Package 'MED'

October 12, 2022

Type Package

Title Mediation by Tilted Balancing

Version 0.1.0

Author Gary Chan <kcgchan@u.washington.edu>

Maintainer Gary Chan <kcgchan@u.washington.edu>

Description Nonparametric estimation and inference for natural direct and indirect effects by Chan, Imai, Yam and Zhang (2016) <arXiv:1601.03501>.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2018-03-22 11:50:15 UTC

R topics documented:

MED-package 1
MED 2
summary.MED 5

Index **7**

`MED-package` *Nonparametric estimation of natural mediation effects by tilted balancing.*

Description

This package provides a user-friendly interface for nonparametric efficient inference of natural mediation effects effects for observational data. The package provides point estimates for average treatment effects, natural direct effects and natural indirect effects. The package also allows inference by consistent variance estimators.

Details

Package:	MED
Type:	Package
Version:	0.1.0
Date:	2015-02-09
License:	GPL (>= 2)

The package includes the following functions:

MED:	Estimate the Natural Indirect and Direct Effects
summary.MED:	summary method for class "MED"

Author(s)

Gary Chan, based on the ATE package developed together with Asad Haris.

Maintainer: Gary Chan <kcgchan@uw.edu>

References

Chan, K. C. G., Imai, K, Yam, S. C. P. and Zhang, Z. (2016). "Efficient Nonpmarametric Estimation of Causal Mediation Effects.", under review.

See Also

MED

MED *Estimate the Natural Indirect and Direct Effects*

Description

The main function for estimating the natural direct and indirect effects. This function creates an MED object which can be used as inputs for generic S3 summary function. This function uses a covariate balancing method which creates weights for each subject, without a need to specify a propensity score, mediator regression or outcome regression models. The main function depends on a Newton-Raphson algorithm with backtracking.

Usage

```
MED (Y, Ti, M, X, theta = 0, verbose = FALSE,
    PIE = FALSE, max.iter = 100, tol = 1e-10,
    backtrack = TRUE, backtrack.alpha = 0.3,
    backtrack.beta = 0.5, ...)
```

Arguments

Y	The response vector of length n. This has to be a numeric vector.
Ti	The vector of treatment assignments of length n. Must be coded as $\{0, 1$ 0,1.
M	A $n \times q$-matrix of mediators M. It can be a vector (a single mediator).
X	A $n \times p$-matrix of covariates X to be matched. This matrix does not need to include an intercept.
theta	A real scalar parameter for the Cressie-Read family of objective functions. The default is $\theta = 0$ (exponential tilting). Other popular examples are $\theta = -1$ (empirical likelihood) and $\theta = 1$ (quadratic loss).
verbose	A logical value indicating whether to print the progress of the function. FALSE by default.
PIE	A logical value indicating whether to estimate the pure indirect effect.
max.iter	The maximum number of iterations for the Newton-Raphson methods. For most problems (e.g. with well-behaved functions ρ and u) convergence of Newton-Raphson should be fairly quick.
tol	The absolute tolerance used to determine a stopping criteria for the Newton-Raphson algorithm.
backtrack	A logical value indicating whether to use backtracking in the Newton-Raphson algorithm.
backtrack.alpha	A scalar parameter for backtracking with $\alpha \in (0, 0.5)$.
backtrack.beta	A scalar parameter for backtracking with $\beta \in (0, 1)$.
...	Additional arguments.

Value

The function reruns an object of type "MED", a list with the following elements

est	The vector of point estimates for the average treatment effect. For a binary treatment it also contains the average difference of treatment effects.
vcov	The estimated variance covariance matrix for the estimates of the treatment effects for each treatment group.
lam	The resulting solution of the main optimization problems, $\hat{\lambda}$, as described in Chan et al.(2015). In the case of a simple, binary treatment study, the object has lam.p and lam.q and when ATT = TRUE, we only have lam.q. For a multiple treatment study design we have lam.mat, a matrix with each $\hat{\lambda}$ corresponding to each treatment arm.
weights	The weights obtained by the balancing covariate method for each treatment group. In the case of ATT = TRUE, we only have weights for the untreated. For binary treatment the list would contain either weights.q or weights.p or both. For multiple treatment effect the list contains a $J \times n$ matrix weights.mat.
gp	A string specifying the type of study design. For binary treatment effect with ATT = FALSE is denoted by group "simple". With ATT = TRUE we have "ATT" and finally "MT" is for multiple treatment arms.

conv	A logical value indicating convergence of Newton-Raphson algorithm.
X,Y, Ti	The data which was used for estimation.
rho, rho1, rho2	The Cressie-Read functions ρ used for estimation along with the first and second derivatives.
FUNu	A function that append a vector of constants to the covariates. Required to make sure that the weights sum to 1 in each group.
J	A scalar indicating the number of treatment arms.
K	A scalar indicating the one plus the dimension of the range space of X.
call	The matched call.

