Package 'cooccurNet'

January 27, 2017

Type Package

Title Co-Occurrence Network

Version 0.1.6

Depends R (>= 3.2.0)

Imports seqinr, igraph, bigmemory, Matrix, foreach, parallel, pryr, knitr, doParallel

Author
Yuanqiang Zou <jerrytsou2001@gmail.com>, Yousong Peng <pys2013@hnu.edu.cn> and Taijiao Jiang <taijiao@moon.ibp.ac.cn>

Maintainer Yuanqiang Zou <jerrytsou2001@gmail.com>

Description
Read and preprocess fasta format data, and construct the co-occurrence network for downstream analyses. This R package is to construct the co-occurrence network with the algorithm developed by Du (2008) <DOI:10.1101/gr.6969007>. It could be used to transform the data with high-dimension, such as DNA or protein sequence, into co-occurrence networks. Co-occurrence network could not only capture the co-variation pattern between variables, such as the positions in DNA or protein sequences, but also reflect the relationship between samples. Although it is originally used in DNA and protein sequences, it could be also used to other kinds of data, such as RNA, SNP, etc.

License GPL-3

LazyData TRUE

VignetteBuilder knitr

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2017-01-27 10:38:24

R topics documented:

---

changeLog *changeLog*

Description

Get the most recent n lines

Usage

```
changeLog(n=20)
```

Arguments

n	numeric, 20 by default, the number of lines will be shown up.

Details

changeLog

Value

list, the most recent n lines

Note

Once you have installed cooccurNet, the change-log can also be viewed from the R prompt.

Examples

```
logs = changeLog(n=20)
```

cooccurNet *cooccurNet: Co-Occurrence network computation for R*

Description

Read and preprocess fasta format data, and construct the co-occurrence network for downstream analyses. This R package is to construct the co-occurrence network with the algorithm developed by Du (2008) <DOI:10.1101/gr.6969007>. It could be used to transform the data with high-dimension, such as DNA or protein sequence, into co-occurrence networks. Co-occurrence network could not only capture the co-variation pattern between variables, such as the positions in DNA or protein sequences, but also reflect the relationship between samples. Although it is originally used in DNA and protein sequences, it could be also used to other kinds of data, such as RNA, SNP, etc.

Details

Index of functions/methods (grouped in a friendly way):

1. readseq(dataFile="", dataType="protein", debug=FALSE)

2. pprocess(data=list(), conservativeFilter=0.95, memory=NULL, debug=FALSE)

3. gencooccur(data=list(), cooccurFilter=NULL, networkFile='cooccurNetwork', module=FALSE, moduleFile='cooccurNetworkModule', property=FALSE, propertyFile='cooccurNetworkProperty', siteCo=FALSE, siteCoFile='siteCooccurr', sampleTimes=100, debug=FALSE, parallel=FALSE)

4. coocnet(dataFile="", dataType="protein", conservativeFilter=0.95, memory=NULL, cooccurFilter=NULL, networkFile='cooccurNetwork', module=FALSE,moduleFile='cooccurNetworkModule', property=FALSE, propertyFile='cooccurNetworkProperty', siteCo=FALSE, siteCoFile='siteCooccurr', sampleTimes=100, debug=FALSE, parallel=FALSE)

5. siteco(dataFile="", dataType="protein", conservativeFilter=0.95, memory=NULL, cooccurFilter=NULL, siteCoFile='siteCooccurr', sampleTimes=100, debug=FALSE, parallel=FALSE)

6. toigraph(networkFile="", networkNames=c())

7. getexample(dataType)

8. getexample_forRCOS(dataType)

9. changeLog(n=20)

Note

Arguments:

1. dataFile
type: character
description: a FASTA data file name with full path.

2. dataType
type: character
description: 'protein' by default, the type of data will be processed. It could be 'DNA', 'RNA', 'protein', 'SNP' or 'other'.

3. conservativeFilter
type: numeric
description: a number in the range of 0~1, 0.95 by default. It's used to filter the highly conservative columns which the ratio of some residue is larger than the conservationFilter.

4. cooccurFilter
type: numeric
description: a number in the range of 0~1. It determines whether two columns are perfect co-occurrence. In default, for the data type of protein, it is set to be 0.9, while for the other data types, it is set to be 1.

5. networkFile
type: character
description: 'cooccurNetwork' be default. It is a file name with full path for storing the co-occurrence network for each row.

