R Functions for fitting RC and Order restricted RC Models

(for details see the paper of Iliopoulos et al.(2006) entitled "Bayesian Estimation of Unrestricted and Order-Restricted Association Models for a Two-Way Contingency Table")

May 16, 2006

Title R Functions for fitting RC and Order restricted RC Models

Version 1.0

Date 11 May 2006

Depends R (>= 2.0.1)

Author G. Iliopoulos, M. Kateri & I. Ntzoufras

Description Functions for fitting RC and Order restricted RC Models using MCMC methods. Details can be found in Iliopoulos et al.˜ (2006).

Maintainer Ioannis Ntzoufras <ntzoufras@aueb.gr>

URL http://www.stat-athens.aueb.gr/˜jbn/papers/paper15.htm

R topics documented:

`Installation` *Installation of the functions*

Description

Installation of the function for the Bayesian analysis of RC and order restricted RC models.

Usage

1...Put the file `rc_install.r in a directory (for example c:\dirname)`
`2...Install the functions using the command`

```
source(,c:/dirname/rc_install.r,)
```

Author(s)

1. Giorgos Iliopoulos, Department of Statistics and Insurance Science, University of Piraeus, Piraeus, Greece, ⟨geh@unipi.gr⟩ .

2. Maria Kateri, Department of Statistics and Insurance Science, University of Piraeus, Piraeus, Greece, ⟨mkateri@unipi.gr⟩ .

3. Ioannis Ntzoufras, Department of Statistics, Athens University of Economics and Business, Athens, Greece, ⟨ntzoufras@aueb.gr⟩ .

References

Iliopoulos, G., Kateri, M. and Ntzoufras, I. (2006). "Bayesian Estimation of Unrestricted and Order-Restricted Association Models for a Two-Way Contingency Table". (under revision for *Computational Statistics and Data Analysis*).

`mcmc.rc` *MCMC algorithm for the unrestricted RC model for an I×J contingency table*

Description

Returns the output of the MCMC algorithm for the RC model

Usage

```
mcmc.rc( y, totiter=2000, burnin=1000, ROW=TRUE, COLUMN=TRUE )
```

Arguments

`y`	I×J table of frequencies given in a `matrix` form
`totiter`	Total number of iterations
`ROW`	If `ROW=TRUE` then estimate row scores otherwise set them constant
`COLUMN`	If `COLUMN=TRUE` then estimate column scores otherwise set them constant

Details

This function implements MCMC algorithm and provides the output of the parameters of the unresticted RC model using the parametrization of Iliopoulos et al.(2006).

Value

A table of dimension $T \times p$ containing the MCMC generated parameters. T denotes the number of total iterations (`totiter`) while p denotes the number of parameters of the RC model (and is equal to $4 + 2I + 2J$). Each row represents the parameters generated in the corresponding iteration of the MCMC algorithm while each column represents the entire generated sample for the corresponding parameter.

See Also

`mcmc.rc.ordinal` , `mcmc.rc.ordinal.zeros` , `rc.boa` .

Examples

```
# dreams data is the illustrative example used in Iliopoulos et al. (2006)
#
# 11000 iterations for the uniform association (U) model for dreams data
res10000u<-mcmc.rc( dreams, totiter=11000, ROW=FALSE, COLUMN=FALSE)
#
# 11000 iterations for the row association (R) model for dreams data
res10000.r    <-mcmc.rc( dreams, totiter=11000, ROW=TRUE, COLUMN=FALSE)
#
# 11000 iterations for the column (C) association model for dreams data
res10000.c    <-mcmc.rc( dreams, totiter=11000, ROW=FALSE, COLUMN=TRUE)
#
# 11000 iterations for the row-column association (RC) model for dreams data
res10000.rc   <-mcmc.rc( dreams, totiter=11000, ROW=TRUE, COLUMN=TRUE)
```

`mcmc.rc.ordinal` *MCMC algorithm for the order restricted RC model for an I×J contingency table*

Description

Returns the output of the MCMC algorithm for the order restricted RC model

Usage

```
mcmc.rc.ordinal( y, totiter=2000,  ROW=TRUE, COLUMN=TRUE )
```

Arguments

`y`	I×J table of frequencies given in a `matrix` form
`totiter`	Total number of iterations
`ROW`	If `ROW=TRUE` then estimate row scores otherwise set them constant
`COLUMN`	If `COLUMN=TRUE` then estimate column scores otherwise set them constant

Details

This function implements MCMC algorithm and provides the output of the parameters of the order restricted RC model using the parametrization of Iliopoulos et al.(2006).

