Package 'localboot'

January 10, 2024

Type Package

Title Local Bootstrap Methods for Various Networks

Version 0.9.2

Date 2024-01-08

Description Network analysis usually requires estimating the uncertainty of
graph statistics. Through this package, we provide tools to bootstrap
various networks via local bootstrap procedure. Additionally, it includes
functions for generating probability matrices, creating network adjacency
matrices from probability matrices, and plotting network structures.
The reference will be updated soon.

License GPL-3

Depends R (>= 3.5.0)

Imports graphics, viridis, Rcpp (>= 1.0.11)

Suggests testthat (>= 3.0.0)

LinkingTo Rcpp (>= 1.0.11), RcppEigen

Encoding UTF-8

RoxygenNote 7.2.3

NeedsCompilation yes

Author Tianhai Zu [aut, cre],
Yichen Qin [aut, ctb]

Maintainer Tianhai Zu <zuti@mail.uc.edu>

Repository CRAN

Date/Publication 2024-01-10 11:33:21 UTC

R topics documented:

localboot-package	*localboot: A Package for Local Bootstrap Methods for Various Networks*

Description

Network analysis usually requires estimating the uncertainty of a graph statistics. Through this package, we provides tools to bootstrap various networks via local bootstrap procedure. Additionally, it includes functions for generating probability matrices, creating network adjacency matrices from probability matrices, and plotting network structures.

Author(s)

Maintainer: Tianhai Zu <zuti@mail.uc.edu>

Authors:

• Yichen Qin <qinyn@ucmail.uc.edu> [contributor]

generate_graphon	*Generate a Graphon Probability Matrix*

Description

This function generates a graphon probability matrix based on a specified graphon type. Users can control the generation process through various parameters.

Usage

```
generate_graphon(size, graph_num = 1, sampling_on_u = TRUE, u_input = NULL)
```

Arguments

size	An integer specifying the size of the network.
graph_num	An integer (default is 1) indicating the graphon type to use. Acceptable values are from 1 to 6.
sampling_on_u	A logical value determining if uniform sampling should be used for 'u'. Defaults to TRUE. If FALSE, a regular sequence from 0 to 1 is used.
u_input	An optional numeric vector that provides specific values for 'u'. If NULL (default), 'u' is generated based on 'sampling_on_u'.

Value

A matrix of probabilities is returned.

Examples

```
# Generate a graphon probability matrix of size 100 using graphon setting 1
P = generate_graphon(100, 1)
```

---

generate_network_P	*Generate Network Adjacency Matrix from Probability Matrix*

Description

This function generates a network adjacency matrix from a given probability matrix. It checks if the input is a valid probability matrix and can produce either a single network or multiple replicates.

Usage

```
generate_network_P(P, replicate = 1, symmetric.out = TRUE, noloop = FALSE)
```

Arguments

P	A square matrix representing the probability matrix, where each element is a probability (between 0 and 1) of an edge between nodes.
replicate	An integer indicating the number of network replicates to generate. Defaults to 1.
symmetric.out	A logical value indicating whether the output matrix should be symmetric. Defaults to TRUE.
noloop	A logical value indicating whether to include self-loops in the network. Defaults to FALSE.

Value

If 'replicate' is 1, returns a single adjacency matrix. If 'replicate' is greater than 1, returns a list of adjacency matrices. Each matrix is a square binary matrix, where 1 indicates the presence of an edge and 0 indicates its absence.

Examples

```
P = generate_graphon(100, 1)
network = generate_network_P(P, replicate = 1, symmetric.out = TRUE)
```

---

localboot	*Local Bootstrap for Network Data*

Description

This function applies a local bootstrap method to network data, represented by an adjacency matrix. It offers various methods and options for bootstrapping, including handling weighted networks and custom distance functions.

Usage

```
localboot(
  A,
  B,
  quantile_n = 0,
  returns = "boot",
  method = "own",
  dist_func = get_dist_default_eigen,
  kowning_u = NULL,
  induced_sampling = TRUE,
  weighted = FALSE,
  getT = NULL,
  user_blist = NULL,
  fast = NULL,
  ...
)
```

Arguments

A	A square adjacency matrix of the network.
B	The number of bootstrap samples to generate.
quantile_n	The quantile used for neighborhood selection in some methods. If set to 0 (default), it's calculated as $(\log(N) / N)^{0.5}$.
returns	Specifies the type of output returned. Possible values are "boot" (default), "p_and_time", "p_and_boot", and "T".
method	The method used for bootstrapping. Options are "own" and "zhu".
dist_func	A function to compute the distance matrix. Default is 'get_dist_default_eigen'.
kowning_u	An optional known 'u' vector for distance calculation.
induced_sampling	
	A logical indicating whether to use induced sampling. Defaults to TRUE.
weighted	A logical indicating if the network is weighted. Defaults to FALSE.
getT	An optional function to apply to each bootstrapped sample.
user_blist	An optional user-provided bootstrap list.
fast	A logical indicating if a faster, approximate method should be used. Automatically set based on network size if NULL.
...	Additional arguments passed to other methods.

Value

Depending on the 'returns' argument, this function can return various types of outputs including bootstrapped networks, estimated probabilities, computation times, and statistics from the 'getT' function.

Examples

```
# Example usage
P = generate_graphon(100, 1)
A = generate_network_P(P, replicate = 1, symmetric.out = TRUE)
result <- localboot(A = A, B = 100, returns = "boot")
```

---

plot_adj   *Plot Adjacency Matrix*

Description

This function creates a plot of an adjacency matrix, where the matrix is displayed as an image.

Usage

```
plot_adj(X, ...)
```

Arguments

X   An adjacency matrix to be plotted.

...   Additional graphical parameters to pass to 'image'.

Value

Generates a plot.

Examples

```
adj_matrix <- matrix(rbinom(100, 1, 0.5), 10, 10)
plot_adj(adj_matrix)
```

---

plot_P   *Plot Probability Matrix*

Description

This function creates a filled contour plot of a probability matrix using the 'viridis' color palette. The plot is created using the 'ggplot2' and 'viridis' libraries.

Usage

```
plot_P(P)
```

Arguments

P   A probability matrix to be plotted.

Value

Generates a filled contour plot.

Examples

```
P_matrix <- matrix(runif(100), 10, 10)
plot_P(P_matrix)
```

Index