Package ‘bsreg’

February 25, 2022

Type Package
Title Bayesian Spatial Regression Models
Version 0.0.2
Date 2022-02-25
Author Nikolas Kuschnig [aut, cre] (<https://orcid.org/0000-0002-6642-2543>)
Maintainer Nikolas Kuschnig <nikolas.kuschnig@wu.ac.at>
Description Fit Bayesian models with a focus on the spatial econometric models.
Depends R (>= 3.5.0)
Imports R6, Matrix, stats, graphics, utils, grDevices
Suggests stochvol, coda
License GPL-3 | file LICENSE
Encoding UTF-8
LazyData true
RoxygenNote 7.1.2
NeedsCompilation no
Repository CRAN
Date/Publication 2022-02-25 14:40:02 UTC

R topics documented:

bsreg-package .......................................................... 2
bm ................................................................. 2
cigarettes .......................................................... 3
coda ................................................................. 4
sample ............................................................... 5
set_mh ............................................................... 5
set_NG ............................................................. 6
set_options ......................................................... 7
set_SAR ............................................................ 8
set_SV ............................................................. 9
tune ............................................................... 10
us_states .......................................................... 11
Index

bsreg-package  
Bayesian Spatial Regression Models

Description

Fit Bayesian models with a focus on the spatial econometric models.

bm

Fit a Bayesian model

Description

Fit a Bayesian model

Usage

bm(x, ...)

## S3 method for class 'formula'
bm(
  x,
  data = NULL,
  n_save = 1000L,
  n_burn = 500L,
  options = set_options(),
  mh = set_mh(),
  verbose = TRUE,
  W,
  X_SLX,
  type = c("lm", "slx", "sar", "sem", "sdm", "sdem", "sv"),
  ...
)

## S3 method for class 'bm'
bm(x, n_save = 1000L, n_burn = 0L, verbose = TRUE, ...)

blm(...)

bslx(...)

bsar(...)

bsem(...)

bm
cigarettes

bsdm(...)  
bsdem(...)  
bsv(...)  

Arguments

x
Formula or bm object to sample with.
...
Not used.
data
A data.frame containing the variables in the model.
n_save, n_burn
Integer scalar. Number of draws for the burn-in period and to store for inference.
options
Settings for the prior setup. See set_options.
mh
Settings to tune the Metropolis-Hastings step. See set_mh.
verbose
Logical scalar. Whether to print status updates.
W
Numeric matrix (or function to construct one) with the spatial connectivities.
X_SLX
Numeric matrix with explanatory variables that should be lagged spatially.
type
Character scalar used to specify the desired model.

Value

Returns a list with draws from the specified Bayesian model and an object to obtain further samples.

Examples

N <- 100L  
beta <- 1:5  
X <- matrix(rnorm(N * 5), N, 5)  
y <- X %*% beta + rnorm(N)  

bm(y ~ X, n_burn = 100, n_draw = 100)

# Reproduce the linear model in Kuschnig (2022)
blm(log(sales) ~ log(price / cpi) + log(ndi / cpi) +  
    factor(name) + factor(year), data = cigarettes)

cigarettes  Cigarette demand

Description

Panel dataset on cigarette demand in 46 US states from 1963 until 1992, see Baltagi and Levin (1992) and Baltagi and Li (2004). Extended with longitude and latitude from the us_states dataset.
Usage

cigarettes

Format

A data.frame object.

References


coda

Methods for coda Markov chain Monte Carlo objects

Description

Methods to convert parameter and/or coefficient draws to coda’s mcmc format for further processing.

Usage

as.mcmc.bm(x, ...)

Arguments

x

A bm object, obtained from bm.

... Other parameters for as.mcmc.

