Package ‘PROFANCY’

February 19, 2015

Version 1.0

Title The package can prioritize candidate disease metabolites based on global functional relationships between metabolites in the context of metabolic pathways

Author Qianlan Yao, Desi Shang, Chunquan Li

Maintainer Qianlan Yao <yaoqianlan@yahoo.com>

Description The package can prioritize the candidate disease metabolites based on the assumption that functionally related metabolites tend to associate with the same or similar diseases in the context of metabolic pathway. The PROFANCY package (1) prioritizes the disease metabolites from global functional similarity and local modularity of the metabolic network; (2) allows users to select default metabolites or input their interested metabolites as seed nodes or candidate nodes (3) can prioritize the candidate metabolites in KEGG or EHMN metabolic network.

Depends R (>= 2.15.2), Matrix, lattice, igraph

Collate getTopDiseaseMetaboliteList.R

LazyData Yes

License GPL (>= 2)

biocViews Disease Metabolites, Network, RandomWalk;

NeedsCompilation no

Repository CRAN

Date/Publication 2013-07-12 08:58:53

R topics documented:

- PROFANCY-package ................................................................. 2
- envData ............................................................................. 2
- getCandidates .................................................................... 3
- getProvidedDiseaseName ...................................................... 4
- getSeed ............................................................................... 5
- getTopDiseaseMetabolites .................................................... 6
- initializeData ....................................................................... 9

Index 10
### PROFANCY-package

**The PROFANCY package**

**Description**

PROFANCY is a R platform to prioritize candidate disease metabolites by PROFANCY method which extend random walk analysis in the metabolic network with functional pathway nodes.

**Introduction**

The main goal of PROFANCY is to prioritize candidate disease metabolites by PROFANCY method. This system provides the rank of candidate diseases metabolites which could be provided by users or this system.

**Author(s)**

Qianlan Yao <yaoqianlan@yahoo.com> Desi Shang <sds_46@163.com> Chunquan Li <lcqbio@aliyun.com>

---

### envData

**The variables in the environment variable envData of the system**

**Description**

The variables in the environment variable envData of the system.

**Format**

An environment variable

**Details**

The environment variable includes the variable DiseaseInfList, EHMMAddPathInfNetwork, KEGGAddPathInfNetwork, MetaboliteInf, ProstateCandidates, ProstateSeeds, etc.

**Author(s)**

Qianlan Yao <yaoqianlan@yahoo.com> Desi Shang <sds_46@163.com> Chunquan Li <lcqbio@aliyun.com>

**See Also**

initializeData
**getCandidates**

*Get candidate metabolites*

---

**Description**

Get candidate metabolites to be prioritized. If candidateDefault is TRUE, this system provided default candidate metabolites which is the metabolites in metabolites network except seed metabolites. Otherwise, users could input candidate metabolites. Then the candidate metabolites are the inputting metabolites which are in the metabolites network.

**Usage**

```r
getCandidates(diseaseName, candidates, network, candidateDefault, seed, seedDefault)
```

**Arguments**

- `diseaseName`: A character of the name of the disease users want to study.
- `network`: A character presents which type of metabolites network should be chose. Users could choose KEGG or EHMN network. The default value is "KEGG" network.
- `seed`: A character vector. The seed metabolites are the known disease metabolites which are used in PROFANCY method. If users have set the seedDefault parameter TRUE thus selected diseaseName provided by system, users do not need to set this parameter. Otherwise, users should input seed metabolites.
- `candidates`: A character vector. If users set the candidateDefault parameter FALSE, users should input candidates. If users do not set this parameter, the candidate metabolites are all metabolites in the network except seed metabolites.
- `seedDefault`: a logical value (TRUE or FALSE). If users set TRUE and have selected the diseaseName provided by system, the seed metabolites are default known disease metabolites. Otherwise, users should input seed metabolites.
- `candidateDefault`: a logical value (TRUE or FALSE). If TRUE, the metabolites in network except seed metabolites will be prioritized. Otherwise, users should input candidate metabolites.

