Package ‘SIHR’

April 9, 2023

Type Package

Title Statistical Inference in High Dimensional Regression

Version 2.0.1

Author Prabrisha Rakshit, Zhenyu Wang, Tony Cai, Zijian Guo

Maintainer Zijian Guo <zijguo@stat.rutgers.edu>

Description The goal of SIHR is to provide inference procedures in the high-dimensional setting for
(1) linear functionals in generalized linear regression ('Cai et al.' (2019) <arXiv:1904.12891>, 'Guo et al.' (2020) <arXiv:2012.07133>, 'Cai et al.' (2021)),
(2) conditional average treatment effects in generalized linear regression,
(3) quadratic functionals in generalized linear regression ('Guo et al.' (2019) <arXiv:1909.01503>).
(4) inner product in generalized linear regression
(5) distance in generalized linear regression.

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.3

URL https://github.com/prabrishar1/SIHR

Imports CVXR, glmnet, stats

NeedsCompilation no

Repository CRAN

Date/Publication 2023-04-09 16:50:02 UTC

R topics documented:

CATE 2
Dist 4
InnProd 6
LF 8
QF 9

Index **12**

CATE *Inference for difference of linear combinations of the regression vectors in high dimensional generalized linear regressions*

Description

Computes the bias-corrected estimator of the difference of linear combinations of the regression vectors for the high dimensional generalized linear regressions and the corresponding standard error.

Usage

```
CATE(
  X1,
  y1,
  X2,
  y2,
  loading.mat,
  model = c("linear", "logistic", "logistic_alter"),
  intercept = TRUE,
  intercept.loading = FALSE,
  beta.init1 = NULL,
  beta.init2 = NULL,
  lambda = NULL,
  mu = NULL,
  prob.filter = 0.05,
  rescale = 1.1,
  alpha = 0.05,
  verbose = FALSE
)
```

Arguments

`X1`	Design matrix for the first sample, of dimension n_1 x p
`y1`	Outcome vector for the first sample, of length n_1
`X2`	Design matrix for the second sample, of dimension n_2 x p
`y2`	Outcome vector for the second sample, of length n_1
`loading.mat`	Loading matrix, nrow=p, each column corresponds to a loading of interest
`model`	The high dimensional regression model, either `"linear"` or `"logistic"` or `"logistic_alter"`
`intercept`	Should intercept(s) be fitted for the initial estimators (default = `TRUE`)
`intercept.loading`	Should intercept term be included for the `loading` (default = `FALSE`)
`beta.init1`	The initial estimator of the regression vector for the 1st data (default = `NULL`)
`beta.init2`	The initial estimator of the regression vector for the 2nd data (default = `NULL`)

`lambda`	The tuning parameter in fitting initial model. If `NULL`, it will be picked by cross-validation. (default = `NULL`)
`mu`	The dual tuning parameter used in the construction of the projection direction. If `NULL` it will be searched automatically. (default = `NULL`)
`prob.filter`	The threshold of estimated probabilities for filtering observations in logistic regression. (default = 0.05)
`rescale`	The factor to enlarge the standard error to account for the finite sample bias. (default = 1.1)
`alpha`	Level of significance to construct two-sided confidence interval (default = 0.05)
`verbose`	Should intermediate message(s) be printed (default = `FALSE`)

Value

A list consists of plugin estimators, debiased estimators, and confidence intervals. For logistic regression, it also returns those items after probability transformation.

`est.plugin.vec`	The vector of plugin(biased) estimators for the linear combination of regression coefficients, length of `ncol(loading.mat)`; corresponding to different column in `loading.mat`
`est.debias.vec`	The vector of bias-corrected estimators for the linear combination of regression coefficients, length of `ncol(loading.mat)`; corresponding to different column in `loading.mat`
`se.vec`	The vector of standard errors of the bias-corrected estimators, length of `ncol(loading.mat)`; corresponding to different column in `loading.mat`
`ci.mat`	The matrix of two.sided confidence interval for the linear combination, dimension of `ncol(loading.mat)` x 2; the row corresponding to different column in `loading.mat`
`prob.debias.vec`	The vector of bias-corrected estimators after probability transformation, length of `ncol(loading.mat)`; corresponding to different column in `loading.mat`.
`prob.se.vec`	The vector of standard errors of the bias-corrected estimators after probability transformation, length of `ncol(loading.mat)`; corresponding to different column in `loading.mat`.
`prob.ci.mat`	The matrix of two.sided confidence interval of the bias-corrected estimators after probability transformation, dimension of `ncol(loading.mat)` x 2; the row corresponding to different column in `loading.mat`.

