

Package ‘NetMix’

July 26, 2019

Type Package

Title Dynamic Mixed-Membership Network Regression Model

Version 0.1.4

Date 2019-07-11

Encoding UTF-8

Author Kosuke Imai [aut, cre],
Tyler Pratt [aut, cre],
Adeline Lo [aut, cre],
Santiago Olivella [aut, cre]

Maintainer Santiago Olivella <olivella@unc.edu>

Description Variational EM estimation of mixed-membership stochastic blockmodel for networks, incorporating node-level predictors of mixed-membership vectors, as well as dyad-level predictors. For networks observed over time, the model defines a hidden Markov process that allows the effects of node-level predictors to evolve in discrete, historical periods. In addition, the package offers a variety of utilities for exploring results of estimation, including tools for conducting posterior predictive checks of goodness-of-fit and several plotting functions. The package implements methods described in Olivella, Pratt and Imai (2019) "Dynamic Stochastic Blockmodel Regression for Social Networks: Application to International Conflicts", available at <http://santiagoolivella.info/wp-content/uploads/2018/07/dSBM_Reg.pdf>.

BugReports <https://github.com/solivella/NetMix/issues>

NeedsCompilation yes

License GPL (>= 2)

Depends R (>= 3.1.0)

SystemRequirements C++11

Suggests doParallel (>= 1.0.14), ergm (>= 3.9.4), foreach (>= 1.4.4),
ggplot2 (>= 3.1.1), network (>= 1.13)

Imports clue (>= 0.3-54), graphics (>= 3.5.2), grDevices (>= 3.5.2),
gtools (>= 3.8.1), igraph (>= 1.2.4.1), lda (>= 1.4.2), Matrix
(>= 1.2-15), methods (>= 3.5.2), Rcpp (>= 0.12.13), RSpectra
(>= 0.14-0), stats (>= 3.5.2), utils (>= 3.5.2)

LinkingTo Rcpp

RoxygenNote 6.1.1

Repository CRAN

Date/Publication 2019-07-26 09:10:02 UTC

R topics documented:

NetMix-package	2
.approxB	4
.mmsbm_fit	5
covFX	6
gof	7
head.mmsbm	9
lazega_dyadic	11
lazega_monadic	11
mmsbm	12
plot.mmsbm	15
predict.mmsbm	16
simulate.mmsbm	18
summary.mmsbm	19

NetMix-package

Dynamic Mixed-Membership Network Regression Model

Description

Variational EM estimation of mixed-membership stochastic blockmodel for networks, incorporating node-level predictors of mixed-membership vectors, as well as dyad-level predictors. For networks observed over time, the model defines a hidden Markov process that allows the effects of node-level predictors to evolve in discrete, historical periods. In addition, the package offers a variety of utilities for exploring results of estimation, including tools for conducting posterior predictive checks of goodness-of-fit and several plotting functions. The package implements methods described in Olivella, Pratt and Imai (2019) "Dynamic Stochastic Blockmodel Regression for Social Networks: Application to International Conflicts", available at <http://santiagoolivella.info/wp-content/uploads/2018/07/dSBM_Reg.pdf>.

Details

The DESCRIPTION file:

Package:	NetMix
Type:	Package
Title:	Dynamic Mixed-Membership Network Regression Model
Version:	0.1.4
Date:	2019-07-11
Encoding:	UTF-8

Author: Kosuke Imai [aut, cre], Tyler Pratt [aut, cre], Adeline Lo [aut, cre], Santiago Olivella [aut, cre]
 Maintainer: Santiago Olivella <olivella@unc.edu>
 Description: Variational EM estimation of mixed-membership stochastic blockmodel for networks, incorporating r
 BugReports: <https://github.com/solivella/NetMix/issues>
 NeedsCompilation: yes
 License: GPL (>= 2)
 Depends: R (>= 3.1.0)
 SystemRequirements: C++11
 Suggests: doParallel (>= 1.0.14), ergm (>= 3.9.4), foreach (>= 1.4.4), ggplot2 (>= 3.1.1), network (>= 1.13)
 Imports: clue (>= 0.3-54), graphics (>= 3.5.2), grDevices (>= 3.5.2), gtools (>= 3.8.1), igraph (>= 1.2.4.1), lda
 LinkingTo: Rcpp
 RoxygenNote: 6.1.1

