Package ‘dyads’

April 16, 2021

Type Package

Title Dyadic Network Analysis

Version 1.1.4

Date 2021-04-16

Author Bonne J.H. Zijlstra <B.J.H.Zijlstra@uva.nl>

Maintainer Bonne J.H. Zijlstra <B.J.H.Zijlstra@uva.nl>

Depends R (>= 3.0.0)

Imports stats, CholWishart, MASS, RcppZiggurat, Rfast, mvtnorm

Suggests plyr

Description Contains functions for the MCMC simulation of dyadic network models j2 (Zijlstra, 2017, <doi:10.1080/0022250X.2017.1387858>) and p2 (Van Duijn, Snijders & Zijlstra, 2004, <doi: 10.1046/j.0039-0402.2003.00258.x>), and the bidirectional (multilevel) counterpart of the the multilevel p2 model as described in Zijlstra, Van Duijn & Snijders (2009) <doi: 10.1348/000711007X255336>, the (multilevel) b2 model.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2021-04-16 10:10:02 UTC

R topics documented:

dyads-package *dyads*

Description

Package for Dyadic Network Analysis.

Details

Package:	dyads
Type:	Package
Title:	Dyadic Network Analysis
Version:	1.1.4
Date:	2021-04-16
Author:	Bonne J.H. Zijlstra <B.J.H.Zijlstra@uva.nl>
Maintainer:	Bonne J.H. Zijlstra <B.J.H.Zijlstra@uva.nl>
Depends:	R (>= 3.0.0)
Imports:	stats, CholWishart, MASS, RcppZiggurat, Rfast, mvtnorm
Suggests:	plyr
Description:	Contains functions for the MCMC simulation of dyadic network models j2 (Zijlstra, 2017, <doi:10.1080/00222
License:	GPL (>= 2)

Index of help topics:

```
b2ML                    MCMC estimates for the (multilevel) b2 model
dyads-package           dyads
j2                      MCMC estimates for the j2 model
p2                      MCMC estimates for the p2 model
```

Includes functions for estimation of the p2 model (van Duijn, Snijders and Zijlstra (2004) <doi:10.1046/j.0039-0402.2003.00258.x>), more specifically the adaptive random walk algorithm (Zijlstra, van Duijn and Snijders (2009) <doi:10.1348/000711007X255336>), for the estimation of the j2 model (Zijlstra (2017) <doi:10.1080/0022250X.2017.1387858>), and for their bidirectional counterpart, b2.

Author(s)

Bonne J.H. Zijlstra Maintainer: Bonne J.H. Zijlstra <B.J.H.Zijlstra@uva.nl>

References

Zijlstra, B.J.H., Duijn, M.A.J. van, and Snijders, T.A.B. (2009). MCMC estimation for the $p_2$ network regression model with crossed random effects. *British Journal of Mathematical and Statistical Psychology*, 62, 143-166. Zijlstra, B.J.H. (2017). Regression of directed graphs on independent effects for density and reciprocity. *Journal of Mathematical Sociology*, 41(4), 185-192.

Examples

```
# create a very small network with covariates for illustrative purposes
S <- c(1,0,1,0,1,1,0,1,0,1)
REC <- (S*-1)+1
D1 <- matrix(c(0,1,0,1,0,1,0,1,0,1,
              0,0,0,1,0,1,0,1,0,1,
              1,1,0,0,1,0,0,0,0,0,
              1,1,1,0,1,0,0,0,0,1,
              1,0,1,0,0,1,1,0,1,0,
              0,0,0,0,0,0,1,1,1,1,
              0,0,0,0,0,1,0,1,0,1,
              1,0,0,0,0,1,1,0,1,1,
              0,1,0,1,0,1,0,1,0,0,
              1,0,1,1,1,0,0,0,0,0), ncol=10)
D2 <- abs(matrix(rep(S,10), byrow = FALSE, ncol= 10) -
            matrix(rep(REC,10), byrow = TRUE, ncol= 10))
R <- D1*t(D1)
Y <- matrix(c(0,1,1,1,1,1,0,0,1,1,
              0,0,0,1,1,1,0,0,1,0,
              1,1,0,1,1,1,0,0,1,1,
              1,1,1,0,1,1,0,1,1,0,
              1,1,1,1,0,1,1,0,1,1,
              0,1,1,1,1,0,1,1,1,0,
              1,0,1,0,1,1,0,1,0,1,
              0,1,1,1,0,1,1,0,1,1,
              1,0,1,0,1,0,1,1,0,1,
              1,1,1,0,0,1,1,1,1,0), ncol=10)

# estimate p2 model
p2(Y,sender= ~ S, receiver =  ~ REC, density = ~ D1 + D2, reciprocity= ~ R,
   burnin = 100, sample = 400, adapt = 10)
# Notice: burn-in, sample size and number of adaptive sequenses are
# much smaller than recommended to keep computation time low.
# recommended code:

p2(Y,sender= ~ S, receiver =  ~ REC, density = ~ D1 + D2, reciprocity= ~ R)
```

