Package ‘bayMDS’

December 16, 2021

Type Package
Title Bayesian Multidimensional Scaling and Choice of Dimension
Version 1.6
Date 2021-12-13
License GPL (>= 2)
Depends R (>= 3.5.0)
Imports Rcpp (>= 1.0.7), progress, ggplot2, shinythemes, shiny, gridExtra, rgl, ggpubr
LinkingTo Rcpp, RcppArmadillo
Encoding UTF-8
RoxygenNote 7.1.2
NeedsCompilation yes
Author Man-Suk Oh [aut, cre], Eun-Kyung Lee [aut]
Maintainer Man-Suk Oh <msoh@ewha.ac.kr>
Repository CRAN
Date/Publication 2021-12-16 10:00:06 UTC

R topics documented:

bayMDSApp ................................................................. 2
bmds ................................................................. 2
bmdsMCMC ............................................................... 5
checkDIST .............................................................. 6
cityDIST .............................................................. 6
distRcpp ............................................................... 7
MDSIC ................................................................. 7
plotDelDist ............................................................ 8
plotObj ............................................................... 9
plotTrace ............................................................ 10
bayMDSApp

Shiny App for exploring the results of `bmds` function

Description

Call Shiny to show the results of Bayesian analysis of multidimensional scaling in a web-based application.

Usage

bayMDSApp(out)

Arguments

out an object of class `bmds`, the output of the `bmds` function

Value

open Shiny app

Examples

```r
data(cityDIST)
out <- bmds(cityDIST, min_p=1, max_p=6 )
if(interactive()){bayMDSApp(out)}
```

bmds

run `bmdsMCMC` for various number of dimensions

Description

Provide object configuration and estimates of parameters, for number of dimensions from `min_p` to `max_p`

Usage

```r
bmds(DIST,min_p=1, max_p=6,nwarm = 1000,niter = 5000,...)
```
Arguments

- **DIST**: symmetric data matrix of dissimilarity measures for pairs of objects
- **min_p**: minimum number of dimensions for object configuration (default=1)
- **max_p**: maximum number of dimensions for object configuration (default=6)
- **nwarm**: number of iterations for burn-in period in MCMC (default=1000)
- **niter**: number of MCMC iterations after burn-in period (default=5000)
- ... arguments to be passed to methods.

Details

**Model**

The basic model for Bayesian multidimensional scaling given in Oh and Raftery (2001) is as follows. Given the number of dimensions $p$, we assume that an observed dissimilarity measure follows a truncated multivariate normal distribution with mean equal to Euclidean distance, i.e.,

$$d_{ij} \sim N(\delta_{ij}, \sigma^2)I(d_{ij} > 0),$$

independently for $i \neq j, i, j = 1, \ldots, n$,

where

- $n$ is the number of objects, i.e., number of rows in DIST
- $d_{ij}$ is an observed dissimilarity measure between objects $i$ and $j$
- $\delta_{ij}$ is the distance between objects $i$ and $j$ in a $p$-dimensional Euclidean space, i.e.,

$$\delta_{ij} = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2}$$

- $x_i = (x_{i1}, \ldots, x_{ip})$ denotes the values of the attributes possessed by object $i$, i.e., the coordinates of object $i$ in a $p$-dimensional Euclidean space.

**Priors**

- Prior distribution of $x_i$ is given as a multivariate normal distribution with mean 0 and a diagonal covariance matrix $\Lambda$, i.e., $x_i \sim N(0, \Lambda)$, independently for $i = 1, \ldots, n$. Note that the zero mean and diagonal covariance matrix is assumed because Euclidean distance is invariant under translation and rotation of $X = \{x_i\}$.

- Prior distribution of the error variance $\sigma^2$ is given as $\sigma^2 \sim IG(a, b)$, the inverse Gamma distribution with mode $b/(a + 1)$.