Index of help topics:

<code>.approxB</code>	Internal functions and generics for 'mmsbm' package
<code>.mmsbm_fit</code>	Fitter Function for dynamic MMSBM Model
<code>NetMix-package</code>	Dynamic Mixed-Membership Network Regression Model
<code>covFX</code>	Generate estimated monadic covariate effects for estimated mmsbm model
<code>gof</code>	Posterior predictive checks using structural network characteristics
<code>head.mmsbm</code>	Identify nodes with most frequent membership in latent groups
<code>lazega_dyadic</code>	Dyadic predictors in the Lazega friendship network (Lazega 2001).
<code>lazega_monadic</code>	Monadic predictors in the Lazega friendship network (Lazega 2001).
<code>mmsbm</code>	Dynamic mixed-membership stochastic blockmodel with covariates
<code>plot.mmsbm</code>	Various visualization tools for 'mmsbm' objects
<code>predict.mmsbm</code>	Predict edges based on estimated mmsbm model
<code>simulate.mmsbm</code>	Simulate a complete sociomatrix from an 'mmsbm' object
<code>summary.mmsbm</code>	Summarize 'mmsbm' object

This section should provide a more detailed overview of how to use the package, including the most important functions.

Author(s)

Kosuke Imai [aut, cre], Tyler Pratt [aut, cre], Adeline Lo [aut, cre], Santiago Olivella [aut, cre]
 Maintainer: Santiago Olivella <olivella@unc.edu>

References

This optional section can contain literature or other references for background information.

See Also

Optional links to other man pages

Examples

```
## Optional simple examples of the most important functions
## Use \dontrun{} around code to be shown but not executed
```

.approxB

Internal functions and generics for mmsbm package

Description

These are various utilities and generic methods used by the main package function.

Usage

```
.approxB(y, d_id, pi_mat)

.getZ(pi_mat)

.cbind.fill(...)

.mpower(mat, p)

.findPerm(block.list, target.mat = NULL, use.perms = TRUE)

.transf(mat)

.pi.hat(X, beta)

.e.pi(pi_l, kappa)
```

Arguments

y	Numeric vector; vector of edge values.
d_id	Integer matrix; two-column matrix with nr. dyads rows, containing zero-based sender (first column) and receiver (second column) node id's for each dyad.
pi_mat	Numeric matrix; row-stochastic matrix of mixed-memberships.
...	Numeric vectors; vectors of potentially different length to be cbind-ed.
mat	Numeric matrix
p	Numeric scalar; power to raise matrix to.

<code>block.list</code>	List of matrices; each element is a square, numeric matrix that defines a block-model,
<code>target.mat</code>	Numeric matrix; reference blockmodel that those in <code>block.list</code> should be aligned to. Optional, defaults to <code>NULL</code> .
<code>use.perms</code>	Boolean; should all row/column permutations be explored when realigning matrices? defaults to <code>TRUE</code> .
<code>X</code>	Numeric matrix; design matrix of monadic predictors.
<code>beta</code>	Numeric array; array of coefficients associated with monadic predictors. It of dimensions Nr. Predictors by Nr. of Blocks by Nr. of HMM states.
<code>pi_l</code>	List of mixed-membership matrices.
<code>kappa</code>	Numeric matrix; matrix of marginal HMM state probabilities.

Details

These functions are meant for internal use only.

Value

See individual return section for each function:

.cbind.fill Matrix of `cbind`'ed elements in `. . .`, with missing values in each vector filled with `NA`.

.mpower Matrix; the result of raising `mat` to the `p` power.

.findPerm List of permuted blockmodel matrices

.transf Matrix with transformed mixed-membership vectors along its rows, s.t. no element is equal to 0.0 or 1.0.

.pi.hat List of predicted mixed-membership matrices, one element per HMM state.

.e.pi Matrix of expected mixed-membership vectors along its rows, with expectation computed over marginal distribution over HMM states for each time period.

