Package 'eiPack'

February 19, 2023

Type Package

Version 0.2-2

Date 2023-02-18

Title Ecological Inference and Higher-Dimension Data Management

Author Olivia Lau <olivia.lau@post.harvard.edu>, Ryan T. Moore <rtm@american.edu>, Michael Kellermann <mrkellermann@gmail.com>

Maintainer Michael Kellermann <mrkellermann@gmail.com>

Depends R (>= 2.0.0)

Imports MASS, coda, msm

Suggests lattice

Description Provides methods for analyzing R by C ecological contingency tables using the extreme case analysis, ecological regression, and Multinomial-Dirichlet ecological inference models. Also provides tools for manipulating higher-dimension data objects.

License GPL (>= 2) | file LICENSE

URL http://www.olivialau.org/software/

NeedsCompilation yes

Repository CRAN

Date/Publication 2023-02-18 23:40:02 UTC

R topics documented:

---

bounds *Deterministic bounds for units satisfying row thresholds*

Description

Calculates the deterministic bounds on the proportion of row members within a specified column.

Usage

```
bounds(formula, data, rows, column, excluded = NULL,
    threshold = 0.9, total = NULL)
```

Arguments

`formula`	a formula of the form `cbind(col1, col2, ...) ~ cbind(row1, row2, ...)`. Column and row marginals must have the same total for each ecological unit.
`data`	a data frame containing the variables specified in `formula` and (optionally) `total`
`rows`	a character vector specifying the rows of interest
`column`	a character string specifying the column marginal of interest
`excluded`	an optional character string (or vector of character strings) specifying the columns to be excluded from the bounds calculation. For example, if the quantity of interest is Democratic share of the two-party vote, non-voters would be excluded.
`threshold`	the minimum proportion of the unit that row members must comprise for the bounds to be calculated for the unit. If `threshold = 0`, bounds will be calculated for all units.
`total`	if row and/or column marginals are given as proportions, `total` identifies the name of the variable in `data` containing the total number of individuals in each unit

Value

A list with elements

bounds	a list of deterministic bounds for all units in which row proportions meet the threshold
intersection	if the intersection of the deterministic bounding intervals is non-empty, the intersection is returned. Otherwise, NA is returned.

Author(s)

Ryan T. Moore <<rtm@american.edu>>

References

Otis Dudley Duncan and Beverley Davis. 1953. "An Alternative to Ecological Correlation." *American Sociological Review* 18: 665-666.

See Also

plot.bounds

---

cover.plot *Unit-level coverage plots for beta parameters from MD EI model*

Description

Generates a plot of central credible intervals for the unit-level beta parameters from the Multinomial-Dirichlet ecological inference model (see ei.MD.bayes).

Usage

```
cover.plot(object, row, column, x = NULL, CI = 0.95,
          medians = TRUE, col = NULL, ylim = c(0,1),
          ylab, lty = par("lty"), lwd = par("lwd"), ...)
```

Arguments

object	output from ei.MD.bayes
row	a character string specifying the row marginal of interest
column	a character string specifying the column marginal of interest
x	an optional covariate to index the units along the x-axis
CI	a fraction between 0 and 1 (defaults to 0.95), specifying the coverage of the central credible interval to be plotted for each unit
medians	a logical value specifying whether to plot the median (defaults to TRUE). If medians = FALSE, the medians are not plotted.

col	an optional vector of colors to be passed to plot and segments. If col is of length two, then the first color is used for plot and the second for segments.
ylim	an optional range for the y-axis (defaults to c(0,1)).
ylab	an optional label for the y-axis (defaults to Proportion of row in column).
lty	an optional line type passed to segments.
lwd	an optional line width argument passed to segments.
...	additional arguments passed to plot.

Value

A plot with vertical intervals indicating the central credible intervals for each ecological unit.

Author(s)

Olivia Lau <<olivia.lau@post.harvard.edu>>

See Also

plot, segments, par

---

densityplot *Density plots for population level parameters*

Description

Generates a density plot for population level quantities of interest output by lambda.MD, lambda.reg, and lambda.reg.bayes. For the Bayesian methods, densityplot plots the kernel density for the draws. For the frequentist lambda.reg method, densityplot plots the canonical Normal density conditional on the mean and standard error output by lambda.reg.

