Package ‘motif4node’

January 29, 2023

Type Package

Title Motif4node, motif analysis using all non-redundant four-node gene circuits

Version 1.0

Author Benjamin Clauss, Mingyang Lu

Maintainer Benjamin Clauss <b.clauss@northeastern.edu>

Description R package for circuit motif analysis.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

NeedsCompilation no

VignetteBuilder knitr

Imports igraph, visNetwork, sRACIPE, scales, ggplot2, cowplot, pheatmap, htmlwidgets, SummarizedExperiment, webshot, combinat, RColorBrewer, grDevices

Suggests knitr

RoxygenNote 7.2.3

Depends R (>= 3.5.0)

R topics documented:

`adj_to_tpo` *Convert an adjacency matrix to a topology*

Description

Convert an adjacency matrix to a topology

Usage

```
adj_to_tpo(adj)
```

Arguments

`adj`	the adjacency martix to be converted to a topology file

`all.circuits` *Data file containing the topology of all 60212 non-redundant four-node gene circuits*

Description

This data contains the topology of all 60212 non-redundant four-node gene circuits.

Usage

```
all.circuits
```

Format

A list of 60212 circuits, each element is a matrix of adjacency matrix. Four columns/rows represent genes "A", "B", "C", and "D". In each adjacency matrix, 1 represents activation, 2 represents inhibition, and 0 represents no interaction.

analysis_circuit_2node

Analysis script to evaluate the state distribution of a two-node circuit motif

Description

Analysis script to evaluate the state distribution of a two-node circuit motif

Usage

```
analysis_circuit_2node(tpo, numModels = 10000)
```

Arguments

`tpo`	Data frame. Topology data of a two-node circuit motif. The data frame contains three columns: Source, Target, Interaction type
`numModels`	Numeric. Number of models to be simulated. Default: 10000

analysis_circuit_4node

Analysis script to evaluate the state distribution of a four-node gene circuit

Description

Analysis script to evaluate the state distribution of a four-node gene circuit

Usage

```
analysis_circuit_4node(rSet, numModels = 10000)
```

Arguments

`rSet`	sRACIPE object. RACIPE simulation data.
`numModels`	Numeric. Number of models to be simulated. Default: 10000

Value

List. plot objects.

`circuit_grouping` *Grouping two-node circuit motifs by their types*

Description

Grouping two-node circuit motifs by their types

Usage

```
circuit_grouping()
```

Value

Vector of factor: types for all two-node motifs.

`dist_ks` *Calculate th KS distance of two gene expression distributions*

Description

Calculate th KS distance of two gene expression distributions

Usage

```
dist_ks(query, reference, experimental)
```

Arguments

`query`	sRACIPE object or PCA matrix of the query data
`reference`	sRACIPE object of the reference data
`experimental`	Logical. T: query is the PCA matrix from an experimental dataset. F: query is an sRACIPE object

Value

the distance between the gene expression distributions from the query and reference.

enrichment_coupling *Motif enrichment analysis for the coupling of two two-node circuit motifs*

Description

Motif enrichment analysis for the coupling of two two-node circuit motifs

Usage

```
enrichment_coupling(
  all.circuits = all.circuits,
  all.scores,
  motif_list,
  new_ind,
  overlap_list,
  decreasing = T,
  if_overlap = 2,
  topCircuits = 600,
  nhill = 20
)
```

Arguments

all.circuits	List of the topologies of all 60212 non-redundant 4-node circuits. Default "all.circuits" from the package data.
all.scores	Data frame. 1st column: scores; 2nd column: circuit index. The data frame is ordered by the scores.
motif_list	List of 2 by 2 integer matrix. 2-node motif info generated by the function: generate_motif_list.
new_ind	Vector of integer. mapping to non-redundant 2-node motifs generated by the function: generate_index_conversion.
overlap_list	Matrix (6 by 6 integers). Info of overlapping from the motif location in the 4-node circuit generated by the function: generate_overlap_data.
decreasing	Logic. Whether circuits are ranked by the scores in a decreasing order (T) or not (F). Default T.
if_overlap	Whether consider two motifs with overlapping (1), without overlapping (0), or both (2). Default 2.
topCircuits	Integer. Number of top circuits for the enrichment analysis. Default 600.
nhill	Integer/Numeric. Hill coefficient for the Hill function as the weighting factor for motif counts, large n makes the Hill function more binary. Default 20.

