Package 'LavaCvxr'

June 4, 2021

Type Package

Title Lava Estimation for the Sum of Sparse and Dense Signals(3 Methods)

Version 1.0.2

Date 2021-05-26

Author Victor Chernozhukov [aut, cre], Christian Hansen [aut, cre], Yuan Liao [aut, cre], Jaeheon Jung [ctb, cre], Yang Liu [ctb, cre]

Maintainer Yang Liu <yl1241@economics.rutgers.edu>

Description
The lava estimation is used to recover signals that is the sum of a sparse signal and a dense signal. The post-lava method corrects the shrinkage bias of lava. For more information on the lava estimation, see Chernozhukov, Hansen, and Liao (2017) <doi:10.1214/16-AOS1434>.

Imports pracma, CVXR

Depends Lavash

License GPL-2

Repository CRAN

NeedsCompilation no

Encoding UTF-8

RoxygenNote 7.1.1

Date/Publication 2021-06-04 07:00:05 UTC

R topics documented:

LavaCvxr 2

Index **5**

LavaCvxr *Lava Estimation for the Sum of Sparse and Dense Signals(3 Methods).*

Description

The lava estimation is used to recover signals that is the sum of a sparse signal and a dense signal. The post-lava method corrects the shrinkage bias of lava. The model is Y=X*B+error, where B can be decomposed into B(theta)=dense part(beta)+sparse part(delta). Lava solves the following problem: min_[beta,delta] 1/n*|Y-X*(beta+delta)|_2^2+lamda2*|beta|_2^2+lambda1*|delta|_1. The final estimator is theta, which is theta=beta+delta. Both tuning parameters lambda1 and lambda2 are chosen using the K-fold cross-validation.

Usage

```
LavaCvxr(
  X,
  Y,
  K,
  Lambda1,
  Lambda2,
  method = c("Profile", "Iteration", "LavaCvxr"),
  Maxiter = 50
)
```

Arguments

X	n by p data matrix, where n and p respectively denote the sample size and the number of regressors.
Y	n by 1 matrix of outcome.
K	the K fold cross validation.
Lambda1	If you choose 'Profile' or 'Iteration', 'Lambda1' should be a vector of candidate values to be evaluated in the cross validation to find an optimal 'Lambda1'. If you choose 'LavaCvxr', 'Lambda1' can be a vector (go through the cross validation to get an optimal value) or an any specific value you choose (without going through the cross validation part).
Lambda2	If you choose 'Profile' or 'Iteration', 'Lambda2' should be a vector of candidate values to be evaluated in the cross validation to find an optimal 'Lambda2'. If you choose 'LavaCvxr', 'Lambda2' can be a vector (go through the cross validation to get an optimal value) or an any specific value you choose (without going through the cross validation part).
method	choose among 'Profile', 'Iteration' and 'LavaCvxr'. 'Profile' computes using the profiled lasso method. 'Iteration' computes using iterating lasso and ridge. 'LavaCvxr' computes using CVXR method to calculate. 'Profile' and 'Iteration' depends on the 'Lavash' function in 'Lavash' package. For more details, please see the document for 'Lavash'.
Maxiter	the maximum number of iterations. The default value is 50. Only used when 'Iteration' is selected.

Details

If you choose 'Profile' method or 'Iteration' method, we recommend using a relatively long vector of Lambda1 (e.g., 50 or 100 values), but a short vector of Lambda2 (e.g., within 10). Higher dimensions of Lambda2 substantially increase the computational time because a 'for' loop is called for Lambda2. 'Profile' and 'Iteration' depends on the 'Lavash' function in 'Lavash' package. For more details, please see the document for 'Lavash'.

Value

An 'output_list' containing the following components:

lava_dense	parameter estimate of the dense component using lava.
lava_sparse	parameter estimate of the sparse component using lava.
lava_estimate	lava_estimate=lava_dense+lave_sparse: final parameter estimate using lava.
postlava_dense	parameter estimate of the dense component using post-lava.
postlava_sparse	parameter estimate of the sparse component using post-lava.
postlava_estimaate	postlava_estimate=postlava_dense+postlava_sparse: final parameter estimate using post-lava.
LAMBDA	[lambda1lava,lambda2lava, lambda1post, lambda2post]: These are the CV-chosen for optimal 'Lambda1' and 'Lambda2' for lava and post-lava or the specific value that you choose without going through the cross validation part.

Author(s)

Victor Chernozhukov, Christian Hansen, Yuan Liao, Jaeheon Jung, Yang Liu

References

Chernozhukov, V., Hansen, C., and Liao, Y. (2017) "A lava attack on the recovery of sums of dense and sparse signals", Annals of Statistics, 45, 39-76

Examples

```
N <- 20
P <- 10
K<-5

X <- matrix(rnorm(n = N * P, mean = 0, sd = 3), nrow = N, ncol = P)
beta_true <- as.matrix(rep(x = 0, times = P) )
delta_true <- as.matrix(rep(x = 0, times = P))
beta_true[1:P]<-0.1
delta_true[1:4] <- c(2, -2, 3, 6)
Y <- X%*%delta_true+X%*%beta_true + rnorm(N, mean = 0, sd = 2)

lambda1<-seq(0.01,2,by=6/20)
lambda2<-c(0.01,0.07,0.2,0.7,3,10,60,1000,6000)
```

```
lava_result<-LavaCvxr(X,Y,K,lambda1,lambda2,method=c('Profile'), Maxiter=50)

lava_result$lava_dense
lava_result$lava_sparse
lava_result$lava_estimate
lava_result$postlava_dense
lava_result$postlava_sparse
lava_result$postlava_estimate
lava_result$LAMBDA
```

Index

∗ **lava**
LavaCvxr, 2

LavaCvxr, 2