Package ‘GmicR’

May 29, 2024

Type Package

Title Combines WGCNA and xCell readouts with bayesian network learning to generate a Gene-Module Immune-Cell network (GMIC)

Version 1.18.0

Author Richard Virgen-Slane

Maintainer Richard Virgen-Slane <RVS.BioTools@gmail.com>

Description This package uses bayesian network learning to detect relationships between Gene Modules detected by WGCNA and immune cell signatures defined by xCell. It is a hypothesis generating tool.

License GPL-2 + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

Imports AnnotationDbi, ape, bnlearn, Category, DT, doParallel, foreach, gRbase, GSEABase, gRain, GOstats, org.Hs.eg.db, org.Mm.eg.db, shiny, WGCNA, data.table, grDevices, graphics, reshape2, stats, utils

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

StagedInstall yes

biocViews Software, SystemsBiology, GraphAndNetwork, Network, NetworkInference, GUI, ImmunoOncology, GeneExpression, QualityControl, Bayesian, Clustering

git_url https://git.bioconductor.org/packages/GmicR

git_branch RELEASE_3_19

git_last_commit 58968ed

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19

Date/Publication 2024-05-29
Auto_WGCNA

Carries out WGCNA with default settings or custom settings

Description

Carries out WGCNA with default settings or custom settings

Usage

Auto_WGCNA(
  datExpr,
  colname_correct = TRUE,
  minModuleSize = 10,
  deepSplit = 4,
  networkType = "signed hybrid",
  TOMType = "unsigned",
  corFnc = "bicor",
  mergeCutHeight = 0.25,
  sft_RsquaredCut = 0.85,
  removeFirst = FALSE,
  reassignThreshold = 1e-06,
  maxBlockSize = 25000,
  nThreads = NULL
)

Arguments

datExpr
  Expression data. A matrix (preferred) or data frame in which columns are genes and rows are samples. NAs are allowed, but not too many. See checkMissingData below and details.

colname_correct
  a logical value. If TRUE (default), "." in gene names will be replaced with ".-". This corrects a name change that is induced by R when creating a data.frame. If FALSE, no changes will be made.
minModuleSize minimum module size for module detection. See cutreeDynamic for more details.
depthSplit integer value between 0 and 4. Provides a simplified control over how sensitive module detection should be to module splitting, with 0 least and 4 most sensitive. See cutreeDynamic for more details.
networkType network type. Allowed values are (unique abbreviations of) "unsigned", "signed", "signed hybrid". See adjacency.
TOMType one of "none", "unsigned", "signed". If "none", adjacency will be used for clustering. If "unsigned", the standard TOM will be used (more generally, TOM function will receive the adjacency as input). If "signed", TOM will keep track of the sign of correlations between neighbors.
corFnc the correlation function to be used in adjacency calculation.
mergeCutHeight dendrogram cut height for module merging.
sft_RsquaredCut desired minimum scale free topology fitting index $R^2$. Default is 0.80.
removeFirst should the first bin be removed from the connectivity histogram?
reassignThreshold p-value ratio threshold for reassigning genes between modules. See Details.
maxBlockSize integer giving maximum block size for module detection. Ignored if blocks above is non-NULL. Otherwise, if the number of genes in datExpr exceeds maxBlockSize, genes will be pre-clustered into blocks whose size should not exceed maxBlockSize.
nThreads non-negative integer specifying the number of parallel threads to be used by certain parts of correlation calculations. This option only has an effect on systems on which a POSIX thread library is available (which currently includes Linux and Mac OSX, but excludes Windows). If zero, the number of online processors will be used if it can be determined dynamically, otherwise correlation calculations will use 2 threads.

Value

Returns a lists containing network input parameters used for WGCNA, WGCNA module information, and quality control plots.

Note

This is a wrapper for WGCNA.

See Also

blockwiseModules
adjacency
Examples

```r
sample_dat_dir <- system.file("extdata", "sample_dat.Rdata", package = "GmicR", mustWork = TRUE)
load(sample_dat_dir)
GMIC_Builder <- Auto_WGCNA(sample_dat, mergeCutHeight = 0.35, minModuleSize = 10)
```

---

**Batch_Net**

Generates a subgraph from query nodes

**Description**
Generates a subgraph from query nodes

**Usage**

```r
Batch_Net(bn_output, Node_ids, relationship_type = "nbr")
```

**Arguments**

- `bn_output`: R object output from `bn_tabu_gen` function
- `Node_ids`: vector containing the nodes for subgraph generation.
- `relationship_type`: the relationship to be returned for the specified query nodes. The options are "mb", "nbr", "parents", "children". Default setting is "nbr".

**Value**
a subgraph containing the selected nodes and relationships.

---

**bn_tabu_gen**

Uses tabu search algorithm to learn the structure of discretized data.

