Package ‘ScaleSpikeSlab’

January 20, 2025

Type Package

Title Scalable Spike-and-Slab

Version 1.0

Date 2022-05-13

Description A scalable Gibbs sampling implementation for high dimensional Bayesian regression with the continuous spike-and-slab prior. Niloy Biswas, Lester Mackey and Xiao-Li Meng, ``Scalable Spike-and-Slab'' (2022) <arXiv:2204.01668>.

License GPL (>= 2)

Imports Rcpp, stats, TruncatedNormal

LinkingTo Rcpp, RcppEigen

RoxygenNote 7.1.2

NeedsCompilation yes

Author Niloy Biswas [aut, cre] (<`https://orcid.org/0000-0001-9081-5702`>),
Lester Mackey [aut],
Xiao-Li Meng [aut]

Maintainer Niloy Biswas <`niloy_biswas@g.harvard.edu`>

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2022-05-18 17:00:07 UTC

Contents

`riboflavin` *Riboflavin GWAS dataset*

Description

Dataset of riboflavin production by Bacillus subtilis containing n = 71 observations of a one-dimensional response (riboflavin production) and p = 4088 predictors (gene expressions). The one-dimensional response corresponds to riboflavin production.

Usage

```
data(riboflavin)
```

Format

A data frame containing a vector y of length 71 (responses) and a matrix X of dimension 71 by 4088 (gene expressions)

Details

The processed dataset is the same as in the R packages qut and hdi.

References

Buhlmann, P., Kalisch, M. and Meier, L. (2014) *High-dimensional statistics with a view towards applications in biology*. Annual Review of Statistics and its Applications **1**, 255–278

Examples

```
data(riboflavin)
y <- as.vector(riboflavin$y)
X <- as.matrix(riboflavin$x)
```

`spike_slab_linear` *spike_slab_linear*

Description

Generates Markov chain targeting the posterior corresponding to Bayesian linear regression with spike and slab priors

Usage

```
spike_slab_linear(
  chain_length,
  X,
  y,
  tau0,
  tau1,
  q,
  a0 = 1,
  b0 = 1,
  rinit = NULL,
  verbose = FALSE,
  burnin = 0,
  store = TRUE,
  Xt = NULL,
  XXt = NULL,
  tau0_inverse = NULL,
  tau1_inverse = NULL
)
```

Arguments

`chain_length`	Markov chain length
`X`	matrix of length n by p
`y`	Response
`tau0`	prior hyperparameter (non-negative real)
`tau1`	prior hyperparameter (non-negative real)
`q`	prior hyperparameter (strictly between 0 and 1)
`a0`	prior hyperparameter (non-negative real)
`b0`	prior hyperparameter (non-negative real)
`rinit`	initial distribution of Markov chain (default samples from the prior)
`verbose`	print iteration of the Markov chain (boolean)
`burnin`	chain burnin (non-negative integer)
`store`	store chain trajectory (boolean)
`Xt`	Pre-calculated transpose of X
`XXt`	Pre-calculated matrix X*transpose(X) (n by n matrix)
`tau0_inverse`	Pre-calculated matrix inverse(I + tau0^2*XXt) (n by n matrix)
`tau1_inverse`	Pre-calculated matrix inverse(I + tau1^2*XXt) (n by n matrix)

Value

Output from Markov chain targeting the posterior corresponding to Bayesian linear regression with spike and slab priors

Examples

```
# Synthetic dataset
syn_data <- synthetic_data(n=100,p=200,s0=5,error_std=2,type='linear')
X <- syn_data$X
y <- syn_data$y

# Hyperparamters
params <- spike_slab_params(n=nrow(X),p=ncol(X))

# Run S^3
sss_chain <- spike_slab_linear(chain_length=4e3,burnin=1e3,X=X,y=y,
tau0=params$tau0,tau1=params$tau1,q=params$q,a0=params$a0,b0=params$b0,
verbose=FALSE,store=FALSE)

# Use posterior probabilities for variable selection
sss_chain$z_ergodic_avg[1:10]
```

spike_slab_logistic *spike_slab_logistic*

Description

Generates Markov chain targeting the posterior corresponding to Bayesian logistic regression with spike and slab priors

Usage

```
spike_slab_logistic(
  chain_length,
  X,
  y,
  tau0,
  tau1,
  q,
  rinit = NULL,
  verbose = FALSE,
  burnin = 0,
  store = TRUE,
  Xt = NULL,
  XXt = NULL
)
```