Usage

```
## S3 method for class 'lambdaMD'
densityplot(x, by = "column", col, xlim, ylim,
             main = "", sub = NULL, xlab, ylab,
             lty = par("lty"), lwd = par("lwd"), ...)
## S3 method for class 'lambdaRegBayes'
densityplot(x, by = "column", col, xlim, ylim,
             main = "", sub = NULL, xlab, ylab,
             lty = par("lty"), lwd = par("lwd"), ...)
## S3 method for class 'lambdaReg'
densityplot(x, by = "column", col, xlim, ylim,
             main = "", sub = NULL, xlab, ylab,
             lty = par("lty"), lwd = par("lwd"), ...)
```

Arguments

x	output from lambda.MD, lambda.reg, or lambda.reg.bayes.
by	character string (defaulting to "column") specifying whether to panel the density plot by "row" or "column" marginal.
col	an optional vector of colors, with length corresponding to the number of marginals selected in by. Defaults to rainbow.
xlim,ylim	optional limits for the x-axis and y-axis, passed to plot.
main,sub	optional title and subtitle, passed to plot.
xlab,ylab	optional labels for the x- and y-axes, passed to plot.
lty,lwd	optional arguments for line type and line width, passed to lines and plot. If either lty or lwd are vectors, it must correspond to the number of row or column marginals selected.
...	additional arguments passed to par.

Value

A plot with density lines for the selected margin (row or column).

Author(s)

Olivia Lau <<olivia.lau@post.harvard.edu>>

See Also

plot, segments, par

---

ei.MD.bayes *Multinomial Dirichlet model for Ecological Inference in RxC tables*

Description

Implements a version of the hierarchical model suggested in Rosen et al. (2001)

Usage

```
ei.MD.bayes(formula, covariate = NULL, total = NULL, data,
            lambda1 = 4, lambda2 = 2, covariate.prior.list = NULL,
            tune.list = NULL, start.list = NULL, sample = 1000, thin = 1,
            burnin = 1000, verbose = 0, ret.beta = 'r',
            ret.mcmc = TRUE, usrfun = NULL)
```

Arguments

formula	A formula of the form cbind(col1, col2, ...) ~ cbind(row1, row2, ...). Column and row marginals must have the same totals.
covariate	An optional formula of the form ~ covariate. The default is covariate = NULL, which fits the model without a covariate.
total	if row and/or column marginals are given as proportions, total identifies the name of the variable in data containing the total number of individuals in each unit
data	A data frame containing the variables specified in formula and total
lambda1	The shape parameter for the gamma prior (defaults to 4)
lambda2	The rate parameter for the gamma prior (defaults to 2)
covariate.prior.list	a list containing the parameters for normal prior distributions on delta and gamma for model with covariate. See 'details' for more information.
tune.list	A list containing tuning parameters for each block of parameters. See 'details' for more information. Typically, this will be a list generated by tuneMD. The default is NULL, in which case fixed tuning parameters are used.
start.list	A list containing starting values for each block of parameters. See 'details' for more information. The default is start.list = NULL, which generates appropriate random starting values.
sample	Number of draws to be saved from chain and returned as output from the function (defaults to 1000). The total length of the chain is sample*thin + burnin.
thin	an integer specifying the thinning interval for posterior draws (defaults to 1, but most problems will require a much larger thinning interval).
burnin	integer specifying the number of initial iterations to be discarded (defaults to 1000, but most problems will require a longer burnin).
verbose	an integer specifying whether the progress of the sampler is printed to the screen (defaults to 0). If verbose is greater than 0, the iteration number is printed to the screen every verboseth iteration.
ret.beta	A character indicating how the posterior draws of beta should be handled: 'r'eturn as an R object, 's'ave as .txt.gz files, 'd'iscard (defaults to r).
ret.mcmc	A logical value indicating how the samples from the posterior should be returned. If TRUE (default), samples are returned as coda mcmc objects. If FALSE, samples are returned as arrays.
usrfun	the name of an optional a user-defined function to obtain quantities of interest while drawing from the MCMC chain (defaults to NULL).

Details

ei.MD.bayes implements a version of the hierarchical Multinomial-Dirichlet model for ecological inference in $R \times C$ tables suggested by Rosen et al. (2001).

Let $r = 1, \ldots, R$ index rows, $C = 1, \ldots, C$ index columns, and $i = 1, \ldots, n$ index units. Let $N_{\cdot ci}$ be the marginal count for column c in unit i and X_{ri} be the marginal proportion for row r in unit i. Finally, let β_{rci} be the proportion of row r in column c for unit i.