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

`.mmsbm_fit`

Fitter Function for dynamic MMSBM Model

Description

This is the interface to the C++ fitter for the dynamic mixed-membership stochastic blockmodel for network regression.

Arguments

<code>z_t</code>	Numeric matrix; transpose of monadic design matrix. Should not include intercept row.
<code>x_t</code>	Numeric matrix; transpose of dyadic design matrix.
<code>y</code>	Numeric vector; vector of edge values. Must have same number of elements as <code>ncol(x_t)</code>
<code>time_id_dyad</code>	Integer vector; zero-based time-period identifier for each dyad.
<code>time_id_node</code>	Integer vector; zero-based time-period identifier for each node.
<code>nodes_per_period</code>	Integer vector; total number of unique nodes observed in each time period.
<code>node_id_dyad</code>	Integer matrix; zero-based sender and receiver identifier per dyad.
<code>mu_b</code>	Numeric matrix; matrix of prior means for elements in blockmodel matrix.
<code>var_b</code>	Numeric matrix; matrix of prior variances for elements in blockmodel matrix.
<code>phi_init</code>	Numeric matrix; matrix of initial mixed-memberships. Nodes along columns.
<code>kappa_init_t</code>	Numeric matrix; matrix of initial marginal HMM state probabilities. Time-periods along columns.
<code>b_init_t</code>	Numeric matrix; square matrix of initial values of blockmodel.
<code>beta_init</code>	Numeric vector; flat array (column-major order) of initial values of monadic coefficients.
<code>gamma_init</code>	Numeric vector; vector of initial values of dyadic coefficients
<code>control</code>	List; see the <code>mmsbm.control</code> argument of <code>mmsbm</code>

Value

Unclassed list with named components; see `Value` of `mmsbm`

Warning

This function is for internal use only. End-users should always resort to `mmsbm`. In particular, that interface post-processes the return value of this internal in important ways.

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

covFX	<i>Generate estimated monadic covariate effects for estimated mmsbm model</i>
-------	---

Description

The function estimates the effect of a shift in monadic covariate values on the probability of edge formation in the network.

Usage

```
covFX(fm, cov, shift, max.val = FALSE)
```

Arguments

fm	An object of class <code>mmsbm</code> , a result of a call to <code>mmsbm</code> .
cov	Character string identifying the monadic covariate to be shifted.
shift	Numeric value specifying the desired increase or decrease in the monadic covariate. The monadic predictor will be shifted by this value for all nodes and time periods.
max.val	An optional numeric value specifying the maximum possible value for the monadic covariate.

Value

List with named components:

Overall Avg. Effect Overall average effect of the covariate shift on the predicted probability of edge formation.

Avg. Effect by Time Vector of average effects of the covariate shift on the predicted probability of edge formation for each time period.

Avg. Effect by Node Vector of average effects of the covariate shift on the predicted probability of edge formation for each node.

Avg. Effect by Dyad Vector of average effects of the covariate shift on the predicted probability of edge formation for each node dyad.

Avg. Effect Dyad-Time Vector of estimated effects of the covariate shift on the predicted probability of edge formation for each node dyad-time unit.

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

Examples

```

library(NetMix)
## Load datasets
data("lazega_dyadic")
data("lazega_monadic")
## Estimate model with 2 groups
set.seed(123)
lazega_mmsbm <- mmsbm(SocializeWith ~ Coworkers,
                      ~ School + Practice + Age,
                      senderID = "Lawyer1",
                      receiverID = "Lawyer2",
                      nodeID = "Lawyer",
                      data.dyad = lazega_dyadic,
                      data.monad = lazega_monadic,
                      n.blocks = 2)

## Compute effect of decreasing every lawyers' age by 10 years
fx_list <- covFX(lazega_mmsbm, cov = "Age", shift = -10)
fx_list[["Overall Avg. Effect of Age"]]

```

gof

Posterior predictive checks using structural network characterics

Description

The function generates a variety of plots that serve as posterior predictive checks on the goodness of fit of a fitted `mmsbm` object.

Usage

```

gof(x, ...)

## S3 method for class 'mmsbm'
gof(x, gof_stat = c("Geodesics", "Degree"),
    level = 0.95, samples = 50, new.data.dyad = NULL,
    new.data.monad = NULL, parametric_mm = FALSE, seed = NULL, ...)

