

Package ‘GENIE3’

April 16, 2019

Type Package

Title GEne Network Inference with Ensemble of trees

Version 1.4.3

Date 2019-01-31

Author Van Anh Huynh-Thu, Sara Aibar, Pierre Geurts

Maintainer Van Anh Huynh-Thu <vahuyh@u.lg.ac.be>

Description This package implements the GENIE3 algorithm
for inferring gene regulatory networks from expression data.

License GPL (>= 2)

LazyData TRUE

Imports stats, reshape2

Suggests knitr, rmarkdown, foreach, doRNG, doParallel, Biobase,
SummarizedExperiment, testthat, methods

NeedsCompilation yes

RoxygenNote 6.0.1

VignetteBuilder knitr

biocViews NetworkInference, SystemsBiology, DecisionTree, Regression,
Network, GraphAndNetwork, GeneExpression

git_url <https://git.bioconductor.org/packages/GENIE3>

git_branch RELEASE_3_8

git_last_commit ae719c7

git_last_commit_date 2019-01-31

Date/Publication 2019-04-15

R topics documented:

GENIE3	2
getLinkList	3
Index	5

GENIE3

*GENIE3***Description**

GENIE3 infers a gene regulatory network (in the form of a weighted adjacency matrix) from expression data, using ensembles of regression trees.

Usage

```
GENIE3(exprMatrix, regulators = NULL, targets = NULL, treeMethod = "RF",
       K = "sqrt", nTrees = 1000, nCores = 1, verbose = FALSE)
```

Arguments

<code>exprMatrix</code>	Expression matrix (genes x samples). Every row is a gene, every column is a sample. The expression matrix can also be provided as one of the Bioconductor classes: <ul style="list-style-type: none"> • <code>ExpressionSet</code>: The matrix will be obtained through <code>exprs(exprMatrix)</code> • <code>RangedSummarizedExperiment</code>: The matrix will be obtained through <code>assay(exprMatrix)</code>, which will extract the first assay (usually the counts)
<code>regulators</code>	Subset of genes used as candidate regulators. Must be either a vector of indices, e.g. <code>c(1, 5, 6, 7)</code> , or a vector of gene names, e.g. <code>c("at_12377", "at_10912")</code> . The default value <code>NULL</code> means that all the genes are used as candidate regulators.
<code>targets</code>	Subset of genes to which potential regulators will be calculated. Must be either a vector of indices, e.g. <code>c(1, 5, 6, 7)</code> , or a vector of gene names, e.g. <code>c("at_12377", "at_10912")</code> . If <code>NULL</code> (default), regulators will be calculated for all genes in the input matrix.
<code>treeMethod</code>	Tree-based method used. Must be either <code>"RF"</code> for Random Forests (default) or <code>"ET"</code> for Extra-Trees.
<code>K</code>	Number of candidate regulators randomly selected at each tree node (for the determination of the best split). Must be either <code>"sqrt"</code> for the square root of the total number of candidate regulators (default), <code>"all"</code> for the total number of candidate regulators, or a strictly positive integer.
<code>nTrees</code>	Number of trees in an ensemble for each target gene. Default: 1000.
<code>nCores</code>	Number of cores to use for parallel computing. Default: 1.
<code>verbose</code>	If set to <code>TRUE</code> , a feedback on the progress of the calculations is given. Default: <code>FALSE</code> .

Value

Weighted adjacency matrix of inferred network. Element w_{ij} (row i , column j) gives the importance of the link from regulatory gene i to target gene j .

Examples

```
## Generate fake expression matrix
exprMatrix <- matrix(sample(1:10, 100, replace=TRUE), nrow=20)
rownames(exprMatrix) <- paste("Gene", 1:20, sep="")
colnames(exprMatrix) <- paste("Sample", 1:5, sep="")

## Run GENIE3
set.seed(123) # For reproducibility of results
weightMatrix <- GENIE3(exprMatrix, regulators=paste("Gene", 1:5, sep=""))

## Get ranking of edges
linkList <- getLinkList(weightMatrix)
head(linkList)
```

getLinkList	<i>getLinkList</i>
-------------	--------------------

Description

getLinkList Converts the weight matrix, as returned by [GENIE3](#), to a sorted list of regulatory links (most likely links first).

Usage

```
getLinkList(weightMatrix, reportMax = NULL, threshold = 0)
```

Arguments

weightMatrix	Weighted adjacency matrix as returned by GENIE3 .
reportMax	Maximum number of links to report. The default value NULL means that all the links are reported.
threshold	Only links with a weight above the threshold are reported. Default: threshold = 0, i.e. all the links are reported.

Value

List of regulatory links in a data frame. Each line of the data frame corresponds to a link. The first column is the regulatory gene, the second column is the target gene, and the third column is the weight of the link.

See Also

[GENIE3](#)

Examples

```
## Generate fake expression matrix
exprMat <- matrix(sample(1:10, 100, replace=TRUE), nrow=20)
rownames(exprMat) <- paste("Gene", 1:20, sep="")
colnames(exprMat) <- paste("Sample", 1:5, sep="")

## Run GENIE3
```

```
weightMat <- GENIE3(exprMat, regulators=paste("Gene", 1:5, sep=""))  
  
## Get ranking of edges  
linkList <- getLinkList(weightMat)  
head(linkList)
```

Index

GENIE3, [2](#), [3](#)
getLinkList, [3](#)