Arguments

chain_length	Markov chain length
X	matrix of length n by p
y	Response

`tau0`	prior hyperparameter (non-negative real)
`tau1`	prior hyperparameter (non-negative real)
`q`	prior hyperparameter (strictly between 0 and 1)
`rinit`	initial distribution of Markov chain (default samples from the prior)
`verbose`	print iteration of the Markov chain (boolean)
`burnin`	chain burnin (non-negative integer)
`store`	store chain trajectory (boolean)
`Xt`	Pre-calculated transpose of X
`XXt`	Pre-calculated matrix X*transpose(X) (n by n matrix)

Value

Output from Markov chain targeting the posterior corresponding to Bayesian logistic regression with spike and slab priors

Examples

```
# Synthetic dataset
syn_data <- synthetic_data(n=100,p=200,s0=5,error_std=2,type='logistic')
X <- syn_data$X
y <- syn_data$y

# Hyperparamters
params <- spike_slab_params(n=nrow(X),p=ncol(X))

# Run S^3
sss_chain <- spike_slab_logistic(chain_length=4e3,burnin=1e3,X=X,y=y,
tau0=params$tau0,tau1=params$tau1,q=params$q,verbose=FALSE,store=FALSE)

# Use posterior probabilities for variable selection
sss_chain$z_ergodic_avg[1:10]
```

`spike_slab_params` *spike_slab_params*

Description

Generates hyperparameters for spike-and-slab

Usage

```
spike_slab_params(n, p)
```

Arguments

`n`	number of observations
`p`	number of covariates

Value

spike-and-slab hyperparameters q, tau0, tau1, a0, b0

Examples

```
hyper_params <- spike_slab_params(n=100,p=200)
print(hyper_params)
```

`spike_slab_probit` *spike_slab_probit*

Description

Generates Markov chain targeting the posterior corresponding to Bayesian probit regression with spike and slab priors

Usage

```
spike_slab_probit(
  chain_length,
  X,
  y,
  tau0,
  tau1,
  q,
  rinit = NULL,
  verbose = FALSE,
  burnin = 0,
  store = TRUE,
  Xt = NULL,
  XXt = NULL,
  tau0_inverse = NULL,
  tau1_inverse = NULL
)
```

Arguments

`chain_length`	Markov chain length
`X`	matrix of length n by p
`y`	Response
`tau0`	prior hyperparameter (non-negative real)
`tau1`	prior hyperparameter (non-negative real)
`q`	prior hyperparameter (strictly between 0 and 1)
`rinit`	initial distribution of Markov chain (default samples from the prior)
`verbose`	print iteration of the Markov chain (boolean)

`burnin`	chain burnin (non-negative integer)
`store`	store chain trajectory (boolean)
`Xt`	Pre-calculated transpose of X
`XXt`	Pre-calculated matrix X*transpose(X) (n by n matrix)
`tau0_inverse`	Pre-calculated matrix inverse(I + tau0^2*XXt) (n by n matrix)
`tau1_inverse`	Pre-calculated matrix inverse(I + tau1^2*XXt) (n by n matrix)

Value

Output from Markov chain targeting the posterior corresponding to Bayesian logistic regression with spike and slab priors

Examples

```
# Synthetic dataset
syn_data <- synthetic_data(n=100,p=200,s0=5,error_std=2,type='probit')
X <- syn_data$X
Xt <- t(X)
y <- syn_data$y

# Hyperparamters
params <- spike_slab_params(n=nrow(X),p=ncol(X))

# Run S^3
sss_chain <- spike_slab_probit(chain_length=4e3,burnin=1e3,X=X,y=y,
tau0=params$tau0,tau1=params$tau1,q=params$q,verbose=FALSE,store=FALSE)

# Use posterior probabilities for variable selection
sss_chain$z_ergodic_avg[1:10]
```

`synthetic_data` *synthetic_data*

Description

Generates synthetic linear and logistic regression data

Usage

```
synthetic_data(
  n,
  p,
  s0,
  error_std,
  type = "linear",
  scale = TRUE,
  signal = "constant"
)
```

Arguments

`n` number of observations

`p` number of covariates

`s0` sparsity (number of non-zero components of the true signal)

`error_std` Standard deviation of the Gaussian noise (linear regression only)

`type` dataset type ('linear' or 'logistic')

`scale` design matrix X has columns mean zero and standard deviation 1 (TRUE or FALSE)

`signal` non-zero components of the true signal ('constant' or 'deacy')

Value

Design matrix, response and true signal vector for linear and logistic regression

Examples

```
syn_data <- synthetic_data(n=100,p=200,s0=5,error_std=2)

# syn_data$X is an n by p design matrix
dim(syn_data$X)

# syn_data$y is a length n response vector
length(syn_data$y)

# syn_data$true_beta is a length n response vector with only the first s0 entries non-zero
all(syn_data$true_beta[1:5]!=0)
all(syn_data$true_beta[-c(1:5)]==0)
```

Index