Package ‘amen’

May 25, 2017

Title Additive and Multiplicative Effects Models for Networks and Relational Data

Version 1.3

Author Peter Hoff, Bailey Fosdick, Alex Volfovsky, Yanjun He

Description Analysis of dyadic network and relational data using additive and multiplicative effects (AME) models. The basic model includes regression terms, the covariance structure of the social relations model (Warner, Kenny and Stoto (1979) <DOI:10.1037/0022-3514.37.10.1742>, Wong (1982) <DOI:10.2307/2287296>), and multiplicative factor models (Hoff(2009) <DOI:10.1007/s10588-008-9040-4>). Four different link functions accommodate different relational data structures, including binary/network data (bin), normal relational data (nrm), ordinal relational data (ord) and data from fixed-rank nomination schemes (frn). Several of these link functions are discussed in Hoff, Fosdick, Volfovsky and Stovel (2013) <DOI:10.1017/nws.2013.17>. Development of this software was supported in part by NIH grant R01HD067509.

Maintainer Peter Hoff <peter.hoff@duke.edu>

License GPL-3

Date 2017-05-23

LazyData true

Depends R (>= 2.10)

Suggests coda,network,knitr

VignetteBuilder knitr

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2017-05-25 00:31:52 UTC

R topics documented:

`amen-package` *Additive and Multiplicative Effects Models for Networks and Relational Data*

Description

Analysis of network and relational data using additive and multiplicative effects (AME) models. The basic model includes regression terms, the covariance structure of the social relations model (Warner, Kenny and Stoto (1979), Wong (1982)), and multiplicative factor effects (Hoff(2009)). Four different link functions accommodate different relational data structures, including binary/network data (bin), normal relational data (nrm), ordinal relational data (ord) and data from fixed-rank nomination schemes (frn). Several of these link functions are discussed in Hoff, Fosdick, Volfovsky and Stovel (2013). Development of this software was supported in part by NICHD grant R01HD067509.

Details

Package:	amen
Type:	Package
Version:	1.3
Date:	2017-05-23
License:	GPL-3

Author(s)

Peter Hoff, Bailey Fosdick, Alex Volfovsky, Yanjun He

Maintainer: Peter Hoff <peter.hoff@duke.edu>

Examples

```
data(YX_frn)
fit<-ame(YX_frn$Y,YX_frn$X,burn=5,nscan=5,odens=1,model="frn")

summary(fit)

plot(fit)
```

---

addhealthc3 *AddHealth community 3 data*

Description

A valued sociomatrix (Y) and matrix of nodal attributes (X) for students in community 3 of the AddHealth study.

- Y: A sociomatrix in which the value of the edge corresponds to an ad-hoc measure of intensity of the relation. Note that students were only allowed to nominate up to 5 male friends and 5 female friends.
- X: Matrix of students attributes, including sex, race (1=white, 2=black, 3=hispanic, 4=asian, 5=mixed/other) and grade.

Usage

```
data(addhealthc3)
```

Format

list

---

addhealthc9 *AddHealth community 9 data*

Description

A valued sociomatrix (Y) and matrix of nodal attributes (X) for students in community 9 of the AddHealth study.

- Y: A sociomatrix in which the value of the edge corresponds to an ad-hoc measure of intensity of the relation. Note that students were only allowed to nominate up to 5 male friends and 5 female friends.
- X: Matrix of students attributes, including sex, race (1=white, 2=black, 3=hispanic, 4=asian, 5=mixed/other) and grade.

Usage

```
data(addhealthc9)
```

Format

list

---

addlines *Add lines*

Description

Add lines to a network plot

Usage

```
addlines(Y,X,col="lightblue",alength=0,...)
```

Arguments

Y	a sociomatrix
X	coordinates of nodes
col	color of lines. Can be a vector of length equal to the number of edges to be drawn
alength	length of arrows to be drawn
...	additional plotting parameters

Author(s)

Peter Hoff

Examples

```
data(addhealthc3)
Y<-addhealthc3$Y
X<-xnet(Y)
netplot(Y,X)
addlines(Y,X,col=Y[Y!=0])
```

ame *AME model fitting routine*

Description

An MCMC routine providing a fit to an additive and multiplicative effects (AME) regression model to relational data of various types

Usage

```
ame(Y, Xdyad=NULL, Xrow=NULL, Xcol=NULL, rvar = !(model=="rrl") ,
cvar = TRUE,  dcor = !symmetric, nvar=TRUE, R = 0, model="nrm",
intercept=!is.element(model,c("rrl","ord")),
symmetric=FALSE,
odmax=rep(max(apply(Y>0,1,sum,na.rm=TRUE)),nrow(Y)), seed = 1, nscan =
10000, burn = 500, odens = 25, plot=TRUE, print = TRUE, gof=TRUE)
```

Arguments

`Y`	an n x n square relational matrix of relations. See model below for various data types.
`Xdyad`	an n x n x pd array of covariates
`Xrow`	an n x pr matrix of nodal row covariates
`Xcol`	an n x pc matrix of nodal column covariates
`rvar`	logical: fit row random effects (asymmetric case)?
`cvar`	logical: fit column random effects (asymmetric case)?
`dcor`	logical: fit a dyadic correlation (asymmetric case)?
`nvar`	logical: fit nodal random effects (symmetric case)?
`R`	integer: dimension of the multiplicative effects (can be zero)
`model`	character: one of "nrm","bin","ord","cbin","frn","rrl" - see the details below
`intercept`	logical: fit model with an intercept?
`symmetric`	logical: Is the sociomatrix symmetric by design?
`odmax`	a scalar integer or vector of length n giving the maximum number of nominations that each node may make - used for "frn" and "cbin" models
`seed`	random seed
`nscan`	number of iterations of the Markov chain (beyond burn-in)
`burn`	burn in for the Markov chain
`odens`	output density for the Markov chain
`plot`	logical: plot results while running?
`print`	logical: print results while running?
`gof`	logical: calculate goodness of fit statistics?

