

Package ‘CARNIVAL’

October 16, 2020

Title A Causal Reasoning tool for Network Identification (from gene expression data) using Integer VALue programming

Version 1.0.0

Author Enio Gjerga <enio.gjerga@gmail.com>
Panuwat Trairatphisarn
Anika Liu
Alberto Valdeolivas
Nikolas Peschke

Maintainer Enio Gjerga <enio.gjerga@gmail.com>

Description An upgraded causal reasoning tool from Melas et al in R with updated assignments of TFs' weights from PROGENy scores.
Optimization parameters can be freely adjusted and multiple solutions can be obtained and aggregated.

URL <https://github.com/saezlab/CARNIVAL>

BugReports <https://github.com/saezlab/CARNIVAL/issues>

Depends R (>= 4.0)

Imports doParallel, readr, viper, AnnotationDbi, Category, ggplot2,
UniProt.ws, lpSolve, igraph

biocViews Transcriptomics, GeneExpression, Network

License Apache License (== 3.0) | file LICENSE

LazyData true

Encoding UTF-8

Suggests knitr, readxl, testthat (>= 2.1.0)

VignetteBuilder knitr

RoxygenNote 7.0.2

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

git_branch RELEASE_3_11

git_last_commit deb8dbb

git_last_commit_date 2020-04-27

Date/Publication 2020-10-16

R topics documented:

runCARNIVAL 2

Index**4**

runCARNIVAL	runCARNIVAL
-------------	-------------

Description

Run CARNIVAL pipeline using to the user-provided list of inputs or run CARNIVAL built-in examples

Usage

```
runCARNIVAL(inputObj=NULL, measObj=measObj, netObj=netObj, weightObj=NULL,
            solverPath=NULL, solver=c("lpSolve", "cplex", "cbc"),
            timelimit=3600, mipGAP=0.05, poolrelGAP=0.0001, limitPop=500,
            poolCap=100, poolIntensity=4, poolReplace=2, alphaWeight=1,
            betaWeight=0.2, threads = 0, dir_name=NULL)
```

Arguments

inputObj	Data frame of the list for target of perturbation - optional or default set to NULL to run invCARNIVAL when inputs are not known
measObj	Data frame of the measurement file (i.e. DoRothEA normalised enrichment scores) - always required
netObj	Data frame of the prior knowledge network - always required
weightObj	Data frame of the additional weight (i.e. PROGENy pathway score or measured protein activities) - optional or default set as NULL to run CARNIVAL without weights
solverPath	Path to executable cbc/cplex file - default set to NULL, in which case the solver from lpSolve package is used
solver	Solver to use: lpSolve/cplex/cbc (Default set to lpSolve)
timelimit	CPLEX/Cbc parameter: Time limit of CPLEX optimisation (in seconds)
mipGAP	CPLEX parameter: the absolute tolerance on the gap between the best integer objective and the objective of the best node remaining. When this difference falls below the value of this parameter, the linear integer optimization is stopped. Default set to 0.05
poolrelGAP	CPLEX/Cbc parameter: Allowed relative gap of accepted solution comparing within the pool of accepted solution (fraction; default: 0.0001)
limitPop	CPLEX parameter: Allowed number of solutions to be generated (default: 500)
poolCap	CPLEX parameter: Allowed number of solution to be kept in the pool of solution (default: 100)
poolIntensity	CPLEX parameter: Intensity of solution searching (0,1,2,3,4 - default: 4)
poolReplace	CPLEX parameter: Replacement strategy of solutions in the pool (0,1,2 - default: 2 = most diversified solutions)
alphaWeight	Objective function: weight for mismatch penalty (default: 1 - will only be applied once measurement file only contains discrete values)
betaWeight	Objective function: weight for node penalty (default: 0.2)

threads	Number of threads to be used for the analysis (default: 0 - meaning all threads)
dir_name	For plotting: define if DOT figure will be exported in the directory defined by this variable. Default set to NULL, meaning that no .DOT figure will be exported. If a valid path is specified, the figure will be exported as 'network_solution.dot' file in the specified directory. If specified dir_name path does not exist, no figure will be generated

Value

The networks and predicted node activities from the CARNIVAL pipeline as a variable which are also optionally saved in the destined result folder

Author(s)

Enio Gjerga

Examples

```
## loading toy example 1

load(file = system.file("toy_inputs_ex1.RData",
package="CARNIVAL"))
load(file = system.file("toy_measurements_ex1.RData",
package="CARNIVAL"))
load(file = system.file("toy_network_ex1.RData",
package="CARNIVAL"))

## lpSolve
res1 = runCARNIVAL(inputObj = toy_inputs_ex1, measObj = toy_measurements_ex1,
netObj = toy_network_ex1)

# ## cbc
# res2 = runCARNIVAL(inputObj = toy_inputs_ex1, measObj = toy_measurements_ex1,
# netObj = toy_network_ex1, solverPath = solverPath,
# solver = "cbc")
#
# ## cplex
# res3 = runCARNIVAL(inputObj = toy_inputs_ex1, measObj = toy_measurements_ex1,
# netObj = toy_network_ex1, solverPath = solverPath,
# solver = "cplex")
```

Index

runCARNIVAL, [2](#)