6. module
type: logical
description: FALSE by default, to check whether the modules in each network of the networkFile would be calculated.

7. moduleFile
type: character
description: 'cooccurNetworkModule' by default. It is a file name with full path for storing the modules for co-occurrence network.

8. property
type: logical
description: FALSE by default, to check whether the properties for each network of the networkFile, including the network diameter, connectivity, ConnectionEffcient and so on, would be calculated.

9. propertyFile
type: character
description: 'cooccurNetworkProperty' by default. It is a file name with full path storing the properties for each network of the networkFile.

10. siteCo
type: logical
description: FALSE by default, to check whether the residue co-occurence file would be calculated.

11. siteCoFile
type: character
description: 'siteCooccurr' by default. It is a file name with full path for storing the RCOS between all pairs of columns, and the related p-values.

12. sampleTimes
type: numeric
description: a integer of permutations in the simulation when calculating the p-values. It should be greater than 100.

13.debug
type: logical
description: FALSE by default, indicates whether the debug message will be displayed or not.

14.memory
type: character

description: the type of matrix, NULL by default. It could be 'memory' or 'sparse'. If it's set to be 'memory', all data would be manipulated in the RAM by using normal matrix and package 'bigmemory'. If it's set to be 'sparse', the package "Matrix" would be used to manipulate massive matrices in memory and initialize huge sparse matrix, which could significantly reduce the RAM consumed. In default, it is set to be NULL, so that the system would determine automatically whether all data is manipulated in the RAM or not, according to the size of data inputted and the RAM available for R.

15.parallel
type: logical
description: FALSE by default. It only supports Unix/Mac (not Windows) system.

Output files:

1. networkFile
filePath = data$networkFile
description: A co-occurrence network (networkFile) was given for each sequence, which could be easily transformed into the format of igraph by the function toigraph().

2. moduleFile
filePath = data$moduleFile
description: The inherent modules(moduleFile) in each co-occurrence network were given for each sequence.

3. propertyFile
filePath = data$propertyFile
description: Some basic network attributes(propertyFile) such as connectivity and clustering coefficient for each network were given.

4. siteCoFile
filePath = data$siteCoFile
description: The extent of co-occurrence between residues(siteCoFile), defined as residue co-occurrence score (RCOS), were given for all pairs of residues.

For more details, please see the 'Vignette: Extending cooccurNet' by using the following command.
vignette("Extending-cooccurNet",package="cooccurNet")

Author(s)

Yuanqiang Zou, Yousong Peng, and Taijiao Jiang

Maintainers: Yuanqiang Zou <jerrytsou2001@gmail.com>

References

Du, X., Wang, Z., Wu, A., Song, L., Cao, Y., Hang, H., & Jiang, T. (2008). Networks of genomic co-occurrence capture characteristics of human influenza A (H3N2) evolution. Genome research, 18(1), 178-187. doi:10.1101/gr.6969007

Examples

```
#example of get example data
dataFile=getexample(dataType="protein")

#example of get file paths for all the files available for testing RCOS.
```

```
#dataFiles = getexample_forRCOS()

#example of readseq()
#read sequences from the sample fasta file
#data = readseq(dataFile=dataFile, dataType="protein")

#example of pprocess()
#preprocess the sequence dataFile
#data_process = pprocess(data=data,conservativeFilter=0.95)

#example of gencooccur()
#generate co-occurrence network and save it into the 'networkFile'
#cooccurNetwork  = gencooccur(data=data_process, cooccurFilter=0.9, networkFile='cooccurNetwork')
#check the 'networkFile' path
#print(cooccurNetwork$networkFile)

#example of coocnet()
#also, you can generate the co-occurrence network by one command
#cooccurNetwork  = coocnet(dataFile=getexample(dataType="protein"), dataType="protein")
#check the 'networkFile' path
#print(cooccurNetwork$networkFile)

#example of siteco()
#you can generate the co-occurrence network siteCoFile by one command
#this command will take long time to calculate the p-value.
#pairwiseCooccur = siteco(dataFile=getexample(dataType="protein"), dataType="protein")
#check the 'siteCoFile' path
#print(pairwiseCooccur$siteCoFile)

#example of toigraph()
#you can transform a network file to the igraph.data.frame
#cooccurNetwork  = coocnet(dataFile=getexample(dataType="protein"),dataType="protein")
#get igraph data frame by specifying the network name
#network_igraph = toigraph(networkFile=cooccurNetwork$networkFile, networkName=c("EPI823725"))
#Plot the network (The package "igraph" must be installed and loaded firstly)
#read the names of network
#networkName = cooccurNetwork$xnames
#Transform all cooccurrence network into the igraph format
#Network_igraph = toigraph(networkFile=cooccurNetwork$networkFile, networkNames=networkName)

#example of changelog()
#logs = changeLog(n=20)
```

---

coocnet *coocnet*

Description

Read and preprocess data, and construct the co-occurrence network in one step.