Details

This command provides posterior inference for parameters in AME models of relational data, assuming one of six possible data types/models:

"nrm": A normal AME model.

"bin": A binary probit AME model.

"ord": An ordinal probit AME model. An intercept is not identifiable in this model.

"cbin": An AME model for censored binary data. The value of 'odmax' specifies the maximum number of links each row may have.

"frn": An AME model for fixed rank nomination networks. A higher value of the rank indicates a stronger relationship. The value of 'odmax' specifies the maximum number of links each row may have.

"rrl": An AME model based on the row ranks. This is appropriate if the relationships across rows are not directly comparable in terms of scale. An intercept, row random effects and row regression effects are not estimable for this model.

Value

`BETA`	posterior samples of regression coefficients
`VC`	posterior samples of the variance parameters
`APM`	posterior mean of additive row effects a
`BPM`	posterior mean of additive column effects b
`U`	posterior mean of multiplicative row effects u
`V`	posterior mean of multiplicative column effects v (asymmetric case)
`UVPM`	posterior mean of UV (asymmetric case)
`ULUPM`	posterior mean of ULU (symmetric case)
`L`	posterior mean of L (symmetric case)
`EZ`	estimate of expectation of Z matrix
`YPM`	posterior mean of Y (for imputing missing values)
`GOF`	observed (first row) and posterior predictive (remaining rows) values of four goodness-of-fit statistics

Author(s)

Peter Hoff

Examples

```
data(YX_frn)
fit<-ame(YX_frn$Y,YX_frn$X,burn=5,nscan=5,odens=1,model="frn")
# you should run the Markov chain much longer than this
```

ame_rep *AME model fitting routine for replicated relational data*

Description

An MCMC routine providing a fit to an additive and multiplicative effects (AME) regression model to replicated relational data of various types.

Usage

```
ame_rep(Y,Xdyad=NULL, Xrow=NULL, Xcol=NULL, rvar = !(model=="rrl")
, cvar = TRUE, dcor = !symmetric, nvar=TRUE,  R = 0, model="nrm",
intercept=!is.element(model,c("rrl","ord")),
symmetric=FALSE,
odmax=rep(max(apply(Y>0,c(1,3),sum,na.rm=TRUE)),nrow(Y[,,1])), seed = 1,
nscan = 10000, burn = 500, odens = 25, plot=TRUE, print = TRUE, gof=TRUE)
```

Arguments

Y	an n x n x T array of relational matrix, where the third dimension correponds to replicates (over time, for example). See model below for various data types.
Xdyad	an n x n x pd x T array of covariates
Xrow	an n x pr x T array of nodal row covariates
Xcol	an n x pc x T array of nodal column covariates
rvar	logical: fit row random effects (asymmetric case)?
cvar	logical: fit column random effects (asymmetric case)?
dcor	logical: fit a dyadic correlation (asymmetric case)?
nvar	logical: fit nodal random effects (symmetric case)?
R	integer: dimension of the multiplicative effects (can be zero)
model	character: one of "nrm","bin","ord","cbin","frn","rrl" - see the details below
intercept	logical: fit model with an intercept?
symmetric	logical: Is the sociomatrix symmetric by design?
odmax	a scalar integer or vector of length n giving the maximum number of nominations that each node may make - used for "frn" and "cbin" models
seed	random seed
nscan	number of iterations of the Markov chain (beyond burn-in)
burn	burn in for the Markov chain
odens	output density for the Markov chain
plot	logical: plot results while running?
print	logical: print results while running?
gof	logical: calculate goodness of fit statistics?

Details

This command provides posterior inference for parameters in AME models of independent replicated relational data, assuming one of six possible data types/models:

"nrm": A normal AME model.

"bin": A binary probit AME model.

"ord": An ordinal probit AME model. An intercept is not identifiable in this model.

"cbin": An AME model for censored binary data. The value of 'odmax' specifies the maximum number of links each row may have.

"frn": An AME model for fixed rank nomination networks. A higher value of the rank indicates a stronger relationship. The value of 'odmax' specifies the maximum number of links each row may have.

"rrl": An AME model based on the row ranks. This is appropriate if the relationships across rows are not directly comparable in terms of scale. An intercept, row random effects and row regression effects are not estimable for this model.

