

Package: rMultiNet (via r-universe)

October 11, 2026

Type Package

Title rMultiNet: An R Package For Multilayer Networks Analysis

Version 0.1.0

Author Chenyu Ren, Ting Li, Zhongyuan Lyu, Dong Xia

Maintainer <chenyu.ren@connect.polyu.hk>

Depends R (>= 4.2.0)

Description <https://github.com/ChenyuzZZ73/rMultiNet>

License MIT

Encoding UTF-8

LazyData true

RoxygenNote 7.2.2

Imports dbscan, ggplot2, glmnet, lpSolve, matlabr, Matrix, plotly,
rTensor, truncnorm

Config/pak/sysreqs cmake make libicu-dev libuv1-dev libssl-dev

Repository <https://chenyuzzz73.r-universe.dev>

Date/Publication 2022-11-30 11:35:26 UTC

RemoteUrl <https://github.com/chenyuzzz73/rmultinet>

RemoteRef HEAD

RemoteSha 0d922beb9abdf7255c65d0378c3097fea2cdf69

Contents

Community_cluster	2
Embedding_network	2
GenerateMMLSM	3
GenerateMMSBM	3
InitializationLSM	4
InitializationMMSBM	4
PowerIteration	5
ProjectedGD	5
SpecClustering	6

Community_cluster	<i>A function for clustering visuilation</i>
-------------------	--

Description

a method to visuilize the result of embedding

Usage

Community_cluster_km(embedding, type, cluster_number)

Arguments

- embedding the embedding result of layers or nodes
- type N for layer n for nodes
- cluster_number the number of cluster

Embedding_network	<i>A function for embedding visuilation</i>
-------------------	---

Description

a method to visuilize the result of embedding

Usage

Embedding_network(network_membership, L, paxis = 2)

Arguments

- network_membership
- L the number of network membership(n or L)
- paxis = 2 the number of axis to plot

GenerateMMLSM	<i>A function for generating a tensor</i>
---------------	---

Description

a method to generate a tensor for multilayer network

Usage

```
logi_fun(x, scale_par)
```

Arguments

scale_par	the scale parameter(default as 1)
n	number of nodes
M	types of networks
d	rank of U matrix
L	number of layers
U_mean	the mean of U with the Norm distribution, sd = 1
cmax	the upper bound of C tensor
int_type	way of generating C tensor included "Uniform" and "Norm"
kernel_fun	way of generating the adjacency tensor included "logit" and "probit"
Pi_network_type	size of network types (default as 1/m, an m-dimensional vector)

GenerateMMSBM	<i>A function for generating a tensor</i>
---------------	---

Description

a method to generate a tensor for multilayer network

Usage

```
generate_sbm(n, K, r, d, membership)
```

Arguments

n	the number of nodes
K	number of communities
r	out-in ratio (q/p, a L-dimensional vector)
d	average degree (a L-dimensional vector)
m	types of networks
L	number of layers

InitializationLSM *A function for initialization*

Description

A function for initialization

Usage

```
Init_UW(tensor_A, n, r, M, Utrue, Wtrue, pertub = 0.1, int_type)
```

Arguments

n	nodes
r	rank of U
M	the number of network types
pertub	the upper bound of Uniform distribution
int_type	way of initialize UW included "spec", "rand" and "warm"
gen_list	adj_tensor, Theta

Value

```
list(adj_tensor,U_int, W_int, delta)
```

InitializationMMSBM *A function for initialization*

Description

A function for initialization

Usage

```
InitializationMMSBM(tnsr, ranks = c(2, 2, 2))
```

Arguments

tnsr	the tensor of network
ranks=NULL	the rank of low_rank tensor

PowerIteration	<i>A function for tensor decomposition</i>
----------------	--

Description

A function for tensor decomposition

Usage

```
norm_vec(x)
```

Arguments

tnsr	adjacency tensor ($n*n*L$)
ranks	rank of core tensor
U_0_list	initialization
delta1	reguliazation for mode1 and mode2
delta2	reguliazation for mode3
max_iter	max times of iteration
tol	covergence tolerance

ProjectedGD	<i>A function for tensor decomposition</i>
-------------	--

Description

A function for tensor decomposition

Usage

```
reg_fun(U, delta)
```

Arguments

Ini_list	list(adj_tensor,U_int, W_int, delta) delta for incoherent parameters: (delta_1,delta_2,delta_3)
Cmax	upper bound
eta_outer	learning rate eta
tmax_outer	number of iterations in gradient descent
p_type	type of link function included "logit", "probit" and "poisson"
rd	if use stochastic sampling ("rand" or "Non")
scale_par	scale parameter (sigma in the paper)
sample_size	

SpecClustering	<i>A function for tensor decomposition</i>
----------------	--

Description

A function for tensor decomposition

Usage

```
SpecClustering(tnsr, rank, embedding_type = "Layer")
```

Arguments

matrx	an n by n matrix
ncol	number of columns of the output matrix U

Value

U n by rank numeric matrix that contains the rank tops eigenvectors of Laplacian matrix as column

Index

Community_cluster, [2](#)
Community_cluster_km
 (Community_cluster), [2](#)

Embedding_network, [2](#)

generate_sbm (GenerateMMSBM), [3](#)
GenerateMMLSM, [3](#)
GenerateMMSBM, [3](#)

Init_UW (InitializationLSM), [4](#)
InitializationLSM, [4](#)
InitializationMMSBM, [4](#)

logi_fun (GenerateMMLSM), [3](#)

norm_vec (PowerIteration), [5](#)

PowerIteration, [5](#)
ProjectedGD, [5](#)

reg_fun (ProjectedGD), [5](#)

SpecClustering, [6](#)