The first stage of the model assumes that the vector of column marginal counts in unit i follows a Multinomial distribution of the form:

$$(N_{\cdot 1i}, \ldots, N_{\cdot Ci}) \sim \text{Multinomial}(N_i, \sum_{r=1}^{R} \beta_{r1i} X_{ri}, \ldots, \sum_{r=1}^{R} \beta_{rCi} X_{ri})$$

The second stage of the model assumes that the vector of β for row r in unit i follows a Dirichlet distribution with C parameters. The model may be fit with or without a covariate.

If the model is fit without a covariate, the distribution of the vector β_{ri} is :

$$(\beta_{r1i}, \ldots, \beta_{rCi}) \sim \text{Dirichlet}(\alpha_{r1}, \ldots, \alpha_{rC})$$

In this case, the prior on each α_{rc} is assumed to be:

$$\alpha_{rc} \sim \text{Gamma}(\lambda_1, \lambda_2)$$

If the model is fit with a covariate, the distribution of the vector β_{ri} is :

$$(\beta_{r1i}, \ldots, \beta_{rCi}) \sim \text{Dirichlet}(d_r \exp(\gamma_{r1} + \delta_{r1} Z_i), d_r \exp(\gamma_{r(C-1)} + \delta_{r(C-1)} Z_i), d_r)$$

The parameters γ_{rC} and δ_{rC} are constrained to be zero for identification. (In this function, the last column entered in the formula is so constrained.)

Finally, the prior for d_r is:

$$d_r \sim \text{Gamma}(\lambda_1, \lambda_2)$$

while γ_{rC} and δ_{rC} are given improper uniform priors if `covariate.prior.list = NULL` or have independent normal priors of the form:

$$\delta_{rC} \sim \text{N}(\mu_{\delta_{rC}}, \sigma^2_{\delta_{rC}})$$

$$\gamma_{rC} \sim \text{N}(\mu_{\gamma_{rC}}, \sigma^2_{\gamma_{rC}})$$

If the user wishes to estimate the model with proper normal priors on γ_{rC} and δ_{rC}, a list with four elements must be provided for `covariate.prior.list`:

- `mu.delta` an $R \times (C-1)$ matrix of prior means for Delta
- `sigma.delta` an $R \times (C-1)$ matrix of prior standard deviations for Delta
- `mu.gamma` an $R \times (C-1)$ matrix of prior means for Gamma
- `sigma.gamma` an $R \times (C-1)$ matrix of prior standard deviations for Gamma

Applying the model without a covariate is most reasonable in situations where one can think of individuals being randomly assigned to units, so that there are no aggregation or contextual effects. When this assumption is not reasonable, including an appropriate covariate may improve inferences; note, however, that there is typically little information in the data about the relationship of any given covariate to the unit parameters, which can lead to extremely slow mixing of the MCMC chains and difficulty in assessing convergence.

Because the conditional distributions are non-standard, draws from the posterior are obtained by using a Metropolis-within-Gibbs algorithm. The proposal density for each parameter is a univariate normal distribution centered at the current parameter value with standard deviation equal to the tuning constant; the only exception is for draws of γ_{rc} and δ_{rc}, which use a bivariate normal proposal with covariance zero.

The function will accept user-specified starting values as an argument. If the model includes a covariate, the starting values must be a list with the following elements, in this order:

- `start.dr` a vector of length R of starting values for Dr. Starting values for Dr must be greater than zero.
- `start.betas` an $R \times C$ by precincts array of starting values for Beta. Each row of every precinct must sum to 1.
- `start.gamma` an $R \times C$ matrix of starting values for Gamma. Values in the right-most column must be zero.
- `start.delta` an $R \times C$ matrix of starting values for Delta. Values in the right-most column must be zero.

If there is no covariate, the starting values must be a list with the following elements:

- `start.alphas` an $R \times C$ matrix of starting values for Alpha. Starting values for Alpha must be greater than zero.
- `start.betas` an $R \times C \times$ units array of starting values for Beta. Each row in every unit must sum to 1.

