dNetPipeline

December 16, 2014

dNetPipeline	*Function to setup the pipeline for finding maximum-scoring subgraph from an input graph and the signficance imposed on its nodes*

Description

dNetPipeline is supposed to finish ab inito maximum-scoring subgraph identification for the input graph with the node information on the significance (p-value or fdr). It returns an object of class "igraph" or "graphNEL".

Usage

```
dNetPipeline(g, pval, method = c("pdf", "cdf", "customised"),
significance.threshold = NULL, nsize = NULL, plot = F, verbose = T)
```

Arguments

g
: an object of class "igraph" or "graphNEL"

pval
: a vector containing input p-values (or fdr). For each element, it must have the name that could be mapped onto the input graph. Also, the names in input "pval" should contain all those in the input graph "g", but the reverse is not necessary

method
: the method used for the transformation. It can be either "pdf" for the method based on the probability density function of the fitted model, or "cdf" for the method based on the cumulative distribution function of the fitted model

significance.threshold
: the given significance threshold. By default, it is set to NULL, meaning there is no constraint. If given, those p-values below this are considered significant and thus scored positively. Instead, those p-values above this given significance threshold are considered insigificant and thus scored negatively

nsize
: the desired number of nodes constrained to the resulting subgraph. It is not nulll, a wide range of significance thresholds will be scanned to find the optimal significance threshold leading to the desired number of nodes in the resulting subgraph. Notably, the given significance threshold will be overwritten by this option.

plot
: logical to indicate whether the histogram plot, contour plot and scatter plot should be drawn. By default, it sets to false for no plotting

verbose
: logical to indicate whether the messages will be displayed in the screen. By default, it sets to true for display

Value

a subgraph with a maximum score, an object of class "igraph" or "graphNEL"

Note

The pipeline sequentially consists of:

- ia) if the method is either "pdf" or "cdf", `dBUMfit` used to fit the p-value distribution under beta-uniform mixture model, and `dBUMscore` used to calculate the scores according to the fitted BUM and the significance threshold.
- ib) if the method is either "customised", then the user input list of fdr (or p-values) and the significance threshold will be directly used for score transformation by `dFDRscore`.
- ii) if there is the desired number of nodes constrained to the resulting subgraph, a wide range of significance thresholds (including rough stage with large intervals, and finetune stage with smaller intervals) will be scanned to find the significance threshold to meet the desired number of nodes.
- iii) `dNetFind` used to find maximum-scoring subgraph from the input graph and scores imposed on its nodes.

See Also

`dBUMfit`, `dBUMscore`, `dFDRscore`, `dNetFind`

Examples

```
# 1) generate an vector consisting of random values from beta distribution
x <- rbeta(1000, shape1=0.5, shape2=1)
names(x) <- as.character(1:length(x))

# 2) generate a random graph according to the ER model
g <- erdos.renyi.game(1000, 1/100)

# 3) produce the induced subgraph only based on the nodes in query
subg <- dNetInduce(g, V(g), knn=0)

# 4) find maximum-scoring subgraph based on the given significance threshold
# 4a) assume the input is a list of p-values (controlling fdr=0.1)
subgraph <- dNetPipeline(g=subg, pval=x, significance.threshold=0.1)
# 4b) assume the input is a list of customised significance (eg FDR directly)
subgraph <- dNetPipeline(g=subg, pval=x, method="customised",
significance.threshold=0.1)

# 5) find maximum-scoring subgraph with the desired node number nsize=20
subgraph <- dNetPipeline(g=subg, pval=x, nsize=20)
```