Examples

```
X1 = matrix(rnorm(100*5), nrow=100, ncol=5)
y1 = -0.5 + X1[,1] * 0.5 + X1[,2] * 1 + rnorm(100)
X2 = matrix(rnorm(90*5), nrow=90, ncol=5)
y2 = -0.4 + X2[,1] * 0.48 + X2[,2] * 1.1 + rnorm(90)
loading1 = c(1, 1, rep(0,3))
loading2 = c(-0.5, -1, rep(0,3))
loading.mat = cbind(loading1, loading2)
Est = CATE(X1, y1, X2, y2, loading.mat, model="linear")
```

```
## compute confidence intervals
ci(Est, alpha=0.05, alternative="two.sided")

## summary statistics
summary(Est)
```

Dist — *Inference for weighted quadratic functional of difference of the regression vectors (excluding the intercept term) in high dimensional generalized linear regressions.*

Description

Inference for weighted quadratic functional of difference of the regression vectors (excluding the intercept term) in high dimensional generalized linear regressions.

Usage

```
Dist(
  X1,
  y1,
  X2,
  y2,
  G,
  A = NULL,
  model = c("linear", "logistic", "logistic_alter"),
  intercept = TRUE,
  beta.init1 = NULL,
  beta.init2 = NULL,
  split = TRUE,
  lambda = NULL,
  mu = NULL,
  prob.filter = 0.05,
  rescale = 1.1,
  tau = c(0.25, 0.5, 1),
  alpha = 0.05,
  verbose = FALSE
)
```

Arguments

X1	Design matrix for the first sample, of dimension n_1 x p
y1	Outcome vector for the first sample, of length n_1
X2	Design matrix for the second sample, of dimension n_2 x p
y2	Outcome vector for the second sample, of length n_1

G	The set of indices, G in the quadratic form
A	The matrix A in the quadratic form, of dimension $\|G\|\times\|G\|$. If NULL A would be set as the $\|G\|\times\|G\|$ submatrix of the population covariance matrix corresponding to the index set G (default = NULL)
model	The high dimensional regression model, either "linear" or "logistic" or "logistic_alter"
intercept	Should intercept(s) be fitted for the initial estimators (default = TRUE)
beta.init1	The initial estimator of the regression vector for the 1st data (default = NULL)
beta.init2	The initial estimator of the regression vector for the 2nd data (default = NULL)
split	Sampling splitting or not for computing the initial estimators. It take effects only when beta.init1 = NULL or beta.init2 = NULL. (default = TRUE)
lambda	The tuning parameter in fitting initial model. If NULL, it will be picked by cross-validation. (default = NULL)
mu	The dual tuning parameter used in the construction of the projection direction. If NULL it will be searched automatically. (default = NULL)
prob.filter	The threshold of estimated probabilities for filtering observations in logistic regression. (default = 0.05)
rescale	The factor to enlarge the standard error to account for the finite sample bias. (default = 1.1)
tau	The enlargement factor for asymptotic variance of the bias-corrected estimator to handle super-efficiency. It allows for a scalar or vector. (default = c(0.25,0.5,1))
alpha	Level of significance to construct two-sided confidence interval (default = 0.05)
verbose	Should intermediate message(s) be printed. (default = FALSE)

Value

est.plugin	The plugin(biased) estimator for the quadratic form of the regression vectors restricted to G
est.debias	The bias-corrected estimator of the quadratic form of the regression vectors
se	Standard errors of the bias-corrected estimator, length of tau; corrsponding to different values of tau
ci.mat	The matrix of two.sided confidence interval for the quadratic form of the regression vector; row corresponds to different values of tau

Examples

```
X1 = matrix(rnorm(100*5), nrow=100, ncol=5)
y1 = -0.5 + X1[,1] * 0.5 + X1[,2] * 1 + rnorm(100)
X2 = matrix(rnorm(90*5), nrow=90, ncol=5)
y2 = -0.4 + X2[,1] * 0.48 + X2[,2] * 1.1 + rnorm(90)
G = c(1,2)
A = matrix(c(1.5, 0.8, 0.8, 1.5), nrow=2, ncol=2)
Est = Dist(X1, y1, X2, y2, G, A, model="linear")
Est$ci
```

InnProd — *Inference for weighted inner product of the regression vectors in high dimensional generalized linear regressions*

Description

Inference for weighted inner product of the regression vectors in high dimensional generalized linear regressions