Value

Returns a coda mcmc object.
Obtain draws from a Bayesian model sampler

**Usage**

```r
sample(x, n_save = 1000L, n_burn = 0L, mh = set_mh(), verbose = TRUE)
```

**Arguments**

- `x`: Bayesian model
- `n_save`: Integer scalar with number of draws to save / burn.
- `n_burn`: Integer scalar with number of burn-in draws.
- `mh`: Settings to tune the Metropolis-Hastings step. See `set_mh`.
- `verbose`: Logical scalar. Whether to print status updates.

**Value**

Returns a numeric matrix with stored draws. The Bayesian model is modified in place.

---

Settings to tune a Metropolis-Hastings step

**Usage**

```r
set_mh(adjust_burn = 0.8, acc_target = c(0.2, 0.45), acc_change = 0.01)
```

**Arguments**

- `adjust_burn`: Numeric scalar with the percentage of burn-in that should be used to tune the MH step.
- `acc_target`: Numeric vector with the lower and upper bound of the target acceptance rate for the MH step.
- `acc_change`: Numeric scalar with the percentage adjustment to the proposal scale for tuning.

**Value**

Returns a list with settings to tune the Metropolis-Hastings step of a Bayesian model.
Examples

```r
set_mh(0.5, c(0.1, 0.5), .05)
```

---

**set_NG**

*Set up a Normal-Gamma prior*

### Description

Set up a Normal-Gamma prior

### Usage

```r
set_NG(
  mu = 0,
  precision = 1e-08,
  shape = 0.01,
  rate = 0.01,
  beta = NULL,
  sigma = NULL
)
```

```r
set_SNG(
  lambda_a = 0.01,
  lambda_b = 0.01,
  theta_scale = 0,
  theta_a = 1,
  lambda = 1,
  tau = 10,
  theta = 0.1
)
```

```r
set_HS(lambda = 1, tau = 1, zeta = 1, nu = 1)
```

### Arguments

- **mu**: Numeric scalar or vector with the prior mean of 'beta'.
- **precision**: Numeric scalar or matrix with the prior precision of 'beta'. Not used for shrinkage priors.
- **shape, rate**: Numeric scalars with the prior shape and rate of 'sigma'.
- **lambda_a, lambda_b**: Numeric scalars with the prior shape and rate of 'lambda'.
- **theta_scale**: Numeric scalar with the proposal scale of 'theta'. Defaults to zero for a fixed value.
- **theta_a**: Numeric scalar with the prior rate of 'theta'.
- **lambda, tau, theta, zeta, nu, beta, sigma**: Numerics with starting values for the respective parameter.
Value

Returns a list with priors and settings.

set_options

Set up Bayesian model priors and settings

Description

Set up Bayesian model priors and settings

Usage

set_options(
  type = c("Independent", "Conjugate", "Shrinkage", "Horseshoe"),
  NG = set_NG(),
  SNG = set_SNG(),
  HS = set_HS(),
  SAR = set_SAR(),
  SLX = set_SLX(),
  SEM = set_SEM(),
  SV = set_SV(),
  ...
)

Arguments

  type               Character scalar with the prior type for the nested linear model.
  NG                 Settings for the Normal-Gamma prior (independent or conjugate). See set_NG.
  SNG                Settings for the Normal-Gamma shrinkage prior (Polson and Scott, 2010). See set_NG.
  HS                 Settings for the Horseshoe shrinkage prior (Makalic and Schmidt, 2015). See set_NG.
  SAR                Settings for the spatial autoregressive setup. See set_SAR.
  SLX                Settings for the spatially lagged explanatory setup. See set_SAR. Note that settings for the spatial term 'theta' are provided to NG instead.
  SEM                Settings for the spatial error setup. See set_SAR.
  SV                 Settings for the stochastic volatility setup. See set_SV.
  ...                Used to provide custom prior elements.

Value

Returns a list with priors and settings for a Bayesian model.

