

# Package ‘LS3MU’

December 9, 2023

**Type** Package

**Title** Laplacian Spike-Slab Selector with Matrix Uncertainty

**Version** 1.0

**Date** 2023-12-09

**Author** Shuyu Guo

**Maintainer** Shuyu Guo <syguo@tamu.edu>

**Description** Considering high-dimensional sparse regression with measurement error in variables, we assume the errors are Normally distributed with mean 0 and errors for each variable have a specific variance. For such problem setting, function 'lsum' in this package can conduct variable selection tasks. It works under EM framework, where the unknown true design matrix and spike-slab prior indicators of regression parameters are treated as latent variables. The final output coefficients are obtained following a decreasing sequence of spike parameters and the paths can be displayed by calling functions in 'LS3MU' package.

**License** GPL (>= 2)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.3

**LinkingTo** Rcpp, RcppEigen

**Imports** Rcpp, RcppEigen, ggplot2, reshape2, stats, MASS

**Suggests** knitr, rmarkdown, SSLASSO, glmnet

**VignetteBuilder** knitr

**NeedsCompilation** yes

**Archs** x64

## R topics documented:

gPlot . . . . .	2
lsum . . . . .	2
pathPlot . . . . .	4
posteriorX . . . . .	5
Index . . . . .	7

gPlot

*gPlot***Description**

Plot maximized  $g(\beta, \theta, \sigma | \beta^{(t)}, \theta^{(t)}, \sigma^{(t)})$  over iterations at a specified spike parameter. The function  $g$  is a concave lower bound for objective log-likelihood function at  $\beta^{(t)}, \theta^{(t)}$  and  $\sigma^{(t)}$  by Jensen's Inequality.  $\beta^{(t+1)}, \theta^{(t+1)}$  and  $\sigma^{(t+1)}$  are obtained by maximizing  $g(\beta, \theta, \sigma | \beta^{(t)}, \theta^{(t)}, \sigma^{(t)})$ . See Vignette for more details.

**Usage**

```
gPlot(fit_obj, spike_param)
```

**Arguments**

fit_obj	Fitted object from function 'lsum'.
spike_param	The value of the spike parameter to be specified, which must be in spike_params designated in 'lsum' function. Each spike_param corresponds to a unique gPlot. If default spike_params is used in function 'lsum', one can check the value of spike_params by calling \$spike_params from the returned object from function 'lsum'.

lsum

*Laplacian Selector for Uncertain Matrix***Description**

For high dimensional sparse regression with additive errors in potential variables, this Laplacian Spike & Slab Selector provides a possible solution of variable selection under EM framework, where both unknown true variables and spike-slab indicators are treated as latent variables. The path of regression coefficients is obtained following a decreasing list of spike parameters.

**Usage**

```
lsum(
  Z,
  y,
  tauList,
  spike_params,
  slab_param,
  beta_init,
  sigma_update = TRUE,
  sigma_init,
  theta_init = 0.5,
  return_g = FALSE,
  a = 1,
  b = NULL,
  omega = 1,
```

```

kappa = 1,
tolerance = 0.01,
max_iter = 500
)

```

### Arguments

<code>Z</code>	$n * p$ covariate matrix with additive error in each column (possibly $p > n$ ). It is assumed the measurement error model takes the form $Z = X + \Xi$ , where $X$ is the unknown true design matrix and $\Xi$ is the matrix of i.i.d Gaussian measurement error of mean 0 and same variance within each column.
<code>y</code>	Numeric response vector from $n$ observations.
<code>tauList</code>	Vector of length $p$ . Corresponding variances of additive measurement error for $p$ columns.
<code>spike_params</code>	Vector of of length $L$ . Decreasing scale parameters of spike prior for beta. <code>spike_params</code> should be less than <code>slab_param</code> . If not specified, <code>spike_params</code> will be assigned default values. See 'Details' for more information.
<code>slab_param</code>	Numeric. Scale parameter $\lambda_1$ of slab prior for beta. If not specified, <code>slab_param</code> = <code>spike_params[1] * 10</code> if <code>spike_params</code> is given; Otherwise, <code>slab_param</code> = 1 by default.
<code>beta_init</code>	Vector. Initial value of regression coefficients beta. <code>beta_init</code> = 0 by default.
<code>sigma_update</code>	Logical. Whether the variance of model error is updated or not. Default is TRUE.
<code>sigma_init</code>	Numeric. The initial value of standard deviation of model error. If not specified, <code>sigma</code> is initialized according to <code>sd(y)</code> .
<code>theta_init</code>	Numeric. The initial value of prior proportion of nonzero beta. <code>theta_init</code> must be in $(0, 1]$ . Default is 0.5.
<code>return_g</code>	Logical. Default is FALSE. If specified TRUE, return a list containing the expected value of log likelihood function w.r.t the current conditional distribution of latent variables.
<code>a, b</code>	Numeric parameters of Beta prior distribution of theta, where $\theta \sim \text{Beta}(a, b)$ . $a = 1$ and $b = p$ where $p = \text{ncol}(Z)$ by default.
<code>omega, kappa</code>	Numeric parameters of Inverse Gamma prior distribution of $\sigma^2$ , where $\sigma^2 \sim \text{IG}(\omega/2, \omega * \kappa/2)$ . $\omega = 1$ and $\kappa = 1$ by default.
<code>tolerance</code>	Numeric. Criterion for early stopping at each <code>spike_param</code> . If $\ \beta_{old} - \beta_{new}\ _2 < \text{tolerance}$ , then break the iteration at the current <code>spike_param</code> .
<code>max_iter</code>	Integer. The maximum iteration number at each <code>spike_param</code> .