Usage

```
coocnet(dataFile = "", dataType = "protein", conservativeFilter = 0.95,
  cooccurFilter = NULL, networkFile = "cooccurNetwork", module = FALSE,
  moduleFile = "cooccurNetworkModule", property = FALSE,
  propertyFile = "cooccurNetworkProperty", siteCo = FALSE,
  siteCoFile = "siteCooccurr", sampleTimes = 100, debug = FALSE,
  parallel = FALSE, memory = NULL)
```

Arguments

dataFile	character, a FASTA data file name with full path.
dataType	character, 'protein' by default, the type of data will be processed. It could be 'DNA', 'RNA', 'protein', 'SNP' or 'other'.
conservativeFilter	numeric, a number in the range of 0~1, 0.95 by default. It's used to filter the highly conservative columns which the ratio of some residue is larger than the conservationFilter.
cooccurFilter	numeric, a number in the range of 0~1. It determines whether two columns are perfect co-occurrence. In default, for the data type of protein, it is set to be 0.9, while for the other data types, it is set to be 1.
networkFile	character, 'cooccurNetwork' be default. It is a file name with full path for storing the co-occurrence network for each row.
module	logic, FALSE by default, to check whether the modules in each network of the networkFile would be calculated.
moduleFile	character, 'cooccurNetworkModule' by default. It is a file name with full path for storing the modules for co-occurrence network.
property	logic, FALSE by default, to check whether the properties for each network of the networkFile, including the network diameter, connectivity, ConnectionEffcient and so on, would be calculated.
propertyFile	character, 'cooccurNetworkProperty' by default. It is a file name with full path storing the properties for each network of the networkFile.
siteCo	logic, FALSE by default, to check whether the residue co-occurence file would be calculated.
siteCoFile	character, 'siteCooccurr' by default. It is a file name with full path for storing the RCOS between all pairs of columns, and the related p-values.
sampleTimes	numeric, an integer of permutations in the simulation when calculating the p-values. It should be greater than 100.
debug	logic, FALSE by default, indicates whether the debug message will be displayed or not.
parallel	logic, FALSE by default. It only supports Unix/Mac (not Windows) system.
memory	character, the type of matrix, NULL by default. It could be 'memory' or 'sparse'. If it's set to be 'memory', all data would be manipulated in the RAM by using normal matrix and package 'bigmemory'. If it's set to be 'sparse', the package "Matrix" would be used to manipulate massive matrices in memory and initialize

huge sparse matrix, which could significantly reduce the RAM consumed. In default, it is set to be NULL, so that the system would determine automatically whether all data is manipulated in the RAM or not, according to the size of data inputted and the RAM available for R.

Value

list, all the output file paths are attributed in it.
The attribute "networkFile" stores the co-occurrence network for each row;
The attribute "moduleFile" is optional. When the module is set to be TRUE, it would be output. It stores the modules for co-occurrence network;
The attribute "propertyFile" is optional. When the property is set to be TRUE, it would be output. It stores the properties for co-occurrence network;
The attribute "siteCoFile" is optional. When the property is set to be TRUE, it would be output. It stores all the pairwise siteCos between columns.