```

Arguments

<code>x</code>	An object of class <code>mmsbm</code> , a result of a call to <code>mmsbm</code> .
<code>...</code>	Currently ignored.
<code>gof_stat</code>	Character vector. Accepts any subset from "Geodesics", "Degree", "Indegree", "Outdegree", "3-Motifs", "Dyad Shared Partners", "Edge Shared Partners", and "Incoming K-stars". See details.
<code>level</code>	Double. Level of credible interval for posterior predictive distribution around structural quantities of interest.

`samples` Integer. Number of sampled networks from model's posterior predictive using `simulate.mmsbm`.

`new.data.dyad` See `simulate.mmsbm`. Enables out-of-sample checking.

`new.data.monad` See `simulate.mmsbm`. Enables out-of-sample checking.

`parametric_mm` See `simulate.mmsbm`.

`seed` See `simulate.mmsbm`.

Details

Goodness of fit of network models has typically been established by evaluating how the structural characteristics of predicted networks compare to those of the observed network. When estimated in a Bayesian framework, this approach is equivalent to conducting posterior predictive checks on these structural quantities of interest. When `new.data.dyad` and/or `new.data.monad` are passed that are different from those used in estimation, this is equivalent to conducting posterior predictive checks out-of-sample.

The set of structural features used to determine goodness of fit is somewhat arbitrary, and chosen mostly to incorporate various first order, second order, and (to the extent possible) third-order characteristics of the network. "Geodesics" focuses on the distribution over observed and predicted geodesic distances between nodes; "Indegree" and "Outdegree" focuses on the distribution over incoming and outgoing connections per node; "3-motifs" focus on a distribution over possible connectivity patterns between triads (i.e. the triadic census); "Dyad Shared Partners" focuses on the distribution over the number of shared partners between any two days; "Edge Shared Partners" is similarly defined, but w.r.t. edges, rather than dyads; and finally "Incoming K-stars" focuses on a frequency distribution over stars with $k=1, \dots$ spokes.

Obtaining samples of the last three structural features can be very computationally expensive, and is discouraged on networks with more than 50 nodes.

Value

A `ggplot` object.

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

Examples

```
library(NetMix)
## Load datasets
data("lazega_dyadic")
data("lazega_monadic")

## Estimate model with 2 groups
set.seed(123)
lazega_mmsbm <- mmsbm(SocializeWith ~ Coworkers,
                      senderID = "Lawyer1",
```

```

receiverID = "Lawyer2",
nodeID = "Lawyer",
data.dyad = lazega_dyadic,
data.monad = lazega_monadic,
n.blocks = 2)

## Plot observed (red) and simulated (gray) distributions over
## geodesic distances
## (typically a larger number of samples would be taken)
gof(lazega_mmsbm, gof_stat = "Geodesics", samples = 2)

```

head.mmsbm

Identify nodes with most frequent membership in latent groups

Description

The function lists the nodes (optionally, node-time periods) that most frequently instantiate membership in each latent group.

Usage

```

## S3 method for class 'mmsbm'
head(x, n = 6, t = NULL, node = TRUE,
      t.correct = FALSE, ...)

```

Arguments

x	An object of class <code>mmsbm</code> , a result of a call to <code>mmsbm</code> .
n	Numeric or integer; specifies how many units will be identified for each group.
t	Optional vector of time periods to be used for assessing latent group membership.
node	Logical; indicates whether latent group memberships should be averaged at the node level. If <code>FALSE</code> , the function returns the node-time period units with highest estimated membership in each latent group.
t.correct	Logical; indicates whether latent group memberships should be corrected for temporal trends. If <code>TRUE</code> , the function returns the node-time period units with highest estimated membership in each latent group.
...	Currently ignored

Value

List of length `n.groups`. Each entry contains a sorted vector of average latent membership probabilities of length `n`.