---

b2ML *MCMC estimates for the (multilevel) b2 model*

Description

Estimates the (multilevel) b2 model parameters, which is the bidirectional counterpart of the multilevel p2 model as described in Zijlstra, Van Duijn and Snijders (2006) <doi: 10.1027/1614-2241.2.1.42>.

Usage

```
b2ML(nets, actor = NULL, density = NULL, adapt = NULL, burnin = NULL, center = NULL,
separate= NULL, densVar = NULL, seed = NULL)
```

Arguments

`nets` List with n dichotomous symmetric dependent networks.

`actor` Optional matrix with a stacked actor covariate, corresponding to the n networks. Multiple actor covariates can be added as a formula object, see example below

`density` Optional matrix with symmetric a stacked density covariate, with dimensions similar to the n dependent networks. Multiple density covariates can be added as a formula object, see example below

`adapt` Optional number of adaptive sequenses (default is 100).

`burnin` Optional specification of number of burn-in iterations (default is 5000).

`center` Optional argument for centering predictors (default is TRUE).

`separate` Optional argument for estimating separate coefficients for the n dependent networks (default is FALSE).

`densVar` Optional argument for estimating densty variance at the network level (default is TRUE).

`seed` Optonal specification of random seed (delfault is 1).

Value

Returns a fitted model of class b2ML, to be opened with the function summary().

Author(s)

Bonne J.H. Zijlstra <b.j.h.zijlstra@uva.nl>

References

Zijlstra, B. J., Van Duijn, M. A., & Snijders, T. A. (2006). The Multilevel p2 Model A random effects model for the analysis of multiple social networks. *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences*, 2(1), 42.

Examples

```
# create two very small networks with covariates for illustrative purposes
Y1 <- matrix(c( 0,1,1,1,1,1,1,1,1,0,
                1,0,1,0,1,1,1,1,1,1,
                1,1,0,0,1,1,1,1,0,1,
                1,0,0,0,1,0,0,1,0,0,
                1,1,1,1,0,1,1,0,1,1,
                1,1,1,0,1,0,1,0,1,1,
                1,1,1,0,1,1,0,1,1,1,
                1,1,1,1,0,0,1,0,0,1,
                1,1,0,0,1,1,1,0,0,1,
                0,1,1,0,1,1,1,1,1,0), ncol=10)
Y2 <- matrix(c( 0,0,1,0,1,1,0,1,0,0,
                0,0,0,0,0,0,0,1,1,0,
                1,0,0,1,0,1,0,1,0,0,
                0,0,1,0,0,0,1,1,0,0,
                1,0,0,0,0,0,1,1,0,0,
                1,0,1,0,0,0,1,1,0,0,
                0,0,0,1,1,1,0,1,0,0,
                1,1,1,1,1,1,1,0,0,1,
                0,1,0,0,0,0,0,0,0,0,
                0,0,0,0,0,0,0,1,0,0), ncol=10)
Y <- list(Y1, Y2)
Aa1 <- c(1,0,1,0,1,1,0,1,0,1)
Aa2 <- c(1,0,0,1,0,0,1,1,0,1)
Aa <- list(Aa1, Aa2)
Aat <- do.call(plyr::rbind.fill.matrix, Aa)
Ab1 <- c(0,0,0,0,0,0,0,0,0,0)
Ab2 <- c(1,1,1,1,1,1,1,1,1,1)
Ab <- list(Ab1, Ab2)
Abt <- do.call(plyr::rbind.fill.matrix, Ab)
Da1 <- abs(matrix(rep(Aa1,10), byrow = FALSE, ncol= 10) -
            matrix(rep(Aa1,10), byrow = TRUE, ncol= 10))
Da2 <- abs(matrix(rep(Aa2,10), byrow = FALSE, ncol= 10) -
            matrix(rep(Aa2,10), byrow = TRUE, ncol= 10))
Da <- list(Da1, Da2)
Dat <- do.call(plyr::rbind.fill.matrix, Da)

# estimate b2 model for two networks
M1 <- b2ML(Y,actor= ~ Aat + Abt, density = ~ Dat, adapt = 10, burnin = 100, densVar = FALSE)
summary(M1)
# Notice: burn-in, and number of adaptive sequenses are
# much smaller than recommended to keep computation time low.
# recommended code:

M1 <- b2ML(Y,actor= ~ Aat + Abt, density = ~ Dat, densVar = FALSE)
summary(M1)


# estimate b2 model for a single network
M2 <- b2ML(list(Y[[1]]),actor= ~ Aat[1:10,], density = ~ Dat[1:10,], adapt = 10, burnin = 100,
densVar = FALSE)
summary(M2)
# Notice: burn-in, and number of adaptive sequenses are
# much smaller than recommended to keep computation time low.
# recommended code:

M2 <- b2ML(list(Y[[1]]),actor= ~ Aat[1:10,], density = ~ Dat[1:10,], densVar = FALSE)
summary(M2)
```

