Package ‘emBayes’

November 7, 2023

Type Package

Title Robust Bayesian Variable Selection via Expectation-Maximization

Version 0.1.4

Date 2023-11-02

Maintainer Yuwen Liu <yuwen@ksu.edu>

Description Variable selection methods have been extensively developed for analyzing high-dimensional omics data within both the frequentist and Bayesian frameworks. This package implemented the spike-and-slab quantile LASSO which has been developed along the line of Bayesian hierarchical model but deeply rooted in the frequentist regularization methods by utilizing the Expectation–Maximization (EM) algorithm. Therefore, the proposed method borrows strength from both the frequentist and Bayesian frameworks while overcoming their respective limitations. The spike-and-slab quantile LASSO can handle data irregularity in terms of skewness and outliers in the disease trait, compared to its nonrobust alternative, the spike-and-slab LASSO, which has also been implemented in the package. The core module of this package is developed in 'C++'.

Depends R (>= 4.2.0)

License GPL-2

Encoding UTF-8

LazyData true

Imports Rcpp, glmnet

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 7.2.3

NeedsCompilation yes

Author Yuwen Liu [aut, cre],
Cen Wu [aut]

Repository CRAN

Date/Publication 2023-11-07 05:10:02 UTC

R topics documented:

---

emBayes-package *Robust Bayesian Variable Selection via Expectation-Maximization*

Description

This package provides the implementation of the spike-and-slab quantile LASSO (ssQLASSO) which combines the strength of Bayesian robust variable selection and the Expectation-Maximization (EM) coordinate descent approach. The alternative method spike-and-slab LASSO (ssLASSO) is also included in the package.

Details

Two user friendly, integrated interface **cv.emBayes()** and **emBayes()** allows users to flexibly choose the variable selection method by specifying the following parameter:

quant: to specify different quantiles when using robust methods.

func: the model to perform variable selection. Two choices are available: "ssLASSO" and "ssQLASSO".

error: to specify the difference between expectations of likelihood of two consecutive iterations. It can be used to determine convergence.

maxiter: to specify the maximum number of iterations.

Function cv.emBayes() returns cross-validation errors based on the check loss, least squares loss and Schwarz Information Criterion along with the corresponding optimal tuning parameters. Function emBayes() returns the estimated intercept, clinical coefficients, beta coefficients, scale parameter, probability parameter, number of iterations and expectation of likelihood at each iteration.

References

Liu, Y. and Wu, C. (2023). Spike-and-Slab Quantile LASSO. (to be submitted)

Ren, J., Zhou, F., Li, X., Ma, S., Jiang, Y., and Wu, C. (2022). Robust Bayesian variable selection for gene–environment interactions. *Biometrics.* doi:10.1111/biom.13670

Ren, J., Du, Y., Li, S., Ma, S., Jiang,Y. and Wu, C. (2019). Robust network-based regularization and variable selection for high dimensional genomics data in cancer prognosis. *Genet. Epidemiol.*, 43:276-291 doi:10.1002/gepi.22194

Wu, C., Zhang, Q., Jiang,Y. and Ma, S. (2018). Robust network-based analysis of the associations between (epi)genetic measurements. *J Multivar Anal.*, 168:119-130 doi:10.1016/j.jmva.2018.06.009

Tang, Z., Shen, Y., Zhang, X., and Yi, N. (2017). The spike-and-slab lasso generalized linear models for prediction and associated genes detection. *Genetics*, 205(1), 77-88 doi:10.1534/genetics.116.192195

Tang, Z., Shen, Y., Zhang, X., and Yi, N. (2017). The spike-and-slab lasso Cox model for survival prediction and associated genes detection. *Bioinformatics*, 33(18), 2799-2807 doi:10.1093/bioinformatics/btx300

Wu, C., and Ma, S. (2015). A selective review of robust variable selection with applications in bioinformatics. *Briefings in Bioinformatics*, 16(5), 873–883 doi:10.1093/bib/bbu046