Examples

set_options("Shrinkage", SNG = set_SNG(lambda_a = 1, lambda_b = 1))
set_SAR **Set up a spatial prior**

Description

Set up a spatial prior

Usage

```r
set_SAR(
  lambda_a = 1.01,
  lambda_b = 1.01,
  lambda = 0,
  lambda_scale = 0.1,
  lambda_min = -1,
  lambda_max = 1 - 1e-12,
  delta_a = 1.01,
  delta_b = 1.01,
  delta = 1,
  delta_scale = 0,
  delta_min = 1e-12,
  delta_max = Inf
)
```

```r
set_SLX(
  lambda_a = 1.01,
  lambda_b = 1.01,
  lambda = 0,
  lambda_scale = 0.1,
  lambda_min = -1,
  lambda_max = 1 - 1e-12,
  delta_a = 1.01,
  delta_b = 1.01,
  delta = 1,
  delta_scale = 0,
  delta_min = 1e-12,
  delta_max = Inf
)
```

```r
set_SEM(
  lambda_a = 1.01,
  lambda_b = 1.01,
  lambda = 0,
  lambda_scale = 0.1,
  lambda_min = -1,
  lambda_max = 1 - 1e-12,
  delta_a = 1.01,
```
delta_b = 1.01,
delta = 1,
delta_scale = 0,
delta_min = 1e-12,
delta_max = Inf
)

Arguments

lambda_a, lambda_b
  Numeric scalars with the prior shapes of the connectivity strength 'lambda'.
lambda, delta
  Numerics with starting values for the respective parameter.
lambda_scale
  Numeric scalar with the proposal scale of 'lambda'.
lambda_min, lambda_max
  Numeric scalars with upper and lower bounds for 'lambda'.
delta_a, delta_b
  Numeric scalars with the prior shapes of the connectivity parameter 'delta'.
delta_scale
  Numeric scalar with the proposal scale of 'delta'. Defaults to zero for a fixed value.
delta_min, delta_max
  Numeric scalars with upper and lower bounds for 'delta'.

Value

Returns a list with priors and settings.

set_SV

Set up a volatility prior

Description

Set up a volatility prior

Usage

set_SV(
  priors,
  mu = 0,
  phi = 0.5,
  sigma = 1,
  nu = Inf,
  rho = 0,
  beta = 0,
  latent0 = 0
)
tune

Arguments

priors Prior settings from specify_priors.
mu, phi, sigma, nu, rho, beta, latent0
   Numerics with starting values for the respective parameter.

Value

Returns a list with priors and settings.

---

tune  
Burn-in and tune a Bayesian model sampler

Description

Burn-in and tune a Bayesian model sampler

Usage

tune(x, n_burn = 1000L, mh = set_mh(), verbose = TRUE)
burn(x, n_burn = 1000L, verbose = TRUE)

Arguments

x  Bayesian model
n_burn  Integer scalar with number of draws to save / burn.
mh  Settings to tune the Metropolis-Hastings step. See set_mh.
verbose  Logical scalar. Whether to print status updates.

Value

Modifies the Bayesian model in place and returns it invisibly.
us_states

United States Historical States

Description


Usage

us_states

Format

A data.frame object.

Source

https://publications.newberry.org/ahcbp/

References

Index

* datasets
  cigarettes, 3
  us_states, 11

as.mcmc, 4
as.mcmc.bm (coda), 4

blm (bm), 2
bm, 2, 4
bsar (bm), 2
bsdem (bm), 2
bsdm (bm), 2
bsem (bm), 2
bslx (bm), 2
bsreg-package, 2
bsv (bm), 2
burn (tune), 10

cigarettes, 3
coda, 4

data.frame, 3

mcmc, 4

sample, 5
set_HS (set_NS), 6
set_mh, 3, 5, 10
set_NS, 6, 7
set_options, 3, 7
set_SAR, 7, 8
set_SEM (set_SAR), 8
set_SLX (set_SAR), 8
set_SNG (set_NS), 6
set_SV, 7, 9
specify_priors, 10

tune, 10

us_states, 11