Usage

```
InnProd(
  X1,
  y1,
  X2,
  y2,
  G,
  A = NULL,
  model = c("linear", "logistic", "logistic_alter"),
  intercept = TRUE,
  beta.init1 = NULL,
  beta.init2 = NULL,
  split = TRUE,
  lambda = NULL,
  mu = NULL,
  prob.filter = 0.05,
  rescale = 1.1,
  tau = c(0.25, 0.5, 1),
  alpha = 0.05,
  verbose = FALSE
)
```

Arguments

`X1`	Design matrix for the first sample, of dimension n_1 x p
`y1`	Outcome vector for the first sample, of length n_1
`X2`	Design matrix for the second sample, of dimension n_2 x p
`y2`	Outcome vector for the second sample, of length n_1
`G`	The set of indices, `G` in the quadratic form
`A`	The matrix A in the quadratic form, of dimension $\|G\|\times\|G\|$. If `NULL` A would be set as the $\|G\|\times\|G\|$ submatrix of the population covariance matrix corresponding to the index set `G` (default = `NULL`)
`model`	The high dimensional regression model, either `"linear"` or `"logistic"` or `"logistic_alter"`
`intercept`	Should intercept(s) be fitted for the initial estimators (default = `TRUE`)

`beta.init1`	The initial estimator of the regression vector for the 1st data (default = `NULL`)
`beta.init2`	The initial estimator of the regression vector for the 2nd data (default = `NULL`)
`split`	Sampling splitting or not for computing the initial estimators. It take effects only when `beta.init1` = `NULL` or `beta.init2` = `NULL`. (default = `TRUE`)
`lambda`	The tuning parameter in fitting initial model. If `NULL`, it will be picked by cross-validation. (default = `NULL`)
`mu`	The dual tuning parameter used in the construction of the projection direction. If `NULL` it will be searched automatically. (default = `NULL`)
`prob.filter`	The threshold of estimated probabilities for filtering observations in logistic regression. (default = 0.05)
`rescale`	The factor to enlarge the standard error to account for the finite sample bias. (default = 1.1)
`tau`	The enlargement factor for asymptotic variance of the bias-corrected estimator to handle super-efficiency. It allows for a scalar or vector. (default = `c(0.25,0.5,1)`)
`alpha`	Level of significance to construct two-sided confidence interval (default = 0.05)
`verbose`	Should intermediate message(s) be printed. (default = `FALSE`)

Value

`est.plugin`	The plugin(biased) estimator for the inner product form of the regression vectors restricted to `G`
`est.debias`	The bias-corrected estimator of the inner product form of the regression vectors
`se`	Standard errors of the bias-corrected estimator, length of `tau`; corrsponding to different values of `tau`
`ci.mat`	The matrix of two.sided confidence interval for the inner product form of the regression vector; row corresponds to different values of `tau`

Examples

```
X1 = matrix(rnorm(100*5), nrow=100, ncol=5)
y1 = -0.5 + X1[,1] * 0.5 + X1[,2] * 1 + rnorm(100)
X2 = matrix(rnorm(90*5), nrow=90, ncol=5)
y2 = -0.4 + X2[,1] * 0.48 + X2[,2] * 1.1 + rnorm(90)
G = c(1,2)
A = matrix(c(1.5, 0.8, 0.8, 1.5), nrow=2, ncol=2)
Est = InnProd(X1, y1, X2, y2, G, A, model="linear")
Est$ci
```

LF *Inference for linear combination of the regression vector in high dimensional generalized linear regression*

Description

Inference for linear combination of the regression vector in high dimensional generalized linear regression

Usage

```
LF(
  X,
  y,
  loading.mat,
  model = c("linear", "logistic", "logistic_alter"),
  intercept = TRUE,
  intercept.loading = FALSE,
  beta.init = NULL,
  lambda = NULL,
  mu = NULL,
  prob.filter = 0.05,
  rescale = 1.1,
  alpha = 0.05,
  verbose = FALSE
)
```

Arguments

`X`	Design matrix, of dimension n x p
`y`	Outcome vector, of length n
`loading.mat`	Loading matrix, nrow=p, each column corresponds to a loading of interest
`model`	The high dimensional regression model, either `"linear"` or `"logistic"` or `"logistic_alter"`
`intercept`	Should intercept be fitted for the initial estimator (default = `TRUE`)
`intercept.loading`	Should intercept term be included for the loading (default = `FALSE`)
`beta.init`	The initial estimator of the regression vector (default = `NULL`)
`lambda`	The tuning parameter in fitting initial model. If `NULL`, it will be picked by cross-validation. (default = `NULL`)
`mu`	The dual tuning parameter used in the construction of the projection direction. If `NULL` it will be searched automatically. (default = `NULL`)
`prob.filter`	The threshold of estimated probabilities for filtering observations in logistic regression. (default = 0.05)

