

Package ‘sparseLTSEigen’

May 9, 2026

Type Package

Title RcppEigen back end for sparse least trimmed squares regression

Version 0.2.0.1

Date 2013-11-13

Depends robustHD ($\geq 0.4.0$)

Imports Rcpp ($\geq 0.9.10$), RcppEigen ($\geq 0.2.0$)

Suggests mvtnorm

LinkingTo Rcpp, RcppEigen

Description Use RcppEigen to fit least trimmed squares regression models with an L1 penalty in order to obtain sparse models.

License GPL (≥ 2)

LazyLoad yes

Author Andreas Alfons [aut, cre]

Maintainer Andreas Alfons <alfons@ese.eur.nl>

NeedsCompilation yes

Repository CRAN

Date/Publication 2020-10-14 16:34:51 UTC

Contents

sparseLTSEigen-package 2

Index 4

sparseLTSEigen-package

RcppEigen back end for sparse least trimmed squares regression

Description

Use RcppEigen to fit least trimmed squares regression models with an L1 penalty in order to obtain sparse models.

Details

```

Package:    sparseLTSEigen
Type:      Package
Version:    0.2.0
Date:      2013-11-13
Depends:    robustHD (>= 0.4.0)
Imports:    Rcpp (>= 0.9.10), RcppEigen (>= 0.2.0)
Suggests:  mvtnorm
LinkingTo:  Rcpp, RcppEigen
License:    GPL (>= 2)
LazyLoad:  yes

```

Index:

sparseLTSEigen-package

RcppEigen back end for sparse least trimmed
squares regression

Note

Package **sparseLTSEigen** provides an alternative back end for sparse least trimmed squares regression from package **robustHD**. The back end built into **robustHD** uses the C++ library Armadillo, whereas this back end uses the C++ library Eigen. The latter is faster, but currently does not work on 32-bit R for Windows.

When **sparseLTSEigen** is loaded, its back end is used automatically for sparse least trimmed squares regression, except on 32-bit R for Windows.

Author(s)

Andreas Alfons [aut, cre]

Maintainer: Andreas Alfons <alfons@ese.eur.nl>

Examples

```
# example is not high-dimensional to keep computation time low
library("mvtnorm")
set.seed(1234) # for reproducibility
n <- 100 # number of observations
p <- 25 # number of variables
beta <- rep.int(c(1, 0), c(5, p-5)) # coefficients
sigma <- 0.5 # controls signal-to-noise ratio
epsilon <- 0.1 # contamination level
Sigma <- 0.5^t(sapply(1:p, function(i, j) abs(i-j), 1:p))
x <- rmvnorm(n, sigma=Sigma) # predictor matrix
e <- rnorm(n) # error terms
i <- 1:ceiling(epsilon*n) # observations to be contaminated
e[i] <- e[i] + 5 # vertical outliers
y <- c(x %*% beta + sigma * e) # response
x[i,] <- x[i,] + 5 # bad leverage points

## fit sparse LTS model
# since package sparseLTSEigen is loaded, its back end based on
# the C++ library Eigen is used rather than the back end built
# into package robustHD, except on 32-bit R for Windows
fit <- sparseLTS(x, y, lambda = 0.05, mode = "fraction")
coef(fit, zeros = FALSE)
```

Index

* **package**

 sparseLTSEigen-package, [2](#)

.CallSparseLTSEigen

 (sparseLTSEigen-package), [2](#)

sparseLTSEigen

 (sparseLTSEigen-package), [2](#)

sparseLTSEigen-package, [2](#)