

# Package ‘dhga’

May 8, 2026

**Type** Package

**Title** Differential Hub Gene Analysis

**Version** 0.1

**Date** 2016-08-31

**Author** Samarendra Das <samarendra4849@gmail.com> and Baidya Nath Mandal <mandal.stat@gmail.com>

**Maintainer** Samarendra Das <samarendra4849@gmail.com>

**Depends** R (>= 3.3.1), VennDiagram

**Description** Identification of hub genes in a gene co-expression network from gene expression data. The differential network analysis for two contrasting conditions leads to the identification of various types of hubs like Housekeeping, Unique to stress (Disease) and Unique to control (Normal) hub genes.

**License** GPL (>= 2)

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2016-09-06 08:12:54

## Contents

Adjacency . . . . .	2
DiffHub . . . . .	3
hub.pval.cutoff . . . . .	4
hub.wgs . . . . .	5
HubPlot . . . . .	5
HubStatus . . . . .	6
pvalue.hub . . . . .	7
rice_normal . . . . .	8
rice_salt . . . . .	9
WeightedGeneScore . . . . .	10

<b>Index</b>	<b>11</b>
--------------	-----------

---

Adjacency

*Construction of Gene Co-expression Network*

---

### Description

The function computes the adjacency matrix using the soft threshold parameter to construct a Gene Co-expression Network

### Usage

```
Adjacency(x, beta, threshold)
```

### Arguments

x	x is a data frame of gene expression values where rows represent genes and columns represent samples/time points under a particular condition/stress/trait.
beta	beta is a soft threshold parameter determined from the scale free property of biological networks (Gene Co-expression Network).
threshold	threshold is a scalar representing the threshold value for the adjacency matrix to construct the Gene Co-expression Network.

### Value

The function returns a list of outputs like adjacency matrix, Node-list and Edge-list for constructing a Gene Co-expression Network at certain threshold level

### Author(s)

Samarendra Das and Baidya Nath Mandal

### Examples

```
data(rice_normal)
beta=6
threshold=0.4
Adjacency(rice_normal,6,0.4)
```

---

DiffHub

*Differential Hub status of the genes in a gene co-expression network*

---

### Description

The function returns hub status of each gene in a gene co-expression network under two contrasting conditions

### Usage

```
DiffHub(x,y,m1,m2,s,beta,alpha, plot=TRUE)
```

### Arguments

x	x is a data frame of gene expression values where rows represent genes and columns represent samples/time point under stress condition.
y	y is a data frame of gene expression values where rows represent genes and columns represent samples/time points under control condition.
m1	m1 is a scalar representing sample size and is less than or equal to number of columns in x.
m2	m2 is a scalar representing sample size and is less than or equal to number of columns in y.
s	s is a scalar representing number of times each of the m samples will be resampled.
beta	beta is a soft threshold parameter determined from the scale free property of biological networks (GCN).
alpha	alpha is a scalar representing statistical level of significance. Default is alpha=0.0001
plot	plot is a character representing whether the hubplot can be drawn (TRUE) or not (FALSE).

### Value

The function returns a list with two components. First component returns the list of genes along with their hub status. Second component gives a table containing number of genes under different categories of hubs viz. housekeeping, unique to stress and unique to normal. The function also returns a venn plot of hub genes and unique hub genes under two conditions.

### Author(s)

Samarendra Das and Baidya Nath Mandal

### Examples

```
data(rice_salt)
data(rice_normal)
DiffHub(rice_salt,rice_normal,18,18,80, 6, alpha=0.0001, plot=TRUE)
```

---

hub.pval.cutoff	<i>Identification of hub genes in a Gene Co-expression Network based on gene connection significance values</i>
-----------------	---

---

### Description

The function returns list of top ranked hub genes under a particular condition/trait

### Usage

```
hub.pval.cutoff(x, beta, m, s, n)
```

### Arguments

x	x is a data frame of gene expression values where rows represent genes and columns represent samples/time point under a particular condition/trait.
beta	beta is a soft threshold parameter determined from the scale free property of biological networks (Gene Co-expression Network).
m	m is a scalar representing sample size and is less than or equal to number of columns in x.
s	s is a scalar representing number of times each of the m samples will be resampled.
n	n is a scalar representing number of top ranked hub genes to be selected under that particular condition/trait

### Value

The function returns a vector containing list of top ranked hub genes for that particular condition/trait

### Author(s)

Samarendra Das and Baidya Nath Mandal

### Examples

```
data(rice_salt)
x=as.data.frame(rice_salt)
beta=6
m=18
s=80
n=20
hub.pval.cutoff(x, beta, m, s, n)
```

---

hub.wgs	<i>Identification of hub genes in a Gene Co-expression Network based on Weighted Gene Score</i>
---------	---

---

**Description**

The function returns the list of hub genes in a Gene Co-expression Network based on Weighted Gene Score without considering statistical significance values.