Value

A table of dimension $T \times p$ containing the MCMC generated parameters. T denotes the number of total iterations (`totiter`) while p denotes the number of parameters of the RC model (and is equal to $4 + 2I + 2J$). Each row represents the parameters generated in the corresponding iteration of the MCMC algorithm while each column represents the entire generated sample for the corresponding parameter.

See Also

`mcmc.rc` , `mcmc.rc.ordinal.zeros` , `rc.boa` .

Examples

```
# dreams data is the illustrative example used in Iliopoulos et al. (2006)
#
# 11000 iterations for the order restricted row association (R.ord) model
res10000.rord<-mcmc.rc.ordinal( dreams, 11000, ROW=TRUE, COLUMN=FALSE  )
#
# 11000 iterations for the order restricted column association (C.ord) model
res10000.cord<-mcmc.rc.ordinal( dreams, 11000, ROW=FALSE, COLUMN=TRUE  )
#
# 11000 iterations for the order restricted row column association (R.ord) model
res10000.rcord<-mcmc.rc.ordinal( dreams, 11000, ROW=TRUE, COLUMN=TRUE  )
```

---

mcmc.rc.ordinal.zeros

MCMC algorithm for the order restricted RC model for an I×J contingency table giving the possibility to set some SPECIFIC scores equal

---

Description

Returns the output of the MCMC algorithm for the order restricted RC model with some specific score equal ($\mu_2 = \mu_1$, $\mu_4 = \mu_3$ $\nu_3 = \nu_2$). WARNING: this function gives the possibility to set equal only the specific scores illustrated in Iliopoulos et al.(2006). If other equalities needed the function should be slightly modified.

Usage

```
mcmc.rc.ordinal.zeros( y, totiter=2000, ROW=TRUE, COLUMN=TRUE,
                       zerodm2=FALSE, zerodm4=FALSE, zerodv3=FALSE )
```

Arguments

`y`	I×J table of frequencies given in a `matrix` form
`totiter`	Total number of iterations
`ROW`	If `ROW=TRUE` then estimate row scores otherwise set them constant
`COLUMN`	If `COLUMN=TRUE` then estimate column scores otherwise set them constant
`zerodm2`	If `TRUE` then the difference $\delta_{\mu_2} = \mu_2 - \mu_1$ is restricted to zero for the fitted model
`zerodm4`	If `TRUE` then the difference $\delta_{\mu_4} = \mu_4 - \mu_3$ is restricted to zero for the fitted model
`zerodv3`	If `TRUE` then the difference $\delta_{\nu_3} = \nu_3 - \nu_2$ is restricted to zero for the fitted model

Details

This function implements MCMC algorithm and provides the output of the parameters of the order restricted RC model with specific equalities in scores using the parametrization of Iliopoulos et al.(2006).

Value

A table of dimension $T \times p$ containing the MCMC generated parameters. T denotes the number of total iterations (`totiter`) while p denotes the number of parameters of the fitted model. Each row represents the parameters generated in the corresponding iteration of the MCMC algorithm while each column represents the entire generated sample for the corresponding parameter.

See Also

`mcmc.rc` , `mcmc.rc.ordinal` , `rc.boa` .

Examples

```
# dreams data is the example using in Iliopoulos et al. (2006)
#
# RC.ord model with (mu2=mu1)
res10000zdm2<-mcmc.rc.ordinal.zeros( dreams, totiter=11000, burnin=1000, ROW=TRUE,
                          COLUMN=TRUE, zerodm2=TRUE, zerodm4=FALSE, zerodv3=FALSE )
#
# RC.ord model with (mu4=mu3)
res10000zdm2<-mcmc.rc.ordinal.zeros( dreams, totiter=11000, burnin=1000, ROW=TRUE,
                          COLUMN=TRUE, zerodm2=FALSE, zerodm4=TRUE, zerodv3=FALSE )
#
# RC.ord model with (v3=v2)
res10000zdv3<-mcmc.rc.ordinal.zeros( dreams, totiter=11000, burnin=1000, ROW=TRUE,
                          COLUMN=TRUE, zerodm2=FALSE, zerodm4=FALSE, zerodv3=TRUE )
#
# RC.ord model with (mu2=mu1, v3=v2)
res10000zdm2v3<-mcmc.rc.ordinal.zeros( dreams, totiter=11000, burnin=1000,
                ROW=TRUE, COLUMN=TRUE, zerodm2=TRUE, zerodm4=FALSE, zerodv3=TRUE )
#
# RC.ord model with (mu2=mu1, mu4=mu3, v3=v2)
res10000zdm2m4v3<-mcmc.rc.ordinal.zeros( dreams, totiter=11000, burnin=1000,
                 ROW=TRUE, COLUMN=TRUE, zerodm2=TRUE, zerodm4=TRUE, zerodv3=TRUE )
#
```

