Package 'eigenmodel'

February 9, 2012

Title Semiparametric factor and regression models for symmetric relational data

Version 1.0

Date 2007-06-25

Author Peter Hoff

Maintainer Peter Hoff <hoff@stat.washington.edu>

Description This package estimates the parameters of a model for
symmetric relational data (e.g., the above-diagonal part of a
square matrix), using a model-based eigenvalue decomposition
and regression. Missing data is accomodated, and a posterior
mean for missing data is calculated under the assumption that
the data are missing at random. The marginal distribution of
the relational data can be arbitrary, and is fit with an ordered probit specification.

License GPL-2

URL http://www.stat.washington.edu/hoff

Repository CRAN

Date/Publication 2007-06-27 11:32:51

R topics documented:

`eigenmodel-package` *Semiparametric factor and regression models for symmetric relational data*

Description

This package estimates the parameters of a model for symmetric relational data (e.g., the above-diagonal part of a square matrix), using a model-based eigenvalue decomposition and regression. Missing data is accomodated, and a posterior mean for missing data is calculated under the assumption that the data are missing at random. The marginal distribution of the relational data can be arbitrary, and is fit with an ordered probit specification.

Details

Package:	eigenmodel
Type:	Package
Version:	1.0
Date:	2007-06-25
License:	GPL Version 2

Author(s)

Peter Hoff <hoff@stat.washington.edu>

References

Hoff (2007) "Modeling homophily and stochastic equivalence in symmetric relational data"

Examples

```
data(YX_Friend)

fit<-eigenmodel_mcmc(Y=YX_Friend$Y,X=YX_Friend$X,R=2,S=750,burn=250)

# in general you should run the Markov chain longer than 750 scans
```

```
plot(fit)


# people familiar with MCMC might want to implement
# their own Markov chains:

Y<-YX_Friend$Y
X<-YX_Friend$X

eigenmodel_setup(R=2)

for(s in 1:100) {   # you should run your chain longer than 100 scans

    Z<-rZ_fc()
    UL<-rUL_fc()
    b<-rb_fc()

                }


#fit_Gen<-eigenmodel_mcmc(Y=Y_Gen,R=3,S=10000)

#fit_Pro<-eigenmodel_mcmc(Y=Y_Pro,R=3,S=10000)
```

`addlines` *Adds lines between nodes to an existing plot of nodes*

Description

Adds lines between nodes to an existing plot of nodes

Usage

```
addlines(U, Y, col = "green", lwd = 1, lty = 1)
```

Arguments

`U`	an n x 2 matrix of node locations
`Y`	a symmetric matrix
`col`	color of the lines
`lwd`	width of the lines
`lty`	line type

Value

NULL

Author(s)

Peter Hoff

`eigenmodel_mcmc` *Approximate the posterior distribution of parameters in an eigenmodel*

Description

Construct approximate samples from the posterior distribution of the parameters and latent variables in an eigenmodel for symmetric relational data.

Usage

```
eigenmodel_mcmc(Y, X = NULL, R = 2, S = 1000, seed = 1, Nss = min(S-burn, 1000), burn = 0)
```

Arguments

`Y`	an n x n symmetric matrix with missing diagonal entries. Off-diagonal missing values are allowed.
`X`	an n x n x p array of regressors
`R`	the rank of the approximating factor matrix
`S`	number of samples from the Markov chain
`seed`	a random seed
`Nss`	number of samples to be saved
`burn`	number of initial scans of the Markov chain to be dropped

Value

a list with the following components:

`Z_postmean`	posterior mean of the latent variable in the probit specification
`ULU_postmean`	posterior mean of the reduced-rank approximating matrix
`Y_postmean`	the original data matrix with missing values replaced by posterior means
`L_postsamp`	samples of the eigenvalues
`b_postsamp`	samples of the regression coefficients
`Y`	original data matrix
`X`	original regressor array
`S`	number of scans of the Markov chain

Author(s)

Peter Hoff

Examples

```
data(YX_Friend)

fit<-eigenmodel_mcmc(Y=YX_Friend$Y,X=YX_Friend$X,R=2,S=750,burn=250)

# in general you  should run the Markov chain longer than 750 scans

plot(fit)

#fit<-eigenmodel_mcmc(Y=Y_Gen,R=3,S=10000)

#fit<-eigenmodel_mcmc(Y=Y_Pro,R=3,S=10000)
```