**Description**
Uses tabu search algorithm to learn the structure of discretized data.
bn_tabu_gen

Usage

bn_tabu_gen(
  Auto_WGCNA_OUTPUT,
  whitelist = NULL,
  blacklist = NULL,
  score = "bde",
  tabu = 50,
  iss = 10,
  cluster = NULL,
  debug = TRUE,
  bootstraps_replicates = 500
)

Arguments

  Auto_WGCNA_OUTPUT
    an R object generated by Auto_WGCNA and discretized using the Data_Prep
    function.
  whitelist
    a data frame with two columns (optionally labeled "from" and "to"), containing
    a set of arcs to be included in the graph.
  blacklist
    a data frame with two columns (optionally labeled "from" and "to"), containing
    a set of arcs not to be included in the graph.
  score
    character string indicating the score used for structure learning. If "bde" (de-
    fault), prior is set to "uniform". If bds is used, the prior is set to "marginal".
  tabu
    a positive integer number, the length of the tabu list used in the tabu function.
  iss
    the imaginary sample size, used by the Bayesian Dirichlet scores (bde and bds)
    It is also known as “equivalent sample size”. The default value is equal to 10.
  cluster
    an optional cluster object from package parallel.
  debug
    a boolean value. If TRUE a lot of debugging output is printed; otherwise the
    function is completely silent.
  bootstraps_replicates
    an integer for the number of bootstraps_replicates used for structure learning.
    Default value is 500

Value

  The learned bayesian network

See Also

  arc.strength
  hc
  score
Examples

```r
GMIC_Builder_disc_dir <- system.file("extdata", "GMIC_Builder_disc.Rdata", package = "GmicR", mustWork = TRUE)
load(GMIC_Builder_disc_dir)

no_cores <- 1
c <- parallel::makeCluster(no_cores)

GMIC_net <- bn_tabu_gen(GMIC_Builder_disc,
    cluster = c,
    bootstraps_replicates = 50, score = "bds")
parallel::stopCluster(c)
```

Data Prep

Discretizes biological assay data in preparation for bayesian network learning

Description

Discretizes biological assay data in preparation for bayesian network learning

Usage

```r
Data_Prep(
    Auto_WGCNA_OUTPUT = NULL,
    Remove_ME0 = FALSE,
    Numeric_Pheno_scores = NULL,
    xCell_Signatures = NULL,
    ibreaks = 60
)
```

Arguments

- **Auto_WGCNA_OUTPUT**: R object generated from Auto_WGCNA function.
- **Remove_ME0**: a logical value. If FALSE (default), ME0 is not removed. If TRUE the eigengene for module 0 is removed prior to analysis.
- **Numeric_Pheno_scores**: a data.frame with rows indicating sample ID and columns representing additional phenotype data to be included in BN learning. If NULL (default) no data will be included. If provided, the data.frame will be merged with MEs and discretized into three levels.
- **xCell_Signatures**: the name of the text file generated by xCell that contains the cell signature scores. If NULL (default) the only module eigenegnes will be processed. If not NULL and if Auto_WGCNA_OUTPUT is NULL, cell signature scores will be discretized.
- **ibreaks**: an integer that indicates the number of ibreaks used for discretization. The default value is 60.
Gmic_viz

Value

a list containing a data.frame with module eigengenes merged with Xcell signature scores and discretized into three levels: L, M, H. If Auto_WGCNA_OUTPUT is NULL, both scaled and discretized cell signatures will be return.

Note

Please verify that the sample name formatting is consistent between both datasets. Rownames in the module eigengenes data.frame and the column names of xCell signatures scores text file are matched for merging. Only samples that are present in both will be processed!

Examples

```r
file_dir<-system.file("extdata", "IRIS_xCell_sig.txt", package = "GmicR", mustWork = TRUE)
Disc_Xcell_sig<-Data_Prep(xCell_Signatures=file_dir, ibreaks = 10)
Disc_Xcell_sig$disc_data
```

Gmic_viz

Visualized network

Description

Visualized network

Usage

Gmic_viz(Auto_WGCNA_Output, Filter_unconnected_ME = TRUE)

Arguments

Auto_WGCNA_Output

R object with GMIC bayesian network

Filter_unconnected_ME

a logical value. If TRUE, the default, unconnected modules will be removed from the final network. If FALSE, all modules will be shown.

Value

a shiny object for network visualization.

Examples

```r
GMIC_Final_dir<-system.file("extdata", "GMIC_Final.Rdata", package = "GmicR", mustWork = TRUE)
load(GMIC_Final_dir)
if(interactive()){
  Gmic_viz(GMIC_Final)
}
GO_Module_NameR

**Description**

GO enrichment for module names

**Usage**

GO_Module_NameR(Auto_WGCNA_OUTPUT, cutoff_size = NULL)

**Arguments**

- **Auto_WGCNA_OUTPUT**: object returned by GSEAGO_Builder function
- **cutoff_size**: integer indication the GO size cut off. If NULL (default) the term with the smallest Pvalue will be returned.

**Value**

a data.frame listing the GO name for each module. Additionally, a table similar to the output of Query_Prep function is returned, appended with module GO names.

**Examples**

