Package 'corto'

October 30, 2019

Type Package

Title Inference of Gene Regulatory Networks

Version 0.99.10

Maintainer Federico M. Giorgi <federico.giorgi@gmail.com>

Description We present 'corto' (Correlation Tool), a simple package to infer gene regulatory networks using DPI (Data Processing Inequality) and bootstrapping to recover edges. An initial step is performed to calculate all significant edges between a list of source nodes (centroids) and target genes. Then all triplets containing two centroids and one target are tested in a DPI step which removes edges. A bootstrapping process then calculates the robustness of the network, eventually re-adding edges previously removed by DPI. The package implements a similar pipeline as ARACNe-AP (Algorithm for the Reconstruction of Accurate Cellular Networks with Adaptive Partitioning) by Giorgi (2016) <doi:10.1093/bioinformatics/btw216> with optimizations to run outside a computing cluster (most notably correlation to infer feature dependencies instead of Mutual Information).

License LGPL-3

Encoding UTF-8

LazyData TRUE

RoxygenNote 6.1.1

Depends R (>= 3.5)

NeedsCompilation no

Imports dplyr, knitr, parallel, pbapply, rmarkdown, stats, utils

VignetteBuilder knitr

Author Federico M. Giorgi [aut, cre],
Daniele Mercatelli [ctb],
Gonzalo Lopez-Garcia [ctb]

Repository CRAN

Date/Publication 2019-10-30 17:00:14 UTC

R topics documented:

`corto` *Calculate a regulon from a data matrix*

Description

This function applies Spearman Correlation and DPI to generate a robust regulon object based on the input data matrix and the selected centroids.

Usage

```
corto(inmat, centroids, nbootstraps = 100, p = 1e-30, nthreads = 1,
  verbose = FALSE, cnvmat = NULL)
```

Arguments

`inmat`	Input matrix, with features (e.g. genes) as rows and samples as columns
`centroids`	A character vector indicating which features (e.g. genes) to consider as centroids (a.k.a. Master Regulators) for DPI
`nbootstraps`	Number of bootstraps to be performed. Default is 100
`p`	The p-value threshold for correlation significance (by default 1E-30)
`nthreads`	The number of threads to use for bootstrapping. Default is 1
`verbose`	Logical. Whether to print progress messages. Default is FALSE
`cnvmat`	An optional matrix with copy-number variation data. If specified, the program will calculate linear regression between the gene expression data in the input matrix (exp) and the cnv data, and target profiles will be transformed to the residuals of each linear model exp~cnv. Default is NULL

Value

A list (object of class regulon), where each element is a centroid

- tfmode: a named vector containing correlation coefficients between features and the centroid
- likelihood: a numeric vector indicating the likelihood of interaction

Examples

```
# Load data matrix inmat (from TCGA mesothelioma project)
load(system.file("extdata","inmat.rda",package="corto",mustWork=TRUE))
# Load centroids
load(system.file("extdata","centroids.rda",package="corto",mustWork=TRUE))
# Run corto
regulon <- corto(inmat,centroids=centroids,nthreads=2,nbootstraps=10,verbose=TRUE)

# In a second example, a CNV matrix is provided. The analysis will be run only
# for the features (rows) and samples (columns) present in both matrices
load(system.file("extdata","cnvmat.rda",package="corto",mustWork=TRUE))
regulon <- corto(inmat,centroids=centroids,nthreads=2,nbootstraps=6,verbose=TRUE,cnvmat=cnvmat,
p=1e-8)
```

---

fcor *A fast correlation function*

Description

A fast correlation function

Usage

```
fcor(inmat, centroids, r)
```

Arguments

inmat	An input matrix with features as rows and samples as columns
centroids	A character vector indicating the centroids
r	A numeric correlation threshold

