

# Package ‘STOPES’

May 27, 2022

**Type** Package

**Title** Selection Threshold Optimized Empirically via Splitting

**Version** 0.2

**Date** 2022-05-20

**Author** Marinela Capanu [aut, cre],  
Mihai Giurcanu [aut, ctb],  
Colin Begg [aut],  
Mithat Gonen [aut]

**Maintainer** Marinela Capanu <capanum@mskcc.org>

**Imports** MASS, cvTools, glmnet, changepoint

**Description** Implements variable selection procedures for low to moderate size generalized linear regressions models. It includes the STOPES functions for linear regression (Capanu M, Giurcanu M, Begg C, Gonen M, Optimized variable selection via repeated data splitting, *Statistics in Medicine*, 2020, 19(6):2167-2184) as well as subsampling based optimization methods for generalized linear regression models (Marinela Capanu, Mihai Giurcanu, Colin B Begg, Mithat Gonen, Subsampling based variable selection for generalized linear models).

**License** GPL-2

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2022-05-27 08:40:13 UTC

## R topics documented:

alasso.cv . . . . .	2
opts . . . . .	3
opts_th . . . . .	4
stopes . . . . .	5

<b>Index</b>	<b>7</b>
--------------	----------

---

alasso.cv	<i>ALASSO variable selection via cross-validation regularization parameter selection</i>
-----------	--

---

### Description

alasso.cv computes the ALASSO estimator.

### Usage

```
alasso.cv(x, y)
```

### Arguments

x	n x p covariate matrix
y	n x 1 response vector

### Value

alasso.cv returns the ALASSO estimate

alasso            the ALASSO estimator

### References

Hui Zou, (2006). "The adaptive LASSO and its oracle properties", JASA, 101 (476), 1418-1429

### Examples

```
p <- 5
n <- 100
beta <- c(2, 1, 0.5, rep(0, p - 3))
x <- matrix(nrow = n, ncol = p, rnorm(n * p))
y <- rnorm(n) + crossprod(t(x), beta)
alasso.cv(x, y)
```

---

opts *Optimization via Subsampling (OPTS)*

---

**Description**

opts computes the OPTS MLE in low dimensional case.

**Usage**

```
opts(X, Y, m, crit = "aic", prop_split = 0.5, cutoff = 0.75, ...)
```

**Arguments**

X	n x p covariate matrix (without intercept)
Y	n x 1 binary response vector
m	number of subsamples
crit	information criterion to select the variables: (a) aic = minimum AIC and (b) bic = minimum BIC
prop_split	proportion of subsample size and sample size, default value = 0.5
cutoff	cutoff used to select the variables using the stability selection criterion, default value = 0.75
...	other arguments passed to the glm function, e.g., family = "binomial"

**Value**

opts returns a list:

betahat	OPTS MLE of regression parameter vector
Jhat	estimated set of active predictors (TRUE/FALSE) corresponding to the OPTS MLE
SE	standard error of OPTS MLE
freqs	relative frequency of selection for all variables

**Examples**

```
require(MASS)
P = 15
N = 100
M = 20
BETA_vector = c(0.5, rep(0.5, 2), rep(0.5, 2), rep(0, P - 5))
MU_vector = numeric(P)
SIGMA_mat = diag(P)

X <- mvrnorm(N, MU_vector, Sigma = SIGMA_mat)
linearPred <- cbind(rep(1, N), X)
Y <- rbinom(N, 1, plogis(linearPred))
```

```
# OPTS-AIC MLE
opts(X, Y, 10, family = "binomial")
```

---

opts\_th *Threshold OPTimization via Subsampling (OPTS\_TH)*

---

### Description

opts\_th computes the threshold OPTS MLE in low dimensional case.

### Usage

```
opts_th(X, Y, m, crit = "aic", type = "binseg", prop_split = 0.5,
        prop_trim = 0.2, q_tail = 0.5, ...)
```

### Arguments

X	n x p covariate matrix (without intercept)
Y	n x 1 binary response vector
m	number of subsamples
crit	information criterion to select the variables: (a) aic = minimum AIC and (b) bic = minimum BIC
type	method used to minimize the trimmed and averaged information criterion: (a) min = observed minimum subsampling trimmed average information, (b) sd = observed minimum using the 0.25sd rule (corresponding to OPTS-min in the paper), (c) pelt = PELT changepoint algorithm (corresponding to OPTS-PELT in the paper), (d) binseg = binary segmentation changepoint algorithm (corresponding to OPTS-BinSeg in the paper), (e) amoc = AMOC method.
prop_split	proportion of subsample size of the sample size; default value is 0.5
prop_trim	proportion that defines the trimmed mean; default value = 0.2
q_tail	quantiles for the minimum and maximum p-values across the subsample cutpoints used to define the range of cutpoints
...	other arguments passed to the glm function, e.g., family = "binomial"

### Value

opts\_th returns a list:

betahat	STOPES MLE of regression parameters
SE	SE of STOPES MLE
Jhat	set of active predictors (TRUE/FALSE) corresponding to STOPES MLE
cutthat	estimated cutpoint for variable selection

pval	marginal p-values from univariate fit
cutpoits	subsample cutpoints
aic_mean	mean subsample AIC
bic_mean	mean subsample BIC

### Examples

```
require(MASS)
P = 15
N = 100
M = 20
BETA_vector = c(0.5, rep(0.5, 2), rep(0.5, 2), rep(0, P - 5))
MU_vector = numeric(P)
SIGMA_mat = diag(P)

X <- mvrnorm(N, MU_vector, Sigma = SIGMA_mat)
linearPred <- cbind(rep(1, N), X)
Y <- rbinom(N, 1, plogis(linearPred))

# Threshold OPTS-BinSeg MLE
opts_th(X, Y, M, family = "binomial")
```

---

stopes	<i>Selection of Threshold OPTimized Empirically via Splitting (STOPES)</i>
--------	--

---

### Description

stopes computes the STOPES estimator.

### Usage

```
stopes(x, y, m = 20, prop_split = 0.50, prop_trim = 0.20, q_tail = 0.90)
```

### Arguments

x	n x p covariate matrix
y	n x 1 response vector
m	number of split samples, with default value = 20
prop_split	proportion of data used for training samples, default value = 0.50
prop_trim	proportion of trimming, default prop_trim = 0.20
q_tail	proportion of truncation samples across the split samples, default values = 0.90

**Value**

stopes returns a list with the STOPE estimates via data splitting using 0.25 method and the PELT method:

beta_stop	the STOPE estimate via data splitting
J_stop	the set of active predictors corresponding to STOPES via data splitting
final_cutpoints	the final cutpoint for STOPES
beta_pelt	the STOPE estimate via PELT
J_pelt	the set of active predictors corresponding to STOPES via PELT
final_cutpoints_PELT	the final cutpoint for PELT
quan_NA	test if the vector of trimmed cutpoints has length 0, with 1 if TRUE and 0 otherwise

**Author(s)**

Marinela Capanu, Mihai Giurcanu, Colin Begg, and Mithat Gonen

**Examples**

```
p <- 5
n <- 100
beta <- c(2, 1, 0.5, rep(0, p - 3))
x <- matrix(nrow = n, ncol = p, rnorm(n * p))
y <- rnorm(n) + crossprod(t(x), beta)
stopes(x, y)
```

# Index

`alasso.cv`, 2

`opts`, 3

`opts_th`, 4

`stopes`, 5