- Hyperpriors for the elements of $\Lambda = diag(\lambda_1, \ldots, \lambda_p)$ are given as $\lambda_j \sim IG(\alpha, \beta_j)$, independently for $j = 1, \ldots, p$.

- We assume prior independence among $X, \Lambda, \sigma^2$.

**Measure of fit**

A measure of fit, called STRESS, is defined as

$$STRESS = \sqrt{\frac{\sum_{i>j} (d_{ij} - \hat{\delta}_{ij})^2}{\sum_{i>j} \delta_{ij}^2}},$$

where $\hat{\delta}_{ij}$ is the Euclidean distance between objects $i$ and $j$, computed from the estimated object configuration. Note that the squared $STRESS$ is proportional to the sum of squared residuals, $SSR = \sum_{i>j} (d_{ij} - \delta_{ij})^2$. 
Value

in bmds object

n  number of objects, i.e., number of rows in DIST
min_p  minimum number of dimensions
max_p  maximum number of dimensions
niter  number of MCMC iterations
nwarm  number of burn-in in MCMC
*  the following lists contains objects from bmdsMCMC for number of dimensions from min_p to max_p

x_bmds  a list of object configurations
minSSR.L  a list of minimum sum of squares of residuals between the observed dissimilarities and the estimated Euclidean distances between pairs of objects
minSSR_id.L  a list of the indecies of the iteration corresponding to minimum SSR
stress.L  a list of STRESS values
e_sigma.L  a list of posterior mean of $\sigma^2$
var_sigma.L  a list of posterior variance of $\sigma^2$
SSR.L  a list of posterior samples of SSR
lam.L  a list of posterior samples of elements of $\Lambda$
sigma.L  a list of posterior samples of $\sigma^2$, the error variance
del.L  a list of posterior samples of $\delta_s$,Euclidean distances between pairs of objects)
cmds.L  a list of object configuration from the classical multidimensional scaling of Togerson(1952)
BMDSp  a list of outputs from bmdsMCMC founction for each number of dimensions

References


Examples

data(cityDIST)
out <- bmds(cityDIST)
Description

run MCMC algorithm given in Oh and Raftery (2001) and return posterior samples of parameters as well as object configuration and other parameter estimates, for a given number of dimensions p

Usage

bmdsMCMC(DIST,p,nwarm = 1000,niter = 5000)

Arguments

- **DIST**: symmetric matrix of dissimilarity measures between objects
- **p**: number of dimensions of object configuration
- **nwarm**: number of iterations for burn-in period in MCMC (default=1000)
- **niter**: number of MCMC iterations after burn-in period (default=5000)

Value

A list of MCMC results

- **x_bmds**: n by p matrix of object configuration that minimizes the sum of squares of residuals(SSR), where n is the number of objects, i.e., n=nrow(DIST)
- **cmds**: n by p matrix of object configuration from the classical multidimensional scaling of Torgerson(1952)
- **minSSR**: minimum of sum of squares of residuals between the observed dissimilarities and the estimated Euclidean distances for pairs of objects
- **minSSR_id**: index of the iteration corresponding to minimum SSR
- **stress**: STRESS computed from minSSR
- **e_sigma**: posterior mean of σ²
- **var_sigma**: posterior variance of σ²
- **SSR.L**: niter dimensional vector of posterior samples of SSR
- **lam.L**: niter by p matrix of posterior samples of elements of Λ
- **sigma.L**: niter dimensional vector of posterior samples of σ²
- **del.L**: niter by n(n − 1)/2 matrix of posterior samples of δ, p-dimensional Euclidean distances between pairs of objects

References

Examples

```r
data(cityDIST)
result = bmdsMCMC(cityDIST, p = 3)
```

---

**checkDIST**

*check the dissimilarity matrix*

**Description**

Check the type of dissimilarity matrix and convert it to a symmetric full matrix for the input of `bmdsMCMC` and `bmds` function.