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

Examples

```

library(NetMix)
## Load datasets
data("lazega_dyadic")
data("lazega_monadic")
## Estimate model with 2 groups
set.seed(123)
lazega_mmsbm <- mmsbm(SocializeWith ~ Coworkers,
                      ~ School + Practice + Status,
                      senderID = "Lawyer1",
                      receiverID = "Lawyer2",
                      nodeID = "Lawyer",
                      data.dyad = lazega_dyadic,
                      data.monad = lazega_monadic,
                      n.blocks = 2)

## Show top 6 lawyers in each estimated latent block
head(lazega_mmsbm)

```

lazega_dyadic	<i>Dyadic predictors in the Lazega friendship network (Lazega 2001).</i>
---------------	--

Description

A dataset containing edges and dyad-level predictors in the network of friendships among lawyers in a New England law firm. More details are available in Lazega (2001).

Usage

```
data(lazega_dyadic)
```

Format

A data frame with 5041 rows and 4 variables:

Lawyer1, Lawyer2 lawyer ID, corresponding to identifiers common to those in `lazega_monadic`;
numeric

SocializeWith value of edge in network; binary

Coworkers are the corresponding lawyers in the same office? boolean

Source

```

https://github.com/Z-co/networkdata/blob/master/networkdata/data/lazega.
rda

```

References

Emmanuel Lazega, *The Collegial Phenomenon: The Social Mechanisms of Cooperation Among Peers in a Corporate Law Partnership*, Oxford University Press (2001).

lazega_monadic	<i>Monadic predictors in the Lazega friendship network (Lazega 2001).</i>
----------------	---

Description

A dataset containing vertex-level predictors in the network of sought-after advise among lawyers in a New England law firm. More details are available in Lazega (2001).

Usage

```
data(lazega_monadic)
```

Format

A data frame with 71 rows and 7 variables:

Lawyer lawyer ID, corresponding to identifiers common to those in `lazega_dyadic`; numeric

Age age, in years; numeric

Gender 1=man; 2=woman; factor

School 1=harvard, yale; 2=ucon; 3= other; factor

Practice 1=litigation; 2=corporate; factor

Seniority time in the firm, in years; numeric

Status 1=partner; 2=associate; factor

Source

Emmanuel Lazega, *The Collegial Phenomenon: The Social Mechanisms of Cooperation Among Peers in a Corporate Law Partnership*, Oxford University Press (2001).

<https://github.com/Z-co/networkdata/blob/master/networkdata/data/lazega.rda>

mmsbm	<i>Dynamic mixed-membership stochastic blockmodel with covariates</i>
-------	---

Description

The function estimates a dynamic mixed-membership stochastic blockmodel that incorporates covariates.

Usage

```
mmsbm(formula.dyad, formula.monad = ~1, senderID, receiverID,
       nodeID = NULL, timeID = NULL, data.dyad, data.monad = NULL,
       n.blocks, n.hmmstates = 1, directed = TRUE,
       missing = "indicator method", mmsbm.control = list())
```