`rc.boa` *Bayesian output analysis for RC association models*

Description

This function is used for obtaining posterior summaries from the MCMC output of functions `mcmc.rc` and `mcmc.rc.ordinal` . Returns the posterior means, medians and mode for the given parametrization. Moreover, these posterior summaries are transformed in order to satisfy standard constraints (sum of scores equal to zero and sum of squared scores equal to one) for comparison reasons. Plots of the MCMC output and the posterior means of the contingency table probabilities under the fitted model are also provided. Finally, AIC, BIC and DIC values are calculated when the original data are supplied.

Usage

```
rc.boa( chain, burnin=0, transformation=TRUE, MODE=TRUE, PLOTS=TRUE,
        data=NULL, model=,RC, )
```

Arguments

`chain`	Matrix of dimension $T \times p$; where T is the total number of MCMC iterations and p is the number of parameters for the fitted association model.
`burnin`	Number of iterations to discard from the calculation of the posterior summaries.
`transformation`	If `TRUE` then the posterior summaries are transformed to satisfy standard parameterization (sum of scores equal to zero and sum of squared scores equal to one) .
`MODE`	If `TRUE` then the parameter values with the maximum posterior value are also provided.
`PLOTS`	If `TRUE` then plots for the evolution of each parameter are provided (trace plot, plot of ergodic means, histogram of the posterior distribution and autocorrelation plot).
`data`	Data for the fitted model given in a $I \times J$ matrix. This object is only used for the calculation of AIC, BIC and DIC.
`model`	Provides the type of the fitted model and it is used to calculate the number of free parameters in AIC and BIC. Available choices are: 'RC' (row column association model), 'R', (row association model) 'C', (column association model) 'U' (uniform association model) and 'I' (independence model).

Details

This function is used for the analysis of the MCMC output of `mcmc.rc` and `mcmc.rc.ordinal` functions. The plotting of each parameter is optional and is determined by a simple menu appearing on screen.

Value

A list object returned with the following variables.

`orig`	Data frame with the posterior mean, median and mode (if `MODE=TRUE`) values.
`transf`	Data frame with the transformed posterior mean, median and mode (if `MODE=TRUE`) values. The transformed values satisfy the standard constraints (sum of scores equal to zero and sum of squared scores equal to one) and are provided for comparison reasons when `transformation=TRUE`.
`fitted.means`	$I \times J$ array giving the posterior means of the cell probabilities for the fitted mode.
`AIC, BIC, DIC`	AIC, BIC and DIC values provided only if the data table is also provided. For AIC and BIC the values for the posterior mean probabilities as well as the posterior means and 2.5% and 97.5% percentiles are provided. For DIC, the estimated value, the deviance evaluated at the posterior mean probabilities, the mean of the deviance values and the number of 'effective' parameters p_m are provided.

See Also

`mcmc.rc` , `mcmc.rc.ordinal` .

Examples

```
#
# calculate the posterior summaries for res10000.rc
# with a burn in equal to 1000 iterations
 t3<-rc.boa(res10000.rc,burnin=1000)
#
 t3<-rc.boa(res10000.rc,burnin=1000, data=dreams)
```

`dreams` *Dreams Dataset*

Description

The object `dreams` is a 5×4 array describing the cross-classification of the severity of dreams' disturbance for boys aged 5 to 15. This classical dataset was introduced by Maxwell (1961) and is the cross-tabulation of the age of the child participating in the study (divided in 5 categories: 5–7, 8–9, 10–11, 12–13, 14–15) and the severity of his dreams disturbance (levels 1-4 from low to high). It was used as an illustrative example in Iliopoulos et al.(2006).

Format

A 5×4 array with

rows representing five age groups (5–7, 8–9, 10–11, 12–13, 14–15 years old) and

columns representing the severity of the dreams disturbance (levels from 1 to 4 from low to high).

Details

This dataset is used as an illustrative example in Iliopoulos et al.(2006).

Source

Maxwell, A.E. (1961). *Analyzing Qualitative Data*. London: Methuen.

Index