Zhou, Y. H., Ni, Z. X., and Li, Y. (2014). Quantile regression via the EM algorithm. *Communications in Statistics-Simulation and Computation*, 43(10), 2162-2172 doi:10.1080/03610918.2012.746980

Ročková, V., and George, E. I. (2014). EMVS: The EM approach to Bayesian variable selection. *Journal of the American Statistical Association*, 109(506), 828-846 doi:10.1080/01621459.2013.869223

Li, Q., Lin, N., and Xi, R. (2010). Bayesian regularized quantile regression. *Bayesian Analysis*, 5(3), 533-556 doi:10.1214/10BA521

George, E. I., and McCulloch, R. E. (1993). Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423), 881-889 doi:10.1080/01621459.1993.10476353

See Also

`cv.emBayes` `emBayes`

---

cv.emBayes — *k-folds cross-validation for 'emBayes'*

Description

This function performs cross-validation and returns the optimal values of the tuning parameters.

Usage

```
cv.emBayes(
  y,
  clin = NULL,
  X,
  quant,
  t0,
  t1,
  k,
  func,
  error = 0.01,
  maxiter = 200
)
```

Arguments

`y`	a vector of response variable.
`clin`	a matrix of clinical factors. It has default value NULL.
`X`	a matrix of genetic factors.
`quant`	value of quantile.
`t0`	a user-supplied sequence of the spike scale s_0.
`t1`	a user-supplied sequence of the slab scale s_1.
`k`	number of folds for cross-validation.
`func`	methods to perform variable selection. Two choices are available: "ssLASSO" and "ssQLASSO".
`error`	cutoff value for determining convergence. The algorithm reaches convergence if the difference in the expected log-likelihood of two iterations is less than the value of error. The default value is 0.01.
`maxiter`	the maximum number of iterations that is used in the estimation algorithm. The default value is 200.

Details

When performing cross-validation for emBayes, function cv.emBayes returns two sets of optimal tuning parameters and their corresponding cross-validation error matrices. The spike scale parameter $CL.s0$ and the slab scale parameter $CL.s1$ are obtained based on the quantile check loss. The spike scale parameter $SL.s0$ and the slab scale parameter $SL.s1$ are obtained based on the least squares loss. The spike scale parameter $SIC.s0$ and the slab scale parameter $SIC.s1$ are obtained based on the Schwarz Information Criterion (SIC). Corresponding error matrices $CL.CV$, $SL.CV$ and $SIC.CV$ can also be obtained from the output.

Schwarz Information Criterion has the following form:

$$SIC = \log \sum_{i=1}^{n} L(y_i - \hat{y_i}) + \frac{\log n}{2n} edf$$

where $L(\cdot)$ is the check loss and edf is the number of close to zero residuals (≤ 0.001).

Value

A list with components:

`CL.s0`	the optimal spike scale under check loss.
`CL.s1`	the optimal slab scale under check loss.
`SL.s0`	the optimal slab scale under least squares loss.
`SL.s1`	the optimal slab scale under least squares loss.
`SIC.s0`	the optimal slab scale under SIC.
`SIC.s1`	the optimal slab scale under SIC.
`CL.CV`	cross-validation error matrix under check loss.
`SL.CV`	cross-validation error matrix under least squares loss.
`SIC.CV`	cross-validation error matrix under SIC.

data *simulated gene expression example data*

Description

Simulated gene expression data for demonstrating the usage of emBayes.

Usage

```
data(data)
```

Format

The data file consists of five components: y, clin, X, quant, coef and clin.coe. The coefficients and clinical coefficients are the true values of parameters used for generating response y. They can be used for performance evaluation.

Details

The data model for generating response

Let y_i be the response of the i-th subject (1≤ i≤ n). We have $z_i = (1, z_{i1}, \ldots, z_{iq})^\top$ being a $(q+1)$-dimensional vector of which the last q components indicate clinical factors and $x_i = (x_{i1}, \ldots, x_{ip})^\top$ denoting a p-dimensional vector of genetic factors. The linear quantile regression model for the τ-th quantile $(0 < \tau < 1)$ is:

$$y_i = z_i^\top \alpha + x_i^\top \beta + \epsilon_i$$

where $\alpha = (\alpha_0, \cdots, \alpha_q)^\top$ contains the intercept and the regression coefficients for the clinical covariates. $\beta = (\beta_1, \cdots, \beta_p)^\top$ are the regression coefficients and random error $\epsilon_i = (\epsilon_1, ..., \epsilon_n)^\top$ is set to follow a T2 distribution and has value 0 at its τ-th quantile.

