database = "organism", match = c("rattus", "human"))

MS_FindKEGG(KEGG_database = "pathway", match = c("glycol", "insulin signal", "akt"),
            organism_code = "rno")
```

MS_FindMappedNodes *Map gene IDs or metabolite IDs onto the network*

Description

This function can be used to find out if a set of genes or metabolites of interest can be mapped onto the MetaboSignal_network.

Usage

```
MS_FindMappedNodes(nodes, network_table, organism_code, organism_name,
                   orthology = TRUE)
```

Arguments

nodes	character vector containing the IDs of the genes or the metabolites to be mapped onto the network. All IDs need to correspond to the same molecular entity (i.e. gene or metabolite). For metabolites, KEGG IDs are required (see function "MS_FindKEGG()"). For genes, entrez IDs or official symbols can be used, but note that all genes need to be in the same ID format (i.e. entrez or symbols). It is preferable to use entrez IDs rather than gene symbols, since some gene symbols are not unique.
network_table	two-column matrix where each row represents an edge between two nodes. See function "MetaboSignal_matrix()".
organism_code	character vector containing the KEGG code for the organism of interest. For example the KEGG code for the rat is "rno". See the function "MS_FindKEGG()". This argument is not required to map metabolites.
organism_name	character vector containing the common name of the organism of interest (e.g. "rat", "mouse", "human", "zebrafish") or taxonomy id. For more details, check: http://docs.mygene.info/en/latest/doc/data.html#species . This argument is not required to map metabolites.
orthology	logical scalar indicating whether the gene nodes of the network_table represent KEGG orthology IDs (FALSE) or specific KEGG gene IDs (TRUE). By default, orthology = TRUE.

Value

A list reporting which genes or metabolites can or cannot be mapped onto the network.

References

Carlson, M. org.Hs.eg.db: Genome wide annotation for Human.R package version 3.2.3.
 Mark, A., Thompson, R. and Wu, C. (2014) mygene: Access MyGene.Info_ services.R package version 1.6.0.
<http://www.kegg.jp/kegg/docs/keggapi.html>

Examples

```
data(MetaboSignal_table)

# Map D-glucose ("cpd:C00031") and taurine ("cpd:C00245") onto the network

MS_FindMappedNodes(nodes = c("cpd:C00031","cpd:C00245"), MetaboSignal_table,
                    orthology = TRUE)

# Map entrez IDs onto the network

MS_FindMappedNodes(nodes = c("303565", "24267", "11114"), MetaboSignal_table,
                    organism_code = "rno", organism_name = "rat", orthology = TRUE)

# Map gene symbols onto the network

MS_FindMappedNodes(nodes = c("G6pc3","Comt"), MetaboSignal_table, organism_code = "rno",
                    organism_name = "rat",orthology = TRUE)
```

MS_GetKEGG_GeneID *Transform entrez IDs or gene symbols into KEGG IDs*

Description

This function allows transforming entrez gene IDs or official gene symbols into KEGG IDs (orthology IDs or organism-specific gene IDs). The transformed KEGG IDs can be stored and used as `source_genes` in the functions `"MetaboSignal_distances()"` or `"MetaboSignal_NetworkCytoscape()"`. This strategy allows reducing the time required to compute these functions.

Usage

```
MS_GetKEGG_GeneID(genes, organism_code, organism_name, output = "vector",
                  orthology = TRUE)
```

Arguments

`genes` character vector containing the entrez IDs or official symbols of the genes of interest. All genes need to be in the same ID format (i.e. entrez or symbols). It is preferable to use entrez IDs rather than gene symbols, since some gene symbols are not unique.

organism_code	character vector containing the KEGG code for the organism of interest. For example the KEGG code for the rat is "rno". See the function "MS_FindKEGG()".
organism_name	character vector containing the common name of the organism of interest (e.g. "rat", "mouse", "human", "zebrafish") or taxonomy id. For more details, check: http://docs.mygene.info/en/latest/doc/data.html#species . This argument is only required when gene symbols are used.
output	character constant indicating whether the function will return a vector containing mapped and transformed KEGG IDs (output = "vector"), or a matrix containing both mapped entrez IDs or gene symbols and their corresponding KEGG IDs (output = "matrix"). By default, output = "vector".
orthology	logical scalar indicating whether the gene IDs will be transformed into orthology IDs or into organism-specific gene IDs. By default, orthology = TRUE.

Value

A character vector containing mapped and transformed KEGG IDs or a matrix containing both mapped entrez IDs or gene symbols and their corresponding KEGG IDs.

References

- Carlson M.(2016). org.Hs.eg.db: Genome wide annotation for Human, Bioconductor, v3.2.3.
 Mark A, et al.(2016). mygene: Access MyGene.Info_ services. Bioconductor, v1.8.0.
<http://www.kegg.jp/kegg/docs/keggapi.html>

Examples

