

Package ‘MaximinInfer’

July 21, 2025

Type Package

Title Inference for Maximin Effects in High-Dimensional Settings

Version 2.0.0

Author Zhenyu Wang; Zijian Guo

Maintainer Zhenyu Wang <zw425@stat.rutgers.edu>

Description Implementation of the sampling and aggregation method for the covariate shift maximin effect, which was proposed in <doi:10.48550/arXiv.2011.07568>. It constructs the confidence interval for any linear combination of the high-dimensional maximin effect.

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

Imports MASS, stats, CVXR, glmnet, intervals, SIHR

Depends R (>= 2.10)

NeedsCompilation no

Repository CRAN

Date/Publication 2023-04-12 09:40:02 UTC

Contents

decide_delta	2
Infer	2
Maximin	3
measure_instability	5
Index	6

decide_delta	<i>Decide ridge penalty data-dependently</i>
--------------	--

Description

To tell if the estimator is stable or not without ridge penalty at first. If instable, it picks a ridge penalty data-dependently.

Usage

```
decide_delta(
  obj,
  gen.size = 500,
  step_delta = 0.1,
  MAX_iter = 100,
  verbose = FALSE
)
```

Arguments

obj	The returned list of Maximin
gen.size	The generating sample size (Default = 500)
step_delta	The step size of searching delta (Default = 0.1)
MAX_iter	Maximum of iterations for searching (Default = 100)
verbose	Print information about delta and reward (Default = FALSE)

Value

delta	The data-dependent ridge penalty
reward.ratio	The ratio of penalized reward over non-penalized reward

Infer	<i>Inference method</i>
-------	-------------------------

Description

Given the returned list of Maximin, compute the Point estimator and Confidence interval.

Usage

```
Infer(
  obj,
  delta = 0,
  gen.size = 500,
  threshold = 0,
  alpha = 0.05,
  alpha.thres = 0.01
)
```

Arguments

<code>obj</code>	returned list of Maximin
<code>delta</code>	The ridge penalty (Default = 0)
<code>gen.size</code>	The generating sample size (Default = 500)
<code>threshold</code>	Should generated samples be filtered or not? if 0, use normal threshold to filter; if 1, use chi-square threshold to filter; if 2, do not filter (Default = 0)
<code>alpha</code>	confidence value to construct confidence interval (Default = 0.05)
<code>alpha.thres</code>	confidence value to select generated samples (Default = 0.01)

Value

<code>weight</code>	The weight vector for groups, of length L
<code>mm.effect</code>	The aggregated maximin effect (coefficients), of length p or $p + 1$
<code>mminfer</code>	The list of length $n.loading$, each contains the point estimator and confidence interval

Maximin

Returns a list that provides materials for later inference method.

Description

Given list of observations, compute the bias-corrected initial estimators and do bias-correction to the regressopm covariance matrix.

Usage

```
Maximin(
  Xlist,
  Ylist,
  loading.mat,
  X0 = NULL,
  cov.shift = TRUE,
  cov0 = NULL,
  intercept = TRUE,
```

```

    intercept.loading = FALSE,
    lambda = NULL,
    verbose = FALSE
)

```

Arguments

<code>Xlist</code>	list of design matrix for source data, of length L
<code>Ylist</code>	list of outcome vector for source data, of length L
<code>loading.mat</code>	Loading matrix, of dimension $n.loading \times p$, each column corresponds to a loading of interest
<code>X0</code>	design matrix for target data, of dimension $n0 \times p$ (default = NULL)
<code>cov.shift</code>	Covariate shifts or not between source and target data (default = TRUE)
<code>cov0</code>	Covariance matrix for target data, of dimension $p \times p$ (default = NULL)
<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>lambda</code>	The tuning parameter in fitting initial model. If NULL, it will be picked by cross-validation. (default = NULL)
<code>verbose</code>	Should intermediate message(s) be printed. (default = FALSE)

Details

The algorithm implemented scenarios with or without covariate shift. If `cov0` is specified, the `X0` will be ignored; if not, while `X0` is specified, `cov0` will be estimated by `X0`. If both are not specified, the algorithm will automatically set `cov.shift` as FALSE.

Value

The returned list contains the following components:

<code>Gamma.plugin</code>	The plugin regression covariance matrix
<code>Gamma.debias</code>	The proposed debiased regression covariance matrix
<code>Var.Gamma</code>	The variance matrix for sampling the regression covariance matrix
<code>fits.info</code>	The list of length L , that contains the initial coefficient estimators and variance of fitted residuals.
<code>Points.info</code>	The list of length L , that contains the initial debiased estimator for linear combinations and its corresponding standard error.

Examples

```

L = 2
n1 = n2 = 100; p = 4
X1 = MASS::mvrnorm(n1, rep(0,p), Sigma=diag(p))
X2 = MASS::mvrnorm(n2, rep(0,p), Sigma=0.5*diag(p))
b1 = seq(1,4)/10; b2 = rep(0.2, p)

```

```

y1 = as.vector(X1*%b1+rnorm(n1)); y2 = as.vector(X2*%b2+rnorm(n2))
loading1 = rep(0.4, p)
loading2 = c(-0.5, -0.5, rep(0,p-2))
loading.mat = cbind(loading1, loading2)
cov0 = diag(p)
mm = Maximin(list(X1,X2),list(y1,y2),loading.mat,cov0=cov0)

# inference
out = Infer(mm, gen.size=10)

```

measure_instability *measurement of instability*

Description

compute the instability measurement given a specific ridge penalty

Usage

```

measure_instability(
  obj,
  delta = 0,
  gen.size = 500,
  threshold = 0,
  alpha.thres = 0.01
)

```

Arguments

obj	The returned list of Maximin
delta	The ridge penalty (Default = 0)
gen.size	The generating sample size (Default = 500)
threshold	Should generated samples be filtered or not? if 0, use normal threshold to filter; if 1, use chi-square threshold to filter; if 2, do not filter. (Default = 0)
alpha.thres	The confidence value to select generated samples (Default = 0.01)

Value

The measurement of instability

Index

decide_delta, [2](#)

Infer, [2](#)

Maximin, [3](#)

measure_instability, [5](#)