Package ‘scenario’

February 17, 2016

Type Package
Title Construct Reduced Trees with Predefined Nodal Structures
Version 1.0
Date 2016-02-15
URL https://github.com/swerwer/scenario

Description Uses the neural gas algorithm to construct
a scenario tree for use in multi-stage stochastic programming.
The primary input is a set of initial scenarios or realizations
of a disturbance. The scenario tree nodal structure must be
predefined using a scenario tree nodal partition matrix.

License GPL (>= 2)
LazyData yes
Imports graphics
Suggests knitr
VignetteBuilder knitr
NeedsCompilation no
Author Sean Turner [aut, cre],
        Stefano Galelli [aut]
Maintainer Sean Turner <swd.turner@gmail.com>
Repository CRAN
Date/Publication 2016-02-17 10:52:20

R topics documented:

  buidtree ................................................................. 2
  checktree ............................................................... 3
  scenario ................................................................. 3

Index 5
Build a scenario tree with a predefined nodal structure.

Description

Uses the neural gas method to build a scenario tree.

Usage

buildtree(x, treeStruct, lambda_0 = 10, lambda_f = 0.01, e_0 = 0.5, e_f = 0.05, jmax = 40000, plot = TRUE)

Arguments

x Matrix of initial scenarios, realizations or ensemble members. Each column stores a scenario, with number of rows equal to number of time steps.
treeStruct Matrix defining the nodal structure of the tree (see example). This is a scenario tree nodal partition matrix.
lambda_0 Upper neighbourhood range parameter. Default = 10.
lambda_f Lower neighborhood range parameter. Default = 0.01.
e_0 Upper adaptation step parameter. Default = 0.5.
e_f Lower adaptation step parameter. Default = 0.05.
jmax Number of iterations. Default = 40000.
plot logical. If TRUE (the default) the final tree is plotted.

Value

Returns a list object containing the initial input scenarios, the input scenarios tree structure, the values of the final reduced scenario tree, and the tree branch probabilities at the end nodes.

References


Examples

# Generate some 25 random realizations of length 4 and reduce to scenario tree.
scenarios <- matrix(rnorm(100), ncol=25)
treeStruct <- rbind(c(1, 1, 1, 1),
                    c(2, 2, 7, 7, 11),
                    c(3, 5, 8, 8, 12),
                    c(4, 6, 9, 10, 13)
                  )
tree <- buildtree(scenarios, treeStruct, jmax = 1000)
checktree

Check the tree structure implied by a scenario tree nodal partition matrix.

Description

Returns a plot showing the nodal structure (not values) of a scenario tree defined using a scenario tree nodal partition matrix.

Usage

checktree(treeStruct)

Arguments

treeStruct     Matrix defining the nodal structure of the tree.

Value

Returns a plot of the scenario tree structure implied by the input nodal partition matrix.

References


Examples

treeStruct <- rbind(c(1, 1, 1, 1, 1),
                    c(2, 2, 7, 7, 11),
                    c(3, 5, 8, 8, 12),
                    c(4, 6, 9, 10, 13))
checktree(treeStruct)

scenario

scenario: Construct reduced trees with a predefined nodal structures

Description

The buildtree function uses the neural gas method to generate a scenario tree of predefined nodal structure. The checktree function plots a scenario tree structure as defined by a nodal partition matrix.
References


Examples

# TEST BY GENERATING SCENARIOS FROM KNOWN CENTROIDS AND THEN
# COMPARING FIT BETWEEN THE GENERATED TREE AND INITIAL CENTROIDS.

# 1. Generate scenarios with known centroids:

centroids <- cbind(c(0,2,3), c(0,2,1), c(0,-2,-3), c(0,-2,-1))
matplot(centroids, type="l", lwd = 3, col = "black", lty = 3)
scenarios <- matrix(rep(centroids, 5), ncol=10) + matrix(rnorm(60, 0, 0.25), ncol=20)
matlines(scenarios, col = "grey")

# 2. Assign and check nodal structure for tree:

treeStruct <- rbind(c(1,1,1,1),
                    c(2,2,5,5),
                    c(3,4,6,7))
checktree(treeStruct)

# 3. Build scenario tree:

tree <- buildtree(scenarios, treeStruct, jMax = 1000)

# 4. Compare original centroids

matlines(centroids,lwd = 3, col = "black", lty = 3)
# Improved convergence is achieved by increasing the number of iterations, jMax.
Index

buildtree, 2, 3
checktree, 3, 3
scenario, 3
scenario-package (scenario), 3