References

Du, X., Wang, Z., Wu, A., Song, L., Cao, Y., Hang, H., & Jiang, T. (2008). Networks of genomic co-occurrence capture characteristics of human influenza A (H3N2) evolution. Genome research, 18(1), 178-187. doi:10.1101/gr.6969007

Examples

```
cooccurNetwork  = coocnet(dataFile=getexample(dataType="protein"), dataType="protein")
```

`gencooccur` *gencooccur*

Description

Construct the co-occurrence network

Usage

```
gencooccur(data = list(), cooccurFilter = NULL,
  networkFile = "cooccurNetwork", module = FALSE,
  moduleFile = "cooccurNetworkModule", property = FALSE,
  propertyFile = "cooccurNetworkProperty", siteCo = FALSE,
  siteCoFile = "siteCooccurr", sampleTimes = 100, debug = FALSE,
  parallel = FALSE)
```

Arguments

`data`	list, returns from the function "pprocess()"
`cooccurFilter`	numeric, a number in the range of 0~1. It determines whether two columns are perfect co-occurrence. In default, for the data type of protein, it is set to be 0.9, while for the other data types, it is set to be 1.

networkFile	character, 'cooccurNetwork' be default. It is a file name with full path for storing the co-occurrence network for each row.
module	logic, FALSE by default, to check whether the modules in each network of the networkFile would be calculated.
moduleFile	character, 'cooccurNetworkModule' by default. It is a file name with full path for storing the modules for co-occurrence network.
property	logic, FALSE by default, to check whether the properties for each network of the networkFile, including the network diameter, connectivity, ConnectionEffcient and so on, would be calculated.
propertyFile	character, 'cooccurNetworkProperty' by default. It is a file name with full path storing the properties for each network of the networkFile.
siteCo	logic, FALSE by default, to check whether the residue co-occurence file would be calculated.
siteCoFile	character, 'siteCooccurr' by default. It is a file name with full path for storing the RCOS between all pairs of columns, and the related p-values.
sampleTimes	numeric, an integer of permutations in the simulation when calculating the p-values. It should be greater than 100.
debug	logic, FALSE by default, indicates whether the debug message will be displayed or not.
parallel	logic, FALSE by default. It only supports Unix/Mac (not Windows) system.

Value

list, all the output file paths are attributed in it.
The attribute "networkFile" stores the co-occurrence network for each row;
The attribute "moduleFile" is optional. When the module is set to be TRUE, it would be output. It stores the modules for co-occurrence network;
The attribute "propertyFile" is optional. When the property is set to be TRUE, it would be output. It stores the properties for co-occurrence network;
The attribute "siteCoFile" is optional. When the property is set to be TRUE, it would be output. It stores all the pairwise siteCos between columns.

References

Du, X., Wang, Z., Wu, A., Song, L., Cao, Y., Hang, H., & Jiang, T. (2008). Networks of genomic co-occurrence capture characteristics of human influenza A (H3N2) evolution. Genome research, 18(1), 178-187. doi:10.1101/gr.6969007

Examples

```
data = readseq(dataFile=getexample(dataType="protein"),dataType="protein")
data_process = pprocess(data=data)
#cooccurNetwork  = gencooccur(data=data_process)
```

getexample *getexample*

Description

Get the example data

Usage

```
getexample(dataType)
```

Arguments

dataType	character, 'protein' by default. It could be 'DNA', 'RNA', 'protein', 'SNP' or 'other'.

Details

getexample

Value

character, the file path of the example data

Note

Both the dataset "DNA" and "protein" are sampled from the Hemagglutinin sequences of human influenza H3N2 viruses, while the dataset "SNP" are simulated.

Examples

```
dataFile = getexample(dataType='protein')
```

getexample_forRCOS *getexample_forRCOS*

Description

Get the test data for testing RCOS method

Usage

```
getexample_forRCOS()
```

Details

getexample_forRCOS

Value

character, the files available for testing RCOS.

Note

File paths for all the files available for testing RCOS. Currently, there is only one file "HA1protein_humanH3N2", the sequences within which are derived from the database of Influenza Virus Resource.

Examples

```
dataFile = getexample_forRCOS()
```

`pprocess` *pprocess*

Description

Filter the conservative columns (defined as the conservative score greater than the "conservativeFilter")

Usage

```
pprocess(data=list(),  conservativeFilter=0.95, debug=FALSE, memory=NULL)
```

Arguments

`data`	list, returns from the function "readseq()".
`conservativeFilter`	numeric, a number in the range of 0~1, 0.95 by default. It's used to filter the highly conservative columns which the ratio of some residue is larger than the conservationFilter.
`debug`	logic, FALSE by default, indicates whether the debug message will be displayed or not.
`memory`	character, the type of matrix, NULL by default. It could be 'memory' or 'sparse'. If it's set to be 'memory', all data would be manipulated in the RAM by using normal matrix and package 'bigmemory'. If it's set to be 'sparse', the package "Matrix" would be used to manipulate massive matrices in memory and initialize huge sparse matrix, which could significantly reduce the RAM consumed. In default, it is set to be NULL, so that the system would determine automatically whether all data is manipulated in the RAM or not, according to the size of data inputted and the RAM available for R.