### Details

Since 'lsum' is built based on EM algorithm, it is possible to converge to local rather than global maximum values. Thus the result can be sensitive to the initial choices of beta, sigma and theta.

In addition, the value of spike and slab parameters can be crucial to variable selection result in practice. We set some default values for `spike_params` and `slab_param`. If `spike_param` is not specified and `slab_param` is given, `spike_params` = `exp(seq(log(slab_L), by=-0.3, length.out=20))[-1]` by default. If `slab_param` is also not specified, the `slab_param` = 1 by default and the `spike_param` is set the same as the previous way. Users can assign any valid values to `spike_params` and `slab_param`.

**Value**

lsum returns a list containing the following values:

beta_path	$L * p$ matrix. Each row is the beta fitted at corresponding spike_param.
beta_indices	Vector. Indices of selected nonzero regression parameters.
beta_values	Vector. Values of selected nonzero regression parameters.
beta_output	Vector of length $p$ . Full output beta including zeros.
sigma_path	Vector of Length $L$ . Estimated sigma at each spike_param.
theta_path	Vector of Length $L$ . Estimated theta at each spike_param.
g_List	List of Length $L$ . Each element of g_List is a list containing values of maximized function $g$ over iterations at corresponding spike_param.
iter_nums	Vector of Length $L$ . Number of Iterations at each spike_param.

**Examples**

```
require(MASS)
n = 200
p = 500

set.seed(1234)
beta_true = c(-3, 2, -1.5, -2, 3, rep(0,p-5))
X = matrix(rnorm(n*p, 0, 1), nrow = n)
epsilon = rnorm(n, 0, 1)
y = X %*% beta_true+epsilon

tau = sample(seq(0.5,0.9,by = 0.1), size = p, replace = TRUE)
Xi = mvrnorm(n, rep(0,p), diag(tau))
Z = X + Xi

lsum_fit = lsum(Z, y, tauList = tau)
```

---

pathPlot

*pathPlot*


---

**Description**

Plot the path of beta, theta or sigma over spike\_params.

**Usage**

```
pathPlot(fit_obj, path_of = "beta")
```

**Arguments**

fit_obj	The fitted object from 'lsum' function.
path_of	Character. path_of = "beta", "theta" or "sigma". Default is "beta".

---

posteriorX	<i>posteriorX</i>
------------	-------------------

---

## Description

In ordinary regression model, assuming each column of the design matrix  $Z$  contains additive Gaussian measurement errors with known variance collected in `tauList`, function `'posteriorX'` returns posterior expectation as well as some related values for the true design matrix  $X$ .

More specifically, if we denote the measurement error matrix as  $\Xi$ , then the measurement error model takes the form:  $Z = X + \Xi$ . Since we have  $y_i | x_i, \beta, \sigma^2 \sim N(x_i^\top \beta, \sigma^2)$  and  $x_i | z_i, \Lambda \sim N(z_i, \Lambda)$  where  $\Lambda$  is the diagonal matrix consists of `tauList`, `posteriorX` computes the expectation of  $X$ , variance of  $x_i$  and expectation of  $X^\top X$  conditioning on  $\beta, y, Z, \Lambda$  and  $\sigma$ .

This is part of E-step in the `'lsum'` function.

## Usage

```
posteriorX(Z, y, beta, sigma, tauList)
```

## Arguments

<code>Z</code>	$n * p$ covariate matrix with additive errors in each column (possibly $p > n$ ). It is assumed the measurement error model takes the form $Z = X + \Xi$ , where $X$ is the unknown true design matrix and $\Xi$ is the matrix of i.i.d Gaussian measurement errors of mean 0 and same variance within each column.
<code>y</code>	Numeric response vector from $n$ observations.
<code>beta</code>	Regression parameter vector of length $p$ .
<code>sigma</code>	Numeric. The standard deviance of regression model error terms.
<code>tauList</code>	Vector of length $p$ . Corresponding variances of additive measurement error for $p$ columns.

## Value

`'posteriorX'` returns a list containing the following values:

<code>hatX</code>	$n * p$ matrix. The expectation of $X$ conditional on $\beta, y, Z, \Lambda$ and $\sigma$ .
<code>hatSigma</code>	$p * p$ matrix. The variance of $x_i$ conditional on $\beta, y, Z, \Lambda$ and $\sigma$ , which is equivalent for each $x_i$ .
<code>hatXX</code>	$p * p$ matrix. The conditional expectation of $X^\top X$ . <code>hatXX = t(hatX) * hatX + hatSigma</code> .

## Examples

```
require(MASS)
n = 200
p = 500

set.seed(1234)
beta_true = c(-3, 2, -1.5, -2, 3, rep(0,p-5))
X = matrix(rnorm(n*p, 0, 1), nrow = n)
epsilon = rnorm(n, 0, 1)
```

```
y = X %% beta_true+epsilon

tau = sample(seq(0.5,0.9,by = 0.1), size = p, replace = TRUE)
Xi = mvrnorm(n, rep(0,p), diag(tau))
Z = X + Xi

post_X = posteriorX(Z, y, beta = beta_true, sigma = 1, tauList = tau)
```

# Index

`gPlot`, [2](#)

`lsum`, [2](#)

`pathPlot`, [4](#)

`posteriorX`, [5](#)