

Package ‘SIHR’

September 26, 2022

Type Package

Title Statistical Inference in High Dimensional Regression

Version 1.1.0

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

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

Description 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) individual treatment effects in generalized linear regression,
(3) quadratic functionals in generalized linear regression ('Guo et al.' (2019) <[arXiv:1909.01503](#)>).

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.0

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

Imports CVXR, glmnet, stats

NeedsCompilation no

Repository CRAN

Date/Publication 2022-09-26 17:50:02 UTC

R topics documented:

ITE	2
LF	4
QF	6
Index	8

ITE *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

```
ITE(
  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 $n_1 \times p$
y1	Outcome vector for the first sample, of length n_1
X2	Design matrix for the second sample, of dimension $n_2 \times p$
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 <code>ncol(loading.mat)</code> ; corresponding to different column in <code>loading.mat</code>
est.debias.vec	The vector of bias-corrected estimators for the linear combination of regression coefficients, length of <code>ncol(loading.mat)</code> ; corresponding to different column in <code>loading.mat</code>
se.vec	The vector of standard errors of the bias-corrected estimators, length of <code>ncol(loading.mat)</code> ; corresponding to different column in <code>loading.mat</code>
ci.mat	The matrix of two.sided confidence interval for the linear combination, dimension of <code>ncol(loading.mat) x 2</code> ; the row corresponding to different column in <code>loading.mat</code>
prob.debias.vec	The vector of bias-corrected estimators after probability transformation, length of <code>ncol(loading.mat)</code> ; corresponding to different column in <code>loading.mat</code> .
prob.se.vec	The vector of standard errors of the bias-corrected estimators after probability transformation, length of <code>ncol(loading.mat)</code> ; corresponding to different column in <code>loading.mat</code> .
prob.ci.mat	The matrix of two.sided confidence interval of the bias-corrected estimators after probability transformation, dimension of <code>ncol(loading.mat) x 2</code> ; the row corresponding to different column in <code>loading.mat</code> .

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 = ITE(X1, y1, X2, y2, loading.mat, model="linear")
```

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

## summary statistics
summary(Est)
```

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

<code>X</code>	Design matrix, of dimension $n \times p$
<code>y</code>	Outcome vector, of length n
<code>loading.mat</code>	Loading matrix, $n \times p$, each column corresponds to a loading of interest
<code>model</code>	The high dimensional regression model, either "linear" or "logistic" or "logistic_alter"
<code>intercept</code>	Should intercept be fitted for the initial estimator (default = TRUE)
<code>intercept.loading</code>	Should intercept term be included for the loading (default = FALSE)
<code>beta.init</code>	The initial estimator of the regression vector (default = NULL)

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

Value

<code>est.plugin.vec</code>	The vector of plugin(biased) estimators for the linear combination of regression coefficients, length of <code>ncol(loading.mat)</code> ; each corresponding to a loading of interest
<code>est.debias.vec</code>	The vector of bias-corrected estimators for the linear combination of regression coefficients, length of <code>ncol(loading.mat)</code> ; each corresponding to a loading of interest
<code>se.vec</code>	The vector of standard errors of the bias-corrected estimators, length of <code>ncol(loading.mat)</code> ; each corresponding to a loading of interest
<code>ci.mat</code>	The matrix of two.sided confidence interval for the linear combination, of dimension <code>ncol(loading.mat) x 2</code> ; each row corresponding to a loading of interest
<code>proj.mat</code>	The matrix of projection directions; each column corresponding to a loading of interest. It will be returned only if <code>verbose</code> set as TRUE

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),
  alpha = 0.05,
  verbose = FALSE
)
```

Arguments

X	Design matrix, of dimension $n \times p$
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)

<code>mu</code>	The dual tuning parameter used in the construction of the projection direction. If NULL it will be searched automatically. (default = NULL)
<code>prob.filter</code>	The threshold of estimated probabilities for filtering observations in logistic regression. (default = 0.05)
<code>rescale</code>	The factor to enlarge the standard error to account for the finite sample bias. (default = 1.1)
<code>tau</code>	The enlargement factor for asymptotic variance of the bias-corrected estimator to handle super-efficiency. It allows for a scalar or vector. (default = $c(0.25, 0.5)$)
<code>alpha</code>	Level of significance to construct two-sided confidence interval (default = 0.05)
<code>verbose</code>	Should intermediate message(s) be printed, the projection direction be returned. (default = FALSE)

Value

<code>est.plugin</code>	The plugin(biased) estimator for the quadratic form of the regression vector restricted to G
<code>est.debias</code>	The bias-corrected estimator of the quadratic form of the regression vector
<code>se</code>	Standard errors of the bias-corrected estimator, length of tau; corresponding to different values of tau
<code>ci.mat</code>	The matrix of two.sided confidence interval for the quadratic form of the regression vector; row corresponds to different values of tau
<code>proj</code>	The projection direction. It will be returned only if verbose set as TRUE

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

ITE, 2

LF, 4

QF, 6