```
# Transform gene symbol Hoga1 (293949) into rat-specific KEGG ID
MS_GetKEGG_GeneID(genes = "Hoga1", organism_code = "rno", organism_name = "rat",
                  orthology = FALSE)

MS_GetKEGG_GeneID(genes = "Hoga1", "rno", "rat", output = "matrix", orthology = FALSE)

# Transform entrez ID 293949 into orthology KEGG ID
MS_GetKEGG_GeneID(genes = "293949", organism_code = "rno", output = "matrix")
```

MS_GetShortestpaths *Calculate shortest paths*

Description

This function calculates the shortest path(s) between any two reachable nodes of a network-table.

Usage

```
MS_GetShortestpaths(network_table, source_node, target_node, mode = "out",
                    type = "first", network = FALSE, interaction_type = NULL)
```

Arguments

network_table	two-column matrix where each row represents and edge between two nodes. See function "MetaboSignal_matrix ()".
source_node	character vector containing the KEGG ID of the node from which the shortest paths will be calculated.
target_node	character vector containing the KEGG ID of node to which the shortest path will be calculated.
mode	character constant indicating whether a directed or an undirected network will be considered. "all" indicates that all the edges of the network will be considered as undirected. "out" indicates that all the edges of the network will be considered as directed. "SP" indicates that all network will be considered as directed except the edges linked to target metabolite, which will be considered as undirected. The difference between the "out" and "SP" options, is that the latter aids reaching target metabolites that are substrate of irreversible reactions. By default, mode = "out".
type	indicates whether all shortest paths or a single shortest path will be considered when there are several shortest paths between the source_node and the target_node. If type = "all", all shortest paths will be considered. If type = "first" a single path will be considered. If type = "bw" the path with the highest betweenness score will be considered. The betweenness score is calculated as the average betweenness of the gene nodes of the path. Using type = "bw" increases the time required to compute this function. By default, type = "first".
network	logical constant indicating the output of the function. If network = TRUE, the shortest path(s) will be represented as a two-column matrix where each row represents and egde between two nodes. If network = FALSE, the shortest path(s) will be represented as a vector (single path), or as a matrix (multiple paths), where each row represents a shortest path. By default, network = FALSE.
interaction_type	optional three-column matrix generated using the function "MS_interactionType()". This argument is ignored when network = FALSE.

Value

By default, a vector or a matrix where each row contains a shortest path from the source_node to the target_node. If network = TRUE, the function returns matrix where each row represents an edge between two nodes. KEGG IDs can be transformed into common names using the function "MS_ChangeNames()".

References

G. Csardi and T. Nepusz (2015). igraph package, The Comprehensive R Archive Network, v1.0.1.

Examples

```
data(MetaboSignal_table)

# Shortest path from HK ("K00844") to a-D-Glucose ("cpd:C00267")

path1 <- MS_GetShortestpaths(MetaboSignal_table, "K00844", "cpd:C00267", mode = "SP")
path2 <- MS_GetShortestpaths(MetaboSignal_table, "K00844", "cpd:C00267", mode = "out")

# Shortest paths from G6PC ("K01084") to pyruvate ("cpd:C00022")
```



```
path3 <- MS_GetShortestpaths(MetaboSignal_table, "K01084", "cpd:C00022", type = "all")
path4 <- MS_GetShortestpaths(MetaboSignal_table, "K01084", "cpd:C00022", type = "bw")

