Package ‘partitionstability’

August 30, 2018

Type Package

Title Provides Implementations of Algorithms Presented in the Article of Ball and Geyer-Schulz (2018)

Version 0.1

Date 2018-08-30

Author Fabian Ball [aut, cre, cph, ctb]

Maintainer Fabian Ball <fabian.ball@kit.edu>

Description Exemplary implementations of Algorithms 1 to 4 of Ball, F and Geyer-Schulz A (2018) <doi:TBA> that can be used to test partition stability given a set of generators for a permutation group that acts on the partitioned set.
The implementations strongly relate to the pseudocode, which means there certainly exist ways to implement the algorithms more efficiently.
However, the main focus lies on supplying supplemental material for the cited article.

License MIT + file LICENSE

Depends R (>= 3.2.0)

RoxygenNote 6.0.1

RdMacros Rdpack

Imports sets, Rdpack

Suggests testthat

R topics documented:

partitionstability-package *partitionstability: Provides Implementations of Algorithms Presented in the Article of Ball and Geyer-Schulz (2018)*

Description

Exemplary implementations of Algorithms 1 to 4 of Ball, F and Geyer-Schulz A (2018) <doi:TBA> that can be used to test partition stability given a set of generators for a permutation group that acts on the partitioned set. The implementations strongly relate to the pseudocode, which means there certainly exist ways to implement the algorithms more efficiently. However, the main focus lies on supplying supplemental material for the cited article.

Details

Algorithms 1 (testStability) and 4 (geq) are nearly one-to-one implementations of the pseudocode. One exception is that we removed the need of a hash map in geq by simply using a vector of the same length as the partition to build our map of cluster ids. However, this implies that cluster ids must be the same as the node ids.

Algorithms 2 and 3 are merged into a single function (computeOrbitPartition) as R does not easily support call be reference.

For details of the algorithms and the background we refer to Ball, Fabian and Geyer-Schulz, Andreas (2018), "Symmetry-based Graph Clustering Partition Stability", Archives of Datascience Series A <vol tba>(<nr tba>), DOI: <doi tba>

Author(s)

Maintainer: Fabian Ball <`fabian.ball@kit.edu`> [copyright holder, contributor]

computeOrbitPartition *computeOrbitPartition*

Description

Implementation of Algorithms 2 and 3 (combined as one function), which computes the orbit partition of a permutation group from a given set of generators.

Usage

```
computeOrbitPartition(S, n)
```

Arguments

`S`	Set of generators
`n`	Length of the set the group $\langle S \rangle$ acts on

Examples

```
library("sets")
p1 <- c(2L, 1L, 3L, 4L, 6L, 5L, 7L, 8L, 9L, 10L)
p2 <- c(1L, 2L, 3L, 4L, 5L, 6L, 7L, 8L, 10L, 9L)
p3 <- c(1L, 9L, 3L, 4L, 5L, 6L, 7L, 8L, 2L, 10L)
S <- list(p1, p2, p3)
computeOrbitPartition(S, 10)
# [1] 1 1 3 4 5 5 7 8 1 1
```

`geq` *geq*

Description

Implementation of Algorithm 4, which tests $P \geq Q$

Usage

```
geq(P, Q)
```

Arguments

`P`	A partition of length n
`Q`	Another partition of length n#'

Examples

```
P = c(1, 1, 1, 2, 2, 2)
P_prime = c(2, 2, 2, 1, 1, 1)
Q = c(4, 4, 1, 3, 3, 2)
R = c(4, 4, 4, 4, 2, 2)
isTRUE(geq(P, Q))
isTRUE(geq(P, P_prime))
isTRUE(geq(P, P_prime) == geq(P_prime, P))
!isTRUE(geq(P, R))
!isTRUE(geq(R, P))
```

`testStability` *testStability*

Description

Implementation of Algorithm 1, which checks if a partition is stabl udner the given symmetry implied by the generators of a permutation group S.

Usage

```
testStability(P, S)
```

Arguments

`P`	A partition
`S`	Set of generators for a group $\langle S \rangle$ that acts on P

Examples

```
p1 <- c(2L, 1L, 3L, 4L, 6L, 5L, 7L, 8L, 9L, 10L)
p2 <- c(1L, 2L, 3L, 4L, 5L, 6L, 7L, 8L, 10L, 9L)
p3 <- c(1L, 9L, 3L, 4L, 5L, 6L, 7L, 8L, 2L, 10L)
S <- list(p1, p2, p3)
P <- rep(1L, 10)
isTRUE(testStability(P, S))
Q <- c(2L, 2L, 1L, 1L, 1L, 1L, 1L, 1L, 3L, 3L)
!isTRUE(testStability(Q, S))
```

Index