Arguments

<code>formula.dyad</code>	A formula object. The variable in <code>data.dyad</code> that contains binary edges should be used as a LHS, and any dyadic predictors can be included on the RHS (when no dyadic covariates are available, use $y \sim 1$). Same syntax as a <code>glm</code> formula.
<code>formula.monad</code>	An optional formula object. LHS is ignored. RHS contains names of nodal attributes found in <code>data.monad</code> .
<code>senderID</code>	Character string. Quoted name of the variable in <code>data.dyad</code> identifying the sender node. For undirected networks, the variable simply contains name of first node in dyad. Cannot contain special charecter "@".
<code>receiverID</code>	Character string. Quoted name of the variable in <code>data.dyad</code> identifying the receiver node. For undirected networks, the variable simply contains name of second node in dyad. Cannot contain special charecter "@".
<code>nodeID</code>	Character string. Quoted name of the variable in <code>data.monad</code> identifying a node in either <code>data.dyad[, senderID]</code> or <code>data.dyad[, senderID]</code> . If not <code>NULL</code> , every node <code>data.dyad[, senderID]</code> or <code>data.dyad[, senderID]</code> must be present in <code>data.monad[, nodeID]</code> . Cannot contain special charecter "@".
<code>timeID</code>	Character string. Quoted name of the variable in both <code>data.dyad</code> and <code>data.monad</code> indicating the time in which network (and correspdng nodal attributes) were observed. The variable itself must be composed of integers. Cannot contain special charecter "@".
<code>data.dyad</code>	Data frame. Sociomatrix in "long" (i.e. dyadic) format. Must contain at least three variables: the sender identifier (or identifier of the first node in an undirected networks dyad), the receiver identifier (or identifier of the second node in an undirected network dyad), and the value of the edge between them. Currently, only edges between zero and one (inclusive) are supported.
<code>data.monad</code>	Data frame. Nodal atributes. Must contain a node identifier matching the names of nodes used in the <code>data.dyad</code> data frame.
<code>n.blocks</code>	Integer value. How many latent groups should be used to estimate the model?
<code>n.hmmstates</code>	Integer value. How many hidden Markov state should be used in the HMM? Defaults to 1 (i.e. no HMM).
<code>directed</code>	Boolean. Is the network directed? Defaults to <code>TRUE</code> .
<code>missing</code>	Means of handling missing data. One of "indicator method" (default) or "list-wise deletion".
<code>mmsbm.control</code>	A named list of optional algorithm control parameters. <ul style="list-style-type: none"> spectral Boolean. Type of initialization algorithm for mixed-membership vectors in static case. If <code>TRUE</code> (default), use spectral clustering with degree correction; otherwise, use <code>kmeans</code> algorithm. init.dyn.gibbs Boolean. Should a collapsed Gibbs sampler of non-regression <code>mmsbm</code> be used to initialize each time period when multiple time periods are observed (instead of a spectral or simple <code>kmeans</code> initialization)? Setting to <code>TRUE</code> will be result in faster estimation that is very sensitive to choice of <code>alpha</code> (see below)

- alpha** Numeric positive value. Concentration parameter for collapsed Gibbs sampler to find initial mixed-membership values in dynamic case when `init.dyn.gibbs=TRUE`. Defaults to 0.5
- seed** RNG seed. Defaults to NULL, which does not seed the RNG
- em_iter** Number of maximum iterations in variational EM. Defaults to 5e3
- opt_iter** Number of maximum iterations of BFGS in M-step. Defaults to 10e3
- mu_b** Numeric vector with two elements: prior mean of blockmodel's main diagonal elements, and and prior mean of blockmodel's offdiagonal elements. Defaults to `c(5.0, -5.0)`
- var_b** Numeric vector with two positive elements: prior variance of blockmodel's main diagonal elements, and and prior variance of blockmodel's offdiagonal elements. Defaults to `c(1.0, 1.0)`
- var_beta** Numeric positive value. (Gaussian) Prior variance of monadic coefficients. Defaults to 5.0.
- var_gamma** Numeric positive value. (Gaussian) Prior variance of dyadic coefficients. Defaults to 5.0.
- eta** Numeric positive value. Concentration hyper-parameter for HMM. Defaults to 10.3
- phi_init_t** Matrix, `n.blocks` by total number of nodes across years. Optional initial values for variational parameters for mixed-membership vectors. Column names must be of the form `nodeid\@year`
- kappa_init_t** Matrix, `n.hmmstates` by number of years. Optional initial values for variational parameters for state probabilities.
- b_init_t** Matrix, `n.blocks` by `n.blocks`. Optional initial values for blockmodel.
- beta_init** Array, predictors by `n.blocks` by `n.hmmstates`. Optional initial values for monadic coefficients.
- gamma_init** Vector. Optional initial values for dyadic coefficients.
- permute** Boolean. Should all permutations be tested to realign initial block models in dynamic case? If FALSE, realignment is done via faster graph matching algorithm, but may not be exact. Defaults to TRUE.
- threads** Numeric integer. Number of available cores for parallization. Defaults to 4
- conv_tol** Numeric value. Absolute tolerance for VI convergence. Defaults to 1e-4
- verbose** Boolean. Should extra information be printed as model iterates? Defaults to FALSE

Value

Object of class `mmsbm`. List with named components:

MixedMembership Matrix of variational posterior of mean of mixed-membership vectors. `nodes` by `n.groups`

BlockModel `n.groups` by `n.groups` matrix of estimated tie log-odds between members of corresponding latent groups. The blockmodel.