Value

A data frame containing the enrichment scores of the coupling between 2-node circuit motifs (39 by 39)

enrichment_coupling_all_cases

A convenient function to perform different motif coupling analyses altogether

Description

A convenient function to perform different motif coupling analyses altogether

Usage

```
enrichment_coupling_all_cases(
  all.circuits,
  all.scores,
  motif_list,
  new_ind,
  overlap_list,
  decreasing = decreasing,
  topCircuits = 600,
  nhill = 20
)
```

Arguments

`all.circuits`	List of the topologies of all 60212 non-redundant 4-node circuits. Default "all.circuits" from the package data.
`all.scores`	Data frame. 1st column: scores; 2nd column: circuit index. The data frame is ordered by the scores.
`motif_list`	List of 2 by 2 integer matrix. 2-node motif info generated by the function: generate_motif_list.
`new_ind`	Vector of integer. mapping to non-redundant 2-node motifs generated by the function: generate_index_conversion.
`overlap_list`	Matrix (6 by 6 integers). Info of overlapping from the motif location in the 4-node circuit generated by the function: generate_overlap_data.
`decreasing`	Logic. Whether circuits are ranked by the scores in a decreasing order (T) or not (F). Default T.
`topCircuits`	Integer. Number of top circuits for the enrichment analysis. Default 600.
`nhill`	Integer/Numeric. Hill coefficient for the Hill function as the weighting factor for motif counts, large n makes the Hill function more binary. Default 20.

Value

List containing the enrichment of coupling between 2-node motifs for all cases (without overlapping, overlapping, and both)

`enrichment_single` *Motif enrichment analysis for single two-node circuit motifs*

Description

Motif enrichment analysis for single two-node circuit motifs

Usage

```
enrichment_single(
  all.circuits = all.circuits,
  all.scores,
  motif_list,
  new_ind,
  decreasing = T,
  topCircuits = 600,
  nhill = 20
)
```

Arguments

`all.circuits`	List of the topologies of all 60212 non-redundant 4-node circuits. Default "all.circuits" from the package data.
`all.scores`	Data frame. 1st column: scores; 2nd column: circuit index. The data frame is ordered by the scores.
`motif_list`	List of 2 by 2 integer matrix. 2-node motif info generated by the function: generate_motif_list.
`new_ind`	Vector of integer. mapping to non-redundant 2-node motifs generated by the function: generate_index_conversion.
`decreasing`	Logic. Whether circuits are ranked by the scores in a decreasing order (T) or not (F). Default T.
`topCircuits`	Integer. Number of top circuits for the enrichment analysis. Default 600.
`nhill`	Integer/Numeric. Hill coefficient for the Hill function as the weighting factor for motif counts, large n makes the Hill function more binary. Default 20.

Value

Data frame containing the enrichment scores of each 2-node circuit motif (39 by 1)

`generate_index_conversion` *Convert redundant indices to the indices for all 39 non-redundant motifs*

Description

Convert redundant indices to the indices for all 39 non-redundant motifs

Usage

```
generate_index_conversion()
```

---

generate_motif_list *Generate a list of all 72 2-node circuit motifs (including redundant ones)*

Description

Generate a list of all 72 2-node circuit motifs (including redundant ones)

Usage

```
generate_motif_list()
```

Value

list of all 72 2-node circuit motifs

---

gen_network_scalefree *Generate a random scale-free gene network consisting the two-node motifs of choice*

Description

Generate a random scale-free gene network consisting the two-node motifs of choice

Usage

```
gen_network_scalefree(num_nodes, motif_list, motif_choice)
```

Arguments

num_nodes	Integer. Number of nodes of the gene network to be generated.
motif_list	List of 2 by 2 integer matrix. 2-node motif info generated by the function: generate_motif_list.
motif_choice	Vector of integer. A vector of indices of the selected two-node circuit motifs

Value

Matrix. Adjacency matrix of the generated network

get_motif_adj *Obtain the adjacency matrix for a specific two-node circuit motif (number should be from 1 to 39)*

Description

Obtain the adjacency matrix for a specific two-node circuit motif (number should be from 1 to 39)

Usage

```
get_motif_adj(number, motif_list)
```

Arguments

`number`	Integer. Index of the two-node circuit motif.
`motif_list`	List of 2 by 2 integer matrix. 2-node motif info generated by the function: generate_motif_list.

Value

Integer matrix. 2 by 2 adjacency matrix.

lin_score *The scoring function for ranking circuits with a linear state distribution*

Description

The scoring function for ranking circuits with a linear state distribution

Usage

```
lin_score(rset)
```

Arguments

`rset`	The sRACIPE object of the simulated circuit.

Value

return(min(score_vector)): returns the score for the linear state distribution.

`map_and_project` *Compare the state distributions of two four-node circuits, find the most matched genes, and project the simulated gene expression data of the 2nd circuit to the PCs of the 1st circuit*

Description

Compare the state distributions of two four-node circuits, find the most matched genes, and project the simulated gene expression data of the 2nd circuit to the PCs of the 1st circuit

Usage

```
map_and_project(rset1, rset2)
```

Arguments

`rset1`	sRACIPE object. RACIPE simulation data for the first circuit
`rset2`	sRACIPE object. RACIPE simulation data for the second circuit

Value

Numeric matrix. PC coordinates of RACIPE simulated gene expression of the second circuit projected onto the PCs of the RACIPE simulated gene expression of the first circuit

`motif_analysis` *A combined motif enrichment analysis for single two-node circuit motifs and motif coupling*

Description

A combined motif enrichment analysis for single two-node circuit motifs and motif coupling

Usage

```
motif_analysis(
  all.circuits = all.circuits,
  all.scores,
  ylim = NULL,
  color_breaks = NULL,
  filename = NULL,
  decreasing = T,
  topCircuits = 600
)
```

Arguments

`all.circuits` List of the topologies of all 60212 non-redundant 4-node circuits. Default "all.circuits" from the package data.