j2 *MCMC estimates for the j2 model*

Description

Estimates j2 model parameters as described in Zijlstra (in press) <doi:10.1080/0022250X.2017.1387858>.

Usage

```
j2(net, sender = NULL, receiver = NULL , density = NULL, reciprocity = NULL,
burnin = NULL, sample = NULL, adapt= NULL, center = NULL, seed = NULL)
```

Arguments

`net`	Directed dichotomous n*n network (digraph).
`sender`	Optional sender covariates of lenght n.
`receiver`	Optinal receiver covariates of length n.
`density`	Optional density covariates of dimensions n*n.
`reciprocity`	Optional symmetric reciprocity covariates of dimensions n*n.
`burnin`	Optional specification of number of burn-in iterations (default is 10000).
`sample`	Optional specification of number of MCMC samples (default is 40000).
`adapt`	Optional number of adaptive sequenses (default is 100).
`center`	Optional boolean argument for centering predictors (default is TRUE).
`seed`	Optonal specification of random seed (delfault is 1).

Value

Returns a matrix with MCMC means, standard deviations, quantiles and effective sample sizes for j2 parameters.

Author(s)

Bonne J.H. Zijlstra <`b.j.h.zijlstra@uva.nl`>

References

Zijlstra, B.J.H. (in press). Regression of directed graphs on independent effects for density and reciprocity. *Journal of Mathematical Sociology*.

Examples

```
# create a very small network with covariates for illustrative purposes
S <- c(1,0,1,0,1,1,0,1,0,1)
REC <- c(0,0,1,1,0,0,1,1,0,0)
D1 <- matrix(c(0,1,0,1,0,1,0,1,0,0,
               0,0,1,1,0,1,0,1,0,1,
               1,1,0,0,1,0,0,0,0,0,
               1,1,1,0,1,0,0,0,0,1,
               1,0,1,0,0,1,1,0,1,1,
               0,0,0,0,0,0,1,1,1,1,
               0,0,0,0,0,1,0,1,0,1,
               1,0,0,0,0,1,1,0,1,1,
               0,1,0,1,0,1,0,1,0,0,
               0,0,1,1,1,0,0,0,0,0), ncol=10)
D2 <- abs(matrix(rep(S,10), byrow = FALSE, ncol= 10) -
            matrix(rep(REC,10), byrow = TRUE, ncol= 10))
R <- D1*t(D1)
Y <- matrix(c(0,0,1,1,1,1,0,0,1,1,
              0,0,0,1,1,1,0,0,1,0,
              1,1,0,1,1,1,0,0,1,1,
              0,1,1,0,1,1,0,1,1,0,
              1,1,1,1,0,1,1,0,1,1,
              0,1,1,1,1,0,1,1,1,0,
              1,0,1,0,1,1,0,1,0,1,
              0,1,1,1,0,1,1,0,1,1,
              1,0,1,0,1,0,1,1,0,1,
              1,1,1,0,0,1,1,1,1,0), ncol=10)

# estimate j2 model
j2(Y,sender= ~ S, receiver =  ~ REC, density = ~ D1 + D2, reciprocity= ~ R,
   burnin = 100, sample = 400, adapt = 10)
# notice: burn-in, sample size and number of adaptive sequenses are
# much smaller than recommended to keep computation time low.
# recommended code:

j2(Y,sender= ~ S, receiver =  ~ REC, density = ~ D1 + D2, reciprocity= ~ R)
```