**Details**

Note if the seedDefault is TRUE, users should not set the seed parameter and the default seeds would be used in this condition. If users want to input seeds, users should set the seedDefault FALSE. If the candidateDefault is TRUE, users should not set the candidate parameter, the default candidate metabolites would be used in this condition. If users want to input their own candidate, candidateDefault should be FALSE.

**Value**

A character vector.
getProvidedDiseaseName

Author(s)
Qianlan Yao <yaoqianlan@yahoo.com> Desi Shang <sds_46@163.com> Chunquan Li <lcqbio@aliyun.com>

See Also
getTopDiseaseMetabolites

Examples

```r
## Not run:  Get disease candidates

## Example 1: Users have chose disease name provided by system (Prostate cancer),
## network is set to "EHMN". The candidates and seed metabolites are default.
Candidates1 <- getCandidates(diseaseName = "Prostate cancer", network = "EHMN",
                            seedDefault = TRUE, candidateDefault = TRUE)
print(Candidates1[1:5])

## Example 2: The disease name is input by users. The seeds are provided by users.
## The network is set to "KEGG". The candidates are provided by users.
path1 <- paste(system.file(package = "PROFANCY"), "localdata/ProstateCandidates.txt", sep = "")
candidateExample <- read.table(path1)
candidateExample <- candidateExample[[1]]
path2 <- paste(system.file(package = "PROFANCY"), "localdata/ProstateSeeds.txt", sep = "")
seedExample <- read.table(path2)
seedExample <- seedExample[[1]]
Candidates2 <- getCandidates(diseaseName = "Prostate cancer", network = "KEGG",
                            seed = seedExample,
                            seedDefault = FALSE, candidates = candidateExample, candidateDefault = FALSE)
print(Candidates2[1:5])

## End(Not run)
```

getProvidedDiseaseName

Get all disease names provided by PROFANCY package

Description
Get all disease names provided by PROFANCY package.

Usage

getProvidedDiseaseName()

Value
A character vector.
getSeed

Author(s)
Qianlan Yao <yaoqianlan@yahoo.com> Desi Shang <sds_46@163.com> Chunquan Li <lcqbio@aliyun.com>

See Also
getTopDiseaseMetabolites

Examples

```r
## Not run:
getSeed(diseasename = "common");
getSeed(diseasename = "rare");
getSeed(diseasename = "unknown");
write.table(result, file = "result.txt")
```

getSeed

Get seed metabolites

Description
Get seed metabolites (known disease metabolites). If seedDefault is TRUE, this system provided
default seed metabolites. Otherwise, users should input the seed metabolites and the seed metabo-
lites are inputting metabolite which are in the metabolites network.

Usage

getSeed(diseasename, network, seed, seedDefault)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>diseasename</td>
<td>A character of the name of the disease users want to study.</td>
</tr>
<tr>
<td>network</td>
<td>A character. Which type of metabolites network should be chose. Users could choose KEGG or EHMN network. The default value is &quot;KEGG&quot; network.</td>
</tr>
<tr>
<td>seed</td>
<td>A character vector. The seed metabolites (the known disease metabolites are used in random walk analysis on the network. If users have set the seedDefault parameter TURE and selected diseasename provided by system, users do not need to set this parameter. Otherwise, users should input seed metabolites.</td>
</tr>
<tr>
<td>seedDefault</td>
<td>a logical value(TRUE or FALSE). If users set TRUE and have selected the diseaseName provided by system, the seed metabolites are default known disease metabolites. Otherwise, users should input seed metabolites.</td>
</tr>
</tbody>
</table>
getTopDiseaseMetabolites

Details

Note if the seedDefault is TRUE, you should not set the seed parameter, the default seeds would be used in this condition. If users want to input seeds, users should set the seedDefault FALSE.

Value

A character vector.

