# Package ‘NetGreg’

July 21, 2025

**Type** Package

**Title** Network-Guided Penalized Regression (NetGreg)

**Version** 0.0.2

**Description** A network-guided penalized regression framework that integrates network characteristics from Gaussian graphical models with partial penalization, accounting for both network structure (hubs and non-hubs) and clinical covariates in high-dimensional omics data, including transcriptomics and proteomics. The full methodological details can be found in our recent preprint by Ahn S and Oh EJ (2025) <doi:10.48550/arXiv.2505.22986>.

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**Depends** R (>= 3.5.0)

**Imports** huge, glmnet, dplyr, stats, plsgenomics

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**NeedsCompilation** no

**Author** Seungjun Ahn [cre, aut] (ORCID: <`https://orcid.org/0000-0002-4816-8924`>),
Eun Jeong Oh [aut] (ORCID: <`https://orcid.org/0000-0001-8949-6564`>)

**Maintainer** Seungjun Ahn <`seungjun.ahn@mountsinai.org`>

**Repository** CRAN

**Date/Publication** 2025-06-03 09:30:12 UTC

## Contents

---

`identifyHubs` *identifyHubs*

---

## Description

A function to identify hub nodes (i.e., genes or proteins) from high-dimensional data using network-based criteria.

## Usage

```
identifyHubs(X, delta, tau, ebic.gamma = 0.1)
```

## Arguments

| | |
|---|---|
| `X` | A data matrix of dimension n x p representing samples (rows) by features (columns). |
| `delta` | A numeric value indicating the proportion of nodes to considered as hubs in a network. |
| `tau` | A user-specified cutoff for the number of hubs. |
| `ebic.gamma` | A numeric value specifying the tuning parameter for the extended Bayesian information criterion (eBIC) used in network estimation. |

## Value

A list containing (1) the selected sparse graph structure and model selection results; (2) a data frame of feature names with their associated network characteristics (e.g., degree centrality); and (3) a character vector of top-ranked hub features (e.g., hub genes or proteins).

## Examples

```
library(plsgenomics)
data(Colon) ## Data from plsgenomics R package
X = data.frame(Colon$X[,1:100]) ## The first 100 genes
Z = data.frame(Colon$X[,101:102]) ## Two clinical covariates
colnames(Z) = c("Z1", "Z2")
Y = as.vector(Colon$X[,1000])  ## Continuous outcome variable

## Apply identifyHubs():
preNG = identifyHubs(X=X, delta=0.05, tau=5, ebic.gamma = 0.1)

## Explore preNG results:
## To display the degree centrality for each node,
## sorted from strongest to weakest.
preNG$assoResults
preNG$hubs ## Returns the names of the identified hub nodes.
```

---

`NetworkGuided` *NetworkGuided*

---

## Description

A main function to obtain network-guided penalized regression coefficient estimates.

## Usage

```
NetworkGuided(Y, X, hubs, Z, nfolds = 5)
```

## Arguments

| | |
|---|---|
| `Y` | A continuous outcome variable. |
| `X` | A data matrix of dimension n x p representing samples (rows) by features (columns). |
| `hubs` | A vector of hubs idenfitied through identifyHubs function from our package. |
| `Z` | A matrix of clinical or demographic covariates. |
| `nfolds` | A user-specified numeric value for k-fold cross-validation. |

## Value

A vector of network-guided penalized regression coefficients.

## Examples

```
library(plsgenomics)
data(Colon) ## Data from plsgenomics R package
X = data.frame(Colon$X[,1:100]) ## The first 100 genes
Z = data.frame(Colon$X[,101:102]) ## Two clinical covariates
colnames(Z) = c("Z1", "Z2")
Y = as.vector(Colon$X[,1000])  ## Continuous outcome variable

## Apply identifyHubs():
preNG = identifyHubs(X=X, delta=0.05, tau=5, ebic.gamma = 0.1)

## Explore preNG results:
hubs = preNG$hubs ## Returns the names of the identified hub nodes.

## Use our main NetworkGuided function, to obtain network-guided
## penalized regression coefficient estimates.
NG = NetworkGuided(Y=Y, X=X, hubs=preNG$hubs, Z=Z, nfolds=5)
NG$coef
```

# Index