```r
GMIC_Builder_dir<-system.file("extdata", "GMIC_Builder.Rdata", package = "GmicR", mustWork = TRUE)
load(GMIC_Builder_dir)
GMIC_Builder<-GO_Module_NameR(GMIC_Builder, cutoff_size = 100)
```

---

GSEAGO_Builder

**Description**

GO enrichment for module names

**Usage**

GSEAGO_Builder(
    Auto_WGCNA_OUTPUT,
    species = "Homo sapiens",
    no_cores = 4,
    ontology = "BP",
    GO_conditional = FALSE,
    colname_correct = TRUE
)

---
**InverseARCs**

**Arguments**

- **Auto_WGCNA_OUTPUT**
  - Output from `Auto_WGCNA` function.
- **species**
  - Either 'Homo sapiens' (default) or 'Mus musculus'.
- **no_cores**
  - Number of cores to use. Default = 4.
- **ontology**
  - String either 'BP' (Biological Process; default), 'CC' (Cellular Component), or 'MF' (Molecular Function).
- **GO_conditional**
  - A logical indicating whether the calculation should condition on the GO structure. Will not be carried out. If TRUE,
- **colname_correct**
  - A logical value. If TRUE (default), '.' in gene names will be replaced with '-'. This corrects a name change that is induced by R when creating a data.frame. If FALSE, no changes will be made.

**Value**

Lists with gene ontology enrichment analysis, performed using GOstats, for each module.

**Note**

gene names must be official gene symbol

**Examples**

```r
GMIC_Builder_dir<-system.file("extdata", "GMIC_Builder.Rdata", package = "GmicR", mustWork = TRUE)
load(GMIC_Builder_dir)
GMIC_Builder$GSEAGO_Builder_Output<-NULL
Test_GMIC_Builder<-GSEAGO_Builder(GMIC_Builder, no_cores = 1)
summary(Test_GMIC_Builder$GSEAGO_Builder_Output)
```

---

**InverseARCs**

Identifies arcs between nodes with inverse relationships

**Description**

Identifies arcs between nodes with inverse relationships

**Usage**

```r
InverseARCs(Output, threshold = -0.3)
```

**Arguments**

- **Output**
  - A data frame containing the output of `BN_Conditions` function.
- **threshold**
  - Number indicating the maximum slope for defining negative relationships. Default level is -0.3.
Query Prep

Value

arcs with inverse relationships

Examples

GMIC_net_dir<-system.file("extdata", "GMIC_net.Rdata",
package = "GmicR", mustWork = TRUE)
load(GMIC_net_dir)
GMIC_Final<-InverseARCs(GMIC_net, threshold = -0.3)

Query Prep

Description

Query Prep

Usage

Query_Prep(
    Auto_WGCNA_OUTPUT,
    numGenes = 500,
    Find_hubs = FALSE,
    nThreads = NULL,
    calculate_intramodularConnectivity = TRUE
)

Arguments

Auto_WGCNA_OUTPUT  R object generated by Auto_WGCNA function.
numGenes  integer indicating the number of random genes to test for hub gene detection. Default is 500.
Find_hubs  logical value. If TRUE, module hub genes will be returned. If FALSE (default), intramodularConnectivity will be returned without hub gene identification.
nThreads  non-negative integer specifying the number of parallel threads to be used by certain parts of correlation calculations. This option only has an effect on systems on which a POSIX thread library is available (which currently includes Linux and Mac OSX, but excludes Windows). If zero, the number of online processors will be used if it can be determined dynamically, otherwise correlation calculations will use 2 threads.
calculate_intramodularConnectivity  a logical value. If TRUE (default), the intramodularConnectivity will be calculated using the intramodularConnectivity function from WGCNA. If FALSE, a table of modules and genes will be returned without intramodularConnectivity values.
**Value**

a data.frame detailing the gene symbols for each module. Gene intramodularConnectivity may also be returned. If detected, hub genes are annotated.

**Examples**

```r
GMIC_Builder_dir<-system.file("extdata", "GMIC_Builder.Rdata", package = "GmicR", mustWork = TRUE)
load(GMIC_Builder_dir)
GMIC_Builder<-Query_Prep(GMIC_Builder, Find_hubs = TRUE)
head(GMIC_Builder$Query)
```

---

**xCell_loader**

Scales and centers data by sample/row in preparation for discretization

**Description**

Scales and centers data by sample/row in preparation for discretization

**Usage**

```r
xCell_loader(File = NULL)
```

**Arguments**

- **File**: the name of the text file generated by xCell that contains the cell signature scores.

**Value**

xCell signatures scaled and centered by sample. For GMIC, ImmuneScore, StromaScore, and MicroenvironmentScore are removed.

**Examples**

```r
file_dir<-system.file("extdata", "IRIS_xCell_sig.txt", package = "GmicR", mustWork = TRUE)
Xcell Sig<-xCell_loader(file_dir)
plot(Xcell_sig$Bcells)
```
Index

adjacency, 3
arc.strength, 5
Auto_WGCNA, 2
Batch_Net, 4
blockwiseModules, 3
bn_tabu_gen, 4
cutreeDynamic, 3
Data_Prep, 6
Gmic_viz, 7
GSEAGO_Builder, 8
hc, 5
InverseARCs, 9
Query_Prep, 10
score, 5
xCell_loader, 11