Package 'VsusP'

June 23, 2024

Title Variable Selection using Shrinkage Priors

Version 1.0.0

Description Implements Bayesian variable selection using shrinkage priors to identify significant variables in high-dimensional datasets. The package includes methods for determining the number of significant variables through innovative clustering techniques of posterior distributions, specifically utilizing the 2-Means and Sequential 2-Means (S2M) approaches. Designed for complex datasets such as those in genomics and epidemiology, VsusP helps in tackling the challenges of variable selection in the presence of high collinearity and high dimensionality. The package aims to simplify the variable selection process with minimal tuning required in statistical analysis.

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.2

Imports bayesreg, stats

Suggests covr, MASS, knitr, rmarkdown, tinytex, testthat (>= 3.0.0)

Config/testthat/edition 3

Maintainer Nilson Chapagain <nilson.chapagain@gmail.com>

URL https://github.com/nilson01/VsusP-variable-selection-using-shrinkage-priors

BugReports https://github.com/nilson01/VsusP-variable-selection-using-shrinkage-priors/issues

VignetteBuilder knitr

NeedsCompilation no

Author Nilson Chapagain [aut, cre] (<https://orcid.org/0000-0003-2962-2949>), Debdeep Pati [aut]

Contents

numNoiseCoeff *Variable selection using shrinkage priors :: numNoiseCoeff*

Description

Variable selection using shrinkage priors :: numNoiseCoeff

Usage

```
numNoiseCoeff(Beta.i, b.i_r)
```

Arguments

Beta.i	N by p matrix consisting of N posterior samples of p variables
b.i_r	tuning parameter value from Sequential 2-means (S2M) variable selection algorithm.

Value

number of noise coefficients of numeric data type

OptimalHbi *Variable selection using shrinkage priors :: OptimalHbi*

Description

OptimalHbi function will take b.i and H.b.i as input which comes from the result of TwoMeans function. It will return H: the optimal value of the tuning parameter.

Usage

```
OptimalHbi(bi, Hbi)
```

Arguments

bi	a vector holding the values of the tuning parameter specified by the user
Hbi	The estimated number of signals corresponding to each b.i of numeric data type

Value

the optimal value (numeric) of tuning parameter and the associated H value

References

Makalic, E. & Schmidt, D. F. High-Dimensional Bayesian Regularised Regression with the BayesReg Package arXiv:1611.06649, 2016

Li, H., & Pati, D. Variable selection using shrinkage priors Computational Statistics & Data Analysis, 107, 107-119.

Examples

```
n <- 100
p <- 20
X <- matrix(rnorm(n * p), n, p)
beta <- exp(rnorm(p))
Y <- as.vector(X %*% beta + rnorm(n, 0, 1))
df <- data.frame(X, Y)
rv.hs <- bayesreg::bayesreg(Y ~ ., df, "gaussian", "horseshoe+", 200, 100)

Beta <- t(rv.hs$beta)
lower <- 0
upper <- 1
l <- 5
S2Mbeta <- Sequential2MeansBeta(Beta, lower, upper, l)

bi <- S2Mbeta$b.i
Hbi <- S2Mbeta$H.b.i
OptimalHbi(bi, Hbi)
```

S2MVarSelection *Variable selection using shrinkage priors :: S2MVarSelection*

Description

S2MVarSelection function will take S2M: a list obtained from the 2Means.variables function and H: the estimated number of signals obtained from the optimal.b.i function. This will give out the important subset of variables for the Gaussian Linear model.

Usage

```
S2MVarSelection(Beta, H = 10)
```

Arguments

`Beta`	matrix consisting of N posterior samples of p variables that is known either to user or from Sequential2Means function
`H`	Estimated number of signals obtained from the optimal.b.i function of numeric data type

Value

a vector containing indices of important subset of variables of dimension H X 1.

References

Makalic, E. & Schmidt, D. F. High-Dimensional Bayesian Regularised Regression with the BayesReg Package arXiv:1611.06649, 2016

Li, H., & Pati, D. Variable selection using shrinkage priors Computational Statistics & Data Analysis, 107, 107-119.

Examples

```
n <- 100
p <- 20
X <- matrix(rnorm(n * p), n, p)
beta <- exp(rnorm(p))
Y <- as.vector(X %*% beta + rnorm(n, 0, 1))
df <- data.frame(X, Y)
# Fit a model using gaussian horseshoe+ for 200 samples
# # recommended n.samples is 5000 and burning is 2000
rv.hs <- bayesreg::bayesreg(Y ~ ., df, "gaussian", "horseshoe+", 200, 100)

Beta <- rv.hs$beta
H <- 12
impVariablesGLM <- S2MVarSelection(Beta, H)
impVariablesGLM
```

S2MVarSelectionV1 *Variable selection using shrinkage priors :: S2MVarSelectionV1*

Description

S2MVarSelectionV1 function will take S2M: a list obtained from the 2Means.variables function and H: the estimated number of signals obtained from the optimal.b.i function. This will give out the important subset of variables for the Gaussian Linear model.

Usage

```
S2MVarSelectionV1(S2M, H = 10)
```

Arguments

`S2M`	List obtained from the 2Means.variables function
`H`	Estimated number (numeric) of signals obtained from the optimal.b.i function

Value

a vector of indices of important subset of variables for the Gaussian Linear modelof shape H X 1

Sequential2Means *Variable selection using shrinkage priors :: Sequential2Means*

Description

Sequential2Means function will take as input X: design matrix, Y : response vector, t: vector of tuning parameter values from Sequential 2-means (S2M) variable selection algorithm. The function will return a list S2M which will hold p: the total number of variables, b.i: the values of the tuning parameter, H.b.i : the estimated number of signals corresponding to each b.i, abs.post.median: medians of the absolute values of the posterior samples.