Value

BETA	posterior samples of regression coefficients
VC	posterior samples of the variance parameters
APM	posterior mean of additive row effects a
BPM	posterior mean of additive column effects b
U	posterior mean of multiplicative row effects u
V	posterior mean of multiplicative column effects v (asymmetric case)
UVPM	posterior mean of UV
ULUPM	posterior mean of ULU (symmetric case)
L	posterior mean of L (symmetric case)
EZ	estimate of expectation of Z matrix
YPM	posterior mean of Y (for imputing missing values)
GOF	observed (first row) and posterior predictive (remaining rows) values of four goodness-of-fit statistics

Author(s)

Peter Hoff, Yanjun He

Examples

```
data(YX_bin_long)
fit<-ame_rep(YX_bin_long$Y,YX_bin_long$X,burn=5,nscan=5,odens=1,model="bin")
# you should run the Markov chain much longer than this
```

`circplot` *Circular network plot*

Description

Produce a circular network plot.

Usage

```
circplot(Y, U = NULL, V = NULL, row.names = rownames(Y),
  col.names = colnames(Y), plotnames = TRUE, vscale = 0.8,
  pscale = 1.75, lcol = "gray", rcol = "brown", ccol = "blue",
  pch = 16, lty = 3, jitter = 0.1 * (nrow(Y)/(1 + nrow(Y))), bty = "n",
  add = FALSE)
```

Arguments

`Y`	(matrix) m by n relational matrix.
`U`	(matrix) m by 2 matrix of row factors of Y.
`V`	(matrix) n by 2 matrix of column factors of Y.
`row.names`	(character vector) names of the row objects.
`col.names`	(character vector) names of the columns objects.
`plotnames`	(logical) plot row and column names.
`vscale`	(scalar) scaling factor for V coordinates.
`pscale`	(scalar) scaling factor for plotting characters.
`lcol`	(scalar or vector) line color(s) for the nonzero elements of Y.
`rcol`	(scalar or vector) node color(s) for the rows.
`ccol`	(scalar or vector) node color(s) for the columns.
`pch`	(integer) plotting character.
`lty`	(integer) line type.
`jitter`	(scalar) a number to control jittering of nodes.
`bty`	(character) bounding box type.
`add`	(logical) add to existing plot

Details

This function creates a circle plot of a relational matrix or social network. If not supplied via `U` and `V`, two-dimensional row factors and column factors are computed from the SVD of `Y`, scaled versions of which are used to plot positions on the outside edge (`U`) and inside edge (`V`) of the circle plot. The magnitudes of the plotting characters are determined by the magnitudes of the rows of `U` and `V`. Segments are drawn between each row object `i` and column object `j` for which `Y[i,j]!=0`.

Author(s)

Peter Hoff

Examples

```
data(IR90s)
circplot(IR90s$dyadvars[,,1])
```

`coldwar` *Cold War data*

Description

Positive and negative relations between countries during the cold war

Format

A list including the following dyadic and nodal variables:

- `cc`: a socioarray of ordinal levels of military cooperation (positive) and conflict (negative), every 5 years;
- `distance`: between-country distance (in thousands of kilometers);
- `gdp`: country gdp in dollars every 5 years;
- `polity`: country polity every 5 years.

Source

Xun Cao : http://polisci.la.psu.edu/people/xuc11

`comtrade` *Comtrade data*

Description

Eleven years of import and export data between 229 countries. The data use the SITC Rev. 1 commodity classification, aggregated at the first level (AG1).

Format

A list consisting of a socioarray `Trade` and a vector `dollars2010` of inflation rates. The socioarray gives yearly trade volume (exports and imports) in dollars for 10 different commodity classes for eleven years between 229 countries. This gives a five-way array. The first index is the reporting country, so `Trade[i,j,t,k,1]` is what `i` reports for exports to `j`, but in general this is not the same as `Trade[j,i,t,k,2]`, what `j` reports as importing from `i`.

Source

http://comtrade.un.org/, http://www.measuringworth.com/

---

design_array *Computes the design socioarray of covariate values*

Description

Computes the design socioarray of covariate values for an AME fit

Usage

```
design_array(Xrow=NULL,Xcol=NULL,Xdyad=NULL,intercept=TRUE,n)
```

Arguments

Xrow	an n x pr matrix of row covariates
Xcol	an n x pc matrix of column covariates
Xdyad	an n x n x pd array of dyadic covariates
intercept	logical
n	number of rows/columns

Value

an n x n x (pr+pc+pd+intercept) 3-way array

Author(s)

Peter Hoff

---

dutchcollege *Dutch college data*

Description

Longitudinal relational measurements and nodal characteristics of Dutch college students, described in van de Bunt, van Duijn, and Snijders (1999). The time interval between the first four measurements was three weeks, whereas the interval between the last three was six weeks.

Format

A list consisting of a socioarray Y and a matrix X of static nodal attributes. The relational measurements range from -1 to 4, indicating the following:

- -1 a troubled or negative relationship
- 0 don't know
- 1 neutral relationship
- 2 friendly
- 3 friendship
- 4 best friends

Source

Linton Freeman

---

el2sm *Edgelist to sociomatrix*

Description

Construction of a sociomatrix from an edgelist

Usage

```
el2sm(el,directed=TRUE,nadiag=all(el[,1]!=el[,2]))
```

Arguments

el	a matrix in which each row contains the indices of an edge and possibly the weight for the edge
directed	if FALSE, then a relation is placed in both entry ij and ji of the sociomatrix, for each edge ij (or ji)
nadiag	put NAs on the diagonal

Value

a sociomatrix

Author(s)

Peter Hoff

Examples

```
Y<-matrix(rpois(10*10,.5),10,10) ; diag(Y)<-NA
E<-sm2el(Y)
el2sm(E) - Y
```

`gofstats` *Goodness of fit statistics*

Description

Goodness of fit statistics evaluating second and third-order dependence patterns

Usage

```
gofstats(Y)
```

Arguments

`Y`	a relational data matrix

Value

a vector of gof statistics

Author(s)

Peter Hoff

Examples

```
data(YX_nrm)

gofstats(YX_nrm$Y)
```

IR90s *International relations in the 90s*

Description

A relational dataset recording a variety of nodal and dyadic variables on countries in the 1990s, including information on conflicts, trade and other variables. Except for the conflict variable, the variables are averages across the decade.