Author(s)

Gary Chan, based on package ATE developed with Asad Haris.

References

Chan, K. C. G., Imai, K, Yam, S. C. P. and Zhang, Z. (2016). "Efficient Nonpmarametric Estimation of Causal Mediation Effects.", under review.

See Also

summary.MED

Examples

```
library(MED)
#binary treatment and binary mediator

set.seed(25)
n <- 200
Z <- matrix(rnorm(4*n),ncol=4,nrow=n)
prop.e <- 1 / (1 + exp(Z[,1] - 0.5 * Z[,2] + 0.25*Z[,3] + 0.1 * Z[,4]))
treat <- rbinom(n, 1, prop.e)
prop.m <- 1 / (1 + exp(-(0.5 - Z[,1] + 0.5 * Z[,2] - 0.9 *Z [,3] + Z[,4] - 1.5 * treat)))
M <- rbinom(n, 1, prop.m)
Y <- 200 + treat + M + 27.4*Z[,1] + 13.7*Z[,2] +
          13.7*Z[,3] + 13.7*Z[,4] + rnorm(n)
X <- cbind(exp(Z[,1])/2,Z[,2]/(1+exp(Z[,1])),
          (Z[,1]*Z[,3]/25+0.6)^3,(Z[,2]+Z[,4]+20)^2)

#estimation of natural mediation effecs
fit1<-MED(Y,treat,M,X)
summary(fit1)
```

summary.MED *Summarizing output of study.*

Description

summary method for class "MED"

Usage

```
## S3 method for class 'MED'
summary(object, ...)

## S3 method for class 'summary.MED'
print(x, ...)
```

Arguments

object	An object of class "MED", usually a result of a call to MED.
x	An object of class "summary.MED", usually a result of a call to summary.MED.
...	Further arguments passed to or from methods.

Details

print.summary.MED prints a simplified output similar to print.summary.lm. The resulting table provides the point estimates, estimated standard errors, 95% Wald confidence intervals, the Z-statistic and the P-values for a Z-test.

Value

The function summary.MED returns a list with the following components

Estimate	A matrix with point estimates along with standard errors, confidence intervals etc. This is the matrix users see with the print.summary.RIPW function.
vcov	The variance-covariance matrix of the point estimates.
Conv	The convergence result of the object.
weights	The weights for each subject in each treatment arm. These are same as the weight component of the "RIPW" object.
call	The call passed on as an argument of the function which is equivalent to object$call.

Author(s)

Gary Chan

See Also

MED

Examples

```
library(MED)
#binary treatment and binary mediator

set.seed(25)
n <- 200
Z <- matrix(rnorm(4*n),ncol=4,nrow=n)
prop.e <- 1 / (1 + exp(Z[,1] - 0.5 * Z[,2] + 0.25*Z[,3] + 0.1 * Z[,4]))
treat <- rbinom(n, 1, prop.e)
prop.m <- 1 / (1 + exp(-(0.5 - Z[,1] + 0.5 * Z[,2] - 0.9 *Z [,3] + Z[,4] - 1.5 * treat)))
M <- rbinom(n, 1, prop.m)
Y <- 200 + treat + M + 27.4*Z[,1] + 13.7*Z[,2] +
          13.7*Z[,3] + 13.7*Z[,4] + rnorm(n)
X <- cbind(exp(Z[,1])/2,Z[,2]/(1+exp(Z[,1])),
          (Z[,1]*Z[,3]/25+0.6)^3,(Z[,2]+Z[,4]+20)^2)

#estimation of natural mediation effecs
fit1<-MED(Y,treat,M,X)
summary(fit1)
```

Index

∗ **package**
MED-package, 1

MED, *2*, 2, *5*
MED-package, 1

print.summary.lm, *5*
print.summary.MED *(*summary.MED*)*, 5

summary.MED, *2*, *4*, *5*, 5