Usage

```
Sequential2Means(
  X,
  Y,
  b.i,
  prior = "horseshoe+",
  n.samples = 5000,
  burnin = 2000
)
```

Arguments

X	Design matrix of dimension n X p, where n = total data points and p = total number of features
Y	Response vector of dimension n X 1
b.i	Vector of tuning parameter values from Sequential 2-means (S2M) variable selection algorithm of dimension specified by user.
prior	Shrinkage prior distribution over the Beta. Available options are ridge regression: prior="rr" or prior="ridge", lasso regression: prior="lasso", horseshoe regression: prior="hs" or prior="horseshoe", and horseshoe+ regression : prior="hs+" or prior="horseshoe+" (String data type)
n.samples	Number of posterior samples to generate of numeric data type
burnin	Number of burn-in samples of numeric data type

Value

A list S2M which will hold Beta, b.i, and H.b.i.

Beta	N by p matrix consisting of N posterior samples of p variables
b.i	the user specified vector holding the tuning parameter values
H.b.i	the estimated number of signals of unmeric data type corresponding to each b.i

References

Makalic, E. & Schmidt, D. F. High-Dimensional Bayesian Regularised Regression with the BayesReg Package arXiv:1611.06649, 2016

Li, H., & Pati, D. Variable selection using shrinkage priors Computational Statistics & Data Analysis, 107, 107-119.

Examples

```
# ----------------------------------------------------------------
# Example 1: Gaussian Model and Horseshoe prior
n <- 100
p <- 20
X <- matrix(rnorm(n * p), n, p)
beta <- exp(rnorm(p))
Y <- as.vector(X %*% beta + rnorm(n, 0, 1))
b.i <- seq(0, 1, 0.05)

# Sequential2Means with horseshoe+ using gibbs sampling
```

```
# recommended n.samples is 5000 and burning is 2000
S2M <- Sequential2Means(X, Y, b.i, "horseshoe+", 200, 100)
Beta <- S2M$Beta
H.b.i <- S2M$H.b.i

# ---------------------------------------------------------------------
# Example 2: Gaussian Model and ridge prior

n <- 100
p <- 20
X <- matrix(rnorm(n * p), n, p)
beta <- exp(rnorm(p))
Y <- as.vector(X %*% beta + rnorm(n, 0, 1))
b.i <- seq(0, 1, 0.05)
# Sequential2Means with ridge regression using gibbs sampling
# recommended n.samples is 5000 and burning is 2000
S2M <- Sequential2Means(X, Y, b.i, "ridge", 200, 100)
Beta <- S2M$Beta
H.b.i <- S2M$H.b.i
```

`Sequential2MeansBeta` *Variable selection using shrinkage prior :: Sequential2MeansBeta*

Description

Sequential2MeansBeta function will take as input Beta : N by p matrix consisting of N posterior samples of p variables, lower : the lower bound of the chosen values of the tuning parameter, upper : the upper bound of the chosen values of the tuning parameter, and l :the number of chosen values of the tuning parameter. The function will return a list S2M which will hold p: the total number of variables, b.i: the values of the tuning parameter, H.b.i : the estimated number of signals corresponding to each b.i, abs.post.median: medians of the absolute values of the posterior samples.

Usage

```
Sequential2MeansBeta(Beta, lower, upper, l)
```

Arguments

`Beta`	N by p matrix consisting of N posterior samples of p variables
`lower`	the lower bound of the chosen values of the tuning parameter of numeric data type.
`upper`	the upper bound of the chosen values of the tuning parameter of numeric data type.
`l`	the number of chosen values of the tuning parameter of numeric data type.

Value

A list S2M which will hold p, b.i, and H.b.i:

`p`	total number of variables in the model
`b.i`	the vector values of the tuning parameter specified by the user
`H.b.i`	the estimated number of signals corresponding to each b.i of numeric data type

References

Makalic, E. & Schmidt, D. F. High-Dimensional Bayesian Regularised Regression with the BayesReg Package arXiv:1611.06649, 2016

Li, H., & Pati, D. Variable selection using shrinkage priors Computational Statistics & Data Analysis, 107, 107-119.

Examples

```
# ----------------------------------------------------------------
# Example 1: Gaussian Model and Horseshoe prior

n <- 100
p <- 20
X <- matrix(rnorm(n * p), n, p)
beta <- exp(rnorm(p))
Y <- as.vector(X %*% beta + rnorm(n, 0, 1))
df <- data.frame(X, Y)

# beta samples for gaussian model using horseshow prior and gibbs sampling
rv.hs <- bayesreg::bayesreg(Y ~ ., df, "gaussian", "horseshoe+", 200, 100)

Beta <- t(rv.hs$beta)
lower <- 0
upper <- 1
l <- 20
S2Mbeta <- Sequential2MeansBeta(Beta, lower, upper, l)
H.b.i <- S2Mbeta$H.b.i

# ----------------------------------------------------------------
# Example 2: normal model and lasso prior

#' n <- 100
p <- 20
X <- matrix(rnorm(n * p), n, p)
beta <- exp(rnorm(p))
Y <- as.vector(X %*% beta + rnorm(n, 0, 1))
df <- data.frame(X, Y)
rv.hs <- bayesreg::bayesreg(Y ~ ., df, "normal", "lasso", 200, 100)

Beta <- t(rv.hs$beta)
lower <- 0
upper <- 1
l <- 15
S2Mbeta <- Sequential2MeansBeta(Beta, lower, upper, l)
H.b.i <- S2Mbeta$H.b.i
```

Index