Format

A list consisting of a socioarray `dyadvars` of dyadic variables and matrix `nodevars` of nodal variables. The dyadic variables include

- total number of conflicts;
- exports (in billions of dollars);
- distance (in thousands of kilometers);
- number of shared IGOs (averages across the years);
- polity interaction.

The nodal variables include

- population (in millions);
- gdp (in billions of dollars);
- polity

Source

Michael Ward.

lazegalaw *Lazega's law firm data*

Description

Several nodal and dyadic variables measured on 71 attorneys in a law firm.

Format

A list consisting of a socioarray `Y` and a nodal attribute matrix `X`.

The dyadic variables in `Y` include three binary networks: advice, friendship and co-worker status.

The categorical nodal attributes in `X` are coded as follows:

- status (1=partner, 2=associate)
- office (1=Boston, 2=Hartford, 3=Providence)
- practice (1=litigation, 2=corporate)
- law school (1=Harvard or Yale, 2=UConn, 3=other)

`seniority` and `age` are given in years, and `female` is a binary indicator.

Source

Linton Freeman

`mhalf` *Symmetric square root of a matrix*

Description

Computes the symmetric square root of a positive definite matrix

Usage

```
mhalf(M)
```

Arguments

`M`	a positive definite matrix

Value

a matrix `H` such that `H^2` equals `M`

Author(s)

Peter Hoff

`netplot` *Network plotting*

Description

Plot the graph of a sociomatrix

Usage

```
netplot(Y,X=NULL,xaxt="n",yaxt="n",xlab="",ylab="",
 lcol="gray",ncol="black",lwd=1,lty=1,pch=16,bty="n",plotnames=FALSE,
 seed=1,
 plot.iso=TRUE,directed=NULL,add=FALSE,...)
```

Arguments

`Y`	a sociomatrix
`X`	coordinates for plotting the nodes
`xaxt`	x-axis type
`yaxt`	y-axis type
`xlab`	x-axis label
`ylab`	y-axis label
`lcol`	edge color
`ncol`	node color (can be node-specific)
`lwd`	line width
`lty`	line type
`pch`	plotting character for nodes (can be node-specific)
`bty`	bounding box type
`plotnames`	plot rownames of Y as node labels
`seed`	random seed
`plot.iso`	include isolates in plot
`directed`	draw arrows
`add`	add to an existing plot region
`...`	additional plotting parameters

Author(s)

Peter Hoff

Examples

```
data(addhealthc3)
Y<-addhealthc3$Y
X<-xnet(Y)
netplot(Y,X)
```

`plot.ame` *Plot results of an AME object*

Description

A set of plots summarizing the MCMC routine for an AME fit, as well as some posterior predictive checks.

Usage

```
## S3 method for class 'ame'
plot(x, ...)
```

Arguments

`x`	the result of fitting an AME model
`...`	additional parameters (not used)

Value

a series of plots

Author(s)

Peter Hoff

`raSab_bin_fc` *Simulate a and Sab from full conditional distributions under bin likelihood*

Description

Simulate a and Sab from full conditional distributions under bin likelihood

Usage

```
raSab_bin_fc(Z, Y, a, b, Sab, SS = round(sqrt(nrow(Z))))
```

Arguments

`Z`	a square matrix, the current value of Z
`Y`	square binary relational matrix
`a`	current value of row effects
`b`	current value of column effects
`Sab`	current value of Cov(a,b)
`SS`	number of iterations

Value

`Z`	new value of Z
`Sab`	new value of Sab
`a`	new value of a

Author(s)

Peter Hoff

`raSab_cbin_fc` *Simulate a and Sab from full conditional distributions under the cbin likelihood*

Description

Simulate a and Sab from full conditional distributions under the cbin likelihood

Usage

```
raSab_cbin_fc(Z, Y, a, b, Sab, odmax, odobs, SS =
round(sqrt(nrow(Z))))
```

Arguments

`Z`	a square matrix, the current value of Z
`Y`	square matrix of ranked nomination data
`a`	current value of row effects
`b`	current value of column effects
`Sab`	current value of Cov(a,b)
`odmax`	a scalar or vector giving the maximum number of nominations for each individual
`odobs`	observed outdegree
`SS`	number of iterations

Value

Z	new value of Z
Sab	new value of Sab
a	new value of a

Author(s)

Peter Hoff

`raSab_frn_fc` *Simulate a and Sab from full conditional distributions under frn likelihood*