# Shortest paths from G6PC ("K01084") to pyruvate ("cpd:C00022") as network

path3 <- MS_GetShortestpaths(MetaboSignal_table, "K01084", "cpd:C00022", type = "all",
                             network = TRUE)
```

MS_interactionType *Get interaction subtype*

Description

This function generates a 3 column-matrix indicating the interaction subtype (e.g. activation, binding) between node pairs involved in the signaling pathways of interest. Each row of the matrix represents an edge between 2 nodes. The first column contains the in-nodes, the second column contains the out-nodes, and the third column contains the interaction subtype.

Usage

```
MS_interactionType(paths, expand_genes = FALSE)
```

Arguments

paths	character vector containing the KEGG IDs for the signaling pathways of interest (organism-specific). For example, the KEGG ID for the pathway "insulin signaling pathway" in the rat is "rno04910".
expand_genes	logical scalar indicating whether the gene nodes will represent orthology IDs (FALSE) or organism-specific gene IDs (TRUE). By default, expand_genes = FALSE.

Value

A three-column matrix where each row represents an edge between two nodes.

References

Zhang, J.D. & Wiemann, S. (2009). KEGGgraph: a graph approach to KEGG PATHWAY in R and Bioconductor. *Bioinformatics*, 25, 1470-1471.

<http://www.kegg.jp/kegg/docs/keggapi.html>

Examples

```
# Get interaction-type matrix of nodes involved in "rno04910" and "rno04151"
interaction_type <- MS_interactionType (paths = c("rno04910", "rno04151"))
```

MS_NodeBW	<i>Get distribution of node betweenness</i>
-----------	---

Description

This function calculates the betweenness of each node of the network.

Usage

```
MS_NodeBW(network_table, mode = "all", normalized = TRUE)
```

Arguments

network_table	two-column matrix where each row represents an edge between two nodes. See function "MetaboSignal_matrix ()".
mode	character constant indicating whether a directed ("out") or undirected ("all") network will be considered. By default, mode = "all".
normalized	logical scalar indicating whether to normalize the betweenness scores. If TRUE, normalized betweenness scores will be returned. If FALSE, raw betweenness scores will be returned. By default, normalized = TRUE.

Value

A numeric vector containing the betweenness of each node of the network. The function also produces a histogram showing the distribution of node betweenness.

References

Csardi, G. & Nepusz, T. (2006). The igraph software package for complex network research. *Inter-Journal, Complex Systems*, 1695.

Examples

```
data(MetaboSignal_table)
MS_NodeBW(MetaboSignal_table)
```

MS_RemoveNode	<i>Remove undesired nodes from the network</i>
---------------	--

Description

This function allows removing undesired nodes of the network-table.

Usage

```
MS_RemoveNode(nodes, network_table)
```

Arguments

nodes character vector containing the node IDs to be removed.
network_table two-column matrix where each row represents an edge between two nodes. See function "MetaboSignal_matrix ()".

Value

A two-column matrix corresponding to the input network-table without the undesired nodes.

Examples

```
data(MetaboSignal_table)

# Remove glucose nodes

GlucoseRemoved <- MS_RemoveNode(nodes = c("cpd:C00267", "cpd:C00221", "cpd:C00031"),
                                MetaboSignal_table)
```

MS_ReplaceNode	<i>Replace nodes of the network</i>
----------------	-------------------------------------

Description

This function allows replacing node IDs of a network-table. It can be used to cluster the IDs of chemical isomers (e.g. alpha-D-glucose ("cpd:C00267"), D-glucose ("cpd:C00031"), and beta-D-glucose ("cpd:C00021")) into a single ID.

Usage

```
MS_ReplaceNode(node1, node2, network_table)
```

Arguments

node1 character vector containing the node IDs to be replaced.
node2 character vector containing the ID that will be used as a replacement.
network_table two-column matrix where each row represents an edge between two nodes. See function "MetaboSignal_matrix ()".

Value

A two-column matrix corresponding to the input network-table with replaced nodes.

Examples

```
data(MetaboSignal_table)