The function will accept user-specified tuning parameters as an argument. The tuning parameters define the standard deviation of the normal distribution used to generate candidate values for each parameter. For the model with a covariate, a bivariate normal distribution is used to generate proposals; the covariance of these normal distributions is fixed at zero. If the model includes a covariate, the tuning parameters must be a list with the following elements, in this order:

- `tune.dr` a vector of length R of tuning parameters for Dr
- `tune.beta` an $R \times (C-1)$ by precincts array of tuning parameters for Beta
- `tune.gamma` an $R \times (C-1)$ matrix of tuning parameters for Gamma
- `tune.delta` an $R \times (C-1)$ matrix of tuning parameters for Delta

If there is no covariate, the tuning parameters are a list with the following elements:

- `tune.alpha` an $R \times C$ matrix of tuning parameters for Alpha
- `tune.beta` an $R \times (C-1)$ by precincts array of tuning parameters for Beta

Value

A list containing

`draws` A list containing samples from the posterior distribution of the parameters. If a covariate is included in the model, the list contains:

- `Dr` Posterior draws for Dr parameters as an R ×sample matrix. If `ret.mcmc = TRUE`, `Dr` is an `mcmc` object.
- `Beta` Posterior draws for beta parameters. Only returned if `ret.beta = TRUE`. If `ret.mcmc = TRUE`, a (R * C * units) × sample matrix saved as an `mcmc` object. Otherwise, a R × C × units × sample array
- `Gamma` Posterior draws for gamma parameters. If `ret.mcmc = TRUE`, a (R * (C - 1)) × sample matrix saved as an `mcmc` object. Otherwise, a R × (C - 1) × sample array
- `Delta` Posterior draws for delta parameters. If `ret.mcmc = TRUE`, a (R * (C - 1)) × sample matrix saved as an `mcmc` object. Otherwise, a R ×(C - 1) × sample array
- `Cell.count` Posterior draws for the cell counts, summed across units. If `ret.mcmc = TRUE`, a (R * C) × sample matrix saved as an `mcmc` object. Otherwise, a R × C × sample array

If the model is fit without a covariate, the list includes:

- `Alpha` Posterior draws for alpha parameters. If `ret.mcmc = TRUE`, a (R * C) × sample matrix saved as an `mcmc` object. Otherwise, a R × C × sample array
- `Beta` Posterior draws for beta parameters. If `ret.mcmc = TRUE`, a (R * C * units) × sample matrix saved as an `mcmc` object. Otherwise, a R × C × units × sample array
- `Cell.count` Posterior draws for the cell counts, summed across units. If `ret.mcmc = TRUE`, a (R * C) × sample matrix saved as an`mcmc` object. Otherwise, a R × C × sample array

`acc.ratios` A list containing acceptance ratios for the parameters. If the model includes a covariate, the list includes:

- `dr.acc` A vector of acceptance ratios for `Dr` draws
- `beta.acc` A vector of acceptance ratios for `Beta` draws
- `gamma.acc` A vector of acceptance ratios for `Gamma` and `Delta` draws

If the model is fit without a covariate , the list includes:

- `alpha.acc` A vector of acceptance ratios for `Alpha` draws
- `beta.acc` A vector of acceptance ratios for `Beta` draws

`usrfun` Output from the optional `usrfn`

`call` Call to `ei.MD.bayes`

Author(s)

Michael Kellermann <<mrkellermann@gmail.com>> and Olivia Lau <<olivia.lau@post.harvard.edu>>

References

Martyn Plummer, Nicky Best, Kate Cowles, and Karen Vines. 2002. *Output Analysis and Diagnostics for MCMC (CODA)*. https://CRAN.R-project.org/package=coda.

Ori Rosen, Wenxin Jiang, Gary King, and Martin A. Tanner. 2001. "Bayesian and Frequentist Inference for Ecological Inference: The $R \times (C-1)$ Case." *Statistica Neerlandica* 55: 134-156.

See Also

lambda.MD, cover.plot, density.plot, tuneMD, mergeMD

`ei.reg` *Ecological regression*

Description

Estimate an ecological regression using least squares.

Usage

```
ei.reg(formula, data, ...)
```

Arguments

`formula`	An R formula object of the form `cbind(c1, c2, ...) ~ cbind(r1, r2, ...)`
`data`	data frame containing the variables specified in `formula`
`...`	Additional arguments passed to `lm`.

Details

For $i \in 1, \ldots, C$, C regressions of the form `c_i ~ cbind(r1, r2, ...)` are performed.

These regressions make use of the accounting identities and the constancy assumption, that $\beta_{rci} = \beta_{rc}$ for all i.