**Usage**

```
hub.wgs(x, beta, n)
```

**Arguments**

x	x is a data frame of gene expression values where rows represent genes and columns represent samples/time.
beta	beta is a soft threshold parameter determined from the scale free property of biological networks (Gene Co-expression Network).
n	n is a scalar representing number of genes to be declared as hub in the Gene Co-expression Network based on Weighted Gene Score.

**Value**

The function returns a list of hub genes in the Gene Co-expression Network

**Author(s)**

Samarendra Das and Baidya Nath Mandal

**Examples**

```
data(rice_salt)
hub.wgs(rice_salt, beta=6, n=20)
```

---

HubPlot	<i>Venn plot of hub genes under two conditions namely stress and control</i>
---------	--

---

**Description**

The function produces a venn diagram of number of hub genes and unique hub genes under two conditions

**Usage**

```
HubPlot(pvalue.stress, pvalue.control, alpha)
```

**Arguments**

`pvalue.stress` `pvalue.stress` is a vector of p-values for genes in the GCN under stress condition. `pvalue.stress` can be obtained from the output of `hub.stress` function.

`pvalue.control` `pvalue.control` is a vector of p-values for genes in the GCN under normal or control condition. `pvalue.control` can be obtained from the output of `hub.control` function.

`alpha` `alpha` is a scalar representing statistical level of significance. Default is `alpha=0.0001`

**Value**

The function produces a venn plot of hub genes and unique hub genes under two conditions

**Author(s)**

Samarendra Das and Baidya Nath Mandal

**Examples**

```
data(rice_salt)
pval.stres <- pvalue.hub(rice_salt, beta=6, m=18, s=80, plot=FALSE)
pvalue.stress <- pval.stres[, 2]

data(rice_normal)
pval.control <- pvalue.hub(rice_normal, beta=6, m=18, s=80, plot=FALSE)
pvalue.control <- pval.control[, 2]
HubPlot(pvalue.stress, pvalue.control, alpha=0.0001)
```

---

HubStatus

*Hub status of the genes in a gene co-expression network*

---

**Description**

The function returns hub status of each gene in a gene co-expression network (GCN)

**Usage**

```
HubStatus(pvalue.stress, pvalue.control, alpha)
```

**Arguments**

- `pvalue.stress` `pvalue.stress` is a vector of p-values for genes in the GCN under stress condition. `pvalue.stress` can be obtained from the output of `hub.stress` function.
- `pvalue.control` `pvalue.control` is a vector of p-values for genes in the GCN under normal or control condition. `pvalue.control` can be obtained from the output of `hub.control` function.
- `alpha` `alpha` is a scalar representing statistical level of significance. Default is `alpha=0.0001`

**Value**

The function returns a list with two components. First component returns the list of genes along with their hub status. Second component gives a table containing number of genes under different categories of hubs viz. housekeeping, unique to stress and unique to normal

**Author(s)**

Samarendra Das and Baidya Nath Mandal

**Examples**

```
data(rice_salt)
pval.stres <- pvalue.hub(rice_salt, beta=6, m=18, s=80, plot=FALSE)
p1 <- pval.stres[, 2]

data(rice_normal)
pval.control <- pvalue.hub(rice_normal, beta=6, m=18, s=80, plot=FALSE)
p2 <- pval.control[, 2]
HubStatus(p1,p2,alpha=0.0001)
```

---

pvalue.hub	<i>Computation of gene connection sigificance values in a Gene Co-expression network</i>
------------	--

---

**Description**

The function computes ths statistical significance values of gene connections in a GCN for a particular condition/trait/stress

**Usage**

```
pvalue.hub(x, beta, m, s, plot=TRUE)
```

**Arguments**

x	x is a data frame of gene expression values where rows represent genes and columns represent samples/time point under a particular condition/trait/stress.
beta	beta is a soft threshold parameter determined from the scale free property of biological networks (Gene Co-expression network).
m	m is a scalar representing sample size and is less than or equal to number of columns in x.
s	s is a scalar representing number of times each of the m samples will be resampled.
plot	plot is a character must either take logical value TRUE/FALSE representing whether the plot of the pvalues of gene connections of all genes in Gene Co-expression Network needs to be constructed or not.

**Value**

The function returns a vector of p-values for all the genes in the GCN

**Author(s)**

Samarendra Das and Baidya Nath Mandal

**Examples**

```
data(rice_salt)
x=as.data.frame(rice_salt)
beta=6
m=18
s=80
pvalue.hub(x, beta, m, s, plot=TRUE)
```

---

rice\_normal

*The gene expression data of rice under control or normal condition*

---

**Description**

rice\_normal data has the gene expression values of 200 genes over 20 samples. The row names of the data are the names of the probes.