Value

A matrix describing which edges were significant in the input matrix matrix according to the r correlation threshold provided

fisherp *Fisher integration of p-values*

Description

This function applies the Fisher integration of pvalues

Usage

```
fisherp(ps)
```

Arguments

ps	a vector of p-values

Value

p.val an integrated p-value

Examples

```
ps<-c(0.01,0.05,0.03,0.2)
fisherp(ps)
```

gsea *GSEA*

Description

This function performs Gene Set Enrichment Analysis

Usage

```
gsea(reflist, set, method = c("permutation", "pareto"), np = 1000,
  w = 1, gsea_null = NULL)
```

Arguments

reflist	named vector of reference scores
set	element set
method	one of 'permutation' or 'pareto'
np	Number of permutations (Default: 1000)
w	exponent used to raise the supplied scores. Default is 1 (original scores unchanged)
gsea_null	a GSEA null distribution (Optional)

Value

A GSEA object. Basically a list of s components:

ES The enrichment score

NES The normalized enrichment socre

ledge The items in the leading edge

p.value The permutation-based p-value

Examples

```
reflist<-setNames(-sort(rnorm(1000)),paste0('gene',1:1000))
set<-paste0('gene',sample(1:200,50))
obj<-gsea(reflist,set,method='pareto',np=1000)
obj$p.value
```

`gsea2` *2-way GSEA GSEA Gene set enrichment analysis of two complementary gene sets using gsea*

Description

2-way GSEA GSEA Gene set enrichment analysis of two complementary gene sets using gsea

Usage

```
gsea2(reflist, set1, set2, method = c("permutation", "pareto"),
  np = 1000, w = 1, gsea_null = NULL)
```

Arguments

`reflist`	named vector of reference scores
`set1`	element set 1
`set2`	element set 1
`method`	one of 'permutation' or 'pareto'
`np`	Number of permutations (Default: 1000)
`w`	exponent used to raise the supplied scores. Default is 1 (original scores unchanged)
`gsea_null`	a GSEA null distribution (Optional)

Value

A list of 2 GSEA objects. Each of which is a list of components:

ES The enrichment score

NES The normalized enrichment socre

ledge The items in the leading edge

p.value The permutation-based p-value

Examples

```
reflist<-setNames(-sort(rnorm(1000)),paste0('gene',1:1000))
set1<-paste0('gene',sample(1:200,50))
set2<-paste0('gene',sample(801:1000,50))
obj<-gsea2(reflist,set1,set2,method='pareto',np=1000)
obj$p.value
```

---

kmgformat *kmgformat - Nice Formatting of Numbers*

Description

This function will convert thousand numbers to K, millions to M, billions to G, trillions to T, quadrillions to P

Usage

```
kmgformat(input, roundParam = 1)
```

Arguments

input	A vector of values
roundParam	How many decimal digits you want

Value

A character vector of formatted numebr names

Examples

```
# Thousands
set.seed(1)
a<-runif(1000,0,1e4)
plot(a,yaxt='n')
kmg<-kmgformat(pretty(a))
axis(2,at=pretty(a),labels=kmg)

# Millions to Billions
set.seed(1)
a<-runif(1000,0,1e9)
plot(a,yaxt='n',pch=20,col="black")
kmg<-kmgformat(pretty(a))
axis(2,at=pretty(a),labels=kmg)
```

p2r *p2r Convert a P-value to the corresponding Correlation Coefficient*

Description

p2r Convert a P-value to the corresponding Correlation Coefficient

Usage

```
p2r(p, n)
```

Arguments

p	the p-value
n	the number of samples

Value

a correlation coefficient

Examples

```
p2r(p=0.08,n=20)
```

p2z *p2z*

Description

This function gives a gaussian Z-score corresponding to the provided p-value Careful: sign is not provided