See Also

emBayes

emBayes *fit a model with given tuning parameters*

Description

This function performs penalized variable selection based on spike-and-slab quantile LASSO (ssQLASSO) or spike-and-slab LASSO (ssLASSO). Typical usage is to first obtain the optimal spike scale and slab scale using cross-validation, then specify them in the 'emBayes' function.

Usage

```
emBayes(y, clin = NULL, X, quant, s0, s1, func, error = 0.01, maxiter = 200)
```

Arguments

`y`	a vector of response variable.
`clin`	a matrix of clinical factors. It has default value NULL.
`X`	a matrix of genetic factors.
`quant`	value of quantile.
`s0`	value of the spike scale s_0.
`s1`	value of the slab scale s_1.
`func`	methods to perform variable selection. Two choices are available: "ssLASSO" and "ssQLASSO".
`error`	cutoff value for determining convergence. The algorithm reaches convergence if the difference in the expected log-likelihood of two iterations is less than the value of error. The default value is 0.01.
`maxiter`	the maximum number of iterations that is used in the estimation algorithm. The default value is 200.

Details

The current version of emBayes supports two types of methods: "ssLASSO" and "ssQLASSO".

- **ssLASSO:** spike-and-slab LASSO fits a Bayesian linear regression through the EM algorithm.
- **ssQLASSO:** spike-and-slab quantile LASSO fits a Bayesian quantile regression (based on asymmetric Laplace distribution) through the EM algorithm.

Users can choose the desired method by setting func="ssLASSO" or "ssQLASSO".

Value

A list with components:

`alpha`	a vector containing the estimated intercept and clinical coefficients.
`intercept`	value of the estimated intercept.
`clin.coe`	a vector of estimated clinical coefficients.
`beta`	a vector of estimated beta coefficients.
`sigma`	value of estimated asymmetric Laplace distribution scale parameter σ.
`theta`	value of estimated probability parameter θ.
`iter`	value of number of iterations.
`ll`	a vector of expectation of likelihood at each iteration.

Examples

```
data(data)
##load the clinical factors, genetic factors, response and quantile data
clin=data$clin
X=data$X
y=data$y
quant=data$quant

##generate tuning vectors of desired range
t0 <- seq(0.01,0.015,length.out=2)
t1 <- seq(0.1,0.5,length.out=2)

##perform cross-validation and obtain tuning parameters based on check loss
CV <- cv.emBayes(y,clin,X,quant,t0,t1,k=5,func="ssQLASSO",error=0.01,maxiter=200)
s0 <- CV$CL.s0
s1 <- CV$CL.s1

##perform BQLSS under optimal tuning and calculate value of TP and FP for selecting beta
EM <- emBayes(y,clin,X,quant,s0,s1,func="ssQLASSO",error=0.01,maxiter=200)
fit <- EM$beta
coef <- data$coef
tp <- sum(fit[coef!=0]!=0)
fp <- sum(fit[coef==0]!=0)
list(tp=tp,fp=fp)
```

`print.cv.emBayes` *print an cv.emBayes result*

Description

Print a summary of an 'cv.emBayes' result

Usage

```
## S3 method for class 'cv.emBayes'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

`x`	cv.emBayes result
`digits`	significant digits in printout.
`...`	other print arguments

Value

Print a list of output from a cv.emBayes object.

See Also

cv.emBayes

---

print.emBayes *print an emBayes result*

Description

Print a summary of an 'emBayes' result

Usage

```
## S3 method for class 'emBayes'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

x	emBayes result
digits	significant digits in printout.
...	other print arguments

Value

Print a list of output from a emBayes object.

See Also

emBayes

Index