Author(s)

Qianlan Yao <yaoqianlan@yahoo.com> Desi Shang <sds_46@163.com> Chunquan Li <lcqbio@aliyun.com>

See Also

getTopDiseaseMetabolites

Examples

## Not run:

Get disease seed metabolites

### Example 1: Users have chose disease name provided by system (Prostate cancer),
### network is set "EHMN". The seeds are provided by default.
seed1<-getSeed(diseaseName="Prostate cancer",network="EHMN",seedDefault=TRUE)
print(seed1[1:5])

### Example 2: The disease name is provided by users. The seeds are provided by users.
### The network is set "KEGG".
diseaseName<-"prostate cancer" ##the disease name provided by users.
path2<-paste(system.file(package="PROFANCY"),"/localdata/ProstateSeeds.txt",sep="")
seedExample<-read.table(path2)
seedExample<-seedExample[[1]]
seed2<-getSeed(diseaseName="Prostate cancer",network="EHMN",seed=seedExample,seedDefault=FALSE)
print(seed2[1:5])

## End(Not run)

getTopDiseaseMetabolites

Get the disease Risk metabolites

Description

This system could get the disease risk metabolites (top ranked candidate metabolites).
getTopDiseaseMetabolites

Usage

getTopDiseaseMetabolites(diseaseName=NULL, network=c("KEGG"), seed=NULL, candidates=NULL, seedDefault=TRUE, candidateDefault=TRUE, showTop=30)

Arguments

diseaseName: A character of the name of the disease users want to study.

network: A character. Which type of metabolites network should be chose. Users could choose KEGG or EHMN network. The default value is "KEGG" network.

seed: A character vector. The seed metabolites are the known disease metabolites are used in random walk analysis on the network. If users have set the seedDefault parameter TRUE and selected diseaseName provided by system, users do not need to set this parameter. Otherwise, users should input seed metabolites.

candidates: A character vector. If users set the candidateDefault parameter FALSE, users should input candidates. If users do not set this parameter, the candidate metabolites are all metabolites in the network except seed metabolites.

seedDefault: a logical value(TRUE or FALSE). If users set TRUE and have selected the diseaseName provided by system, the seed metabolites are default known disease metabolites. Otherwise, users should input seed metabolites.

candidateDefault: a logical value(TRUE or FALSE). If TRUE, the metabolites in network except seed metabolites will be prioritized. Otherwise, users should input candidate metabolites.

showTop: An integer. The number of top ranked candidate metabolites users want to show.

Details

The function getTopDiseaseMetabolites could get the top ranked candidate metabolites by prioritization of the disease candidate metabolites using known disease metabolites as seed node to perform random walk on metabolite network with virtual pathway nodes.

Note that the diseaseName we provided can be obtained by getProvidedDiseaseName. If the seedDefault is TRUE, users should not set the seed parameter and the default seeds would be used in this condition. If users want to input seeds, users should set the seedDefault FALSE. If the candidateDefault is TRUE, users should not set the candidates parameter, the default candidate metabolites would be used in this condition. If users want to input their own candidates, candidateDefault should be FALSE.

Value

A dataframe.

Author(s)

Qianlan Yao <yaoqianlan@yahoo.com> Desi Shang <sds_46@163.com> Chunquan Li <lcqbio@aliyun.com>

See Also

getProvidedDiseaseName
Examples

## Not run:

```
### Prioritize candidate metabolites

## Example 1: Users have chose disease name provided by system and network is set "EHMNN".
The seeds and candidates are default.

ProvidedDiseaseName <- getProvidedDiseaseName() # get all disease name provided.

diseaseName <- ProvidedDiseaseName[17]  # choose the disease you want to study.
ex1 <- getTopDiseaseMetabolites(diseaseName = diseaseName, network = "EHMNN",
    seedDefault = TRUE, showTop = 30, candidateDefault = TRUE)
print(ex1[1:5,])
write.table(ex1, "example1.txt", quote = FALSE, row.names = FALSE, sep = "\t")
```