MonadCoef Array of estimated coefficient values for monadic covariates. Has `n.groups` columns, and `n.hmmstates` slices.

DyadCoef Vector estimated coefficient values for dyadic covariates

TransitionKernel Matrix of estimated HMM transition probabilities

Kappa Matrix of marginal probabilities of being in an HMM state at any given point in time. `n.hmmstates` by years (or whatever time interval networks are observed at)

LowerBound Value of the lower bound at final iteration

niter Final number of VI iterations

converged Convergence indicator; zero indicates failure to converge

NodeIndex Order in which nodes are stored in all return objects

monadic.data, dyadic.data, directed Original values of parameters used during estimation

forms Values of formal arguments passed in original function call

call Original (unevaluated) call

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

Examples

```
library(NetMix)
## Load datasets
data("lazega_dyadic")
data("lazega_monadic")
## Estimate model with 2 groups
set.seed(123)
lazega_mmsbm <- mmsbm(SocializeWith ~ Coworkers,
                     ~ School + Practice + Status,
                     senderID = "Lawyer1",
                     receiverID = "Lawyer2",
                     nodeID = "Lawyer",
                     data.dyad = lazega_dyadic,
                     data.monad = lazega_monadic,
                     n.blocks = 2)
```

plot.mmsbm

Various visualization tools for 'mmsbm' objects

Description

The function provides a variety of plotting options for a fitted `mmsbm` object.

Usage

```
## S3 method for class 'mmsbm'
plot(x, type = "groups", FX = NULL, ...)
```

Arguments

x	An object of class <code>mmsbm</code> , a result of a call to <code>mmsbm</code> .
type	character string denoting the type of plot. The default, "groups," plots the estimated matrix of group by group edge formation probabilities as a network plot, with nodes representing groups (sized proportional to relative membership) and edge colors encoding probability of between-group ties. "blockmodel" plots the same information, but using a tile plot instead of a network plot. "membership" plots average membership in each latent group by time period. "effect" provides a series of plots showing the estimated effect of a shift in monadic covariate values.
FX	with <code>type == "effect"</code> ; a list resulting from a call to <code>covFX</code> .
...	Currently ignored

Value

The requested plot object.

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

Examples

```
library(NetMix)
## Load datasets
data("lazega_dyadic")
data("lazega_monadic")
## Estimate model with 2 groups
set.seed(123)
lazega_mmsbm <- mmsbm(SocializeWith ~ Coworkers,
                     ~ School + Practice + Status,
                     senderID = "Lawyer1",
                     receiverID = "Lawyer2",
                     nodeID = "Lawyer",
                     data.dyad = lazega_dyadic,
                     data.monad = lazega_monadic,
                     n.blocks = 2)

## Plot blockmodel as network
plot(lazega_mmsbm)
```


Description

The function produces expected posterior edges based on estimated parameters and (optionally new) predictor data

Usage

```
## S3 method for class 'mmsbm'
predict(object, new.data.dyad = NULL,
        new.data.monad = NULL, parametric_mm = FALSE, forecast = FALSE,
        type = c("link", "response"), ...)
```

Arguments

object	Object of class mmsbm.
new.data.dyad	An optional data.frame object.
new.data.monad	An optional data.frame object.
parametric_mm	boolean. Should the variational posterior be used for sampling the mixed-memberships (FALSE), or should the mixed-memberships be formed using the parameters in the monadic regression equation (TRUE)? Defaults to FALSE. If (new.data.monad forecast) = TRUE, setting this to FALSE will produce an error.
forecast	Boolean. Should prediction forecast one step into the future? Defaults to FALSE.
type	Character string. The default is to use the linear predictor of edges. The alternative "response" returns predicted probabilities.
...	Currently ignored

Value

If new.data.dyad = NULL, vector of length nrow(object\$dyadic.data). Else, vector of length nrow(new.data.dyad).

