

Package ‘rgml0’

May 9, 2026

Type Package

Title Random Graphical Model Estimation under L0 Penalty

Version 0.0.1

Description Provides functions for estimating sparse precision matrices using a random graphical model framework under an L0-style penalty. The method evaluates candidate theta values and returns both continuous and binary precision matrices representing inferred network structures.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Imports MASS

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Mahdiah Farzin Asanjan [aut, cre],
Vilda Purutcuoglu [aut],
Ernst Wit [aut]

Maintainer Mahdiah Farzin Asanjan <mah.farzin2010@gmail.com>

Repository CRAN

Date/Publication 2026-04-28 19:40:07 UTC

Contents

rgml0	2
Index	4

 rgml0

Random graphical model estimation under L0-penalty

Description

Estimates sparse precision matrices from a covariance matrix using an iterative sparse-step approximation under an L0-style penalty. The method evaluates a grid of theta values and returns both continuous and binary precision matrices representing the inferred network structure.

Usage

```
rgml0(
  S,
  theta.jump = 0.1,
  n.iter = 5,
  gamma0 = 1000,
  gammastop = 1e-04,
  IMsteps = 2,
  gammastep = 1.1,
  force.zero = TRUE,
  threshold = 1e-07,
  use_diag = FALSE
)
```

Arguments

S	A numeric covariance matrix.
theta.jump	Step size for candidate theta values.
n.iter	Number of inner iterations.
gamma0	Initial gamma value.
gammastop	Stopping value for gamma.
IMsteps	Currently unused.
gammastep	Factor used to decrease gamma.
force.zero	Logical flag (currently unused).
threshold	Threshold for binarizing the precision matrix.
use_diag	Logical flag (currently unused).

Value

A list where each element contains:

- precision_matrix** Estimated precision matrix.
- binary_precision_matrix** Thresholded binary matrix.
- theta** Theta value used.
- sample_index** Index of theta.

Examples

```
S <- diag(3)
out <- rgml0(S, theta.jump = 0.4, n.iter = 1,
             gamma0 = 1, gammastop = 0.1, gammastep = 2)
length(out)
```

Index

rgml0, 2