**Usage**

```
data(rice_normal)
```

**Format**

A data frame with 200 rows and 20 columns.

**Details**

rice\_normal data has the gene expression values of 200 genes identified by Wang et al. (2013). After loading the data, the dataset is available in R in the name of rice\_normal.

**Source**

Wang J, Chen L, Wang Y, Zhang J, Liang Y, Xu D (2013) A Computational systems biology study for understanding salt tolerance mechanism in Rice. PLoS one 8(6): e64929. doi:10.1371/journal.pone.0064929.

**Examples**

```
data(rice_normal)
```

---

rice\_salt

*The gene expression data of rice under salinity stress condition*

---

**Description**

rice\_salt data has the gene expression values of 200 genes over 20 samples. The row names of the data are the names of the probes.

**Usage**

```
data("rice_salt")
```

**Format**

A data frame with 200 rows and 20 columns.

**Details**

rice\_salt data has the gene expression values of 200 genes identified by Wang et al. (2013). After loading the data, the dataset is available in R in the name of rice\_salt object.

**Source**

Wang J, Chen L, Wang Y, Zhang J, Liang Y, Xu D (2013) A Computational systems biology study for understanding salt tolerance mechanism in Rice. PLoS one 8(6): e64929. doi:10.1371/journal.pone.0064929.

**Examples**

```
data(rice_salt)
```

---

WeightedGeneScore	<i>Computation of weighted gene score of genes in Gene Co-expression Network</i>
-------------------	--

---

**Description**

The function computes the weighted gene scores of all genes in a Gene Co-expression Network

**Usage**

```
WeightedGeneScore(x, beta, plot=TRUE)
```

**Arguments**

x	x is a data frame of gene expression values where rows represent genes and columns represent samples/time points under a particular condition/stress/trait.
beta	beta is a soft threshold parameter determined from the scale free property of biological networks (Gene Co-expression Network).
plot	plot is a character must either take logical value TRUE/FALSE representing whether the plot of the gene connection degrees of all genes in Gene Co-expression Network needs to be constructed or not.

**Value**

The function returns a vector of weighted gene scores of genes (nodes) in the Gene Co-expression Network along with the gene connectivity plot.

**Author(s)**

Samarendra Das and Baidya Nath Mandal

**Examples**

```
data(rice_normal)
beta=6
WeightedGeneScore(rice_normal, beta, plot=FALSE)
```

# Index

- \* **Adjacency**
    - Adjacency, 2
  - \* **Edgelist**
    - Adjacency, 2
  - \* **Nodelist**
    - Adjacency, 2
  - \* **Weighted Gene Score**
    - hub.wgs, 5
  - \* **co-expression**
    - Adjacency, 2
    - DiffHub, 3
    - hub.pval.cutoff, 4
    - hub.wgs, 5
    - HubPlot, 5
    - HubStatus, 6
    - pvalue.hub, 7
    - WeightedGeneScore, 10
  - \* **datasets**
    - rice\_normal, 8
    - rice\_salt, 9
  - \* **gene**
    - Adjacency, 2
    - DiffHub, 3
    - hub.pval.cutoff, 4
    - hub.wgs, 5
    - HubPlot, 5
    - HubStatus, 6
    - pvalue.hub, 7
    - WeightedGeneScore, 10
  - \* **hub gene**
    - DiffHub, 3
    - hub.pval.cutoff, 4
    - hub.wgs, 5
    - HubPlot, 5
    - HubStatus, 6
    - pvalue.hub, 7
  - \* **hub status**
    - DiffHub, 3
    - HubStatus, 6
  - \* **level of significance**
    - DiffHub, 3
    - HubPlot, 5
    - HubStatus, 6
  - \* **network**
    - Adjacency, 2
    - DiffHub, 3
    - hub.pval.cutoff, 4
    - hub.wgs, 5
    - HubPlot, 5
    - HubStatus, 6
    - pvalue.hub, 7
    - WeightedGeneScore, 10
  - \* **p value**
    - hub.pval.cutoff, 4
    - pvalue.hub, 7
  - \* **plot**
    - pvalue.hub, 7
    - WeightedGeneScore, 10
  - \* **top rank**
    - hub.pval.cutoff, 4
  - \* **venn plot**
    - HubPlot, 5
  - \* **weighted gene score**
    - WeightedGeneScore, 10
- Adjacency, 2
- DiffHub, 3
- hub.pval.cutoff, 4
- hub.wgs, 5
- HubPlot, 5
- HubStatus, 6
- pvalue.hub, 7
- rice\_normal, 8
- rice\_salt, 9
- WeightedGeneScore, 10