`eigenmodel_setup` *Setup constants and starting values for an eigenmodel fit*

Description

Setup constants and starting values for an eigenmodel fit

Usage

```
eigenmodel_setup(R = 0, seed = 1, em_env = .GlobalEnv)
```

Arguments

`R`	non-negative integer rank of the approximating matrix
`seed`	a random seed
`em_env`	enviromnemt within which to do the fitting

Value

NULL

Author(s)

Peter Hoff

plot.eigenmodel_post *Plot the output of an eigenmodel fit*

Description

A graphical display of MCMC output and posterior estimates of model parameters in an eigenmodel fit. Includes 95 percent quantile-based posterior confidence intervals of regression coefficients.

Usage

```
plot.eigenmodel_post(x, ...)
```

Arguments

x	an object of class eigenmodel_post
...	additional plotting options

Value

NULL

Author(s)

Peter Hoff

rb_fc *Sample from the full conditional distribution of the regression coefficients*

Description

Sample from the full conditional distribution of the regression coefficients in an eigenmodel

Usage

```
rb_fc(E = Z - ULU(UL))
```

Arguments

E	a symmetric matrix

Value

a p x 1 vector

Author(s)

Peter Hoff

rmvnorm *Sample from the multivariate normal distribution*

Description

Sample from the multivariate normal distribution

Usage

```
rmvnorm(mu, Sig2)
```

Arguments

mu	a p x 1 vector
Sig2	a p x p positive definite matrix

Value

a p x 1 vector

Author(s)

Peter Hoff

Examples

```
rmvnorm( c(0,0,0),diag(rep(3,1)) )
```

rUL_fc *Sample UL from its full conditional distribution*

Description

Samples the components of a reduced rank approximating matrix from their full conditional distributions

Usage

```
rUL_fc(E = Z - XB(X, b))
```

Arguments

E	an n x n symmetric matrix to be modeled with a reduced rank matrix

Value

A list with the following components:

U	an n x r matrix of eigenvectors
L	an r x r diagonal matrix of eigenvalues

Author(s)

Peter Hoff

---

rZ_fc *Sample from the full conditional distribution of the probit latent variables*

Description

Sample from the full conditional distribution of the latent variables in the ordered probit model

Usage

```
rZ_fc(EZ = XB(X, b) + ULU(UL), MH = TRUE)
```

Arguments

EZ	a symmetric matrix with elements equal to the expected values of the latent variables
MH	whether or not to do a Metropolis update in addition to the Gibbs sampling

Value

a symmetric matrix

Author(s)

Peter Hoff

ULU *Computes a matrix from its eigenvalue decomposition*

Description

Computes a matrix from its eigenvalue decomposition

Usage

```
ULU(UL)
```

Arguments

UL	a list with first component "U", an n x r matrix and the second component "L" an r x r diagonal matrix

Value

an n x n matrix

Author(s)

Peter Hoff

XB *Computes a sociomatrix of regression effects*

Description

Computes a sociomatrix of regression effects

Usage

```
XB(X, b)
```

Arguments

X	an n x n x p array
b	a p x 1 vector

Value

an n x n matrix

Author(s)

Peter Hoff

YX_Friend *Sex, race and friendship data from a 12th grade classroom*

Description

A list in which Y encodes the presence of a friendship tie between 90 12th graders. The array X indicates pairs of the same sex and of the same race.

Usage

```
data(YX_Friend)
```

Source

http://www.cpc.unc.edu/projects/addhealth/design

Examples

```
data(YX_Friend)
```

Y_Gen *Relations between words in the 1st chapter of Genesis*

Description

The i,j th entry of this matrix is the numerical count of the number of times word i was next to word j in the first chapter of Genesis.

Usage

```
data(Y_Gen)
```

Examples

```
data(Y_Gen)
```

Y_impute *Impute missing values of a sociomatrix*

Description

Impute missing values of a sociomatrix

Usage

```
Y_impute()
```

Details

Imputes missing values of a sociomatrix from a matrix of latent variables and an ordered-probit specification.

Value

symmetric matrix

Author(s)

Peter Hoff

Y_Pro *Butland's protein-protein interaction data*

Description

Butland's protein-protein interaction data

Usage

```
data(Y_Pro)
```

References

Butland et al (2005) "Interaction network containing conserved and essential protein complexes in Escherichia coli"

Examples

```
data(Y_Pro)
```

Index