The accounting identities include

- –defining the population cell fractions β_{rc} such that $\sum_{c=1}^{C} \beta_{rc} = 1$ for every r
- –$\sum_{c=1}^{C} \beta_{rci} = 1$ for $r = 1, \ldots, R$ and $i = 1, \ldots, n$
- –$T_{ci} = \sum_{r=1}^{R} \beta_{rci} X_{ri}$ for $c = 1, \ldots, C$ and $i = 1 \ldots, n$

Then regressing

$$T_{ci} = \beta_{rc} X_{ri} + \epsilon_{ci}$$

for $c = 1, \ldots, C$ recovers the population parameters β_{rc} when the standard linear regression assumptions apply, including $E[\epsilon_{ci}] = 0$ and $Var[\epsilon_{ci}] = \sigma_c^2$ for all i.

Value

A list containing

call	the call to ei.reg
coefficients	an $R \times C$ matrix of estimated population cell fractions
se	an $R \times C$ matrix of standard errors for coefficients.
cov.matrices	A list of the C scaled variance-covariance matrices for each of the ecological regressions

Author(s)

Olivia Lau <<olivia.lau@post.harvard.edu>> and Ryan T. Moore <<rtm@american.edu>>

References

Leo Goodman. 1953. "Ecological Regressions and the Behavior of Individuals." *American Sociological Review* 18:663–664.

---

ei.reg.bayes *Ecological regression using Bayesian Normal regression*

Description

Estimate an ecological regression using Bayesian normal regression.

Usage

```
ei.reg.bayes(formula, data, sample = 1000, weights = NULL, truncate=FALSE)
```

Arguments

formula	An R formula object of the form cbind(c1, c2, ...) ~ cbind(r1, r2, ...)
data	data frame containing the variables specified in formula
sample	number of draws from the posterior
weights	a vector of weights
truncate	if TRUE, imposes a proper uniform prior on the unit hypercube for the coefficients; if FALSE, an improper uniform prior is assumed

Details

For $i \in 1, \ldots, C$, C Bayesian regressions of the form c_i ~ cbind(r1, r2, ...) are performed. See the documentation for ei.reg for the accounting identities and constancy assumption underlying this Bayesian linear model.

The sampling density is given by

$$y|\beta, \sigma^2, X \sim N(X\beta, \sigma^2 I)$$

The improper prior is $p(\beta, \sigma^2|X) \propto \sigma^{-2}$.

The proper prior is $p(\beta, \sigma^2|x) \propto I(\beta \in [0, 1]) \times \sigma^{-2}$.

Value

A list containing

call	the call to ei.reg.bayes
draws	A, $R \times C \times$ sample array containing posterior draws for each population cell fraction

Author(s)

Olivia Lau <<olivia.lau@post.harvard.edu>> and Ryan T. Moore <<rtm@american.edu>>

References

Leo Goodman. 1953. "Ecological Regressions and the Behavior of Individuals." *American Sociological Review* 18:663–664.

---

lambda.MD *Calculate shares using data from MD model*

Description

Calculates the population share of row members in a particular column as a proportion of the total number of row members in the selected subset of columns.

Usage

```
lambda.MD(object, columns, ret.mcmc = TRUE)
```

Arguments

object	an R object of class eiMD, output from ei.MD.bayes
columns	a character vector of column names to be included in calculating the shares
ret.mcmc	a logical value indicating how the samples from the posterior should be returned. If TRUE (default), samples are returned as mcmc objects. If FALSE, samples are returned as arrays.

Details

This function allows users to define subpopulations within the data and calculate the proportion of individuals within each of the columns that defines that subpopulation. For example, if the model includes the groups Democrat, Republican, and Unaffiliated, the argument columns = c(``Democrat", ``Republican") will calculate the two-party shares of Democrats and Republicans for each row.

Value

Returns either a (R * included columns) $\times$ samples) matrix as an mcmc object or a ($R\times$ included columns $\times$ samples) array.