Description

Simulate a and Sab from full conditional distributions under frn likelihood

Usage

```
raSab_frn_fc(Z, Y, YL, a, b, Sab, odmax, odobs, SS =
round(sqrt(nrow(Z))))
```

Arguments

Z	a square matrix, the current value of Z
Y	square matrix of ranked nomination data
YL	list of ranked individuals, from least to most preferred in each row
a	current value of row effects
b	current value of column effects
Sab	current value of Cov(a,b)
odmax	a scalar or vector giving the maximum number of nominations for each individual
odobs	observed outdegree
SS	number of iterations

Value

Z	new value of Z
Sab	new value of Sab
a	new value of a

Author(s)

Peter Hoff

`rbeta_ab_fc` *Gibbs sampling of additive row and column effects and regression coefficient*

Description

Simulates from the joint full conditional distribution of (a,b,beta)

Usage

```
rbeta_ab_fc(Z, Sab, rho, X, mX, mXt, XX, XXt, Xr, Xc, s2 = 1)
```

Arguments

`Z`	n X n (latent) normal relational matrix, with multiplicative effects subtracted out
`Sab`	row and column covariance
`rho`	dyadic correlation
`X`	n x n x p covariate array
`mX`	design matrix (matricizied version of X)
`mXt`	dyad-transposed design matrix
`XX`	regression sums of squares
`XXt`	crossproduct sums of squares
`Xr`	row sums for X
`Xc`	column sums for X
`s2`	dyadic variance

Value

`beta`	regression coefficients
`a`	additive row effects
`b`	additive column effects

Author(s)

Peter Hoff

rbeta_ab_rep_fc — *Gibbs sampling of additive row and column effects and regression coefficient with independent replicate relational data*

Description

Simulates from the joint full conditional distribution of (a,b,beta), assuming same additive row and column effects and regression coefficient across replicates.

Usage

```
rbeta_ab_rep_fc(Z.T,Sab,rho,X.T,s2=1)
```

Arguments

Z.T	n x n x T array, with the third dimension for replicates. Each slice of the array is a (latent) normal relational matrix, with multiplicative effects subtracted out
Sab	row and column covariance
rho	dyadic correlation
X.T	n x n x p x T covariate array
s2	dyadic variance

Value

beta	regression coefficients
a	additive row effects
b	additive column effects

Author(s)

Peter Hoff, Yanjun He

rmvnorm — *Simulation from a multivariate normal distribution*

Description

Simulates a matrix where the rows are i.i.d. samples from a multivariate normal distribution

Usage

```
rmvnorm(n, mu, Sigma, Sigma.chol = chol(Sigma))
```

Arguments

`n`	sample size
`mu`	multivariate mean vector
`Sigma`	covariance matrix
`Sigma.chol`	Cholesky factorization of `Sigma`

Value

a matrix with `n` rows

Author(s)

Peter Hoff

`rrho_mh` *Metropolis update for dyadic correlation*

Description

Metropolis update for dyadic correlation

Usage

```
rrho_mh(E, rho, s2 = 1)
```

Arguments

`E`	square residual relational matrix
`rho`	current value of rho
`s2`	current value of s2

Value

a new value of rho

Author(s)

Peter Hoff

`rrho_mh_rep` *Metropolis update for dyadic correlation with independent replicate data*

Description

Metropolis update for dyadic correlation with independent replicate data.

Usage

```
rrho_mh_rep(E.T, rho, s2 = 1)
```

Arguments

`E.T`	Array of square residual relational matrix series. The third dimension of the array is for different replicates. Each slice of the array according to the third dimension is a square residual relational matrix.
`rho`	current value of rho
`s2`	current value of s2

Value

a new value of rho

Author(s)

Peter Hoff, Yanjun He

`rs2_fc` *Gibbs update for dyadic variance*

Description

Gibbs update for dyadic variance

Usage

```
rs2_fc(E, rho)
```

Arguments

`E`	square residual relational matrix
`rho`	current value of rho

Value

a new value of s2

Author(s)

Peter Hoff

`rs2_rep_fc` *Gibbs update for dyadic variance with independent replicate relational data*

Description

Gibbs update for dyadic variance with independent replicate relational data

Usage

```
rs2_rep_fc(E.T, rho)
```

Arguments

`E.T`	Array of square residual relational matrix series. The third dimension of the array is for different replicates. Each slice of the array according to the third dimension is a square residual relational matrix
`rho`	current value of rho

Value

a new value of s2

Author(s)

Peter Hoff, Yanjun He

rUV_fc *Gibbs sampling of U and V*

Description

A Gibbs sampler for updating the multiplicative effect matrices U and V

Usage

```
rUV_fc(E, U, V, rho, s2 = 1, shrink=TRUE)
```

Arguments

E	square residual relational matrix
U	current value of U
V	current value of V
rho	dyadic correlation
s2	dyadic variance
shrink	adaptively shrink the factors with a hierarchical prior

Value

U	a new value of U
V	a new value of V

Author(s)

Peter Hoff

rUV_rep_fc *Gibbs sampling of U and V*

Description

A Gibbs sampler for updating the multiplicative effect matrices U and V, assuming they are the same across replicates.