Usage

```
p2z(p)
```

Arguments

p	a p-value

Value

z a Z score

Examples

```
p<-0.05
p2z(p)
```

plot_gsea *Plot GSEA results*

Description

This function generates a GSEA plot from a gsea object

Usage

```
plot_gsea(gsea.obj, twoColors = c("red", "blue"), plotNames = FALSE,
  colBarcode = "black", title = "Running Enrichment Score",
  bottomTitle = "List Values", bottomYlabel = "Signature values",
  ext_nes = NULL, omit_middle = FALSE)
```

Arguments

gsea.obj	GSEA object produced by the gsea function
twoColors	the two colors to use for positive[1] and negative[2] enrichment scores
plotNames	Logical. Should the set names be plotted?
colBarcode	The color of the barcode
title	String to be plotted above the Running Enrichment Score
bottomTitle	String for the title of the bottom part of the plot
bottomYlabel	String for the Y label of the bottom plot
ext_nes	Provide a NES from an external calculation
omit_middle	If TRUE, will not plot the running score (FALSE by default)

Value

Nothing, a plot is generated in the default output device

Examples

```
reflist<-setNames(-sort(rnorm(1000)),paste0('gene',1:1000))
set<-paste0('gene',sample(1:200,50))
obj<-gsea(reflist,set,method='pareto',np=1000)
plot_gsea(obj)
```

plot_gsea2 *Plot 2-way GSEA results*

Description

This function generates a GSEA plot from a gsea object

Usage

```
plot_gsea2(gsea.obj, twoColors = c("red", "blue"), plotNames = FALSE,
  title = "Running Enrichment Score", bottomTitle = "List Values",
  bottomYlabel = "Signature values")
```

Arguments

gsea.obj	GSEA object produced by the gsea function
twoColors	the two colors to use for positive[1] and negative[2] enrichment scores, and of the barcodes
plotNames	Logical. Should the set names be plotted?
title	String to be plotted above the Running Enrichment Score
bottomTitle	String for the title of the bottom part of the plot
bottomYlabel	String for the Y label of the bottom plot (FALSE by default)

Value

Nothing, a plot is generated in the default output device

Examples

```
reflist<-setNames(-sort(rnorm(1000)),paste0('gene',1:1000))
set1<-paste0('gene',sample(1:200,50))
set2<-paste0('gene',sample(801:1000,50))
obj<-gsea2(reflist,set1,set2,method='pareto',np=1000)
plot_gsea2(obj)
```

r2p *r2p Convert Correlation Coefficient to P-value*

Description

r2p Convert Correlation Coefficient to P-value

Usage

```
r2p(r, n)
```

Arguments

r	the correlation coefficient
n	the number of samples

Value

a numeric p-value

Examples

```
r2p(r=0.4,n=20) # 0.08
```

---

slice *Slice*

Description

This function prints a slice of a matrix

Usage

```
slice(matrix)
```

Arguments

matrix	A matrix

Value

A visualization of the first 5 rows and columns of the input matrix

Examples

```
set.seed(1)
example<-matrix(rnorm(1000),nrow=100,ncol=10)
slice(example)
```

stouffer *Stouffer integration of Z scores*

Description

This function gives a gaussian Z-score corresponding to the provided p-value Careful: sign is not provided

Usage

```
stouffer(x)
```

Arguments

x	a vector of Z scores

Value

Z an integrated Z score

Examples

```
zs<-c(1,3,5,2,3)
stouffer(zs)
```

wstouffer *Weighted Stouffer integration of Z scores*

Description

This function gives a gaussian Z-score corresponding to the provided p-value Careful: sign is not provided

Usage

```
wstouffer(x, w)
```

Arguments

x	a vector of Z scores
w	weight for each Z score

Value

Z an integrated Z score

Examples

```
zs<-c(1,-3,5,2,3)
ws<-c(1,10,1,2,1)
wstouffer(zs,ws)
```

---

z2p *z2p*

Description

This function gives a gaussian p-value corresponding to the provided Z-score

Usage

```
z2p(z)
```

Arguments

z	a Z score

Value

a p-value

Examples

```
z<-1.96
z2p(z)
```

Index