

Package ‘dmGWAS’

February 19, 2015

Type Package

Title dmGWAS package

Version 3.0

Date 2015-02-16

Author Quan Wang and Peilin Jia

Maintainer

Quan Wang <quan.wang@vanderbilt.edu> and Peilin Jia <peilin.jia@vanderbilt.edu>

Depends R (>= 2.9.0), igraph

Description

dmGWAS_3.0 implement our upgraded algorithm EW_dmGWAS, which searches for dense modules in a node- and edge-weighted PPI network.

License GPL (>= 2)

URL <http://bioinfo.mc.vanderbilt.edu/dmGWAS>

R topics documented:

chooseModule	1
dmGWAS_3.0	2
dms	3
estimate_lambda	4
generate_edge_weight	5
generate_graph	5

Index 7

chooseModule	<i>Choosing dense modules</i>
--------------	-------------------------------

Description

This function will choose the top modules specified by users for downstream analyses. Sub-network can also be plotted if plot=TRUE.

Usage

```
chooseModule(res.list, top = 0.01, plot = FALSE)
```

Arguments

<code>res.list</code>	The results from function <i>dms</i>
<code>top</code>	Either a percentage (<1) or an integer (>=1)
<code>plot</code>	To plot the sub-network or not

Details

The parameter *top* could be either a percentage (<1) or an integer (>=1). For example, if *top* is 0.01, the top 1% modules will be chosen; if *top* is 50, the top 50 modules will be chosen.

Value

A list contains both the chosen modules and the sub-network constructed from the chosen modules.

References

Jia P, Zheng S, Long J, Zheng W, and Zhao Z (2011) dmGWAS: dense module searching for genome-wide association studies in protein-protein interaction networks. *Bioinformatics* 27(1):95-102

Wang Q, Yu H, Zhao Z, and Jia P. EW_dmGWAS: Edge-weighted dense module search for genome-wide association studies and gene expression profiles. Manuscript in preparation

See Also

[dms](#)

Examples

```
# res.list <- dms(network, geneweight, expr1, expr2)
# chooseModule(res.list, top = 0.01, plot = FALSE)
```

dmGWAS_3.0

dmGWAS_3.0 searches for dense modules in a node- and edge-weighted PPI network.

Description

dmGWAS_3.0 is an upgraded version of dmGWAS. It constructs a node- and edge-weighted PPI network, performs dense module searching, generates simulation data from random networks, normalizes module scores using simulation data, removes unqualified modules, and orders resultant modules according to their significance.

Details

Package:	dmGWAS
Type:	Package
Version:	3.0
Date:	2015-02-16
License:	GPL (>= 2)

This package takes three types of data as input: a list of genes with association p-values, gene expression profiling in both case and control samples, and a human PPI network. `generate_graph` constructs a node- and edge-weighted PPI network. `dms` performs dense module search upon the node- and edge-weighted PPI network.

Author(s)

Quan Wang <quan.wang@vanderbilt.edu> and Peilin Jia <peilin.jia@vanderbilt.edu>

References

Jia P, Zheng S, Long J, Zheng W, and Zhao Z (2011) dmGWAS: dense module searching for genome-wide association studies in protein-protein interaction networks. *Bioinformatics* 27(1):95-102
 Wang Q, Yu H, Zhao Z, and Jia P. EW_dmGWAS: Edge-weighted dense module search for genome-wide association studies and gene expression profiles. Manuscript in preparation

Examples

```
#res.list <- dms(network, geneweight, expr1, expr2, r=0.1)
#chooseModule(res.list, top=0.01, plot=T)
```

dms	<i>Dense module search function</i>
-----	-------------------------------------

Description

`dms` constructs a node- and edge-weighted PPI network, performs dense module searching, generates simulation data from random networks, normalizes module scores using simulation data, removes un-qualified modules, and orders resultant modules according to their significance.