Author(s)

Michael Kellermann <<mrkellermann@gmail.com>> and Olivia Lau <<olivia.lau@post.harvard.edu>>

See Also

ei.MD.bayes

---

lambda.reg *Calculate shares using data from regression model*

Description

Calculates the population share of row members in a particular column

Usage

```
lambda.reg(object, columns)
```

Arguments

object	An R object of class eiReg, the output from ei.reg
columns	a character vector of column names to be included in calculating the shares

Details

Standard errors are calculated using the delta method as implemented in the library msm. The arguments passed to deltamethod in msm include

- ga list of transformations of the form ~ x1 / (x1 + x2 + + ... + xk), ~ x2 / (x1 + x2 + ... + xk), etc.. Each x_c is the estimated proportion of all row members in column c, $\hat{\beta}_{rc}$
- meanthe estimated proportions of the row members in the specified columns, as a proportion of the total number of row members, $(\hat{\beta}_{r1}, \hat{\beta}_{r2}, ..., \hat{\beta}_{rk})$.
- cova diagonal matrix with the estimated variance of each $\hat{\beta}_{rc}$ on the diagonal. Each column marginal is assumed to be independent, such that the off-diagonal elements of this matrix are zero. Estimates come from object$cov.matrices, the estimated covariance matrix from the regression of the relevant column. Thus,

$$\text{cov} = \begin{matrix} Var(\hat{\beta}_{r1}) & 0 & 0 & \dots \\ 0 & Var(\hat{\beta}_{r2}) & 0 & \dots \\ 0 & 0 & Var(\hat{\beta}_{r3}) & \dots \\ \vdots & \vdots & \vdots & \ddots \end{matrix}$$

Value

Returns a list with the following elements

call	the call to lambda.reg
lambda	an $R \times k$ matrix where k is the number of columns included in the share calculation
se	standard errors calculated using the delta method as implemented in the library msm

Author(s)

Ryan T. Moore <<rtm@american.edu>>

See Also

ei.reg

---

lambda.reg.bayes *Calculate shares using data from Bayesian regression model*

Description

Calculates the population share of row members in selected columns

Usage

```
lambda.reg.bayes(object, columns, ret.mcmc = TRUE)
```

Arguments

object	An R object of class eiRegBayes, the output from ei.reg.bayes
columns	a character vector indicating which column marginals to be included in calculating the shares
ret.mcmc	If TRUE, posterior shares are returned as an mcmc object.

Value

If ret.mcmc = TRUE, draws are returned as an mcmc object with dimensions sample $\times C$. If ret.mcmc = FALSE, draws are returned as an array with dimensions $R \times C \times$ samples array.

Author(s)

Ryan T. Moore <<rtm@american.edu>>

See Also

ei.reg.bayes

mergeMD *Combine output from multiple eiMD objects*

Description

Allows users to combine output from several chains output by ei.MD.bayes

Usage

```
mergeMD(list, discard = 0)
```

Arguments

list	A list containing the names of multiple eiMD objects generated from the same model.
discard	The number of draws to discard from the beginning of each chain. Default is to retain all draws.

Value

Returns an eiMD object of the same format as the input.

Author(s)

Michael Kellermann <<mrkellermann@gmail.com>>

References

Martyn Plummer, Nicky Best, Kate Cowles, and Karen Vines. 2002. *Output Analysis and Diagnostics for MCMC (CODA)*. https://CRAN.R-project.org/package=coda.

Ori Rosen, Wenxin Jiang, Gary King, and Martin A. Tanner. 2001. "Bayesian and Frequentist Inference for Ecological Inference: The $R \times C$ Case." *Statistica Neerlandica* 55: 134-156.

See Also

ei.MD.bayes

plot.bounds *Plot of deterministic bounds for units satisfying row thresholds*

Description

Plots the deterministic bounds on the proportion of row members within a specified column.

Usage

```
## S3 method for class 'bounds'
plot(x, row, column, labels = TRUE, order = NULL,
    intersection = TRUE, xlab, ylab, col = par("fg"),
    lty = par("lty"), lwd = par("lwd"), ...)
```

Arguments

x	output from bounds
row	a character string specifying the row of interest
column	a character string specifying the column of interest
labels	a logical toggle specifying whether precinct labels should be printed above interval bounds
order	an optional vector of values between 0 and 1 specifying the order (left-to-right) in which interval bounds are plotted
intersection	a logical toggle specifying whether the intersection of all plotted bounds (if it exists) should be plotted
xlab, ylab, ...	additional arguments passed to plot
col, lty, lwd	additional arguments passed to segments

Value

A plot with vertical intervals indicating the deterministic bounds on the quantity of interest, and (optionally) a single horizontal interval indicating the intersection of these unit bounds.