`rescale`	The factor to enlarge the standard error to account for the finite sample bias. (default = 1.1)
`alpha`	Level of significance to construct two-sided confidence interval (default = 0.05)
`verbose`	Should intermediate message(s) be printed. (default = `FALSE`)

Value

`est.plugin.vec`	The vector of plugin(biased) estimators for the linear combination of regression coefficients, length of `ncol(loading.mat)`; each corresponding to a loading of interest
`est.debias.vec`	The vector of bias-corrected estimators for the linear combination of regression coefficients, length of `ncol(loading.mat)`; each corresponding to a loading of interest
`se.vec`	The vector of standard errors of the bias-corrected estimators, length of `ncol(loading.mat)`; each corresponding to a loading of interest
`ci.mat`	The matrix of two.sided confidence interval for the linear combination, of dimension `ncol(loading.mat)` x 2; each row corresponding to a loading of interest
`proj.mat`	The matrix of projection directions; each column corresponding to a loading of interest.

Examples

```
X = matrix(rnorm(100*5), nrow=100, ncol=5)
y = -0.5 + X[,1] * 0.5 + X[,2] * 1 + rnorm(100)
loading1 = c(1, 1, rep(0, 3))
loading2 = c(-0.5, -1, rep(0, 3))
loading.mat = cbind(loading1, loading2)
Est = LF(X, y, loading.mat, model="linear")

## compute confidence intervals
ci(Est, alpha=0.05, alternative="two.sided")

## summary statistics
summary(Est)
```

QF — *Inference for quadratic forms of the regression vector in high dimensional generalized linear regressions*

Description

Inference for quadratic forms of the regression vector in high dimensional generalized linear regressions

Usage

```
QF(
  X,
  y,
  G,
  A = NULL,
  model = c("linear", "logistic", "logistic_alter"),
  intercept = TRUE,
  beta.init = NULL,
  split = TRUE,
  lambda = NULL,
  mu = NULL,
  prob.filter = 0.05,
  rescale = 1.1,
  tau = c(0.25, 0.5, 1),
  alpha = 0.05,
  verbose = FALSE
)
```

Arguments

X	Design matrix, of dimension n x p
y	Outcome vector, of length n
G	The set of indices, G in the quadratic form
A	The matrix A in the quadratic form, of dimension $\|G\|\times\|G\|$. If NULL A would be set as the $\|G\|\times\|G\|$ submatrix of the population covariance matrix corresponding to the index set G (default = NULL)
model	The high dimensional regression model, either "linear" or "logistic" or "logistic_alter"
intercept	Should intercept be fitted for the initial estimator (default = TRUE)
beta.init	The initial estimator of the regression vector (default = NULL)
split	Sampling splitting or not for computing the initial estimator. It take effects only when beta.init = NULL. (default = TRUE)
lambda	The tuning parameter in fitting initial model. If NULL, it will be picked by cross-validation. (default = NULL)
mu	The dual tuning parameter used in the construction of the projection direction. If NULL it will be searched automatically. (default = NULL)
prob.filter	The threshold of estimated probabilities for filtering observations in logistic regression. (default = 0.05)
rescale	The factor to enlarge the standard error to account for the finite sample bias. (default = 1.1)
tau	The enlargement factor for asymptotic variance of the bias-corrected estimator to handle super-efficiency. It allows for a scalar or vector. (default = c(0.25,0.5,1))
alpha	Level of significance to construct two-sided confidence interval (default = 0.05)
verbose	Should intermediate message(s) be printed. (default = FALSE)

Value

est.plugin	The plugin(biased) estimator for the quadratic form of the regression vector restricted to G
est.debias	The bias-corrected estimator of the quadratic form of the regression vector
se	Standard errors of the bias-corrected estimator, length of tau; corrsponding to different values of tau
ci.mat	The matrix of two.sided confidence interval for the quadratic form of the regression vector; row corresponds to different values of tau

Examples

```
X = matrix(rnorm(100*5), nrow=100, ncol=5)
y = X[,1] * 0.5 + X[,2] * 1 + rnorm(100)
G = c(1,2)
A = matrix(c(1.5, 0.8, 0.8, 1.5), nrow=2, ncol=2)
Est = QF(X, y, G, A, model="linear")
## compute confidence intervals
ci(Est, alpha=0.05, alternative="two.sided")

## summary statistics
summary(Est)
```

Index

CATE, 2

Dist, 4

InnProd, 6

LF, 8

QF, 9