Usage

```
rUV_rep_fc(E.T,U,V,rho,s2=1,shrink=TRUE)
```

Arguments

`E.T`	Array of square residual relational matrix series with additive effects and covariates subtracted out. The third dimension of the array is for different replicates. Each slice of the array according to the third dimension is a square residual relational matrix.
`U`	current value of U
`V`	current value of V
`rho`	dyadic correlation
`s2`	dyadic variance
`shrink`	adaptively shrink the factors with a hierarchical prior

Value

`U`	a new value of U
`V`	a new value of V

Author(s)

Peter Hoff, Yanjun He

`rUV_sym_fc` *Gibbs sampling of U and V*

Description

A Gibbs sampler for updating the multiplicative effect matrices U and V in the symmetric case. In this case `U%*%t(V)` is symmetric, so this is parameterized as `V=U%*%L` where `L` is the diagonal matrix of eigenvalues of `U%*%t(V)`.

Usage

```
rUV_sym_fc(E, U, V, s2 = 1, shrink=TRUE)
```

Arguments

`E`	square residual relational matrix
`U`	current value of U
`V`	current value of V
`s2`	dyadic variance
`shrink`	adaptively shrink the factors with a hierarchical prior

Value

`U`	a new value of U
`V`	a new value of V

Author(s)

Peter Hoff

Examples

```
U0<-matrix(rnorm(30,2),30,2) ; V0<-U0%*%diag(c(3,-2))
E<- U0%*%t(V0) + matrix(rnorm(30^2),30,30)
rUV_sym_fc
```

---

rwish *Simulation from a Wishart distribution*

Description

Simulates a random Wishart-distributed matrix

Usage

```
rwish(S0, nu = dim(S0)[1] + 1)
```

Arguments

S0	a positive definite matrix
nu	a positive integer

Value

a positive definite matrix

Author(s)

Peter Hoff

Examples

```
## The expectation is S0*nu

S0<-rwish(diag(3))

SS<-matrix(0,3,3)
for(s in 1:1000) { SS<-SS+rwish(S0,5) }

SS/s

S0*5
```

rZ_bin_fc *Simulate Z based on a probit model*

Description

Simulates a random latent matrix Z given its expectation, dyadic correlation and a binary relational matrix Y

Usage

```
rZ_bin_fc(Z, EZ, rho, Y)
```

Arguments

`Z`	a square matrix, the current value of Z
`EZ`	expected value of Z
`rho`	dyadic correlation
`Y`	square binary relational matrix

Value

a square matrix , the new value of Z

Author(s)

Peter Hoff

rZ_cbin_fc *Simulate Z given fixed rank nomination data*

Description

Simulates a random latent matrix Z given its expectation, dyadic correlation and censored binary nomination data

Usage

```
rZ_cbin_fc(Z, EZ, rho, Y, odmax, odobs)
```

Arguments

Z	a square matrix, the current value of Z
EZ	expected value of Z
rho	dyadic correlation
Y	square matrix of ranked nomination data
odmax	a scalar or vector giving the maximum number of nominations for each individual
odobs	observed outdegree

Details

simulates Z under the constraints (1) Y[i,j]=1, Y[i,k]=0 => Z[i,j]>Z[i,k] , (2) Y[i,j]=1 => Z[i,j]>0 , (3) Y[i,j]=0 & odobs[i]<odmax[i] => Z[i,j]<0

Value

a square matrix, the new value of Z

Author(s)

Peter Hoff

---

rZ_frn_fc *Simulate Z given fixed rank nomination data*

Description

Simulates a random latent matrix Z given its expectation, dyadic correlation and fixed rank nomination data

Usage

```
rZ_frn_fc(Z, EZ, rho, Y, YL, odmax, odobs)
```

Arguments

Z	a square matrix, the current value of Z
EZ	expected value of Z
rho	dyadic correlation
Y	square matrix of ranked nomination data
YL	list of ranked individuals, from least to most preferred in each row
odmax	a scalar or vector giving the maximum number of nominations for each individual
odobs	observed outdegree

Details

simulates Z under the constraints (1) Y[i,j]>Y[i,k] => Z[i,j]>Z[i,k] , (2) Y[i,j]>0 => Z[i,j]>0 , (3) Y[i,j]=0 & odobs[i]<odmax[i] => Z[i,j]<0

Value

a square matrix, the new value of Z

Author(s)

Peter Hoff

`rZ_nrm_fc` *Simulate missing values in a normal AME model*

Description

Simulates missing values of a sociomatrix under a normal AME model

Usage

```
rZ_nrm_fc(Z, EZ, rho,s2, Y)
```

Arguments

`Z`	a square matrix, the current value of Z
`EZ`	expected value of Z
`rho`	dyadic correlation
`s2`	dyadic variance
`Y`	square relational matrix

Value

a square matrix, equal to Y at non-missing values

Author(s)

Peter Hoff

rZ_ord_fc — *Simulate Z given the partial ranks*

Description

Simulates a random latent matrix Z given its expectation, dyadic correlation and partial rank information provided by W

Usage

```
rZ_ord_fc(Z, EZ, rho, Y)
```

Arguments

Z	a square matrix, the current value of Z
EZ	expected value of Z
rho	dyadic correlation
Y	matrix of ordinal data

Value

a square matrix, the new value of Z

Author(s)

Peter Hoff

rZ_rrl_fc — *Simulate Z given relative rank nomination data*

Description

Simulates a random latent matrix Z given its expectation, dyadic correlation and relative rank nomination data

Usage

```
rZ_rrl_fc(Z, EZ, rho, Y, YL)
```

Arguments

Z	a square matrix, the current value of Z
EZ	expected value of Z
rho	dyadic correlation
Y	square matrix of ranked nomination data
YL	list of ranked individuals, from least to most preferred in each row

Details

simulates Z under the constraints (1) Y[i,j]>Y[i,k] => Z[i,j]>Z[i,k]

Value

a square matrix, the new value of Z

Author(s)

Peter Hoff

`sampsonmonks` *Sampson's monastery data*

Description

Several dyadic variables measured on 18 members of a monastery.