## Example 2: The disease name is provided by users. The seeds and candidates are provided by users. The network is set "KEGG".

diseaseName <- "prostate cancer"  # the disease name provided by users.

path1 <- paste(system.file(package = "PROFANCY"), "/localdata/ProstateCandidates.txt", sep = "")
candidateExample <- read.table(path1)
candidateExample <- candidateExample[1]

path2 <- paste(system.file(package = "PROFANCY"), "/localdata/ProstateSeeds.txt", sep = "")
seedExample <- read.table(path2)
seedExample <- seedExample[1]

ex2 <- getTopDiseaseMetabolites(diseaseName = diseaseName, network = "KEGG",
    seed = seedExample,
    candidates = candidateExample, seedDefault = FALSE, showTop = 30,
    candidateDefault = FALSE)
print(ex2[1:5,])
write.table(ex2, "example2.txt", quote = FALSE, row.names = FALSE, sep = "\t")

## Example 3: Disease name chose in our provided. The seeds are provided by default. Network is set "EHMNN".

path1 <- paste(system.file(package = "PROFANCY"), "/localdata/ProstateCandidates.txt", sep = "")
candidateExample <- read.table(path1)
candidateExample <- candidateExample[1]

ex3 <- getTopDiseaseMetabolites(diseaseName = "Prostate cancer", network = "EHMNN",
    candidates = candidateExample, seedDefault = TRUE, showTop = 30,
    candidateDefault = FALSE)
print(ex3[1:5,])
write.table(ex3, "example3.txt", quote = FALSE, row.names = FALSE, sep = "\t")

## Example 4: Disease name chose in our provided. The seeds are provided by users. The candidates are provided by default. Network is set "KEGG".

path2 <- paste(system.file(package = "PROFANCY"), "/localdata/ProstateSeeds.txt", sep = "")
seedExample <- read.table(path2)
seedExample <- seedExample[1]

ex4 <- getTopDiseaseMetabolites(diseaseName = "Prostate cancer", network = "KEGG",
    seed = seedExample, seedDefault = FALSE, showTop = 30, candidateDefault = TRUE)
print(ex4[1:5,])
write.table(ex4, "example4.txt", quote = FALSE, row.names = FALSE, sep = "\t")
initializeData  initialize environment envData

Description
initialize environment envData.

Usage
initializeData()

Details
The environment variable envData save many information. We can use the function `ls` to see the variable and use `ls(envData)` to see information in it, which include DiseaseInfList, EHMNAAddPathInfNetwork, KEGGAddPathInfNetwork, MetaboliteInf, ProstateCandidates, ProstateSeeds etc. We can use the function `get` to obtain one of them.

Author(s)
Qianlan Yao <yaoqianlan@yahoo.com> Desi Shang <sds_46@163.com> Chunquan Li <lcqbio@aliyun.com>

Examples
initializeData()

## see information in environment variable envData
ls(envData)

## obtain one variable in environment variable envData
KEGGAddPathInfNetwork<-get("KEGGAddPathInfNetwork",envir=envData)
Index

*Topic file
  envData, 2
  getCandidates, 3
  getProvidedDiseaseName, 4
  getSeed, 5
  getTopDiseaseMetabolites, 6
  initializeData, 9

DiseaseInfList (envData), 2

EHMNAAddPathInfNetwork (envData), 2
  envData, 2
  get, 9
  getCandidates, 3
  getProvidedDiseaseName, 4, 7
  getSeed, 5
  getTopDiseaseMetabolites, 4–6, 6
  initializeData, 2, 9

KEGGAddPathInfNetwork (envData), 2

ls, 9

MetaboliteInf (envData), 2

PROFANCY (PROFANCY-package), 2
PROFANCY-package, 2
ProstateCandidates (envData), 2
ProstateSeeds (envData), 2