Package 'persistence'

July 23, 2025

Type Package

Title Optimal Graph Partition using the Persistence

Version 0.1.0

Description
Calculate the optimal vertex partition of a graph using the persistence as objective function. These subroutines have been used in Avellone et al. <doi:10.1007/s10288-023-00559-z>.

License GPL (>= 2)

Encoding UTF-8

SystemRequirements C++20

Suggests igraph

RoxygenNote 7.3.2

NeedsCompilation yes

Collate 'persistence-exports.R' 'cluster_milano.R' 'global_persistence.R' 'local_persistence.R'

Author Alessandro Avellone [aut, cre], Paolo Bartesaghi [aut], Stefano Benati [aut], Rosanna Grassi [aut]

Maintainer Alessandro Avellone <alessandro.avellone@unimib.it>

Repository CRAN

Date/Publication 2025-05-21 08:30:02 UTC

Contents

persistence-package *Persistence*

Description

Given a non-oriented graph, calculates the optimal vertex partition using the persistence as the objective function.

Details

See manual entries.

Author(s)

Maintainer: Alessandro Avellone <alessandro.avellone@unimib.it>

Authors:

- Paolo Bartesaghi <paolo.bartesaghi@unimi.it>
- Stefano Benati <stefano.benati@unitn.it>
- Rosanna Grassi <rosanna.grassi@unimib.it>

cluster_milano *cluster Milano*

Description

Calculates the partition with maximum global null-adjuted persistence.

Usage

```
cluster_milano(vertex, edge_list, seed = NULL)
```

Arguments

vertex	the vertices of the graph, whose label are integers and they must be consistent with the edge sets.
edge_list	the graph edge list in the form of an integer matrix with two columns.
seed	As some steps of the algorithm are random, users may experiments with different seeds of random numbers.

Value

A list containing:

membeship The optimal vertex partition.

value The null-adjusted persistence of the partition.

seed The used seed to generate random numbers.

Examples

```
library(persistence)
library(igraph)

edg = c(1, 2, 1, 3, 1, 4, 2, 3, 3, 4, 4, 5, 5, 6, 5, 7, 5, 8, 5, 9, 6, 7, 6, 8, 7, 9, 8, 9)
print(length(edg) / 2.0)
vertex = unique(edg)
edg = t(matrix(as.integer(edg), nrow = 2 ))
rete <- graph_from_edgelist(edg, directed = FALSE)
plot(rete)
seed <- sample(1:as.integer(.Machine$integer.max),1, replace= FALSE)
r = cluster_milano(vertex, edg, seed=seed)
print(paste("The optimal null-adjusted persistence is: ", r$measure))
print(paste("The optimal persistence probability is: ", r$measure + 1))
```

---

global_persistence *global persistence*

Description

Given a partition of the graph vertices, it calculates the global persistence as the sum of the persistences of the single clusters. Persistence can be referred to the null-adjusted o to the probability.

Usage

```
global_persistence(vertex, edge_list, membership, H0 = TRUE)
```

Arguments

vertex	the vertices of the graph, whose label are integers and they must be consistent with the edge sets.
edge_list	the graph edge list in the form of an integer matrix with two columns.
membership	An integer vector representing the vertex membership: x_i = k if i in C_k.
H0	If true, it calculates the null-adjusted persistence, if false, the persistence probability.

Value

value A list containing the following:

value The global persistence of the partition.

clusters_value The local persistence of each cluster. If for some k we have v_k = NaN, then C_k is empty in the input membership.

Examples

```
library(persistence)
library(igraph)

edg = c(1, 2, 1, 3, 1, 4, 2, 3, 3, 4, 4, 5, 5, 6, 5, 7, 5, 8, 5, 9, 6, 7, 6, 8, 7, 9, 8, 9)
print(length(edg) / 2.0)
vertex = unique(edg)
edg = t(matrix(as.integer(edg), nrow = 2 ))
rete <- graph_from_edgelist(edg, directed = FALSE) # I graph this matrix
plot(rete)

membership = c(1, 1, 1, 1, 2, 2, 2, 2, 2)
v1 = global_persistence(vertex, edg, membership, H0=TRUE)
print(paste("global null-adjusted persistence: ", v1$value))
print(paste("null-adjusted persistence per cluster: ", v1$clusters_value))
```

`local_persistence` *local_persistence*

Description

Given the incidence vector of a vertex subset, it calculates the persistence probability or the null-adjusted persistence of C.

Usage

```
local_persistence(vertex, edge_list, cluster, H0 = TRUE)
```

Arguments

`vertex`	the vertices of the graph, whose label are integers and they must be consistent with the edge sets
`edge_list`	the graph edge list in the form of an integer matrix with two columns
`cluster`	A binary vector representing the incidence vector of the cluster: x_i = 1 if i in C, 0 otherwise.
`H0`	if true, it calculates the null-adjusted persistence, if false, the persistence probability.

Value

the value of the null-adjusted persistence if H0 = T, the value of the persistence probability if H0 = F

Examples

```
#' library(persistence)
library(igraph)

edg = c(1, 2, 1, 3, 1, 4, 2, 3, 3, 4, 4, 5, 5, 6, 5, 7, 5, 8, 5, 9, 6, 7, 6, 8, 7, 9, 8, 9)
print(length(edg) / 2.0)
vertex = unique(edg)
edg = t(matrix(as.integer(edg), nrow = 2 ))
rete <- graph_from_edgelist(edg, directed = FALSE) # I graph this matrix
plot(rete)

cluster = rep(0, length(vertex))
v1 = c(1, 2, 3, 4)
cluster[v1] = 1
f1 = local_persistence(vertex, edg, cluster, H0 = TRUE)
f2 = local_persistence(vertex, edg, cluster, H0 = FALSE)
```

Index