Format

A socioarray whose dimensions represent nominators, nominatees and relations. Each monk was asked to rank up to three other monks on a variety of positive and negative relations. A rank of three indicates the "highest" ranking for a particular relational variable. The relations `like_m2` and `like_m1` are evaluations of likeing at one and two timepoints previous to when the other relations were measured.

Source

Linton Freeman

`sheep` *Sheep dominance data*

Description

Number of dominance encounters between 28 female bighorn sheep. Cell (i,j) records the number of times sheep i dominated sheep j. From Hass (1991).

Format

A list consisting of the following:

- `dom`: a directed socioarray recording the number of dominance encounters.
- `age`: the age of each sheep in years.

Source

Linton Freeman

simY_bin *Simulate a network, i.e. a binary relational matrix*

Description

Simulates a network, i.e. a binary relational matrix

Usage

```
simY_bin(EZ, rho)
```

Arguments

EZ	square matrix giving the expected value of the latent Z matrix
rho	dyadic correlation

Value

a square binary matrix

Author(s)

Peter Hoff

simY_frn *Simulate an relational matrix based on a fixed rank nomination scheme*

Description

Simulate an relational matrix based on a fixed rank nomination scheme

Usage

```
simY_frn(EZ, rho, odmax, YO)
```

Arguments

EZ	a square matrix giving the expected value of the latent Z matrix
rho	dyadic correlation
odmax	a scalar or vector giving the maximum number of nominations for each node
YO	a square matrix identifying where missing values should be maintained

Value

a square matrix, where higher values represent stronger relationships

Author(s)

Peter Hoff

`simY_nrm` *Simulate a normal relational matrix*

Description

Simulates a normal relational matrix

Usage

```
simY_nrm(EY, rho, s2)
```

Arguments

EY	square matrix giving the expected value of the relational matrix
rho	dyadic correlation
s2	dyadic variance

Value

a square matrix

Author(s)

Peter Hoff

`simY_ord` *Simulate an ordinal relational matrix*

Description

Simulates an ordinal relational matrix having a particular marginal distribution

Usage

```
simY_ord(EZ, rho, Y)
```

Arguments

EZ	square matrix giving the expected value of the latent Z matrix
rho	scalar giving the within-dyad correlation
Y	ordinal relational data matrix

Value

a square matrix

Author(s)

Peter Hoff

`simY_rrl` *Simulate an relational matrix based on a relative rank nomination scheme*

Description

Simulate an relational matrix based on a relative rank nomination scheme

Usage

```
simY_rrl(EZ, rho, odobs, YO)
```

Arguments

`EZ`	a square matrix giving the expected value of the latent Z matrix
`rho`	dyadic correlation
`odobs`	a scalar or vector giving the observed number of nominations for each node
`YO`	a square matrix identifying where missing values should be maintained

Value

a square matrix, where higher values represent stronger relationships

Author(s)

Peter Hoff

simZ — *Simulate Z given its expectation and covariance*

Description

Simulate Z given its expectation and covariance

Usage

```
simZ(EZ, rho, s2 = 1)
```

Arguments

`EZ`	expected value of Z
`rho`	dyadic correlation
`s2`	dyadic variance

Value

a simulated value of Z

Author(s)

Peter Hoff

sm2el — *Sociomatrix to edgelist*

Description

Construction of an edgelist from a sociomatrix

Usage

```
sm2el(sm,directed=TRUE)
```

Arguments

`sm`	a sociomatrix with possibly valued relations
`directed`	if TRUE, only use the upper triangular part of the matrix to enumerate edges

Value

an edglist

Author(s)

Peter Hoff

Examples

```
Y<-matrix(rpois(10*10,.5),10,10) ; diag(Y)<-NA
E<-sm2el(Y)
el2sm(E) - Y
```

`summary.ame` *Summary of an AME object*

Description

Summary method for an AME object

Usage

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

Arguments

`object`	the result of fitting an AME model
`...`	additional parameters (not used)

Value

a summary of parameter estimates and confidence intervals for an AME fit

Author(s)

Peter Hoff

Xbeta — *Linear combinations of submatrices of an array*

Description

Computes a matrix of expected values based on an array X of predictors and a vector beta of regression coefficients.

Usage

```
Xbeta(X, beta)
```

Arguments

X	an n by n by p array
beta	a p by 1 vector

Value

An n by n matrix

Author(s)

Peter Hoff

xnet — *Network embedding*

Description

Compute an embedding of a sociomatrix into a two-dimensional space.