Author(s)

Ryan T. Moore <<rtm@american.edu>>

See Also

bounds

read.betas *Function to read in eiMD parameter chains saved to disk*

Description

In ei.MD.bayes, users have the option to save parameter chains for the unit-level betas to disk rather than returning them to the workspace. This function reconstructs the parameter chains by reading them back into R and producing either an array or an mcmc object.

Usage

```
read.betas(rows, columns, units, dir = NULL, ret.mcmc = TRUE)
```

Arguments

rows	a character vector of the row marginals to be read back in
columns	a character vector of the column marginals to be read back in
units	a character of numeric vector with the units to be read back in
dir	an optional character string identifying the directory in which parameter chains are stored (defaults to getwd)
ret.mcmc	a logical value specifying whether to return the parameters as an mcmc object (defaults to TRUE)

Value

If ret.mcmc = TRUE, an mcmc object with row names corresponding to the parameter chains. If ret.mcmc = FALSE, an array with dimensions named according to the selected rows, columns, and units.

Author(s)

Olivia Lau <olivia.lau@post.harvard.edu>

See Also

ei.MD.bayes,mcmc

redistrict *Redistricting Monte-Carlo data*

Description

Precinct-level observations for a hypothetical jurisdiction with four proposed districts.

Usage

```
data(redistrict)
```

Format

A table containing 150 observations and 9 variables:

precinct precinct identifier

district proposed district number

avg.age average age

per.own percent homeowners

black number of black voting age persons

white number of white voting age persons

hispanic number of hispanic voting age persons

total total number of voting age persons

dem Number of votes for the Democratic candidate

rep Number of votes for the Republican candidate

no.vote Number of non voters

Source

Daniel James Greiner

senc *Party registration in south-east North Carolina*

Description

Registration data for White, Black, and Native American voters in eight counties of south-eastern North Carolina in 2001.

Usage

```
data(senc)
```

Format

A table containing 212 observations and 18 variables:

county county name

precinct precinct name

total number of registered voters in precinct

white number of White registered voters

black number of Black registered voters

natam number of Native American registered voters

dem number of registered Democrats

rep number of registered Republicans

other number of registered voters without major party affiliation

whdem number of White registered Democrats

whrep number of White registered Republicans

whoth number of White registered voters without major party affiliation

bldem number of Black registered Democrats

blrep number of Black registered Republicans

bloth number of Black registered voters without major party affiliation

natamdem number of Native American registered Democrats

natamrep number of Native American registered Republicans

natamoth number of Native American registered voters without major party affiliation

Source

Excerpted from North Carolina General Assembly 2001 redistricting data, https://www.ncleg.gov/Redistricting/BaseData200

---

tuneA *Tuning parameters for alpha hyperpriors in RxC EI model*

Description

Tuning parameters for hyperpriors in RxC EI model

Usage

```
data(tuneA)
```

Format

A table containing 3 rows and 3 columns.

tuneB — *Tuning parameters for the precinct level parameters in the RxC EI model*

Description

A vector containing tuning parameters for the precinct level parameters in the RxC EI model.

Usage

```
data(tuneB)
```

Format

A vector of length 3 x 2 x 150 containing the precinct level tuning parameters for the redistricting sample data.

Examples

```
data(tuneB)
tuneB <- array(tuneB[[1]], dim = c(3, 2, 150))
```

tuneMD — *Generate tuning parameters for MD model*

Description

An adaptive algorithm to generate tuning parameters for the MCMC algorithm implemented in `ei.MD.bayes`. Since we are drawing each parameter one at a time, target acceptance rates are between 0.4 to 0.6.

Usage

```
tuneMD(formula, covariate = NULL, data, ntunes = 10,
    totaldraws = 10000, ...)
```

Arguments

`formula`	A formula of the form `cbind(col1, col2, ...) ~ cbind(row1, row2, ...)` with rows as the predictor and columns as the response
`covariate`	An R formula for the optional covariate in the form `~ x`
`data`	data frame containing the variables specified in `formula` and `covariate`
`ntunes`	number of times to iterate the tuning algorithm
`totaldraws`	number of iterations for each tuning run
`...`	additional arguments passed to `ei.MD.bayes`

Value

A list containing matrices of tuning parameters.

Author(s)

Olivia Lau <<olivia.lau@post.harvard.edu>>

See Also

ei.MD.bayes

Index