**Usage**

```r
checkDIST(dist, ...)```

**Arguments**

- `dist`: Dissimilarity measures for pairs of objects
- `...`: Arguments to be passed to methods

**Value**

A full matrix of dissimilarity measures

**Examples**

```r
x <- matrix(rnorm(100), nrow = 5)
dist(x)
checkDIST(dist(x))
```

---

**cityDIST**

*Airline distances between cities*

**Description**

Airline distances between 30 principal cities of the world. Cities are located on the surface of the earth, a three-dimensional sphere, and airplanes travel on the surface of the earth.

**References**


**Examples**

```r
data(cityDIST)
```
distRcpp

**Description**

calculate Euclidean distances between rows of matrix X

**Usage**

distRcpp(X)

**Arguments**

X
data matrix

**Value**
distance matrix

**Examples**

```r
x <- matrix(rnorm(100), nrow = 5)
distRcpp(x)
```

MDSIC

**Description**

compute and plot MDSIC, a Bayesian selection criterion, given in Oh and Raftery (2001) based on the output of the function bmds

**Usage**

MDSIC(x, plot = TRUE, ...)

**Arguments**

x an object of class bmds, the output of the function bmds

plot TRUE/FALSE, if TRUE plot the number of dimensions versus MDSIC (default=TRUE)

... arguments to be passed to methods
To compute MDSIC, output of the function `bmds` for `min_p=1` is needed for sequential calculation of MDSIC.

Value

- a list of MDSIC results
  - `mdsic` MDSIC, for p =1,...,max_p
  - `llike` log likelihood term in MDSIC, for p=1,...,max_p
  - `penalty` penalty term in MDSIC, for p=1,...,max_p

References


Examples

```r
data(cityDIST)
out <- bmds(cityDIST, min_p=1, max_p=5 )
MDSIC(out)
```

Description

plot Delta (estimated Euclidean distance from bmds) vs DIST (observed dissimilarity measure) for pairs of objects

Usage

`plotDelDist(out)`

Arguments

- `out` the output of the function `bmdsMCMC`

Value

plot of delta vs. dist
Examples

data(cityDIST)
result <- bmdsMCMC(cityDIST,p=3,nwarm=500,niter=3000)
plotDelDist(result)

plotObj

plot object configuration

Description

plot object configuration in a Euclidean space of two selected dimensions

Usage

plotObj(out, ...)

Arguments

out the output of the function bmdsMCMC
...

arguments to be passed to methods

Value

plot of object configuration

Examples

data(cityDIST)
result <- bmdsMCMC(cityDIST,p=3,nwarm=500,niter=3000)
plotObj(result)
plotTrace

trace plots of MCMC samples

Description
plot trace plots of MCMC samples of parameters for visual inspection of MCMC convergence

Usage
plotTrace(out, para = c("del"), linecolor = "blue", ...)

Arguments
out
the output of the function bmdsMCMC
para
names of the parameters for trace plots. It should be any subvector of c("del","sigma", "lambda") (default=c("del"))
linecolor
line color. The default color is blue.
...
arguments to be passed to methods

Details
Notes
• If "del" is in para, trace plots of the Euclidean distances from 4 randomly selected pairs will be given
• If "lambda" is in para, trace plots of the first four elements of Lambda, the diagonal prior variance of objects, will be given
• If "sigma" is in para, trace plot and ACF(Auto Correlation Function) plot of sigma, the error-variance will be given

Value
trace plots of delta, sigma and lambda

Examples
data(cityDIST)
result <- bmdsMCMC(cityDIST,p=3, nwarm=500, niter=3000)
plotTrace(result, para=c("del","sigma", "lambda"))
Index

* datasets
  cityDIST, 6
bayMDSApp, 2
bmds, 2
bmdsMCMC, 5
checkDIST, 6
cityDIST, 6
distRcpp, 7
MDSIC, 7
plotDelDist, 8
plotObj, 9
plotTrace, 10