Usage

```
dms(network, geneweight, expr1, expr2 = NULL, d=1, r = 0.1, lambda = "default")
```

Arguments

network	A data frame containing a symbolic edge list of the PPI network.
geneweight	A data frame containing two columns: the first is unique gene identifier (should be coordinate with the node symbol used in PPI); the second is gene-based p-value derived from GWAS.
expr1	A data frame containing gene expression data from case samples. The first column is gene identifier (should be coordinate with the node symbol used in PPI).
expr2	A data frame containing gene expression data from control samples. The first column should be the same as expr1.
d	An integer used to define the order of neighbour genes to be searched. This parameter is always set up as 1 in dmGWAS_3.0, but could be 1 or 2 in dmGWAS_1.0 and dmGWAS_2.X.
r	A float indicating the cut-off for increment during module expanding process. Greater r will generate smaller module. Default is 0.1.
lambda	A float between 0 and 1 to balance node and edge weights. dmGWAS_NEW will estimate it by default.

Value

A list containing all important data including the node- and edge-weighted network used for searching, resultant dense module list, module score matrix containing S_m and S_n , and randomization data for normalization. A resultant file '*.RData' is also automatically saved in the working folder for future record.

References

Jia P, Zheng S, Long J, Zheng W, and Zhao Z (2011) dmGWAS: dense module searching for genome-wide association studies in protein-protein interaction networks. *Bioinformatics* 27(1):95-102

Wang Q, Yu H, Zhao Z, and Jia P. EW_dmGWAS: Edge-weighted dense module search for genome-wide association studies and gene expression profiles. Manuscript in preparation

Examples

```
#res.list <- dms(network, geneweight, expr1, expr2, r=0.1)
```

estimate_lambda *Estimate the parameter lambda*

Description

Lambda is a parameter to balance node and edge weight when expanding modules. This function will estimate it when it is not specified. This function is generally called by function `dms`. Typically users do not need to call it.

Usage

```
estimate_lambda(G)
```

Arguments

G G is a node- and edge-weighted PPI network, which can be generated from function `generate_graph`.

Value

A float between 0 and 1.

See Also

`generate_graph`

Examples

```
# G <- generate_graph(expr1, expr2, network, geneweight)
# lambda <- estimate_lambda(G)
```

generate_edge_weight
Compute edge weight

Description

`generate_edge_weight` computes differential gene co-expression (i.e., the change of gene co-expression between case and control samples) to infer the edge weight of PPI network. `generate_edge_weight` is generally called by function `generate_graph`. Typically users do not need to call it.

Usage

```
generate_edge_weight(expr1, expr2, network, geneweight)
```

Arguments

<code>expr1</code>	A data frame containing gene expression data from case samples. The first column is gene identifier (should be coordinate with the node symbol used in PPI).
<code>expr2</code>	A data frame containing gene expression data from control samples. The first column should be the same as <code>expr1</code> .
<code>network</code>	A data frame containing a symbolic edge list of the PPI network.
<code>geneweight</code>	A data frame containing two columns: the first is unique gene identifier (should be coordinate with the node symbol used in PPI); the second is gene-based p-value derived from GWAS.

Value

A co-expression matrix is returned.

Examples

```
# edgeweight <- generate_edge_weight(expr1, expr2, network, geneweight)
```

generate_graph *Build background PPI Network*

Description

`generate_graph` constructs a node- and edge-weighted PPI network for function `dms`. This function is generally called by function `dms`. Typically users do not need to call it.

Usage

```
generate_graph(expr1, expr2 , network, geneweight)
```

Arguments

<code>expr1</code>	A data frame containing gene expression data from case samples. The first column is gene identifier (should be coordinate with the node symbol used in PPI).
<code>expr2</code>	A data frame containing gene expression data from control samples. The first column should be the same as <code>expr1</code> .
<code>network</code>	A data frame containing a symbolic edge list of the PPI network.
<code>geneweight</code>	A data frame containing two columns: the first is unique gene identifier (should be coordinate with the node symbol used in PPI); the second is gene-based p-value derived from GWAS.

Value

A node- and edge-weighted PPI network.

Examples

```
# G<- generate_graph (expr1, expr2, network, geneweight)
```

Index

*Topic **package**

dmGWAS_3.0, 2

chooseModule, 1

dmGWAS (*dmGWAS_3.0*), 2

dmGWAS_3.0, 2

dms, 2, 3, 3–5

estimate_lambda, 4

generate_edge_weight, 5, 5

generate_graph, 3, 4, 5, 5