`all.scores` Data frame. 1st column: scores; 2nd column: circuit index. The data frame is ordered by the scores.

`ylim` Vector of numerics (2). Y axis limit for single motif enrichment. Default: NULL.

`color_breaks` Vector that defines color scaling for pheatmap. Default: NULL

`filename` Character. Prefix of filenames for plotting. Default: NULL. If provided, plots are also saved to files.

`decreasing` Logical. Whether circuits are ranked by the scores in a decreasing order (T) or not (F). Default T.

`topCircuits` Integer. Number of top circuits for the enrichment analysis. Default 600.

Value

List of plotting objects for single motif and motif coupling enrichment analyses.

`plot_adj` *Plot a network from the adjacency matrix*

Description

Plot a network from the adjacency matrix

Usage

```
plot_adj(adj)
```

Arguments

`adj` Matrix (4 by 4, integer). Adjacency matrix of a four node circuit.

Value

empty

plot_motif *Plot a specific circuit motif*

Description

Plot a specific circuit motif

Usage

```
plot_motif(number, motif_list)
```

Arguments

`number`	Integer. Index of a two-node circuit motif
`motif_list`	List of 2 by 2 integer matrix. 2-node motif info generated by the function: generate_motif_list.

Value

empty

plot_net *Network plotting function*

Description

Network plotting function

Usage

```
plot_net(tf_links = tf_links)
```

Arguments

`tf_links`	Data frame of circuit edge list. Three columns: Source, Target, Interaction Types – 1: Activation; 2: Inhibition

Value

empty

plot_RACIPE *Scatterplots of PCA and Gene expression from any 4-node circuit*

Description

Scatterplots of PCA and Gene expression from any 4-node circuit

Usage

```
plot_RACIPE(rset)
```

Arguments

rset	The sRACIPE object of the simulated circuit

Value

list(g,p)

simu_rnorm *RACIPE simulations with random kinetic parameters from Gaussian distributions*

Description

RACIPE simulations with random kinetic parameters from Gaussian distributions

Usage

```
simu_rnorm(rset, numModels = 10000)
```

Arguments

rset	sRACIPE object. sRACIPE output from a standard RACIPE simulation
numModels	Numeric. Number of models to be simulated. Default: 10000.

sim_4node *Simulate one of the 4-node circuits from the 60212 unique circuits*

Description

Simulate one of the 4-node circuits from the 60212 unique circuits

Usage

```
sim_4node(index, Gaussian = F, numModels = 10000, all.circuits = all.circuits)
```

Arguments

`index`	Numeric index number of 4-node circuit to be simulate. Takes values from 1:60212
`Gaussian`	Logical. If T, kinetic parameters will be sampled from a gaussian distribution. If F, kinetic parameters will be sampled from a uniform distribution
`numModels`	Numeric. Number of models to be simulated. Default: 10000
`all.circuits`	List. The topology of all 60212 circuit motifs. Default "all.circuits" from the package data.

Value

rset: sRACIPE object. RACIPE simulation results for a circuit

single_motif_permute *Generate permutations for p-value calculations*

Description

Generate permutations for p-value calculations

Usage

```
single_motif_permute(
  all.circuits = all.circuits,
  all.scores,
  decreasing = T,
  topCircuits = 600,
  nhill = 20,
  num_perm = 1000
)
```

Arguments

`all.circuits` List of the topologies of all 60212 non-redundant 4-node circuits. Default "all.circuits" from the package data.

`all.scores` Data frame. 1st column: scores; 2nd column: circuit index. The data frame is ordered by the scores.

`decreasing` Logical. Whether circuits are ranked by the scores in a decreasing order (T) or not (F). Default T.

`topCircuits` Integer. Number of top circuits for the enrichment analysis. Default 600.

`nhill` Integer/Numeric. Hill coefficient for the Hill function as the weighting factor for motif counts, large n makes the Hill function more binary. Default 20.

`num_perm` Integer. Number of permutations. (1000)

Value

Numeric vector. Adjusted p values for the enrichement of each two-node circuit motif (a total of 39)

`trig_score` *The scoring function for ranking circuits with a triangular state distribution*

Description

The scoring function for ranking circuits with a triangular state distribution

Usage

```
trig_score(rset)
```

Arguments

`rset` The sRACIPE object of the simulated circuit.

Value

return(min(score_vector)): returns the score for the triangular state distribution.

`z_score` *Calculate z-score for a specific distance*

Description

Calculate z-score for a specific distance

Usage

```
z_score(distances, score)
```

Arguments

`distances`	list of distances
`score`	Numeric. score to calculate the z-score

Value

z-score

Index