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

Examples

```
library(NetMix)
## Load datasets
data("lazega_dyadic")
data("lazega_monadic")
## Estimate model with 2 groups
set.seed(123)
lazega_mmsbm <- mmsbm(SocializeWith ~ Coworkers,
                     ~ School + Practice + Status,
                     senderID = "Lawyer1",
```

```

receiverID = "Lawyer2",
nodeID = "Lawyer",
data.dyad = lazega_dyadic,
data.monad = lazega_monadic,
n.blocks = 2)

## Get in-sample predicted edge probabilities
lazega_preds <- predict(lazega_mmsbm, type = "response")

```

simulate.mmsbm *Simulate a complete sociomatrix from an mmsbm object*

Description

The function generates one sample network from the posterior predictive of the model represented by a fitted `mmsbm` object.

Usage

```

## S3 method for class 'mmsbm'
simulate(object, nsim = 1, seed = NULL,
         new.data.dyad = NULL, new.data.monad = NULL, parametric_mm = FALSE,
         ...)

```

Arguments

<code>object</code>	An object of class <code>mmsbm</code> , a result of a call to <code>mmsbm</code>
<code>nsim</code>	Number of networks to simulate
<code>seed</code>	RNG seed.
<code>new.data.dyad</code>	An optional <code>data.frame</code> object. If not <code>NULL</code> , use these dyadic predictor values instead of those used to fit the original model.
<code>new.data.monad</code>	An optional <code>data.frame</code> object. See <code>new.data.dyad</code> .
<code>parametric_mm</code>	Boolean. Should the variational posterior be used for sampling the mixed-memberships (<code>FALSE</code>), or should the mixed-memberships be formed using the parameters in the monadic regression equation (<code>TRUE</code>)? Defaults to <code>FALSE</code> . If <code>is.null(new.data.monad)=FALSE</code> , setting this to <code>FALSE</code> will produce an error.
<code>...</code>	Currently ignored

Value

List of length `nsim` of simulated networks. If `new.data.dyad = NULL`, each element is a vector of length `nrow(object$dyadic.data)`. Else, vector of length `nrow(new.data.dyad)`. If `seed` is not `NULL`, return object includes its value as attribute `"seed"`.

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

Examples

```
library(NetMix)
## Load datasets
data("lazega_dyadic")
data("lazega_monadic")
## Estimate model with 2 groups
set.seed(123)
lazega_mmsbm <- mmsbm(SocializeWith ~ Coworkers,
                     ~ School + Practice + Status,
                     senderID = "Lawyer1",
                     receiverID = "Lawyer2",
                     nodeID = "Lawyer",
                     data.dyad = lazega_dyadic,
                     data.monad = lazega_monadic,
                     n.blocks = 2)

## Simulate 5 new networks
lazega_sim <- simulate(lazega_mmsbm, nsim = 5, seed = 123)
```

summary.mmsbm *Summarize 'mmsbm' object*

Description

The function summarizes the output of a dynMMSBM model object

Usage

```
## S3 method for class 'mmsbm'
summary(object, ...)
```

Arguments

object	An object of class mmsbm, a result of a call to mmsbm.
...	Currently ignored

Value

List with named components:

N Total number of dyad-time period observations.

Number of Clusters Number of latent groups included in the dynMMSBM model.

Percent of Observations in Each Cluster Average membership in each latent group, across all node-time periods.

Edge Formation Probabilities `n.groups` by `n.groups` matrix of estimated edge formation probabilities between latent groups.

Dyadic Coefficients Vector of estimated coefficient values for dyadic covariates.

Monadic Coefficients Array of estimated coefficient values for monadic covariates. Has `n.groups` columns, and `n.hmmstates` slices.

Markov State Probabilities Average HMM state probabilities across all time periods.

Author(s)

Kosuke Imai (imai@harvard.edu), Tyler Pratt (tyler.pratt@yale.edu), Santiago Olivella (olivella@unc.edu)

Examples

```
library(NetMix)
## Load datasets
data("lazega_dyadic")
data("lazega_monadic")
## Estimate model with 2 groups
set.seed(123)
lazega_mmsbm <- mmsbm(SocializeWith ~ Coworkers,
                     ~ School + Practice + Status,
                     senderID = "Lawyer1",
                     receiverID = "Lawyer2",
                     nodeID = "Lawyer",
                     data.dyad = lazega_dyadic,
                     data.monad = lazega_monadic,
                     n.blocks = 2)

## Summarize estimated model
summary(lazega_mmsbm)
```