# Cluster D-glucose isomers ("cpd:C00267", "cpd:C00221", "cpd:C00031")

GlucoseClustered <- MS_ReplaceNode(node1 = c("cpd:C00267", "cpd:C00221"),
                                   node2 = "cpd:C00031", MetaboSignal_table)
```

MS_ToCytoscape

*Export network in cytoscape format***Description**

The function exports a network file ("CytoscapeNetwork.txt") and two attribute files ("CytoscapeAttributesType.txt", "CytoscapeAttributesTarget.txt"), which can be imported into cytoscape to visualize the network. The first attribute file allows customizing the nodes of the network based on the molecular entity they represent: metabolites, metabolic-genes, or signaling-genes. The second attribute file allows discriminating a set of nodes of interest ("target") from any other node ("untarget") of the network.

Usage

```
MS_ToCytoscape (network_table, organism_code, names = TRUE, target_nodes = NULL,
                interaction_type, file_name = "Cytoscape")
```

Arguments

network_table	two-column matrix where each row represents and edge between two nodes. See function "MetaboSignal_matrix ()".
organism_code	character vector containing the KEGG code for the organism of interest. For example the KEGG code for the rat is "rno". See the function "MS_FindKEGG()".
names	logical scalar indicating whether the metabolite or gene KEGG IDs will be transformed into common metabolite names or gene symbols. Reaction IDs remain unchanged. By default, names = TRUE.
target_nodes	character vector containing the IDs of the target nodes to be discriminated from the other nodes of the network. This argument is optional.
interaction_type	optional three-column matrix generated using the function "MS_interactionType()".
file_name	character vector that allows customizing the name of the exported files. By default, the file_name = "Cytoscape".

Value

A data frame where each row represents an edge between two nodes. The function also generates and exports a network file ("CytoscapeNetwork.txt") and two attribute files ("CytoscapeAttributesType.txt", "CytoscapeAttributesTarget.txt"), which can be imported into cytoscape to visualize the network.

References

Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N. & Ideker, B.S.T. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Research*, 13, 2498-2504.

Examples

```
Glucolysis <- MetaboSignal_matrix(metabo_paths = "mmu00010", organism_name = "mouse")
MS_ToCytoscape(Glucolysis, organism_code = "mmu")
```

neglected_genes	<i>Genes removed during tissue-expression filtering</i>
-----------------	---

Description

This vector contains the IDs of the genes that were removed from the MetaboSignal_table due to adipose-tissue filtering in the example of the vignette. Notice that due to updates in KEGG/hpar, this vector might be different to the one generated when running the vignette.

Usage

```
data(neglected_genes)
```

Format

Vector

Value

Vector

rno_pathways	<i>Examples of metabolic and signaling rat KEGG pathways</i>
--------------	--

Description

This list contains examples of metabolic (first element of the list) and signaling (second element of the list) rat KEGG pathways.

Usage

```
rno_pathways
```

Format

List

Value

List

subnetwork

Shortest-path subnetwork

Description

This network-table corresponds to the bw-ranked shortest-path subnetwork reported in the example of the vignette. Notice that due to KEGG/hpar updates, this network might be different than the one generated when running the vignette.

Usage

```
data(subnetwork)
```

Format

Matrix

Value

Matrix

Index

directionality_reactions, [2](#)

hpaNormalTissue, [3](#)

MetaboSignal_distances, [3](#)

MetaboSignal_matrix, [5](#)

MetaboSignal_NetworkCytoscape, [7](#)

MetaboSignal_table, [9](#)

MetaboSignal_tableUnfiltered, [10](#)

MS_ChangeNames, [10](#)

MS_FilterNetwork, [11](#)

MS_FindKEGG, [12](#)

MS_FindMappedNodes, [13](#)

MS_GetKEGG_GeneID, [14](#)

MS_GetShortestpaths, [15](#)

MS_interactionType, [17](#)

MS_NodeBW, [18](#)

MS_RemoveNode, [18](#)

MS_ReplaceNode, [19](#)

MS_ToCytoscape, [20](#)

neglected_genes, [21](#)

rno_pathways, [21](#)

subnetwork, [22](#)