---

p2 *MCMC estimates for the p2 model*

Description

Estimates p2 model parameters with the adaptive random walk algorithm as described in Zijlstra, Van Duijn and Snijders (2009) <doi: 10.1348/000711007X255336>.

Usage

```
p2(net, sender = NULL, receiver = NULL, density = NULL, reciprocity = NULL,
burnin = NULL, sample = NULL, adapt = NULL, seed = NULL)
```

Arguments

net Directed dichotomous n*n network (digraph).

sender Optional sender covariates of lenght n.

receiver Optinal receiver covariates of length n.

density Optional density covariates of dimensions n*n.

reciprocity Optional symmetric reciprocity covariates of dimensions n*n.

burnin Optional specification of number of burn-in iterations (default is 10000).

sample Optional specification of number of MCMC samples (default is 40000).

adapt Optional number of adaptive sequenses (default is 100).

seed Optonal specification of random seed (delfault is 1).

Value

Returns a matrix with MCMC means, standard deviations, quantiles and estimated effective sample sizes for p2 parameters.

Author(s)

Bonne J.H. Zijlstra <b.j.h.zijlstra@uva.nl>

References

Zijlstra, B.J.H., Duijn, M.A.J. van, and Snijders, T.A.B. (2009). MCMC estimation for the $p_2$ network regression model with crossed random effects. *British Journal of Mathematical and Statistical Psychology*, 62, 143-166.

Examples

```
# create a very small network with covariates for illustrative purposes
S <- c(1,0,1,0,1,1,0,1,0,1)
REC <- (S*-1)+1
D1 <- matrix(c(0,1,0,1,0,1,0,1,0,1,
              0,0,0,1,0,1,0,1,0,1,
              1,1,0,0,1,0,0,0,0,0,
              1,1,1,0,1,0,0,0,0,1,
              1,0,1,0,0,1,1,0,1,0,
              0,0,0,0,0,0,1,1,1,1,
              0,0,0,0,0,1,0,1,0,1,
              1,0,0,0,0,1,1,0,1,1,
              0,1,0,1,0,1,0,1,0,0,
              1,0,1,1,1,0,0,0,0,0), ncol=10)
D2 <- abs(matrix(rep(S,10), byrow = FALSE, ncol= 10) -
            matrix(rep(REC,10), byrow = TRUE, ncol= 10))
R <- D1*t(D1)
Y <- matrix(c(0,1,1,1,1,1,0,0,1,1,
              0,0,0,1,1,1,0,0,1,0,
              1,1,0,1,1,1,0,0,1,1,
              1,1,1,0,1,1,0,1,1,0,
              1,1,1,1,0,1,1,0,1,1,
              0,1,1,1,1,0,1,1,1,0,
              1,0,1,0,1,1,0,1,0,1,
              0,1,1,1,0,1,1,0,1,1,
              1,0,1,0,1,0,1,1,0,1,
              1,1,1,0,0,1,1,1,1,0), ncol=10)

# estimate p2 model
p2(Y,sender= ~ S, receiver =  ~ REC, density = ~ D1 + D2, reciprocity= ~ R,
   burnin = 100, sample = 400, adapt = 10)
# Notice: burn-in, sample size and number of adaptive sequenses are
# much smaller than recommended to keep computation time low.
# recommended code:

p2(Y,sender= ~ S, receiver =  ~ REC, density = ~ D1+ D2, reciprocity= ~ R)
```

Index