Usage

```
xnet(Y, fm = suppressWarnings(require("network")), seed = 1)
```

Arguments

Y	(square matrix) The sociomatrix.
fm	(logical scalar) If TRUE, the Fruchterman-Reingold layout will be used (requires the network package).
seed	(integer) The random seed (the FR layout is random).

Details

Coordinates are obtained using the Fruchterman-Reingold layout if the package `network` is installed, and otherwise uses the first two eigenvectors the sociomatrix.

Value

(matrix) A matrix of two-dimensional coordinates.

Author(s)

Peter Hoff

Examples

```
data(addhealthc3)
Y<-addhealthc3$Y
X<-xnet(Y)
netplot(Y,X)
```

`YX_bin` *binary relational data and covariates*

Description

a synthetic dataset that includes binary relational data as well as information on eight covariates

Usage

```
data(YX_bin)
```

Format

The format is: List of 2 $ Y: num [1:100, 1:100] NA 0 0 0 0 0 0 0 0 1 ... $ X: num [1:100, 1:100, 1:8] 1 1 1 1 1 1 1 1 1 1 ... ..- attr(*, "dimnames")=List of 3 .. ..$: NULL .. ..$: NULL .. ..$: chr [1:8] "intercept" "rgpa" "rsmoke" "cgpa" ...

Examples

```
data(YX_bin)
gofstats(YX_bin$Y)
```

YX_bin_long *binary relational data and covariates*

Description

a synthetic dataset that includes longitudinal binary relational data as well as information on covariates

Usage

```
data(YX_bin_long)
```

Format

a list

Examples

```
data(YX_bin_long)
gofstats(YX_bin_long$Y[,,1])
```

YX_cbin *Censored binary nomination data and covariates*

Description

a synthetic dataset that includes relational data where the number of nominations per row is censored at 10, along with information on eight covariates

Usage

```
data(YX_cbin)
```

Format

The format is: List of 2 $ Y: num [1:100, 1:100] NA 0 0 0 1 0 0 0 0 3 ... $ X: num [1:100, 1:100, 1:8] 1 1 1 1 1 1 1 1 1 1 ... ..- attr(*, "dimnames")=List of 3 .. ..$: NULL .. ..$: NULL .. ..$: chr [1:8] "intercept" "rgpa" "rsmoke" "cgpa" ...

Examples

```
data(YX_cbin)
gofstats(YX_cbin$Y)
```

YX_frn *Fixed rank nomination data and covariates*

Description

a synthetic dataset that includes fixed rank nomination data as well as information on eight covariates

Usage

```
data(YX_frn)
```

Format

The format is: List of 2 $ Y: num [1:100, 1:100] NA 0 0 0 1 0 0 0 0 3 ... $ X: num [1:100, 1:100, 1:8] 1 1 1 1 1 1 1 1 1 1 ... ..- attr(*, "dimnames")=List of 3 .. ..$: NULL .. ..$: NULL .. ..$: chr [1:8] "intercept" "rgpa" "rsmoke" "cgpa" ...

Examples

```
data(YX_frn)
gofstats(YX_frn$Y)
```

YX_nrm *normal relational data and covariates*

Description

a synthetic dataset that includes continuous (normal) relational data as well as information on eight covariates

Usage

```
data(YX_nrm)
```

Format

The format is: List of 2 $ Y: num [1:100, 1:100] NA -4.05 -0.181 -3.053 -1.579 ... $ X: num [1:100, 1:100, 1:8] 1 1 1 1 1 1 1 1 1 1 ... ..- attr(*, "dimnames")=List of 3 .. ..$: NULL .. ..$: NULL .. ..$: chr [1:8] "intercept" "rgpa" "rsmoke" "cgpa" ...

Examples

```
data(YX_nrm)
gofstats(YX_nrm$Y)
```

YX_ord   *ordinal relational data and covariates*

Description

a synthetic dataset that includes ordinal relational data as well as information on seven covariates

Usage

```
data(YX_ord)
```

Format

The format is: List of 2 $ Y: num [1:100, 1:100] NA 0 3 0 3 1 0 1 1 0 ... $ X: num [1:100, 1:100, 1:7] 1 1 1 1 1 1 1 1 1 1 ... ..- attr(*, "dimnames")=List of 3 .. ..$: NULL .. ..$: NULL .. ..$: chr [1:7] "rgpa" "rsmoke" "cgpa" "csmoke" ...

Examples

```
data(YX_ord)
gofstats(YX_ord$Y)
```

YX_rrl   *row-specific ordinal relational data and covariates*

Description

a synthetic dataset that includes row-specific ordinal relational data as well as information on five covariates

Usage

```
data(YX_rrl)
```

Format

The format is: List of 2 $ Y: num [1:100, 1:100] NA 0 3 0 3 1 0 1 1 0 ... $ X: num [1:100, 1:100, 1:5] 1 1 1 1 1 1 1 1 1 1 ... ..- attr(*, "dimnames")=List of 3 .. ..$: NULL .. ..$: NULL .. ..$: chr [1:5] "cgpa" "csmoke" "igrade" "ismoke" ...

Examples

```
data(YX_rrl)
gofstats(YX_rrl$Y)
```

`zscores` *rank-based z-scores*

Description

Computes the normal scores corresponding to the ranks of a data vector

Usage

```
zscores(y)
```

Arguments

`y` a numeric vector

Value

a numeric vector

Author(s)

Peter Hoff

Index