Value

list, contains the original data matrix, frequency matrix and other informations.

References

Du, X., Wang, Z., Wu, A., Song, L., Cao, Y., Hang, H., & Jiang, T. (2008). Networks of genomic co-occurrence capture characteristics of human influenza A (H3N2) evolution. Genome research, 18(1), 178-187. doi:10.1101/gr.6969007

Examples

```
data = readseq(dataFile=getexample(dataType="protein"),dataType="protein")
data_process = pprocess(data=data,conservativeFilter=0.95)
```

`readseq` *readseq*

Description

Reading a sequence file in fasta format

Usage

```
readseq(dataFile="", dataType="protein", debug=FALSE)
```

Arguments

`dataFile`	character, a FASTA data file name with full path.
`dataType`	character, 'protein' by default, the type of data will be processed. It could be 'DNA', 'RNA', 'protein', 'SNP' or 'other'.
`debug`	logic, FALSE by default, to indicate whether the debug message will be displayed or not.

Value

list,contains the original data matrix and some other information.

Examples

```
data = readseq(dataFile=getexample(dataType="protein"), dataType="protein")
```

siteco *siteco*

Description

Read and preprocess data, and calculate the pairwise site co-occurrence in one step

Usage

```
siteco(dataFile = "", dataType = "protein", conservativeFilter = 0.95,
  cooccurFilter = NULL, siteCoFile = "siteCooccurr", sampleTimes = 100,
  debug = FALSE, parallel = FALSE, memory = NULL)
```

Arguments

dataFile	character, a FASTA data file name with full path.
dataType	character, 'protein' by default, the type of data will be processed. It could be 'DNA', 'RNA', 'protein', 'SNP' or 'other'.
conservativeFilter	numeric, a number in the range of 0~1, 0.95 by default. It's used to filter the highly conservative columns which the ratio of some residue is larger than the conservationFilter.
cooccurFilter	numeric, a number in the range of 0~1. It determines whether two columns are perfect co-occurrence. In default, for the data type of protein, it is set to be 0.9, while for the other data types, it is set to be 1.
siteCoFile	character, 'siteCooccurr' by default. It is a file name with full path for storing the RCOS between all pairs of columns, and the related p-values.
sampleTimes	numeric, an integer of permutations in the simulation when calculating the p-values. It should be greater than 100.
debug	logic, FALSE by default, indicates whether the debug message will be displayed or not.
parallel	logic, FALSE by default. It only supports Unix/Mac (not Windows) system.
memory	character, the type of matrix, NULL by default. It could be 'memory' or 'sparse'. If it's set to be 'memory', all data would be manipulated in the RAM by using normal matrix and package 'bigmemory'. If it's set to be 'sparse', the package "Matrix" would be used to manipulate massive matrices in memory and initialize huge sparse matrix, which could significantly reduce the RAM consumed. In default, it is set to be NULL, so that the system would determine automatically whether all data is manipulated in the RAM or not, according to the size of data inputted and the RAM available for R.

Value

list, the output file path of 'sitecoFile' is attributed in it. The file stores all the pairwise siteCos between columns.

References

Du, X., Wang, Z., Wu, A., Song, L., Cao, Y., Hang, H., & Jiang, T. (2008). Networks of genomic co-occurrence capture characteristics of human influenza A (H3N2) evolution. Genome research, 18(1), 178-187. doi:10.1101/gr.6969007

Examples

```
#pairwiseCooccur = siteco(dataFile=getexample(dataType="protein"), dataType="protein")
```

`toigraph` *toigraph*

Description

transform a network file to the igraph.data.frame type by specifying a network file name and a network name

Usage

```
toigraph(networkFile="", networkNames=c())
```

Arguments

`networkFile`	character, a file name with full path for storing the co-occurrence network for each row and generated from cooc() or gencooccur().
`networkNames`	character, a vector of network names

Value

a list of igraph::graph.data.frame object ordered by the input network names.

Examples

```
#cooccurNetwork  = cooc(dataFile=getexample(dataType="protein"),dataType="protein")
#network_igraph = toigraph(networkFile=cooccurNetwork$networkFile, networkName=c("EPI823725")
```

Index